Structural Optimization by Genetic Algorithms with Tournament Selection

By Jiaping Yang[1] and Chee Kiong Soh[2]

Abstract: A new approach to optimization design concerning the configurations of structures using genetic algorithm (GA) with a tournament selection strategy has been proposed. The tournament selection strategy is used as a replacement for the commonly used fitness-proportional selection strategy to drive the GA so as to improve the fitness of each succeeding generation more efficiently. Numerical results for three examples reveal that a significant reduction of computation cost has been achieved in the newly proposed GA with tournament selection, as compared to the widely used GA with fitness-proportional selection and other hybrid GA approaches. Also, it has verified that the tournament selection performs well over the fitness-proportional selection and other hybrid techniques in enhancing GA search efficiency.

INTRODUCTION

Most optimization designs of structures are currently based on numerical optimization methods where a lot of gradient information is required (Soh and Yang 1996). In this paper, a new approach using a genetic-based evolution algorithm without gradient requirement is applied to the configuration optimization of truss structures. Genetic-based evolution algorithms were originally introduced by Holland (1975), and later refined by De Jong (1975), Goldberg (1989), and others. The algorithms imitate the evolutionary process with a particular focus on genetic mechanisms. The simple genetic algorithm (GA) approach (Goldberg 1989) operates on a group of individuals, where each individual represents a solution, encoding all the design variables. The evolution of generations are carried out by applying the genetic operators (selection, crossover, and mutation). Over the past few years, this innovative algorithm has proved robust in solving very complex optimization problems (Davis 1991).

The GA approach for engineering optimization was introduced by Goldberg (1987). Recently, it has seen significant applications in structural optimization designs. However, most of the applications (Hajela 1990; Deb 1991; Jenkins 1991, 1992; Lin and Hajela 1992; Rajeev and Kirshnamoorthy 1992; Adeli and Cheng 1993, 1994; Koumousis and Georgiou 1994; Soh and Yang 1996; Yang 1996) are based on the simple GA approach; one character of this approach is use of the fitness-proportional selection scheme. The principle of this selection scheme is to give good individuals, i.e., higher fitness value, a higher chance of passing their genes to the succeeding generation. One may think of each individual in the current population as a segment in a wheel of fortune with segment size according to their fitness (often referred to as the roulette wheel selection). The selection probability P_i of an individual i is given by either $P_i = \Phi_i/\Sigma_{j=1}^{n} \Phi_j$ or $P_i = \Phi_i/\Phi_{avg}$, where Φ_i and Φ_{avg} are the ith fitness function value and the average fitness function value, respectively. The wheel is spun twice to determine two mating parents for the crossover operation. (This will be illustrated in a later figure.)

As shown in some GA-related research (Thierens and Goldberg 1994), the choice of selection schemes is important to improve GA behavior in the search process. Therefore, there are some researchers who attempted to employ several variations of the roulette wheel selection scheme. They did so because theoretical studies on these variations have shown their superiority over the roulette wheel selection (De Jong 1975; Baker 1987; Michalewicz 1992). For example, Chapman et al. (1994) adopted an improved derivative of the roulette wheel selection called stochastic universal sampling selection (Baker 1987) in their GA approach for optimization designs. Grierson and Pak (1993) as well as Wu and Chow (1995) used a rank-based selection scheme (Whitley 1989) for the reproduction operation. In this scheme individuals are selected according to their relative rank in the population rather than their individual fitness relative to the population average or the sum of population fitnesses. Their motivation is to avoid the drawback of roulette wheel selection, which may result in the rather fit (but nonoptimal) solutions quickly dominating the population and undermining population diversity, leading to a premature stagnation of the GA search process.

In this paper, a new GA-based evolution approach with the tournament selection scheme is applied to the configuration optimization design of structures. The objective is to make a comparative analysis of the GA with the tournament selection (GATS) approach and GA with the roulette wheel selection (GARWS) approach (Yang 1996). Furthermore, comparison is also made with our earlier work on hybrid fuzzy controlled GA with roulette wheel selection (FC-GARWS) approach (Soh and Yang 1996). The following two sections outline the general formulation of the configuration optimization design of truss structures and characteristics of the GA approach. Then, the optimization procedure GATS is described. Three configuration optimization design examples are presented to demonstrate the effectiveness of the proposed GATS approach. Finally, we conclude with a summary of the benefit of the proposed GATS approach.

FORMULATION FOR CONFIGURATION OPTIMIZATION OF TRUSSES

The general formulation for the configuration optimization design of trusses can be mathematically expressed as follows: Find the particular set of **R** and **A** that minimize a function $W(\mathbf{R}, \mathbf{A})$, which usually represents either structural weight or volume. This minimum design must also satisfy the inequality constraints that limit design variable size and structural responses. Thus the problem is

$$\text{minimize } W(\mathbf{R}, \mathbf{A}) \tag{1}$$

$$\text{subject to } G_i^L \le G_i(\mathbf{R}, \mathbf{A}) \le G_i^U,\ i = 1, 2, \ldots, q \tag{2}$$

[1]Res. Engr., School of Appl. Sci., Nanyang Technol. Univ., Nanyang Ave., Singapore 639798.
[2]Assoc. Prof., School of Civ. and Struct. Engrg., Nanyang Technol. Univ., Nanyang Ave., Singapore 639798.

Note. Discussion open until December 1, 1997. To extend the closing date one month, a written request must be filed with the ASCE Manager of Journals. The manuscript for this paper was submitted for review and possible publication on August 22, 1996. This paper is part of the ***Journal of Computing in Civil Engineering***, Vol. 11, No. 3, July, 1997. ©ASCE, ISSN 0887-3801/97/0003-0195–0200/$4.00 + $.50 per page. Paper No. 14062.

J. Comput. Civ. Eng. 1997.11:195-200.

where **R** and **A** = vectors of joint coordinates and member cross-sectional areas, respectively. Bounds on the constraint functions $G_i(\mathbf{R}, \mathbf{A})$ involve (1) joint coordinates; (2) member cross-sectional areas; (3) member allowable stresses; (4) joint displacements; and (5) member buckling strength. Here q is the total number of the constraints, and L and U are superscripts denoting the lower and upper bounds, respectively.

CHARACTERISTICS OF GENETIC ALGORITHMS

GA has several important features. First, GA's simplicity and directness of characters make the representation of design, which must be encoded in a chromosome, easy for a variety of domain problems (Goldberg 1989). Second, the GA operators acting on a chromosome at random are actually applied to a set or population of design rather than a single design point. This enables GA to explore the search space from many different points, simultaneously, and find an optimum by a more "global" search strategy instead of a localized gradient search or "hill-climbing" approach. Third, the fact that no gradient information is required avoids the mathematical complexity of numerical optimization methods. Fourth, the inherent ability of GA in implicit parallel computing can reduce the length of time during the evolution process. Finally, the GA approach has demonstrated certain aspects of intelligence characterized by human beings. That is, it exploits "best" inheritance accumulated during the evolution in a way that efficiently trades off the need to explore new regions of the search space with the need to focus on a high-performance region of that space. Thus the GA approach has been considered an alternative optimization tool for a wide variety of optimization problems.

GA OPTIMIZATION PROCEDURE

The fundamental mechanisms leading the GA search process are the equivalents of natural selection, crossover, and mutation. GA deals with a population that is a collection of individuals, and the chromosome of each individual represents a candidate solution, i.e., an optimum design in the present problem. The GA approach flowchart for configuration optimization of trusses is shown in Fig. 1, where the parameter $G_{\max}$ in the stop criterion is the maximum number of generations allowed. This stop criterion is prescribed by the designers in advance. Essentially, it assumed that there is no more improvement in finding an optimum solution after the permitted maximum number of generations.

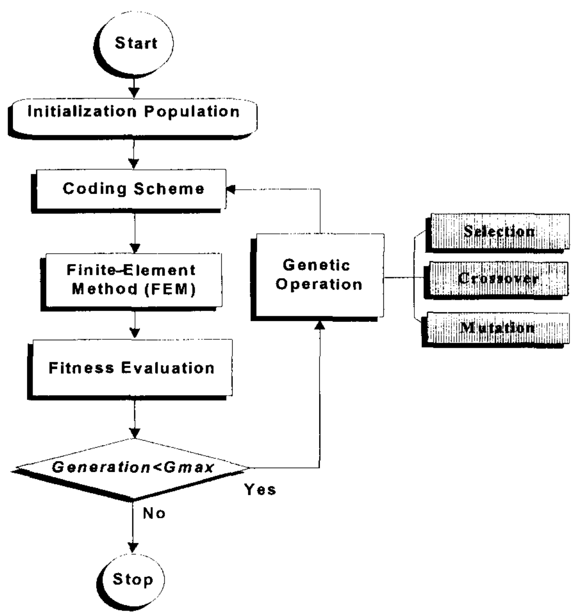

FIG. 1. Flowchart of GA Approach

An optimum solution to the problem described by (1) and (2), derived according to the GA evolutionary computation process as shown in Fig. 1, can be sought by: (1) producing the initial population; (2) defining a coding scheme for all the design variables; (3) running the finite-element analysis on these design variables; (4) evaluating the fitness function with respect to the objective function; and (5) performing the computation with the genetic operators.

In our study, the initial population is created randomly. The most commonly used binary coding method is adopted. It directly transforms all design variables **R** and **A** in each design solution (i.e., an individual) into a concatenation of binary strings, in which each design variable is mapped to one binary string with a corresponding length. These binary strings construct a chromosome for the individual. The fitness or "goodness" of each individual with respect to the objective function of the problem is evaluated by the finite-element method. In addition, GA can be used directly for unconstrained problems only. Thus the structural configuration optimal design, which is a constrained optimum problem, needs to be transformed to an unconstrained problem by the introduction of the exterior penalty function. Mathematically, the objective function $W(\mathbf{R}, \mathbf{A})$ can be represented by $\Phi(\mathbf{R}, \mathbf{A})$ in the following form:

$$\Phi(\mathbf{R}, \mathbf{A}) = W(\mathbf{R}, \mathbf{A}) + \sum_{j=1}^{p} c_j \sum_{i=1}^{n_j} \left[\frac{|G_{ij}|}{|G_{ij}^a|} - 1\right]^2 \quad (q = p \times n_j)$$

$$G_{ij}^a = G_{ij}^L, \quad \text{when } G_{ij} < 0 \tag{3a}$$

$$G_{ij}^a = G_{ij}^U, \quad \text{when } G_{ij} \geq 0 \tag{3b}$$

$$\left(\frac{|G_{ij}|}{|G_{ij}^a|} - 1\right) = \max\left(\frac{|G_{ij}|}{|G_{ij}^a|} - 1, 0\right) \tag{3c}$$

where c_j = a set of penalty parameters; p = total number of constraint types; and n_j = total number of the jth constraint type. The penalty parameters are obtained by trials. They are the same as those in earlier work (Soh and Yang 1996; Yang 1996). As GA seeks to maximize $\Phi(\mathbf{R}, \mathbf{A})$, a rescaling evaluation function is defined as follows:

$$\text{fitness} = \begin{cases} C_{avg}(\text{generation}) - \Phi(\mathbf{R}, \mathbf{A}) & \text{when } \Phi(\mathbf{R}, \mathbf{A}) < C_{avg} \\ 0 & \text{when } \Phi(\mathbf{R}, \mathbf{A}) \geq C_{avg} \end{cases} \tag{4}$$

where C_{avg} = average value of $\Phi(\mathbf{R}, \mathbf{A})$ of the current generation. It varies with each generation, and prevents a vector of the design variables with fitness less than or equal to C_{avg} from entering a new "mating pool" of individuals; these individuals will be used in the next generation. When GA seeks to maximize the fitness, $\Phi(\mathbf{R}, \mathbf{A})$ will arrive at a minimum value. Hence, the objective function $W(\mathbf{R}, \mathbf{A})$ could be minimized. Such a fitness evaluation objective function can adequately measure the performance of an individual (a candidate solution).

TOURNAMENT SELECTION STRATEGY

The evolution of GA from generation to generation is performed by three main GA operators: selection, crossover, and mutation operators. Here, our focus is on the description of selection operator. The crossover and mutation operators are referred to in Goldberg (1989) and Michalewicz (1992).

GA uses a strategy to select individuals from the population and insert them into a mating pool. Individuals from the mating pool are used to generate new offspring, which are the basis of the next generation. As the individuals in the mating pool are the ones whose genes will be inherited by the next generation, it is desirable that the mating pool consist of good individuals. A selection strategy in GA is simply a process that favors the selection of better individuals in the population for the mating pool (Goldberg and Deb 1991).

J. Comput. Civ. Eng. 1997.11:195-200.

There are two important issues in the evolution process of the genetic search: population diversity and selective pressure (Whitley 1989). Population diversity means that the genes from the already discovered good individuals are exploited while the promising new areas of the search space continue to be explored. Selective pressure is the degree to which the better individuals are favored: whereby the higher the selective pressure, the more the better individuals are favored. This selective pressure drives the GA to improve population fitness over succeeding generations. The convergence rate of a GA is largely determined by the selective pressure and population diversity. In general, higher selective pressures result in higher convergence rates. However, if the selective pressure is too high, there is an increased chance of the GA prematurely converging to a local optimal solution because the population diversity of the search space to be explored is lost. If the selective pressure is too low, the convergence rate will be slow and the GA will take an unnecessarily long time to find the optimal solution because more genes are explored in the search. An ideal selection strategy should be one that is able to adjust its selective pressure and population diversity so as to fine-tune the GA search performance.

Whitley (1989) pointed out that the fitness-proportional selection (i.e., roulette wheel selection) is likely to lead to two problems (1) stagnation of the search because it lacks selective pressure; and (2) premature convergence of the search because it causes the search to narrow down too quickly. Unlike the roulette wheel selection, the tournament selection strategy provides selective pressure by holding a tournament competition among N_{ts} individuals (frequently $N_{ts} = 2$) (Goldberg and Deb 1991). The best individual (the winner) from this tournament is the one with the highest fitness Φ_{winner} of the N_{ts} tournament competitors, and the winner is then inserted into the mating pool. The tournament competition is repeated until the mating pool for generating new offspring is filled. The mating pool, comprising the tournament winners, has a higher average fitness than the average population fitness. The fitness difference provides the selective pressure, which drives the GA to improve the fitness of each succeeding generation. Fig. 2 illustrates the roulette wheel selection and tournament selection strategies. Goldberg and Deb have compared the different selection schemes under idealized assumptions and seem to favor tournament selection over roulette wheel selection. Thus, the tournament selection strategy is used in this study.

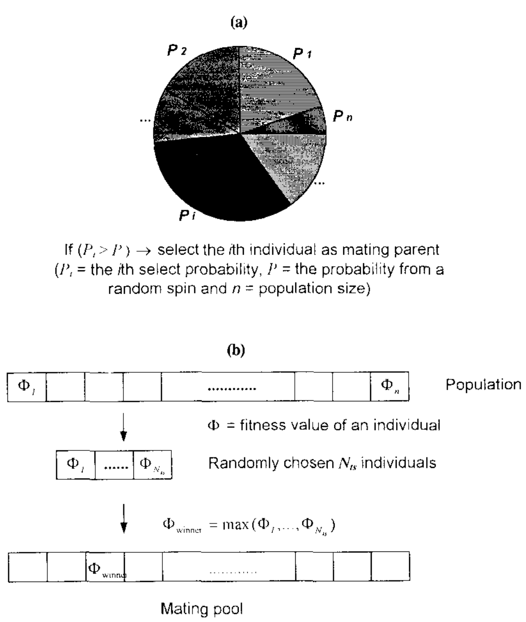

FIG. 2. Illustration of: (a) Roulette Wheel Selection; (b) Tournament Selection

Due to its evolutionary nature, GA will search for optimal solutions without regard to the specific inner working of the problems. Once one solution (an individual) violates some constraints, it will be thrown out. From a design standpoint, it may seem logical to prevent the search process from considering an invalid solution. But in GA, variety is its innate nature. During the early genetic evolution process, there are a large number of individuals or chromosomes that almost satisfy "all" constraints, except one or two. A change in one or two design variables (strings) may produce a solution with a higher fitness value. This means that throwing out these solutions at the early evolution stage may result in loss of some important information, which might eventually lead to the optimal solution.

To keep potentially good individuals in the mating pool, a modified evaluation method (Jenkins 1992) was used in this study by replacing the hard constraints with "soften" constraints to increase the GA's performance. A set of small transition parameters, ε_i, defining the transition between the satisfied and unsatisfied constraints is used in the soften constraints. For example, the constraints on the member allowable stress, member buckling strength, and joint allowable displacement given in (2) can be transformed into the following inequalities:

$$\frac{|\sigma_i|}{|\sigma_i^a|} - (1 + \varepsilon_1) \leq 0 \tag{5a}$$

$$\frac{|\sigma_i^b|}{|\sigma_i^{ab}|} - (1 + \varepsilon_2) \leq 0 \tag{5b}$$

$$\frac{|\Delta_i|}{|\Delta_i^a|} - (1 + \varepsilon_3) \leq 0 \tag{5c}$$

These parameters (with an initial value of 0.01) will then decrease sequentially with each 10 generations until they reach the predefined tolerance levels (e.g., 0.0001). When the constraint conditions given in (5) are satisfied, the relevant individual is not necessarily removed from the mating pool, even though it does not strictly satisfy all the inequalities in (2). Thus the individual still has the opportunity for selection, crossover, and mutation in the next generation.

NUMERICAL EXAMPLES

Three examples are presented to demonstrate the efficiency of the proposed GATS approach for configuration optimization of truss structures. The computations were carried out on a PC486. In this study, we used the genetic algorithm with two-point crossover and mutation operators. The population size was set at 40, and crossover and mutation probabilities were 0.87 and 0.004, respectively. These GA parameters are close to those typically used by GA researchers (Goldberg 1989; Whitley 1989; Davis 1991). Goldberg has proved that higher crossover probability and lower mutation probability are good for GA to converge to an optimal solution efficiently. They were also adopted in Soh and Yang (1996) and Yang (1996). The other detailed design data are given in Table 1.

TABLE 1. Design Information for Examples 1, 2, and 3

Items (1)	Example 1 (2)	Example 2 (3)	Example 3 (4)
Young's modulus (E)	0.1E8 psi	0.1E8 psi	210 kN/mm^2
Allowable member stress (σ)	±0.2E5 psi	±0.4E6 psi	165 N/mm^2
Density (ρ)	0.1 lb cu in.	0.1 lb cu in.	0.1 kg/cm^3
Allowable joint displacement (Δ)	None	None	±20.0 mm at joints 1, 17, 23, 29, 35
Lower and upper cross-sectional area size (A)	3.5–18.0 sq in.	0.01–1.0 sq in.	150–1,000 mm^2

Note: 1 psi = 6.89 kPa; 1 in. = 25.4 mm; 1 sq in. = 6.4516 cm^2; and 1 lb/cu in. = 0.0277 kg/cm^3.

J. Comput. Civ. Eng. 1997.11:195-200.

Example 1: 18-Member Cantilever Plane Steel Truss

This is a statically determinate cantilever plane steel truss that has previously been used to illustrate the FC-GARWS approach (Soh and Yang 1996) (see Fig. 3). All members of the structure have been categorized into four groups, as shown in Table 2. The single loading condition is a set of vertical loads (P = 88.96 kN) acting on the upper joints of the truss. The lower joints 3, 5, 7, and 9 were allowed to move in any direction in the X-Y plane. Two different joint condition situations are considered. Cases 1 and 2 assumed that the location of joint 11 can be unchanged and changed, respectively, in the Y-direction. Thus, for case 1 there is a total of 12 design variables, which include four sizing variables and eight independent coordinate variables. For case 2 there is a total of 13 design variables, which include four sizing variables and nine independent coordinate variables. The goal is to design an optimal configuration for the truss to produce a minimum weight design that meets both the allowable stress and the buckling constraints. The buckling of member i is computed as $-4EA_i/L_i^2$(L_i is the length of the ith member).

A brief description on how the GATS approach solved Case 1 is given to illustrate GATS optimization search process.

At the beginning of operation, namely generation 0, GATS randomly initialized a population of 40 individuals. Then these 40 individuals were investigated one by one, repeating the following three-step process (which is carried out by the three modules shown in Fig. 1, namely "Coding Scheme," "Finite-Element Method," and "Fitness Evaluation," respectively):

1. To transform each binary string of an individual into the relevant sizing and coordinate variables based on the given lower and upper bounds.
2. To perform a finite-element analysis on these design variables.
3. To evaluate the fitness function value by checking the objective function and constraint conditions on structural performance.

Subsequently, the 40 groups of N_{ts} individuals are randomly chosen as the tournament competitors. The 40 winners with the higher fitness values are then inserted into a mating pool. Based on the crossover probability P_c and mutation probability P_m, the genetic operations were performed on the mating pool so as to produce 40 new individuals (offsprings). These 40 offspring constituted a new population. After that GATS went back to the next generation (generation 1) to investigate the 40 new individuals by repeating the three-step process (i.e., the transformation of design variables, structural analysis, and evaluation), as described earlier. Such a cycle of generations was repeated until the permitted maximum number of generations was reached.

Comparison of the results obtained using GATS with our earlier solutions obtained using FC-GARWS (Soh and Yang 1996) and those obtained using a simpler GARWS (Yang 1996) are shown in Tables 2 and 3. The tables show that the proposed GATS approach gives a better performance over the FC-GARWS approach in search efficiency. The reduction in computing time is 60 and 58% in terms of number of iterations, i.e., number of generations required, compared with the 52 and 37% of the FC-GARWS approach for the two cases, respectively, with respect to the simpler GARWS approach.

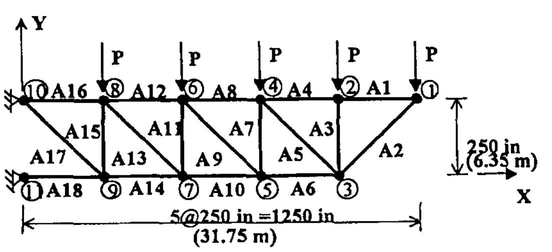

FIG. 3. 18-Member Cantilever Plane Steel Truss

TABLE 2. Optimal Results for 18-Member Plane Truss under Case 1

Design variables A_i (sq in.), X_j and Y_j (in.) (1)	Results Obtained Using GARWS (Yang 1996) (2)	FC-GARWS (Soh and Yang 1996) (3)	GATS (4)
$A_1 = A_4 = A_8 = A_{12} = A_{16}$	12.61	12.59	12.33
$A_2 = A_6 = A_{10} = A_{14} = A_{18}$	18.10	17.912	17.97
$A_3 = A_7 = A_{11} = A_{15}$	5.47	5.5	5.60
$A_5 = A_9 = A_{13} = A_{17}$	3.54	3.553	3.66
X_3	914.5	909.79	907.19
Y_3	183.04	184.47	184.23
X_5	646.98	640.31	643.29
Y_5	147.36	147.83	149.23
X_7	414.24	410.03	413.93
Y_7	100.37	96.99	102.0
X_9	200.04	200.87	202.07
Y_9	31.9	31.96	30.90
Weight (lb)	4,552.8	4,531.9	4,520.0
Total number of iterations (generation)	75	36	30
Reduction in iterations	—	52%	60%

Note: 1 lb = 4.448 N; 1 in. = 25.4 mm; and 1 sq in. = 6.452 cm.^{2}

TABLE 3. Optimal Results for 18-Member Plane Truss under Case 2

Design variables A_i sq in., X_j and Y_j (in.) (1)	Results Obtained Using GARWS (Yang 1996) (2)	FC-GARWS (Soh and Yang 1996) (3)	GATS (4)
$A_1 = A_4 = A_8 = A_{12} = A_{16}$	10.86	10.98	10.68
$A_2 = A_6 = A_{10} = A_{14} = A_{18}$	15.70	15.54	15.95
$A_3 = A_7 = A_{11} = A_{15}$	8.34	8.35	8.23
$A_5 = A_9 = A_{13} = A_{17}$	3.54	3.51	3.53
X_3	889.13	890.31	883.27
Y_3	138.02	134.92	138.56
X_5	607.28	608.22	607.14
Y_5	101.17	106.2	107.48
X_7	377.39	384.31	386.97
Y_7	51.67	56.46	54.09
X_9	190.58	188.75	194.65
Y_9	−7.45	−6.98	−9.74
Y_{11}	−53.82	−38.38	−53.16
Weight (lb)	4,424.6	4,404.6	4,419.3
Total number of iterations (generation)	71	45	30
Reduction in iterations	—	37%	58%

Note: 1 lb = 4.448 N; 1 in. = 25.4 mm; and 1 sq in. = 6.452 cm.^{2}

Example 2: 25-Member Transmission Tower Space Truss

All the members of a 25-member transmission tower space truss, as shown in Fig. 4, were classified into eight groups to represent eight different kinds of cross-sectional areas. Two load conditions and member grouping are given in Tables 4 and 5, respectively. In addition, the truss is required to remain symmetric with respect to both the X-Z plane and the Y-Z plane. The coordinates of joints 1 and 2 were held constant, and joints 7–10 were required to lie in the X-Y plane. Due to the symmetry of the structure, there are a total of 13 design

J. Comput. Civ. Eng. 1997.11:195-200.

variables, which include eight sizing variables and five independent coordinates variables (X_4, Y_4, Z_4, X_8, and Y_8) for each load case. The design objective is to minimize the weight of the structure subject to constraints on the member stress shown in Table 1 and buckling of $-100.01\pi EA_i/8L_i^2$. The results of the configuration optimization are given in Table 6. A lighter

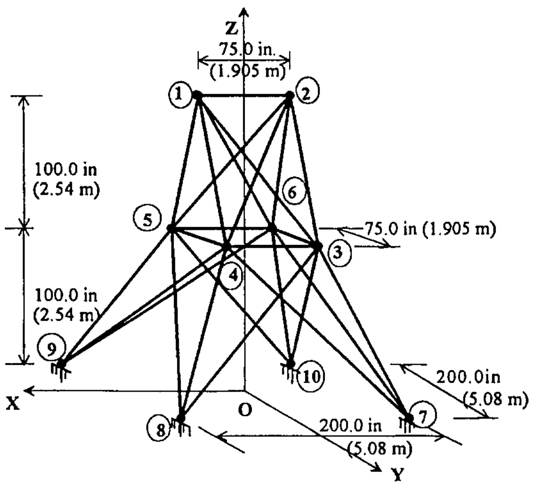

FIG. 4. 25-Member Transmission Tower Space Truss

TABLE 4. Load Conditions for 25-Member Space Truss (1 kip = 4.448 kN)

Load cases (1)	Loading (2)	Force at Joints (kip) 1 (3)	2 (4)	3 (5)	6 (6)
1	Fx	0.0	0.0	0.0	0.0
	Fy	20.0	−20.0	0.0	0.0
	Fz	−5.0	−5.0	0.0	0.0
2	Fx	1.0	0.0	0.5	0.5
	Fy	10.0	10.0	0.0	0.0
	Fz	−5.0	10.0	0.0	0.0

TABLE 5. Member Group for 25-Member Space Truss

Group (1)	Member (end joints) (2)
A_1	1-2
A_2	1-3, 2-4, 2-5, 1-6
A_3	1-4, 2-3, 2-6, 1-5
A_4	3-6, 4-5
A_5	3-4, 5-6
A_6	6-7, 3-10, 5-8, 4-9
A_7	3-8, 4-7, 6-9, 5-10
A_8	3-7, 4-8, 5-9, 6-10

TABLE 6. Optimal Results for 25-Member Space Truss

Geometric variables X, Y, and Z (in.) (1)	Results Obtained Using GARWS (Yang 1996) (2)	FC-GARWS (Soh and Yang 1996) (3)	GATS (4)
X_4	22.63	21.98	22.48
Y_4	41.99	43.57	48.91
Z_4	98.84	96.84	100.59
X_8	15.58	14.13	25.21
Y_8	82.37	81.14	98.13
Weight (lb)	137.2	132.3	131.3
Total number of iterations (generation)	47	38	30
Reduction in iterations	—	19%	36%

Note: 1 lb = 4.448 N; 1 in. = 25.4 mm.

truss weight was obtained after 30 iterations (generation) using the GATS approach as compared with 47 and 38 iterations (generation) using the GARWS (Yang 1996) and FC-GARWS approaches (Soh and Yang 1996), respectively.

Example 3: 112-Member Dome Space Truss

The space structure depicted in Fig. 5 has 112 members. The members have been categorized into two groups. For clarity, the joint numbers (circled) and the group numbers of the members are partially shown in the figure. The structure has been subjected to a vertical loading at each unsupported joint. The detailed data are −5.0 kN for joint 1; −0.4 kN for joints 17, 23, 29, and 35; −1.2 kN for joints 16, 18, 22, 24, 28, 30, 34, and 36; and −2.0 kN for other joints. All the unsupported joints were allowed to move vertically within an allowable limit defined by the functional requirements. Thus, there are a total of five design variables, which include two sizing variables and three independent coordinates ($Z1$, $Z2$, and $Z3$) as shown in Fig. 5. The objective is to minimize the weight of the structure subject to constraints on stress, displacement and buckling. The stress and displacement constraints are shown in Table 1. The buckling stress constraint σ_i^b of member i is computed as follows:

$$\text{if } \lambda_i > C\text{: elastic buckling, } \sigma_i^b = k_b \frac{\pi^2 E}{\lambda_i^2} \tag{6}$$

$$\text{if } \lambda_i < C\text{: plastic buckling, } \begin{cases} \sigma_i^b = \sigma_y \left(1 - \dfrac{\lambda_i^2}{2C^2}\right) \Big/ n \\ n = \dfrac{5}{3} + \dfrac{3\lambda_i}{8C} - \dfrac{\lambda_i^3}{8C^3} \end{cases} \tag{7}$$

where $\lambda_i = L_i/r_i$, $r = 0.4993 \cdot A^{0.6777}$, respectively, for the tubular sections; σ_y = yield stress given as $\sigma_{\text{allowable}}/0.6$; $C = \sqrt{2\pi^2 E/\sigma_y}$; and $k_b = 12/13$. An optimum design having a minimum steel weight of 3,262.3 kg (as shown in Table 7) was obtained after 30 generations using the GATS approach. Compared with the optimum steel weight of 3,451.0 and 3,327.8 kg, obtained using the GARWS (Yang 1996) and FC-GARWS approaches (Soh and Yang 1996), the proposed GATS ap-

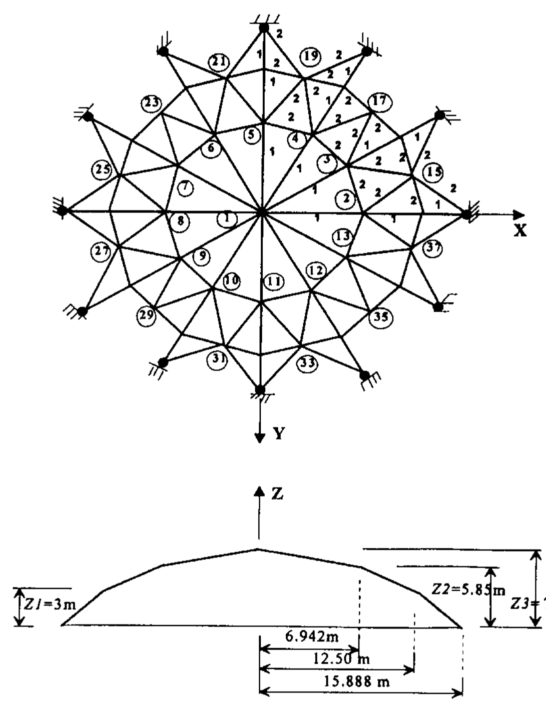

FIG. 5. 112-Member Dome Space Truss

TABLE 7. Optimal Results for 112-Member Dome

Design variable A_i (mm²) and Z_j (m) (1)	Results Obtained Using GARWS (Yang 1996) (2)	FC-GARWS (Soh and Yang 1996) (3)	GATS (4)
A_1	588.41	597.80	618.90
A_2	575.556	538.83	511.09
Z1	2.84	2.85	2.78
Z2	6.14	6.11	5.89
Z3	7.52	7.45	8.00
Total number of iterations (generation)	72	41	30
Reduction in iterations		43%	58%
Weight (kg)	3,451.0	3,327.8	3,262.30

proach has resulted in 58% reduction in the required number of iterations (generation), compared with 43% obtained by FC-GARWS, with respect to GARWS. Again, this example verified that the proposed GATS approach performs better over both the GARWS (Yang 1996) and the FC-GARWS approach (Soh and Yang 1996) in terms of weight and search efficiency.

CONCLUSIONS

The paper presents a new GA search approach incorporated with the tournament selection strategy for optimization of truss configurations. By means of GA combined with the finite-element analysis, this approach can be applied to finding the optimal configuration of structures. Compared with the current widely used GA approach with the roulette wheel selection (for example, GARWS and FC-GARWS), the proposed GATS approach is able to search for an optimum solution in a more efficient way. Thus it enables the approach to have a greater potential for solving a wide variety of optimization problems. Numerical results obtained here also proved that the GATS approach works well and is able to obtain better solutions within a lower number of generations for the configuration optimization design of truss structures. It has verified that tournament selection performs well over the roulette wheel selection (fitness-proportional selection) and other hybrid techniques in enhancing GA search efficiency. When the objective function involves structural material and labor costs, the final configuration optimal design may be different. More work on multiobjective optimization design problems using GA is needed.

APPENDIX. REFERENCES

Adeli, H., and Cheng, N. T. (1993). "Integrated genetic algorithm for optimization of space structures." *J. Aerosp. Engrg.*, ASCE, 6(4), 315–328.

Adeli, H., and Cheng, N. T. (1994). "An augmented Lagrangian genetic algorithms for structural optimization." *J. Aerosp. Engrg.*, ASCE, 7(1), 104–118.

Baker, J. E. (1987). "Reducing bias and inefficiency in the selection algorithms." *Proc., 2nd Int. Conf. on Genetic Algorithms*, Grefenstette, ed., 14-21, Lawrence Erlbaum Associates, Hillsdale, N.J.

Chapman, C. D., Saitou, K., and Jakiela, M. J. (1994). "Genetic algorithms as an approach to configuration and topology design." *J. Mech. Des.*, 116, 1005–1012.

Davis, L. (1991). *Handbook of genetic algorithms*. Van Nostrand Reinhold, New York, N.Y.

Deb, K. (1991). "Optimal design of a welded beam via genetic algorithms." *AIAA J.*, 29(11), 2013–2015.

De Jong, K. (1975). "An analysis of the behavior of a class of genetic adaptive systems," PhD thesis, Univ. of Michigan, Ann Arbor, Mich.

Goldberg, D. E. (1987). "Computer-aided gas pipeline operation using genetic algorithms and rule learning, part I and II." *J. Engrg. with Comp.*, 3, 35–58.

Goldberg, D. E. (1989). *Genetic algorithms in search, optimization and machine learning*. Addison-Wesley, Reading, Mass.

Goldberg, D. E., and Deb, K. (1991). "A comparative analysis of selection schemes used in genetic algorithms." *Found. of Genetic Algorithms*, I, 69–93.

Grierson, D. E., and Pak, W. H. (1993). "Optimal sizing, geometrical and topological design using a genetic algorithm." *J. Struct. Optimization*, 6, 151–159.

Hajela, P. (1990). "Genetic search: an approach to the non-convex optimization problem." *AIAA J.*, 28(7), 1205–1210.

Holland, J. H. (1975). *Adaptation in natural and artificial systems*. University of Michigan Press, Ann Arbor, Mich.

Jenkins, W. M. (1991). "Towards structural optimization via the genetic algorithms." *Comp. and Struct.*, 40(5), 1321–1327.

Jenkins, W. M. (1992). "Plane frame optimum design environment based on genetic algorithms." *J. Struct. Engrg.*, ASCE, 118(11), 3103–3112.

Koumousis, V. K., and Georgiou, P. G. (1994). "Genetic algorithms in discrete optimization of steel truss roofs." *J. Comp. in Civ. Engrg.*, ASCE, 8(3), 309–325.

Lin, C. Y., and Hajela, P. (1992). "Genetic algorithms in optimization problems with discrete and integer design variables." *Engrg. Optimization*, 19(4), 309–327.

Michalewicz, Z. (1992). *Genetic algorithms + data structures = evolution programs*. Springer-Verlag, New York, N.Y.

Rajeev, S., and Kirshnamoorthy, C. S. (1992). "Discrete optimization of structures using genetic algorithms." *J. Struct. Engrg.*, ASCE, 118(5), 1233–1250.

Soh, C. K., and Yang, J. P. (1996). "Fuzzy controlled genetic algorithm search for shape optimization." *J. Comp. in Civ. Engrg.*, ASCE, 10(2), 143–150.

Thierens, D., and Goldberg, D. E. (1994). "Convergence models of genetic algorithms selection schemes." *Parallel problem solving from nature—PPSN III*, V. Davidor et al. eds., Springer-Verlag, Berlin, Germany, 119–129.

Whitley, D. (1989). "The GENITOR algorithm and selection pressure: why rank-based allocation of reproductive trials is best." *Proc., 3rd Int. Conf. on Genetic Algorithms*, Schaffer, ed., Morgan Kaufmann Publishers, Los Altos, Calif., 10–19.

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

Yang, J. P. (1996). "Development of genetic algorithm based approach for structural optimization," PhD thesis, Nanyang Technol. Univ., Singapore.

J. Comput. Civ. Eng. 1997.11:195-200.