Optimal Truss-Structure Design using Real-Coded Genetic Algorithms

Kalyanmoy Deb, Surendra Gulati,

Kanpur Genetic Algorithms Laboratory (KanGAL)

Department of Mechanical Engineering

Indian Institute of Technology

Kanpur, PIN 208 016, India

deb@iitk.ernet.in

Sekhar Chakrabarti

Department of Civil Engineering

Indian Institute of Technology

Kanpur, PIN 208 016, India

chakra@iitk.ernet.in

ABSTRACT

Optimization of truss-structures for finding optimal cross-sectional size, topology, and configuration of 2-D and 3-D trusses to achieve minimum weight is carried out using real-coded genetic algorithms (GAs). All the above three optimization techniques have been made possible by using a novel representation scheme. Although the proposed GA uses a fixed-length vector of design variables representing member areas and change in nodal coordinates, a simple member exclusion principle is introduced to obtain differing topologies. Moreover, practical considerations, such as inclusion of important nodes in the optimized structure is taken care of by using a concept of basic and non-basic nodes. Stress, deflection, and kinematic stability considerations are also handled using constraints. In a number of 2-D and 3-D trusses, the proposed technique finds intuitively optimal or near-optimal trusses, which are also found to have smaller weight than those that are reported in the literature.

1 Introduction

Optimal design of truss-structures has always been an active area of research in the field of search and optimization. Various techniques based on classical optimization methods have been developed to find optimal truss-structures (Dorn, Gomory, and Greenberg, 1964; Haug and Arora, 1989; Krish, 1989; Ringertz, 1985; Topping, 1983; Vanderplaat and Moses, 1972). However, most of these techniques can be classified into three main categories: (i) Sizing, (ii) Configuration, and (iii) Topology optimization.

In the sizing optimization of trusses, cross-sectional areas of members are considered as design variables and the coordinates of the nodes and connectivity among various members are considered to be fixed (Goldberg and Samtani, 1986). The resulting optimization problem is a nonlinear programming (NLP) problem. The sizing optimization problem is extended and made practically useful by restricting the member cross-sectional areas to take only certain pre-specified discrete values (Rajeev and Krishnamoorthy, 1992).

In the configuration optimization of trusses, the change in nodal coordinates are kept as design variables (Imai and Schmit, 1981). In most studies, simultaneous optimization of sizing and configuration has been used. The resulting problem is also a NLP problem with member area and change in nodal coordinates as variables.

In the topology optimization, the connectivity of members in a truss is to be determined (Krish, 1989; Ringertz, 1985). Classical optimization methods have not been used adequately in topology optimization, simply because they lack efficient ways to represent connectivity of members.

Although the above three optimization problems are discussed separately, the most efficient way to design truss-structures optimally is to consider all three optimization methods simultaneously. In most attempts, multi-level optimization methods have been used (Dobbs and Felton, 1969; Ringertz, 1985). In such a method, when topology optimization is performed, member areas and the truss configuration are assumed to be fixed. Once an optimized topology is found, the member areas and/or configuration of the obtained topology are optimized. It is obvious that such a multi-level optimization technique may not always provide the globally best design, since both these problems are not linearly separable.

Genetic algorithms (GAs) have also been in all the above three optimization problems. Goldberg and Samtani (1986) and Rajeev and Krishnamoorthy (1992) have used only size optimization, whereas Hejela, Lee, and Lin (1993) have used

a two-level optimization scheme of first finding multiple optimal topologies and then finding the optimal member areas for each of the truss topologies. Rajan (1995) has used all three optimization methods to only two 2-D truss design problems. Member connectivity is coded by using boolean variables (1 for presence and 0 for absence). Member areas and change in nodal displacements are used separately as variables. Since a separate binary string is used to denote presence and absence of all members, the representation scheme is highly sensitive to this binary string, thereby introducing artificial nonlinearity in the optimization problem.

In this paper, we have used a representation scheme which naturally allows all three optimizations to be used simultaneously. Moreover, the representation scheme also allows a canonical real-coded genetic algorithm (GA) to be used directly. In order to make the solutions practically useful, a concept of basic and non-basic nodes is introduced, which emphasizes important nodes to be present in the optimized solution. Stress, deflection, and kinematic stability considerations are also added as constraints to find functionally useful trusses. The proposed technique is applied to a number of truss-structure design problems. The optimized trusses are compared with that reported in the literature.

2   Proposed Methodology

In a truss-structure design, certain nodes are important and must exist in any feasible design. And certain nodes are added for load sharing and are optional. The important nodes are usually the ones which carry a load (a force) or which support the truss. These information are usually specified by the user (or designer) and must be honored in the design process of a truss-structure. These important nodes are named as the *basic* nodes, in this paper. On the other hand, the optional nodes are sometimes used in a truss to help distribute the stresses better on individual members. These nodes are named as *non-basic* nodes. Thus, the objective in a truss-structure design, (with all three design optimization methods) is to find which optional nodes are necessary in a truss, what coordinates of these optional nodes and which members must be present so that a goal (often, the weight of the truss) is optimized by satisfying certain constraints (often, the stresses in members and displacements of nodes).

The proposed algorithm assumes a *ground* structure, which is a complete truss with all possible member connections among all nodes (basic or non-basic) in the structure. Thus, in a truss having n nodes, there are a total of $m = \binom{n}{2}$ different members possible. A ground structure[1] is a collection of all these members.

A truss in the proposed GA is represented by specifying a cross-sectional area for each member in the ground structure. Thus, a solution represented in the GA population is a vector of m real numbers within two specified limits. Although every solution in a GA population will have m cross-

sectional areas, its phenotype (the truss itself) may not have all m members. The presence or absence of a member in the ground structure is determined by comparing the cross-sectional area of the member with a user-defined small critical cross-sectional area, ϵ. If an area is smaller than ϵ, that member is assumed to be absent in the realized truss. This is how trusses with differing topologies can be obtained with a fixed-length representation of the truss member areas. This representation scheme has another advantage. Since member areas are directly used, the values higher than ϵ specify the actual member cross-sectional area. It is interesting to note that the critical area ϵ and lower and upper bounds on the cross-sectional areas must be so selected that, although working in the range $(\boldsymbol{A}^{\min}, \boldsymbol{A}^{\max})$, there is an adequate probability of making a unwanted member absent in a solution. We have chosen $\boldsymbol{A}^{\min} = -\boldsymbol{A}^{\max}$ and a small positive value for ϵ, so that there is equal probability of any member being present or absent in a truss. If a member is absolutely essential in a truss, the genetic operators quickly make the corresponding member area in all solutions in a population positive. This reduces the chance of making the member absent in children trusses.

In subsequent discussions, we denote m as the number of members present in a realized truss and not the total number of members in the ground structure, for clarity.

With the above discussion, we now present the formulation of the truss-structure optimization problem as a nonlinear programming (NLP) problem:

$$
\begin{aligned}
\text{Minimize} \quad & f(\boldsymbol{A}) = \sum_{j=1}^{m} \rho_j \ell_j \boldsymbol{A}_j \\
\text{Subject to} \quad & G1 \equiv \text{Truss is acceptable to the user} \\
& G2 \equiv \text{Truss is kinematically stable} \\
& G3 \equiv S_j - \sigma_j(\boldsymbol{A}, \boldsymbol{\xi}) \geq 0 \qquad j = 1, 2, \ldots, m \\
& G4 \equiv \delta_k^{\max} - \delta_k(\boldsymbol{A}, \boldsymbol{\xi}) \geq 0 \quad k = 1, 2, \ldots, n \\
& G5 \equiv \boldsymbol{A}_i^{\min} \leq \boldsymbol{A}_i \leq \boldsymbol{A}_i^{\max} \quad i = 1, 2, \ldots, m \\
& G6 \equiv \boldsymbol{\xi}_i^{\min} \leq \boldsymbol{\xi}_i \leq \boldsymbol{\xi}_i^{\max} \qquad i = 1, 2, \ldots, n'
\end{aligned}
\tag{1}
$$

In the above NLP problem, the design variables are the cross-sectional areas of members present in a truss (denoted as $\boldsymbol{A}$) and the coordinates of all n' non-basic nodes (denoted as $\boldsymbol{\xi}$). The parameters S_j and $\delta_k^{\max}$ are the allowable strength of the j-th member and the allowable deflection of the k-th node, respectively. We describe each of the above terms in the following:

Objective function: In this paper, we have considered the weight of the overall truss as the objective function, whereas other criteria such as reliability and dynamic characteristics can also be considered. The parameter ρ_j and ℓ_j are the material density and length of j-th member, respectively.

Constraint G1: The user specifies the location and the number of basic nodes for supports and loads. Thus, a feasi-

---

[1]It is absolutely not necessary to have all $\binom{n}{2}$ members in a ground structure. Problem knowledge can be used to discard some members in the ground structure. We shall show later how a good choice of a ground structure aids in finding better trusses using a GA.

ble truss must have all the basic nodes. This constraint is checked first. If any one of the basic nodes is absent in the truss, a large constant penalty is assigned to the solution and no further calculation of objective function or constraints is done.

Constraints G2: Since trusses with different topologies are created by genetic operators, trusses which are not kinematically stable can also be generated. Trusses must be kinematically stable so that it does not generate into a mechanism. One of the ways to check the kinematic stability of a truss is to check the positive-definiteness of the stiffness matrix created from the member connectivities. If the matrix is positive-definite, the truss is kinematically stable. However, the computation of the positive-definiteness of a matrix of a reasonable size is enormous. We reduce the frequency of such computations by first checking the Grubler's criterion (Ghosh and Mallik, 1988):

$$\text{Degree-of-freedom (D-O-F)} = 2n - m - n_l, \quad (2)$$

where n_l is the number of degrees-of-freedom lost at the support nodes. If the D-O-F is non-positive, the corresponding truss is a not a mechanism. Since a truss has to be a non-mechanism, we first check the above simple criterion. If the truss is a mechanism, we penalize the solution by assigning a large value which is proportional to the D-O-F obtained by the above equation. Thereafter, the corresponding truss is not sent to FEM routine for further calculations such as stiffness matrix, stresses and displacements. If the truss is not a mechanism, we then sent the truss to the FEM routine and check the positive-definiteness of the stiffness matrix. If the matrix is not positive-definite, a large penalty proportional to the violation of positive-definiteness is assigned to the solution and no further calculation of stress or deflection is made.

Constraints G3: In a feasible truss, all members must have stresses within the allowable strength of the material. Since, usually a truss is subjected to a number of different loading conditions applied separately, these constraints must be used for each loading condition. Since the trusses of various topologies are created on the fly, some of them may be statically determinate and some of them may be statically indeterminate. Thus, we have used a finite element method (FEM) to calculate the stresses and deflection in a truss. It is also noteworthy that since each truss is different in its topology, the members and nodes of the truss is needed to be automatically numbered before calling a FEM routine. FEM procedure is developed for 2D as well as 3D trusses. A suitable automatic node numbering scheme is also developed.

In order to have significant effect of all constraints, we normalize all constraints shown above in the following manner so that all constraint violations get equal importance:

$$G1 \equiv \frac{S_j}{\sigma_j(\boldsymbol{A}, \boldsymbol{\xi})} - 1 \geq 0. \quad (3)$$

In the case of any constraint violation (that is, if $S_j \leq \sigma_j(\boldsymbol{A}, \boldsymbol{\xi})$), a bracket-operator penalty term (Deb, 1995; Rao, 1984) is added to the objective function.

Constraints G4: Like in Constraints G1, all nodes (basic or non-basic) in the truss must not deflect more than the allowable limit due to the application of loads. Like Constraints G1, these constraints are also normalized and the constraint violation (if any) is added to the objective function by using a bracket-operator penalty term.

Constraints G5 and G6: Since real-coded GA allows the variables to be bounded within specified limits, these constraints will be automatically satisfied.

The fitness of a solution is dependent on the constraint violations and thus calculated as follows:

$$F(\boldsymbol{A}, \boldsymbol{\xi}) = \begin{cases} 10^9, & \text{if G1 is violated,} \\ 10^8(\text{constraint violation}), \\ \quad \text{if G2 is violated with D-O-F constraint,} \\ 10^7(\text{constraint violation}), \\ \quad \text{if G2 is violated with} \\ \quad \text{positive-definiteness constraint,} \\ f(\boldsymbol{A}, \boldsymbol{\xi}) + 10^5 \sum_{j=1}^{m} |\langle G3_j \rangle| \\ \quad + 10^5 \sum_{k=1}^{n} |\langle G4_k \rangle| \\ \quad \text{otherwise.} \end{cases}$$

$$(4)$$

In the above expression, the operator $\langle \ \rangle$ is the bracket-operator penalty term.

In the real-coded GA, the binary tournament selection without replacement is used. Since a truss is represented with the cross-sectional values directly, we use the simulated binary crossover (SBX) operator (Deb and Agrawal, 1995) and a real-valued mutation operator (Deb, 1997), both of which work on floating-point values. We terminate a GA simulation when a pre-specified number of generations is elapsed.

3 Results

In all simulations presented in this section, we have used a crossover probability of 0.9 and a mutation probability of 0.1. The population size used in a simulation is dependent on the number of members in the ground structure. Since an initial random population is always used, it is expected that the required population size would depend on the problem complexity. It is intuitive that in truss-structure design problems, as the number of members in the ground structure increases there exist many different topologies with almost equal overall weight. This suggests that with the increase in members in the ground structure, the resulting NLP problem becomes multi-modal and hence a large population size is necessary to find optimal or near-optimal solutions.

In the following, we discuss the performance of GAs in solving various 2-D and 3-D truss structures and compare the obtained solutions with the best solutions available in the literature.

3.1 11-Member, Six-Node Truss

The 11-member, six-node ground structure is shown in Figure 1. In all figures showing trusses, the dimensions are in inches. Topology and size optimization are carried out for this truss using 11 variables corresponding to members areas. Following design parameters are used in the problem:

Young's modulus	=	10^4 ksi
Density (ρ)	=	0.1 lb/in^3
Allowable compressive strength	=	25 ksi
Allowable tensile strength	=	25 ksi
Allowable displacement	=	2 in
$A^{\min}, A^{\max}$	=	$-35.0, 35.0$ in^2
Critical area (ϵ)	=	0.09 in^2

After 225 generations, the best truss-structure obtained using GAs with a population size of 220 has an overall weight of 4,899.15 lbs. The optimized truss satisfies all constraints and is shown in Figure 2. Although there were 11 members and 6 nodes in the ground structure, the GA is able to find a structure with only 6 members and 5 nodes. For the optimized truss, the deflection of the node furthest away from the supports (intersection of members 2 and 5) in x and y directions are $-.562$ and 2.000 inches, respectively. Since the deflection in y direction is equal to the maximum allowable deflection, it can be argued that the obtained truss is either optimal or near-optimal. This is because any further reduction in cross-sectional area in any member will make the truss weak and will result in a deflection at the critical node more than the allowable limit.

Figure 3 shows that the best solution in the initial population had a weight over 9,000 lbs and was feasible. The figure also shows how the GA with a population size of 110 finds a truss of weight 4,950.75 lbs, which is a little higher than that obtained using a population size of 220.
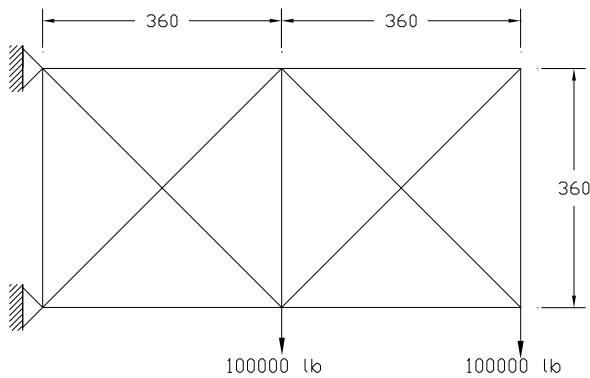
The member areas obtained using the proposed GA is compared with the best-known solution available (Ringertz, 1985), which used a multi-level linear and nonlinear programming method, where the same topology with 6 members and 5 nodes was obtained. Table 1 shows that our GA is able to obtain a truss with slightly smaller weight than that reported in Ringertz (1985). It is also interesting to note that although both weights are very similar, the combination of member areas in both trusses is a little different. Comparing to another GA implementation on the same problem (but with discrete member areas) (Rajan, 1995) which found a truss with a weight of 4,962.1 lbs, our solution is much better.

Next, we make the member areas to take only discrete values (in the step of 1 in^2). In this case, we use the discrete version of the SBX and mutation operators (Deb, 1997), thereby allowing only discrete values to be created using crossover and mutation operators. The optimized truss obtained using GA has the same topology as in Figure 2, but now has an overall weight of 4,912.85 lbs. The best known solution in the literature for the discrete case has the same topology and has a weight of 4,942.70 lbs (Hajela, Lee, and Lin, 1993), which is about 30 lbs more than that obtained by our algorithm. The corresponding member areas are presented in Table 1. Since our GAs have found a truss which requires 1 in^2 less area in both members 4 and 5 (which are the largest members in the truss) compared to that in Hajela, Lee, and Lin (1993), the overall weight is smaller. However, to make the truss safer from stress considerations, the cross-sectional area in member 0 had to be increased by 2 in^2.

3.2 Two-Tier Truss

Two-tier, 39-member, 12-node ground structure (Figure 4) is used for the following optimization studies:

1. Sizing and topology optimization

2. Sizing, shape and topology optimization

The overlapping members are shown with a small gap in the figure for clarity. Symmetry about middle vertical member is
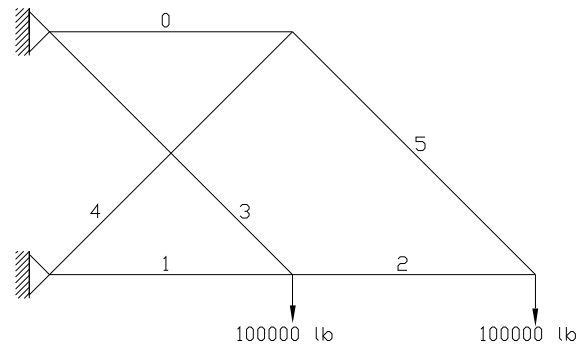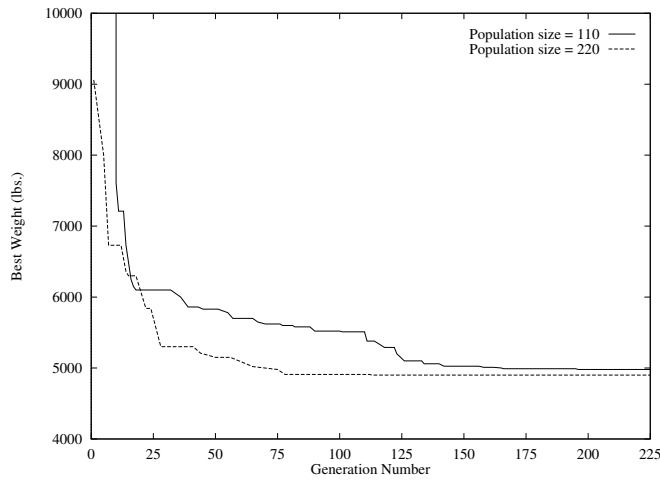

Figure 1   The 11-member, six-node ground structure.

Figure 2   Optimized truss obtained from 11-member, six-node ground structure.

Figure 3   The improvement in the best solution versus generation number.

Table 1   Member areas of the optimized truss for 2-D, 11-member, six-node ground structure.

| Member number (refer Figure 2) | Area of members (in^2) | | | |
| | Continuous Areas | | Discrete Areas | |
	Proposed	Ringertz (1985)	Proposed	Hajela, Lee, and Lin (1993)
0	29.68	30.10	30	28
1	22.07	22.00	24	24
2	15.30	15.00	16	16
3	06.09	06.08	6	6
4	21.44	21.30	20	21
5	21.29	21.30	21	22
Weight of Truss (lb)	4899.15	4900.00	4912.85	4942.7

assumed, thereby reducing the number of variables to 21. The material properties and maximum allowable deflection are the same as in the previous problem, except allowable strength is 25 ksi. Lower and upper bounds of member areas of -2.25 to 2.25 in^2 are used and a critical area of 0.05 in^2 is chosen.

3.2.1   Sizing and Topology Optimization

Simultaneous optimization of sizing and topology is carried out taking 21 continuous variables corresponding to 39 members after considering symmetry. Optimized topology corresponding to a simulation run with population size of 630 is shown in Figure 5 and member areas are listed in Table 2. Of the 39 members and 12 nodes in the ground structure, only 17 members and 10 nodes are retained by our GA. Starting from a weight of 570 lbs found in the best solution in the initial population, the GA has found a truss with a weight of 198 lbs. Since no study of this 2-tier truss is available in the literature, we cannot compare our solution with any other method. Nevertheless, Figure 5 shows that optimized truss is intuitively a much better truss than the ground structure. Moreover, all the

critical members carrying large loads have utilized the material maximally so that the stress developed in each of them is almost equal to the allowable strength of material (20 ksi), as shown in below:

Member: (2,3)   (8,9)   (10,11)   (12,13)   (16)

Stress (ksi):   19.987   19.972   20.000   19.980   19,960

This suggests that the obtained truss is a near-optimal solution.

3.2.2   Sizing, Topology and Shape Optimization

In simultaneous optimization of sizing, configuration, and topology, cross-sectional area of each member, number of members in the truss, and coordinates of the non-basic nodes (nodes that do not carry a load and nodes that are not support nodes) are kept as decision variables. Seven extra nodal displacement variables, in addition to 21 member area variables discussed previously, are considered here. Nodal displacement of 7 non-basic nodes in x and y directions with respect to their original coordinates in the ground structure are de-
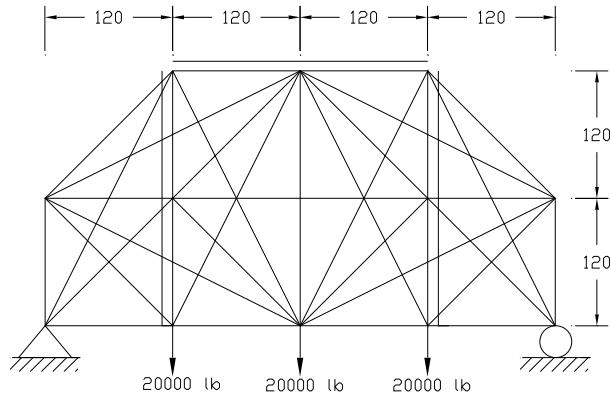
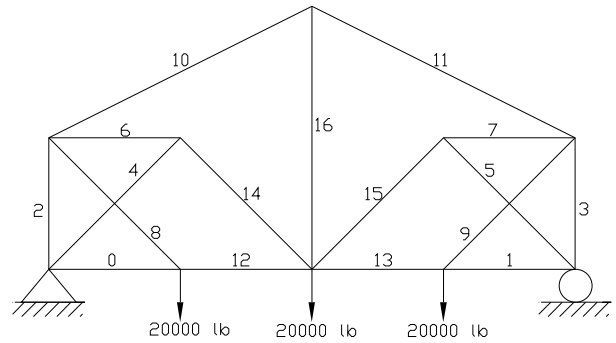**Figure 4**   **Two-tier, 39-member, 12-node ground structure.**

Figure 5   **Optimized truss for two-tier, 39-member, 12-node ground structure for sizing and topology consideration.**

Table 2   **Member areas for the optimized truss structure in the case of sizing and topology optimization.**

Member	0,1	2,3	4,5	6,7	8,9	10,11	12,13	14,15	16
Area (in^2)	0.050	1.501	0.052	0.050	1.416	1.118	1.001	0.050	1.002

noted as variables. Using symmetry about the vertical member at the center of the trusses, we reduce the number of these nodal displacement variables to 7 (the top-most node at the center is assumed to have a fixed x coordinate). These extra variables are assumed to vary within $(-120, 120)$ inch.

The optimized nodal configuration and topology corresponding to a GA run with 1,680 population size (up to 300 generations) is shown in Figure 6, and optimized member areas are listed in Table 3. This truss requiring only 15 members and 9 nodes has a weight of 192.19 lbs, which is 3% smaller than that obtained using only sizing and topology optimization. The shape of the truss is also different from that

obtained in the previous subsection.

3.3   3-D, 25-member, 10-Node Truss

The ground structure for 10-node, 3-D truss which consists of 25 members, taken from the literature (Haug and Arora, 1989) is shown in Figure 7. Young's modulus and density of the material are the same as before. However, an allowable tensile and compressive strength of 40 ksi is used. An allowable deflection of 0.35 inch is used. The lower and upper bounds on the member areas are assumed to be -3 to 3 in^2 and a critical area of 0.005 in^2 is chosen.

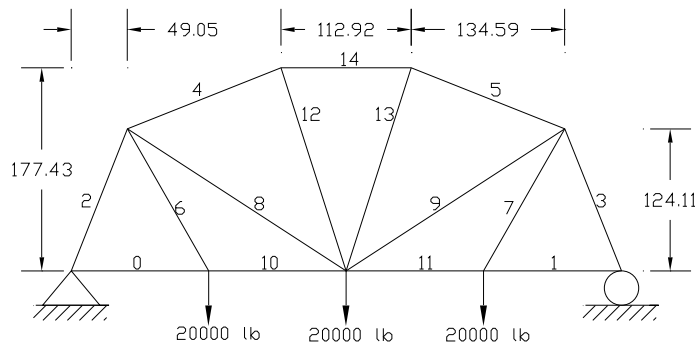Members are grouped considering the symmetry on opposite sides and cross-members to be symmetric on all the sides,
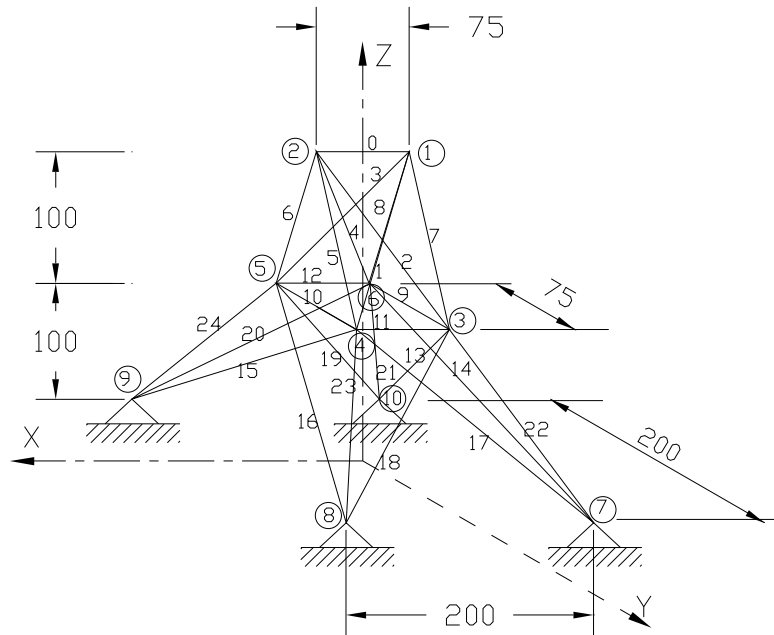

Figure 6   **Optimized truss for two-tier, 39-member, 12-node ground structure for sizing, topology, and configuration consideration.**

Table 3 Member areas for the optimized truss structure in the case of sizing, topology, and configuration optimization.

Member	0,1	2,3	4,5	6,7	8,9	10,11	12,13	14
Area (in^2)	0.595	1.615	1.293	1.155	0.051	1.166	0.504	1.358

Figure 7   3-D, 25-member, 10-node ground structure.

thus reducing the number of variables to 7. This grouping is done in the following way (refer Figure 7):

Group:	0	1	2	3
Member:	(0)	(1,2,3,4)	(5,6,7,8)	(9,10,11,12)

Group:	4	5	6
Member:	(13,14,15,16)	(17,18,19,20)	(21,22,23,24)

This truss is optimized for two separate loading conditions. In loading case 1, a force vector $(0; 20,000; -5,000)$ lbs is applied on node 1 and a force vector $(0; -20,000; -5,000)$ lbs is applied on node 2. In loading case 2, four force vectors are applied: $(1,000; 10,000; -5,000)$ on node 1, $(0; 10,000; -5,000)$ on node 2, and $(500; 0; 0)$ on nodes 3 and 6. The same loading cases were also used in Haug and Arora (1989). However, they performed only the size optimization on the fixed ground structure, a much simpler optimization problem than ours. Since, we have used both size and topology optimization, we cannot compare our results with that study.

The optimized truss topology obtained using our GA with a population size 280 has a weight of 544.852 lbs. Out of seven groups of member areas used in the ground structure,

the optimized truss has only five groups (Groups 0 and 3, totalling five members have been deleted). Member areas corresponding to this solution are listed in Table 4. This truss is also optimal or near-optimal, because the deflection of top two nodes (1 and 2) is equal to the maximum allowable deflection (0.35 inch), which suggests that the truss has adjusted its member areas and member connectivity in such a way which makes the weight of the truss minimum by allowing the deflection to reach the allowable limit.

4   Conclusions

In this paper, we have developed a GA-based optimization procedure for designing 2-D and 3-D truss structures. Nodes in a truss are classified into two categories: (i) Basic nodes, which are used to support the truss or to apply a load, (ii) Non-basic nodes, which do not support the truss nor they bear any load. The concept of basic and non-basic is introduced to emphasize creation of user-satisfactory trusses and also to reduce the computational time by not performing expensive FEM analysis for unsatisfactory trusses. The trusses of varying topology (connectivity among members) is obtained with a fixed-length vector representation of member areas and with an implicit exclusion of small area members. This way any

Table 4   Member areas for the optimized truss structure.

Group	0	1	2	3	4	5	6
Member area (in^2)	–	2.037	2.969	–	0.699	1.644	2.658

member having an area smaller than a critical area is considered to be absent in the corresponding solution. This representation scheme allows conventional GA operators to be used directly. Moreover, since the member areas are used as variables, simultaneous sizing as well as topology optimization are achieved. In the case of simultaneous application of all three optimization methods with sizing, topology, and configuration considerations, additional variables corresponding to change in nodal coordinates have been added.

In a number of different truss-structure problems ranging from 2-D, 6-node trusses to two-tier, 39-member truss to 3-D 25-member trusses, the proposed algorithm has been able to find trusses which are better than those reported in the literature and which utilizes material properties or deflection limits optimally. These results suggest the use of the proposed technique in other truss-structure design problems, where a complete optimization with optimal sizing, topology, and configuration is desired.

Acknowledgement

References

Deb, K. (1997).   Mechanical component design using genetic algorithms.   *Evolutionary algorithms in engineering applications* (Eds.   Dipankar Dasgupta and Z. Michalewicz) New York: Springer-Verlag (pp. 495–512).

Deb, K. (1995). *Optimization for engineering design: Algorithms and examples.* New Delhi: Prentice-Hall.

Deb, K. and Agrawal, R. B. (1995).   Simulated binary crossover for continuous search space. *Complex Systems, 9* 115–148.

Dobbs, M. W. and Felton, L. P. (1969).   Optimization of truss geometry. *ASCE Journal of Structural Division, 95*, (ST10), 2105–2118.

Dorn, W. S., Gomory, R. E., and Greenberg, H. J. (1964).   Automatic design of optimal structures.   *Journal de Mecanique, 3* (6). 25–52.

Ghosh, A. and Mallik, A. K. (1988). *Theory of mechanisms and machines*. New Delhi: Affiliated East-West Press.

Goldberg, D. E. and Samtani, M. P. (1986). Engineering optimization via genetic algorithms, *Proceedings of the Ninth Conference on Electronic Computations*, ASCE, pp. 471–482.

Hajela, P., Lee, E., and Lin, C. Y. (1993). Genetic algorithms in structural topology optimization. *Topology Optimization of Structures*, pp. 117–133.

Haug, E. J. and Arora J. S. (1989). *Introduction to optimal design.* New York: McGraw Hill.

Imai, K. and Schmith, L. A. (1981). Configuration optimization of trusses. *Journal of Structural Division, ASCE, 107*, (ST5). 745–756.

Krish, U. (1989).   Optimal topologies of truss structures. *Computer Methods in Applied Mechanics and Engineering, 72*, 15–28.

Rajan, S. D. (1995). Sizing, shape and topology optimization of trusses using genetic algorithm, *Journal of Structural Engineering, 121*, (10), 1480–1487.

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

Rao, S. S. (1984). *Optimization: Theory and applications*, New York: Wiley.

Ringertz, U. T. (1985). On topology optimization of trusses, *Engineering optimization, 9*, 209–218.

Topping, B. H. V. (1983). Shape optimization of skeletal structures: A review. *Journal of Structural Engineering, 11*, 1933–1951.

Vanderplaat, G. N. and Moses, F. (1972). Automated design of Trusses for Optimal Geometry, *Journal of Structural Division*, (ST3), 671–690.