Structural Optimization 14, 24–35 © Springer-Verlag 1997

Research Papers

Optimal design of steel structures using standard sections

M.-W. Huang and J.S. Arora

Optimal Design Laboratory, College of Engineering, The University of Iowa, Iowa City, IA 52242, USA

Abstract The problem of optimal structural design having linked discrete variables is addressed. For such applications, when a discrete value for a variable is selected, values for other variables linked to it must also be selected from a table. The design of steel structures using available sections is a major application area of such problems. Three strategies that combine a continuous variable optimization method with a genetic algorithm, simulated annealing, and branch and bound method are presented and implemented into a computer program for their numerical evaluation. Three structural design problems are solved to study the performance of the proposed methods. CPU times for solution of the problems with discrete variables are large. Strategies are suggested to reduce these times.

1 Introduction

In many structural engineering applications, optimization problems have design variables that can assume only some predetermined values. Such variables are called discrete. When some of the variables are discrete and others are continuous, we have the so-called mixed variable nonlinear programming problem. Several optimization methods have been developed and tested to deal with such problems since the 1960s. A detailed review of the methods has been presented by Arora *et al.* (1994). Some recent developments in this area can also be found in the work of Huang (1995) and Huang and Arora (1995, 1997).

The design of structural steel frames using the AISC (American Institute of Steel Construction) standard sections is an example of discrete variable optimization problems. However, this represents a special class of discrete variable problems in which the section properties (e.g. section area, moment of inertia, section dimensions, etc.) are not independent of each other. Once a value for one design property is specified, each of the remaining properties must also be assigned a unique value. These properties are linked to each other via a table of commercially available sections, such as Table 1 of AISC sections (note that the table lists only a few of the standard sections; the manual includes several hundred sections). For example, the data in the first row of Table 1 must be used when the standard section W36×300 is selected during the optimization process. A gradient-based optimization method may not be appropriate for such applications because relationships among the properties cannot be expressed analytically. If each property is treated as an independent design variable, the final solution would generally be unacceptable since the properties would have values that cannot coexist. It is also not possible to use one of the properties as the only design variable because other properties cannot be calculated analytically using just that property.

There are other engineering applications where linked discrete variables occur, such as engine types, electric motor types, bolt type, gear type, crank shaft type, etc. A detailed review of the approaches that have been used for this important class of applications has been presented recently by Arora and Huang (1996). An overview of that work is presented later in this paper.

Purpose of this research is to propose, develop, implement and evaluate methods for structural optimization problems having linked discrete design variables (design properties). Three strategies are proposed that combine the use of a continuous variable optimization method with a branch and bound method, simulated annealing or genetic algorithm, to create a solution process.

Section 2 contains the problem formulation and AISC requirements. The linked discrete variables are defined in several ways for different strategies. Cost and constraint functions are also described. Section 3 contains an overview of the approaches that have been used recently for problems with linked discrete variables. Section 4 presents three new approaches based on different definitions of design variables and the optimization algorithms used. Section 5 describes the structural design problems used to test the implementations. Section 6 contains numerical results with different strategies and their comparative evaluation. Finally, Section 7 contains some concluding remarks.

2 Problem formulation and AISC specification

The design of steel structures is formulated as a nonlinear programming problem in this section. For this problem, each member must be chosen to have the shape and dimensions that are available in the AISC tables. The design variables for the structure are identified, a cost function is defined based on minimizing the weight of the structure, and the constraints are explained.

2.1 Design variables

The AISC manual contains tables for available members of various shapes. In this study, we use only the open sections such as I-sections (including W-shapes, M-shapes, S-shapes and HP-shapes). The test problems described later in Section 5 are solved using the 187 W-shape members. Depending on the optimization strategies used (described later), the design variables can be defined in one of the following three ways.

Table 1. Some AISC standard sections (out of 187 W-shape sections)

	A	d	t_w	b	t_f	I_x	S_x	r_x	I_y	S_y	r_y
W36×300	88.30	36.74	0.945	16.665	1.680	20300	1110	15.20	1300	156	3.830
W36×280	82.40	36.52	0.885	16.595	1.570	18900	1030	15.10	1200	144	3.810
W36×260	76.50	36.26	0.840	16.550	1.440	17300	953	15.00	1090	132	3.780
W36×245	72.10	36.08	0.800	16.510	1.350	16100	895	15.00	1010	123	3.750
W36×230	67.60	35.90	0.760	16.470	1.260	15000	837	14.90	940	114	3.730
W36×210	61.80	36.69	0.830	12.180	1.360	13200	719	14.60	411	67.5	2.580
W36×194	57.00	36.49	0.765	12.115	1.260	12100	664	14.60	375	61.9	2.560

A: cross-sectional area (in^2), d: depth (in), t_w: web thickness (in), b: flange width (in), t_f: flange thickness (in), I_x: moment of inertia about the x-x axis (in^4), S_x: elastic section modulus about the x-x axis (in^3), r_x: radius of gyration with respect to the x-x axis (in), I_y: moment of inertia about the y-y axis (in^4), S_y: elastic section modulus about the y-y axis (in^3), r_y: radius of gyration with respect to the y-y axis (in)

2.2 Cross-section type 1

Four design variables for each section. They are the depth d, flange width b, web thickness t_{w}, and the flange thickness t_{f}. Figure 1 shows this type of a cross-section and associated design variables. The total number of design variables is 4NG, where NG is the total number of member groups in the structure (a group is a collection of members having the same cross-sectional and material properties). The design variables for the entire structure are

$$\mathbf{x} = (\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_{\mathrm{NG}})^T = \left[(d, b, t_{\mathrm{w}}, t_{\mathrm{f}})_1, (d, b, t_{\mathrm{w}}, t_{\mathrm{f}})_2, \ldots, (d, b, t_{\mathrm{w}}, t_{\mathrm{f}})_{\mathrm{NG}}\right]^T . \quad (1)$$

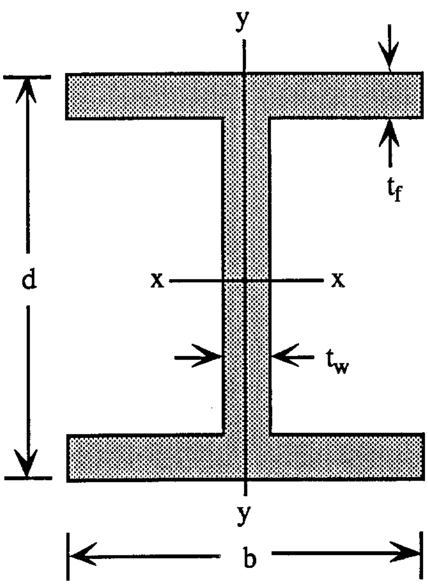

Fig. 1. Cross-section of a steel member

2.3 Cross-section type 2

One design variable for each member. In the present formulation, only the moment of inertia about the x-axis (I_x) is used as the sole design variable for each group (other section properties such as the section area can also be used as the primary variable), i.e.

$$\mathbf{x} = (x_1, x_2, \ldots, x_{\mathrm{NG}})^T = (Ix_1, Ix_2, \ldots, Ix_{\mathrm{NG}}) . \quad (2)$$

For this type of design variables, other section properties must be related to I_x somehow.

2.3.1 Cross-section type 3

One design variable for each member. Only the AISC section number N is used as the sole design variable for each group, i.e.

$$\mathbf{x} = (x_1, x_2, \ldots, x_{\mathrm{NG}})^T = (N_1, N_2, \ldots, N_{\mathrm{NG}})^T . \quad (3)$$

When a section number for a member is specified, all the section properties can be obtained from the table of available sections.

2.4 Cost function

The cost function which is to be minimized is taken as the weight of the structure (in kips) and is expressed as

$$W(\mathbf{x}) = \sum_{i=1}^{\mathrm{NG}} \rho_i A_i L_i , \quad (4)$$

where the subscript i denotes the group number, ρ_i is the material weight density (kips/in^3), A_i is the cross-sectional area (in^2), and L_i is the sum of the lengths (in) of all members in the i-th group. Other costs, such as fabrication, life-cycle, maintenance and operations costs, can be added to the cost function, if they are known and can be expressed in terms of the design variables.

2.5 Constraints

At any design point, the structure must be analyzed to evaluate the constraints, such as the stress constraints (i.e. axial, bending, shear, and combined action stresses), constraints to prevent local buckling, constraints on overall member buckling, displacement constraints, and explicit bounds on the design variables. They are explained in the following paragraphs. These constraints are usually implicit functions of the design variables because explicit expressions for them cannot be written in terms of them.

2.5.1 Stress constraints

The stress constraints are based on the AISC specifications which state that the steel members subjected to compression, bending, and shear forces should satisfy

$$\frac{f_a}{F_a} + \frac{C_m f_b}{(1 - f_a/F_e) F_b} \le 1 , \quad (5)$$

$$\frac{f_a}{0.6 F_y} + \frac{f_b}{F_b} \le 1 , \quad (6)$$

$$\frac{f_s}{F_s} \le 1 . \quad (7)$$

When $\frac{f_a}{F_a} \leq 0.15$, the following equation may be used in lieu of (5) and (6):

$$\frac{f_a}{F_a} + \frac{f_b}{F_b} \leq 1\,, \tag{8}$$

where f_a, f_b, and f_s are the calculated axial, bending, and shear stresses respectively, and F_a, F_b, and F_s correspond to those given in the AISC manual (kips/in^2); F_y is the yield stress (kips/in^2), F_e' is the Euler stress (kips/in^2) divided by a factor of safety and C_m is a coefficient whose value is dependent on the type of joint. AISC recommends using $C_m = 0.85$ for compression members in frames having joint translation. For truss members, $f_b = f_s = 0$.

2.5.2 Displacement constraints

Limits are imposed on displacements of the joints as

$$|\delta_i| \leq \delta_{i\mathrm{U}}\,, \quad \delta_{i\mathrm{U}} > 0\,, \tag{9}$$

where δ_i is the calculated displacement (inch) of the i-th degree of freedom and $\delta_{i\mathrm{U}}$ is its upper limit.

2.5.3 Constraints to prevent local buckling

The AISC manual also specifies some requirements on the dimension and length of a member. The laterally unsupported length L of the compression flange of members must satisfy the following two constraints:

$$L \leq \frac{20000}{F_y}\frac{wt_f}{h}\,, \tag{10}$$

$$L \leq \frac{76w}{\sqrt{F_y}}\,. \tag{11}$$

The dimensions of the cross-section must also satisfy

$$\frac{w}{2t_f} \leq \frac{95}{\sqrt{F_y}}\,, \tag{12}$$

$$\frac{h-2t_f}{t_w} \leq \frac{14000}{\sqrt{F_y(F_y+16.5)}}\,, \tag{13}$$

where F_y has units of ksi in (10) to (13). Note that these four constraints cannot be imposed when only the moment of inertia is used as the sole design variable (cross-section type 2).

2.5.4 Design variable constraints

Upper and lower bound constraints are imposed on all design variables as $x_{i\mathrm{L}} \leq x_i \leq x_{i\mathrm{U}}$, $i = 1,\dots,n$, where n is the number of design variables, and $x_{i\mathrm{L}}$ and $x_{i\mathrm{U}}$ are, respectively, lower and upper bounds for the i-th variable.

2.6 Standard steel sections

The section properties of the AISC standard steel sections are stored in a file and accessed by the optimization program when searching for a discrete solution. Table 1 shows a description for some of the AISC sections. The table is augmented with the values for α, β, and γ for the sections (to be used in the solution process that is explained later). These are calculated as follows:

$$\alpha = \frac{A}{\sqrt{I_x}}\,, \quad \beta = \frac{S_x}{(I_x)^{0.75}}\,, \quad \gamma = \frac{r_y}{(I_x)^{0.25}}\,, \tag{14}$$

where S_x and r_y are the section modulus and the radius of gyration, respectively.

3 Overview of the literature

A detailed review of literature for structural optimization problems with linked discrete variables has been recently presented by Arora and Huang (1996). In this section an overview of that literature is presented.

Interest in discrete variable structural optimization goes back to 1960s when Toakley (1968) applied integer linear programming (ILP) methods for optimal design of plastic and elastic structures. The design of plastic structures was formulated as an LP problem which was transformed to an integer variable problem. The elastic design of determinate trusses subjected to displacement constraints was also formulated as a mixed integer-continuous variable LP problem. The algorithms used were: Gomory's cutting plane method (Gomory 1960), branch and bound method (implicit enumeration) and heuristic techniques. Reinschmidt (1971) also formulated plastic design of building frames as an ILP problem and solved it using Geoffrion's (1967) branch and bound method. Elastic design of trusses subjected to stress and displacement constraints was formulated as a nonlinear programming (NLP) problem which was linearized and solved as a sequence of linear IP problems. The same problem was solved directly without linearization or introduction of integer variables using the branch and bound method by Cella and Logcher (1971). A filtered pattern search was used during the branching phase of the algorithm where each trial design was analysed using the approximate reanalysis approach to reduce the computational effort. A direct method combining Box's algorithm and Hooke and Jeeve's method was used by Lai and Auchenbach (1973). Liebman *et al.* (1981) transformed the problem to a sequence of unconstrained problems that were solved using an integer discrete gradient algorithm. An enumeration algorithm for discrete variable optimization of trusses with stress and displacement constraints was developed by Hua (1983). The method exploits the structure of the problem to develop heuristics that reduce the size of enumeration.

More recently, two phase approaches have been used to obtain a discrete solution for the problem with linked discrete variables. In the first phase, the problem is formulated and solved as a standard nonlinear program. In the second phase, the continuous solution is used as a starting point and a discrete variable optimization method is used to obtain the final solution. The solution approaches have been divided into four broad categories based on how the problem is formulated and solved (Arora and Huang 1996). The first two approaches use only one design variable for each member of the structure, the third one uses mixed single and multiple design variable formulations, and the fourth one uses rounding-off procedures for the final solution.

3.1 Single variable approaches with approximations

In this approach, one of the section properties (e.g. area or moment of inertia) is treated as a continuous design variable for each member (Type 2) and other properties are approximately linked to it by a curve fitting procedure. With this formulation, gradient-based optimization methods can be used to solve the problem. Main drawbacks of the approach are: (i) all the design code constraints cannot be imposed during the optimization process, (ii) the approximate representation of section properties is not accurate for all the available sections, and (iii) when new section become available, the approximate relationships between the properties must be rederived. One may get around this dilemma to some extent by dividing the available sections into smaller groups. Each group would contain only the sections for which more accurate relationships can be developed for various properties.

Based on the foregoing ideas, Grierson and coworkers have extensively developed and demonstrated procedures for discrete variable optimization. For example, Grierson and Lee (1984) formulated the problem of optimal design of 2D frameworks using the section area as the only design variable. Database of available sections was extensively analyzed and divided into several data sets. The optimization process consisted of two phases. Phase I used 3 iterations of a continuous variable optimality criterion method to obtain a good starting point for Phase II. The available sections were not assigned in this phase. In Phase II, a discrete solution was obtained using a specialized discrete variable algorithm. Grierson and Lee (1986) extended their previous work on optimization of steel frames to include constraints under ultimate load conditions. Grierson and Cameron (1989) described features of a computer program required for practical applications in design of steel frameworks. A two phase procedure was demonstrated for a 2D mill crane building framework. Cameron *et al.* (1991) extended previous work for discrete variable optimization of 2D frameworks to design of 3D frameworks. Chan (1992) and Chan *et al.* (1995) developed a procedure for optimal design of tall building steel frameworks using available sections. In the procedure, the drift constraint, member strength constraints and interstorey drift constraints were imposed in a sequential manner by optimizing separate problems. The last phase of the solution process involved final discrete member specification. For the continuous solution, penalty on the structural weight for each member to be specified a higher available section was calculated. A few members that had the least penalty on the weight were assigned discrete available sections, and continuous variable optimization was performed again with the reduced set of design variables. The procedure was continued until all members had been assigned discrete sections. A 3D unsymmetrical frame was used to demonstrate the optimization procedure. This adaptive member selection procedure worked quite well and is similar to the one proposed by Arora (1989) and demonstrated for optimum design of truss structures.

Using the section number as the only design variable, Balling (1991) implemented a simulated annealing (SA) method for optimum design of 3D steel frameworks using available sections. To reduce the number of analyses required in SA, approximation concepts were used to reanalyse the structure (Barthelemy and Haftka 1993). An exact analysis was performed for the final design to check its feasibility. A member-by-member search strategy based on random number generation was used to come up with a good initial design for the SA procedure. In this strategy, the members were allowed to change within four neighbouring sections. In the second phase, the SA strategy repeatedly generated candidate designs in a neighbourhood of the current design by randomly perturbing one of the discrete variables. If the candidate design was infeasible while the current design was either feasible or less infeasible, it was rejected right away. The final solution for an unsymmetric 3D 6-storey frame subjected to 3 loading conditions was compared to the one obtained with the linearized branch and bound (LBB) method of Hager and Balling (1988).

May and Balling (1991, 1992) developed a filter for the SA strategy that blocks many of the poor designs and speeds-up the search process. A candidate design was passed through the filter when a calculated probability was larger than a random number; otherwise the candidate was rejected and a new candidate was generated. The strategy retained the essence of the SA method where worse designs were occasionally passed through the filter. The new method was shown to be more efficient than the standard SA and slightly more efficient than the LBB.

3.2 Single variable approaches without approximations

Instead of using approximate relationships between the section properties, the table containing data for all the available sections can be used directly in structural optimization. For example, the section area A can be used as the sole discrete design variable (type 2) and, when structural analysis is needed, the table can be searched to obtain proper values of other section properties. Therefore, optimal design problem formulation becomes more general in which other properties are related to the sole design variable via the property table. However, these relationships are not continuous or differentiable, and so, a gradient-based method cannot be used.

Another approach would be to use the available section number as the integer design variable for each member. Once the section number is specified, all its properties can be obtained from the appropriate row of the table and used in all the calculations. Liebman *et al.* (1981) have used this approach for optimal design of steel frames. The constrained optimization problem is transformed to an unconstrained one using the interior penalty functions. The problem is then solved using the integer gradient direction method of Glankwahmdee *et al.* (1979). Three example problems are solved: a reinforced concrete beam and two framed structures. The table of available sections needs to be rearranged such that the section areas are in an ascending order. All the design code constraints can be explicitly checked. An initial feasible point is needed to start the search process. The method is simple to implement; however, it can be quite time-consuming on the computer because integer gradient evaluation as well as step size calculation can require a large number of analyses. Amir and Hasegawa (1989) have also used an approach that is quite similar to the one used by Liebman *et al.* (1981). In their approach, all continuous variable are also discretized. Some modifications to the approach are suggested to improve the search process; i.e. if the process fails along the calculated discrete search direction, then some neighbourhood points are searched for improved solutions. Three example problems are solved: a hollow rectangular simply supported beam, a reinforced concrete beam, and a mill building structure.

Simulated annealing and genetic algorithms (stochastic methods) can also be used to solve problems with linked discrete variables. The methods are known to be slow; however, an advantage is that the gradients of functions are not required. Therefore analytical relationships among the properties need not be provided and the property table can be used directly. The section number can be used as the sole design variable. All the design code constraints can be checked since all the section properties are precisely known. In their pure form, these approaches have not been used for design of steel frames because they are extremely time-consuming. The techniques, however, can be combined with other methods to reduce the computational burden.

3.3 Mixed single and multiple variable approaches

Another approach to deal with dependent design properties is to treat some of them as independent design variables. Hager and Balling (1988) have developed a two-phase procedure for optimum design of planar steel frames using multiple section properties as design variables along with a branch and bound method. The procedure was demonstrated on a 3-bay, 8 storey frame. In Phase I of the procedure, a continuous variable optimization problem was formulated and solved using the section area, strong axis moment of inertia and section modulus as the design variables. The final continuous solution, however, is not likely to be close to any of the discrete sections. This difficulty was mitigated by adding enveloping constraints to the continuous optimization problem. Using all the "economy sections", a convex hull was constructed which essentially defined new linear constraints for the problem, forcing the final solution to be close to the available discrete sections. In Phase II, a modified branch and bound method (BBM) was used to determine the discrete solution. To reduce the size of the enumeration in BBM, small neighbourhoods consisting of 3 or 4 sections around the continuous optimum design for each member were defined. Even with that strategy, the number of trial designs was quite large, requiring enormous computational effort for structural analysis. To overcome this difficulty, the problem was linearized about the continuous solution using the three design variables of Phase I. Thus, the Phase II problem became an LP problem, and so, the method was called the linear branch and bound method (LBB). The problem was solved using the Simplex method of linear programming. Then some more members were assigned discrete sections, and the procedure was continued until all the members had been assigned available sections. The difficulty of infeasibility of linearized problems during BBM was encountered and a procedure to overcome it was discussed. It is noted that all the design code constraints cannot be imposed due to the selection of the linearization variables for the problem. Balling and Fonseca (1989) extended the LBB method to optimize 3D steel frames. Five section properties were used as design variables instead of the three for 2D frames. They were: area, two moments of inertia and two section moduli.

3.4 Rounding-off methods

If a problem can be formulated and solved with continuous design variables, then the simplest way to obtain a discrete solution is to use a rounding-off procedure. For structural design problems, the variables are usually rounded-up resulting in a conservative design. An alternate approach would be to increase only some variables to their upper discrete neighbours and decreased others to their lower neighbours. The main difficulty with this approach would be the selection of the variables that should be increased or the variables that should be decreased. Huang and Arora (1995, 1996) describe a dynamic rounding-up method which increases only one variable at a time to its upper discrete neighbour. The selected variable is then fixed and the problem is optimized again, allowing other variables to change. This process is repeated until all variables have been assigned discrete values.

In another approach (Al-Saadoun and Arora 1989), four design variables are used to formulate and solve the problem using an SQP method. For the final solution, each member is selected from the available ones using one of the following two criteria: (i) selection based on optimum depth and section modulus, and (ii) selection based on optimum section modulus and minimum area of cross-section. Another approach is suggested by Arora (1989, pp. 491–495) for the optimal design of trusses. The discrete member selection process is dynamic and works as follows: Once a continuous solution is obtained, a member that gives the least penalty for the cost function due to discrete specification, is selected from the available sections. The sensitivity of the cost function to design variables is used to calculate this penalty for the cost function. Keeping the selected members as fixed, the problem is then reoptimized using the continuous variable method. The process is continued until all members have been selected from available sections.

4 Three design strategies

In this section, three strategies are presented for optimization of steel structures using available sections. All the strategies have two phases and are based on the following algorithms: genetic algorithm, simulated annealing, branch and bound method and sequential quadratic programming (SQP) method. Details of these methods have been presented elsewhere, so they are not presented here (Arora *et al.* 1994; Arora and Huang 1996; Huang 1995; Huang and Arora 1995, 1996, 1997).

The proposed strategies fall into the class of mixed single and multiple variable approaches of Section 3.3. They are defined as: GADSS - Genetic Algorithm for Design of Steel Structures, SADSS - Simulated Annealing for Design of Steel Structures, and BBMDSS - Branch and Bound Method for Design of Steel Structures. In all the methods, each group of members is assigned four design variables: the depth, flange width, flange thickness and the web thickness of each cross-section. In GADSS, an SQP method is used in Phase I to find a continuous solution. A candidate section set (a subset of the AISC table) is created for each member based on this continuous solution. Then a genetic algorithm is used to solve the discrete variable optimization problem in Phase II (Huang and Arora 1997). The algorithm SADSS is similar to algorithm GADSS except that a simulated annealing method is used during discrete variable optimization phase (Huang and Arora 1996b).

In Algorithm BBMDSS, an SQP method is also used to find a continuous solution for the four design variable formulation, as for the foregoing two procedures. Then a candidate section set is created for each member such that the moment of inertia, section area and section modulus are within 5% of

Table 2. Grouping information for the 200-member truss

Group	Members numbers ($k = 0, 1$ and $j = 0, 1, 2, 3, 4$)	Notes
1	1 to 5	Horizontal, top.
2	5+3j	Verticals.
3	19 to 24	Horizontal, interior.
4	18+7k, 56+7k, 94+7k, 132+7k, 170+7k	Horizontal, exterior.
5	26+3j	Verticals.
6	6+k, 9+k, 12+k, 15+k, 27+k, 30+k, 33+k, 36+k	Diagonals.
7	39 to 42	Horizontals.
8	43+3j	Verticals.
9	57 to 62	Horizontal, interior.
10	64+3j	Verticals.
11	44+k, 47+k, 50+k, 53+k, 65+k, 68+k, 71+k, 74+k	Diagonals.
12	77 to 80	Horizontals.
13	81+3j	Verticals.
14	95 to 100	Horizontal, interior.
15	102+3j	Verticals.
16	82+k, 85+k, 88+k, 91+k, 103+k, 106+k, 109+k, 112+k	Diagonals.
17	115 to 118	Horizontals.
18	119+3j	Verticals.
19	133 to 138	Horizontal, interior.
20	140+3j	Verticals.
21	120+k, 123+k, 126+k, 129+k, 141+k, 144+k, 147+k, 150+k	Diagonals.
22	153 to 156	Horizontals.
23	157+3j	Verticals.
24	171 to 176	Horizontal, interior.
25	178+3j	Verticals.
26	158+k, 161+k, 164+k, 167+k, 179+k, 182+k, 185+k, 188+k	Diagonals.
27	191 to 194	Horizontals.
28	195, 197, 198, 200	Inclined, at base.
29	196, 199	Verticals, at base.

the ones for the continuous solution. In Phase II, the problem is formulated using the moment of inertia as the sole discrete design variable for a member. Approximate relationships between the section properties are used at this stage. In developing these relationships, the section dimensions corresponding to the continuous solution of Phase I are used. The allowable discrete values for the moment of inertia for each member are specified by the selected set of sections for the member. Selected sections are arranged in the ascending order of the value for the moment of inertia. The problem is now solved using a branch and bound method (Huang and Arora 1996) where the discrete variables can have nondiscrete values during the solution process. After each local minimization, members are assigned discrete sections from the selected set according to the values of the moment of inertia. At the end of the branch and bound method, when a discrete solution has been obtained, the allowable section set is updated based on the final values for the moment of inertia, section modulus and the section area of each member. Then, the BBM is repeated to obtain a new discrete solution (the parameter IT used in later presentations indicates the number of such iterations). The process is continued until the solution cannot be improved further.

4.0.1 Algorithm GADSS: genetic algorithm for the design of steel structures

Step 1. Formulate the structural design problem. Each group has four design variables: depth, flange width, flange thickness and web thickness (Type 1 variables).

Step 2. Assuming all design variables to be continuous and not linked to each other, solve the optimization problem using the SQP algorithm (Huang and Arora 1996). Save the design variable and cost function values at solution as $\mathbf{x}^*$ and f^*, respectively.

Step 3. According to $\mathbf{x}^*$, calculate the cross-sectional area A_i^* and the moment of inertia I_i^* (about the x-x axis) for each group; $i = 1, \ldots,$ NG; NG is the number of member groups in the structure.

Step 4. For each group, search AISC steel sections (Table 1 with all 187 sections). If a section from the table satisfies any of the following two criteria, then include the section number in the "candidate section set" for the i-th group (the subscript "aisc" indicates the data from Table 1):

$$(a)\ \ 0.3A_i^* \le A_{\text{aisc}} \le 5A_i^*\,,$$

$$(b)\ \ 0.3I_i^* \le I_{\text{aisc}} \le 5I_i^*\,. \tag{15}$$

Table 3. Results with GADSS

No.	Problem name	COST	CPU	NCF	NGF
1	10-bar truss	2.648	1098.37	43846	361100
2	2-bay 6-storey frame	10.123	11653.00	24465	1246360
3	200-bar truss	12.744	3408.31	27289	69369

Table 4. Results with SADSS

No.	Problem name	COST	CPU	NCF	NGF
1	10-bar truss	3.851	283.36	91550	166541
2	2-bay 6-storey frame	11.974	1469.50	93109	196265
3	200-bar truss	15.254	17569.06	289280	405565

Step 5. Using the candidate section sets from Step 4, solve the discrete variable optimization problem by using the genetic algorithm described by Huang and Arora (1997). During the optimization process, only the section number is used as the sole design variable for each group (type 3 variable). The section number gives all the section properties needed in structural analysis and constraint evaluation. The design variable and cost function values at solution are saved as $\mathbf{x}^*_{\text{discr}}$ and f^*_{discr}, respectively.

Note that in Step 4, sections with area or moment of inertia close to that from the continuous solution are chosen to create the discrete section subsets. This reduces the search domain and enables the genetic algorithm to focus on the more suitable sections. Note also that, depending upon the application, a criterion that is different from the one in (15) may be used to generate candidate sets. For the test problems, the population size in the algorithm is set to 2NV with a minimum of 100 and maximum of 300 (NV is the number of design variables).

4.1 Algorithm SADSS. Simulated annealing for design of steel structures

This algorithm is the same as GADSS except that a simulated annealing method described by Huang and Arora (1997) is used in Step 5. For each simulated annealing iteration 100 trial designs are used.

4.2 Algorithm BBMDSS. Branch and bound method for design of steel structures

Step 1. Formulate the structural design problem. Each group has four design variables: depth, flange width, flange thickness and web thickness (type 1 variables).

Step 2. Assuming all design variables to be continuous and not linked to each other, solve the optimization problem by using the SQP algorithm (Huang and Arora 1996). Save the design variable and cost function values at solution as $\mathbf{x}^*$ and f^*, respectively. Set IT = 0

Step 3. Set IT = IT + 1.
According to $\mathbf{x}^*$, calculate the cross-sectional area A^*_i, moment of inertia I^*_i (about the x-x axis), section modulus S^*_i (about the x-x axis) and radius of gyration r^*_i (about the y-y axis) for each group; $i = 1, \ldots,$ NG; NG is the number of member groups in the structure. Then calculate the following three parameters:

$$\alpha^*_i = \frac{A^*_i}{(I^*_i)^{0.50}},\ \beta^*_i = \frac{S^*_i}{(I^*_i)^{0.75}},\ \gamma^*_i = \frac{r^*_i}{(I^*_i)^{0.25}}. \qquad (16)$$

Step 4. For each group, search the standard steel sections (Table 1 with all 187 sections). If a section from the table satisfies the first two of the following criteria and satisfies the third one for groups in which buckling is considered, then include the moment of inertia I_{aisc} into the "candidate section set" for the i-th group (the subscript "aisc" indicates the data from Table 1):

(a) $0.95\alpha^*_i \le \alpha_{\text{aisc}} \le 1.05\alpha^*_i$,

(b) $0.95\beta^*_i \le \beta_{\text{aisc}} \le 1.05\beta^*_i$,

(c) $0.95\gamma^*_i \le \gamma_{\text{aisc}} \le 1.05\gamma^*_i$. (17)

The moment of inertia I_{aisc}'s in the candidate section set are rearranged in ascending order.

Step 5. Using the member section subsets from Step 4, solve the discrete variable optimization problem by using the branch and bound method described by Huang and Arora (1997). During the optimization process, only the moment of inertia I_i (about the $x-x$ axis) is used as the sole design variable for the i-th group (Type 2 variable). For structural analysis, cross-sectional area A_i, section modulus S_i (about the $x-x$ axis) and radius of gyration r_i (about the $y-y$ axis) for each group are calculated as follows:

$$A_i = \alpha^*_i (I_i)^{0.50},\ S_i = \beta^*_i (I_i)^{0.75},\ r_i = \gamma^*_i (I_i)^{0.25}. \quad (18)$$

The design variable and cost function values at the solution are saved as $\mathbf{x}^*_{\text{discr}}$ and f^*_{discr} respectively.

Step 6. If $|f^* - f^*_{\text{discr}}| \le \varepsilon$, where ε is a small positive value defined by the user, then stop. Otherwise, set $f^* = f^*_{\text{discr}}$, $\mathbf{x}^* = \mathbf{x}^*_{\text{discr}}$ and go to Step 3.

In BBMDSS, the discrete value sets are created according to (17). This allows the calculated values of A_i, S_i and r_i to remain accurate no matter which section is chosen from the discrete subset during the optimization process.

Table 5. Results with BBMDSS for first iteration

No.	Problem name	COST	NS	CPU	NCF	NCG	NGF	NTG
1	10-bar truss	2.057	16	18.18	1094	1079	1094	2158
2	2-bay 6-storey frame	5.636	130	315.11	6507	4663	6496	47559
3	200-bar truss	14.338	126	302.72	2899	1518	2898	15277

Table 6. Results with BBMDSS for second iteration

No.	Problem name	COST	NS	CPU	NCF	NCG	NGF	NTG
1	10-bar truss	1.722	32	17.21	1185	1161	1185	2477
2	2-bay 6-storey frame	5.399	70	36.42	1494	461	1489	4752
3	200-bar truss	14.271	64	139.91	852	636	850	11796

Table 7. Results with BBMDSS for third iteration

No.	Problem name	COST	NS	CPU	NCF	NCG	NGF	NTG
1	10-bar truss	1.722	32	19.02	1180	1162	1180	3479
2	2-bay 6-storey frame	5.384	58	33.28	1674	363	1668	3717
3	200-bar truss	13.788	48	147.32	1381	504	1376	10394

5 Test problem

The following three test problems are used to evaluate the three algorithms. They are modified from the problems given by Haug and Arora (1979). The objective of each problem is to minimize the weight of the structure. All members must be selected from the 187 AISC W- Shape steel sections. Final designs for the problems are given in Section 6. The number of constraints for each problem is calculated as follows: number of displacement constraints = NLC∗NOD, number of stress constraints = 2NLC∗NM, and number of local buckling constraints = 4NG, where NLC = number of loading conditions, NOD = number of degrees of freedom, and NM = number of members.

5.1 *Problem 1: Design of a 10-bar cantilever truss*

Figure 2 shows the geometry and dimensions of a 10-bar cantilever truss. This structure has 6 joints (4 are free) and 8 degrees of freedom. One loading condition is imposed for the structure: 100 kips acting in the negative y direction at node points 2 and 4. The displacement limit is set to ± 2 in. along each degree of freedom. Other problem data are given as follows: modulus of elasticity: 30000 ksi, material density: 0.283 lb/cu. in., and yield stress: 36 ksi. This problem has 68 constraints and 40 design variables of type 1 (10 of Type 2 or 3).

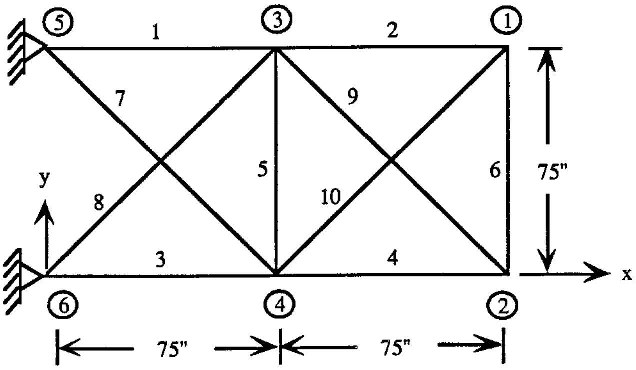

Fig. 2. 10-bar space truss

5.2 *Problem 2. Design of a 2-bay 6-storey frame*

Figure 3 shows the geometry and dimensions of a 2-bay 6-storey frame. This structure has 30 members divided into the following 18 groups: (1, 2), (6, 7), (11, 12), (16, 17), (21, 22), (26, 27), (3, 5), (8, 10), (13, 15), (18, 20), (23, 25), (28, 30), (4), (9), (14), (19), (24) and (29). It has 21 joints and 54 degrees of freedom (x, y translations and a rotation at joints 1 through 18). Two loading conditions are imposed: (1) uniformly distributed load of 0.3333 kip/in in the negative y direction on elements 1, 7, 11, 17, 21 and 27, and 0.0833 kip/in in the negative y direction on elements 2, 6, 12, 16, 22 and 26; and (2) uniformly distributed load of 0.0833 kip/in in the negative y direction on elements 1, 2, 6, 7, 11, 12, 16, 17, 21, 22, 26 and 27, and loads of 9 kips each at nodes 1, 4, 7, 10, 13 and 16 in the positive x direction. The displacement limit is set to ± 2 in. along each degree of freedom. Other problem data are given as follows: modulus of elasticity: 30000 ksi, material density: 0.283 lb/cu. in., and yield stress: 36 ksi. This problem has 300 constraints and 72 design variables of type 1 (18 of type 2 or 3).

5.3 *Problem 3. Design of a 200-member plane truss*

Figure 4 shows the geometry and dimensions of a 200-member plane truss. The 200 members are divided into 29 groups as shown in Table 2. This structure has 77 joints and 150 degrees of freedom. Three loading conditions are imposed on the structure: (1) one kip acting in the positive x direction at node points 1, 6, 15, 20, 29, 34, 43, 48, 57, 62, 71; (2) 10 kips acting in the negative y direction at node points 1, 2, 3, 4, 5, 6, 8, 10, 12, 14, 15, 16, 17, 18, 19, 20, 22, 24,..., 71, 72, 73, 74, 75; and (3) loading conditions 1 and 2 acting together. Some problem data are given as follows: modulus of elasticity: 30000 ksi, material density: 0.283 lb/cu. in., and yield stress: 36 ksi. This problem has 1316 constraints and 116 design variables of type 1 (29 of type 2 or 3).

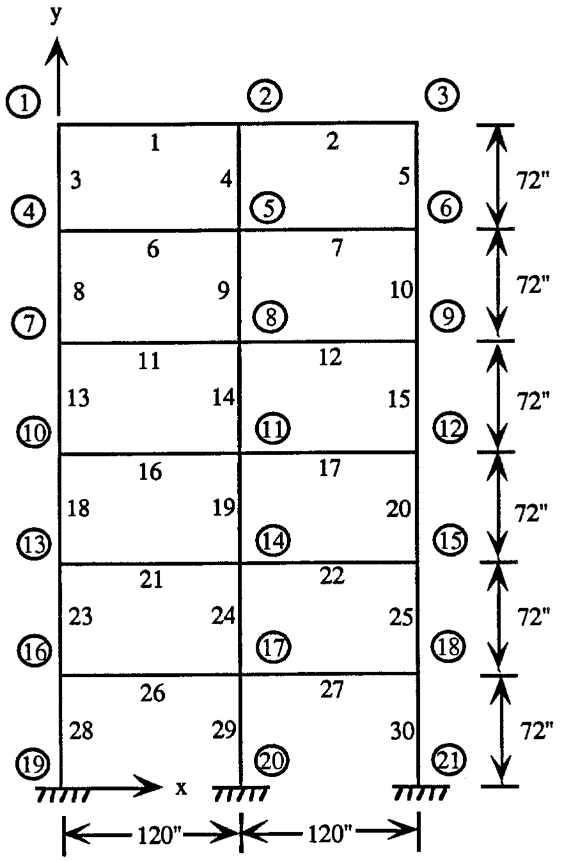

Fig. 3. 2-bay 6-storey plane frame

6 Numerical results

The following data are collected and reported for comparative evaluation of the algorithms:

COST: cost function value
NS: number of subproblems solved
CPU: execution time (seconds)
RCPU: relative execution time with respect to that for the continuous problem
NCF: number of cost function evaluation
NCG: number of gradients evaluated of the cost function
NGF: number of calls for constraint function evaluations
NTG: total number of gradients evaluations
NIT: number of iterations

6.1 *Algorithms GADSS and SADSS*

Results with algorithms GADSS and SADSS are shown in Tables 3 and 4, respectively. Note that the data in the tables do not include measurements from Steps 1-2 of each algorithm. Steps 1-2 are the same for all three algorithms (a continuous subproblem is formulated and solved to prepare the discrete subsets). The results for Steps 1-2 are given later in Table 9 with the SQP method. The cost function values for all the three test problems are smaller with GADSS as

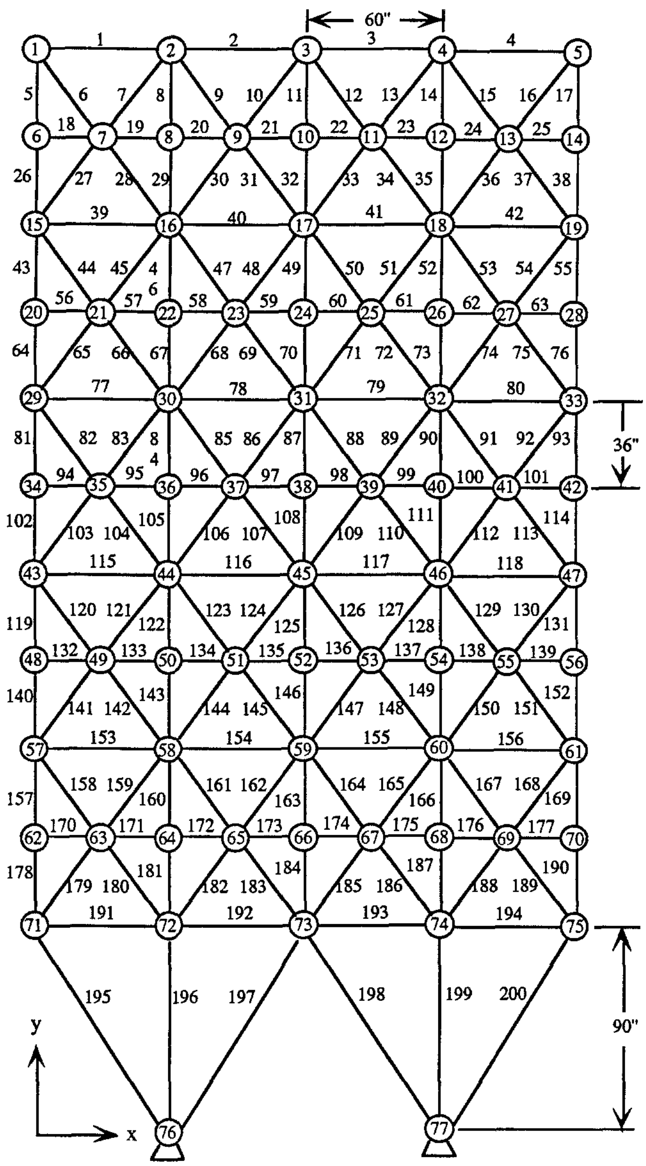

Fig. 4. 200-member plane truss

compared to SADSS. We also used the genetic algorithm to solve the same problems without first preparing the reduced discrete subsets. The results showed that, when all the available sections were used as the allowable discrete subsets, the cost function values at the solution and the CPU times were much higher. The simulated annealing (without the reduced discrete subsets) had a similar behaviour. This shows that the selection of allowable discrete sets is critical for the genetic algorithm and simulated annealing.

Table 8. Results with BBMDSS for the fourth iteration

No.	Problem name	COST	NS	CPU	NCF	NCG	NGF	NTG
1	10-bar truss	1.722	32	19.33	1180	1162	1180	3479
2	2-bay 6-storey frame	5.384	50	92.26	6840	988	6834	7083
3	200-bar truss	13.680	62	186.96	1353	774	1347	14664

Note that in both GADSS and SADSS, random designs are generated. For each design, the constraint functions are evaluated first. If the candidate design is infeasible, it is rejected and a new design is generated. If a design is rejected based on infeasibility, its cost function is not evaluated. Thus in Tables 3 and 4, the number of cost function evaluations is always smaller than the number of calls for the constraint function evaluation.

Table 9. Results with SQP method for the continuous variable formulation

No.	Problem name	COST	NIT	CPU	NCF	NCG	NGF	NTG
1	10-bar truss	1.386	53	2.85	158	53	158	912
2	2-bay 6-storey frame	5.136	108	26.64	478	108	478	4606
3	200-bar truss	6.742	74	56.61	202	74	202	2562

Table 10. Comparison of algorithms

No.	Problem name (NV, NC)		GADSS	SADSS	BBMDSS	Continuous solution
1.	10-bar truss (40, 68)	COST	2.65	3.85	1.72	1.39
		CPU	1101	286	57	2.9
		RCPU	380	99	20	1
2.	2-bay 6-story frame (72, 300)	COST	10.12	11.97	5.38	5.14
		CPU	11680	1497	412	26.6
		RCPU	439	56	15	1
3.	200-bar truss (116, 1316)	COST	12.74	15.25	13.79	6.74
		CPU	3465	17653	647	56.6
		RCPU	61	312	11	1

6.2 *Algorithm BBMDSS*

Tables 5-8 show the results for all three problems with BBMDSS which uses an iterative procedure to improve the discrete solutions. At each iteration (IT), a discrete solution is found which is used to create an improved discrete subset for each design variable, and the problem is re-solved. This process continues until the cost function value cannot be reduced. For all three problems, the good solutions are found in less than five iterations (IT < 5). One disadvantage of BBMDSS is that the local buckling constraints cannot be imposed, as noted previously. This algorithm does not use the dimensions of the cross-section as design variables, so calculations for these constraints are not possible. Although, at any design point, the dimensions can be obtained from the AISC database, there is still no viable way to impose these constraints without making the function nondifferentiable (Step 5 of BBMDSS uses a branch and bound method which requires differentiable functions). One solution would be to limit the members in the discrete sets to the ones that satisfy the local buckling constraints.

Table 11. AISC sections for 10-bar truss

Group No.	GADSS	SADSS	BBMDSS IT = 1	BBMDSS IT = 2	BBMDSS IT = 3
1	W 14×30	W 27×84	W 14×30	W 16×36	
2	W 6×15	W 8×28	W 6×20	W 6×20	
3	W 10×39	W 21×44	W 10×45	W 10×45	
4	W 6×15	W 16×36	W 6×20	W 6×20	
5	W 12×26	W 6×15	W 6×20	W 6×20	Same as
6	W 6×15	W 10×33	W 6×20	W 6×20	IT = 2
7	W 10×49	W 16×67	W 10×49	W 6×20	
8	W 10×49	W 24×68	W 6×20	W 6×20	
9	W 10×54	W 16×67	W 4×13	W 4×13	
10	W 12×53	W 24×68	W 5×19	W 5×19	
Cost	2.648	3.851	2.057	1.722	1.722

6.3 Discussion of results

Results for the problems with continuous variables (without the use of available sections) using the SQP method are given in Table 9. Comparison of the cost function values and execution times with all the algorithms are given in Table 10. Final section designations for the problems are given in Tables 11 to 13. Table 10 also contains data for the continuous solutions. Also, CPU times for BBMDSS are based on three iterations of the algorithm for each problem. Note that when AISC sections are used, the cost function value is usually much higher than that for the continuous solution since some of the sections at the solution may be over-designed with respect to some of the section properties. For example, when a section with a large moment of inertia is needed, the chosen section may have a cross-sectional area that is much larger than what is needed.

Table 12. AISC sections for 2-bay 6-storey frame

Group No.	GADSS	SADSS	BBMDSS IT = 1	BBMDSS IT = 2	BBMDSS IT = 3	BBMDSS IT = 4
1	W 12×53	W 14×61	W 8×21	W 8×21	W 8×21	
2	W 12×79	W 10×49	W 8×28	W 8×28	W 8×28	
3	W 12×53	W 10×49	W 10×26	W 10×26	W 10×26	
4	W 14×61	W 27×84	W 12×26	W 12×26	W 12×26	
5	W 12×53	W 27×84	W 14×26	W 14×26	W 14×26	
6	W 14×61	W 27×84	W 10×19	W 10×19	W 10×19	
7	W 18×40	W 18×35	W 10×15	W 10×17	W 10×17	
8	W 6×15	W 10×22	W 10×17	W 8×10	W 8×10	Same as
9	W 12×26	W 21×44	W 10×12	W 12×16	W 12×14	IT = 3
10	W 10×22	W 12×26	W 14×22	W 14×22	W 14×22	
11	W 14×30	W 18×35	W 16×26	W 16×26	W 16×26	
12	W 16×40	W 21×44	W 16×26	W 16×26	W 16×26	
13	W 6×15	W 8×24	W 12×26	W 8×10	W 8×10	
14	W 8×24	W 14×30	W 12×19	W 12×19	W 12×19	
15	W 8×24	W 14×34	W 18×35	W 14×22	W 14×22	
16	W 18×35	W 14×30	W 18×35	W 16×31	W 16×31	
17	W 14×30	W 27×84	W 18×35	W 16×31	W 16×31	
18	W 16×40	W 18×35	W 24×62	W 24×62	W 24×62	
Cost	10.1225	11.9740	5.6358	5.3990	5.3839	5.3839

The three proposed strategies, SADSS, GADSS and BBMDSS, perform fairly well for all the test problems in the sense that they all find a solution; i.e. they are reliable. The CPU times needed to solve the problems are large compared to those for the continuous variable problems. This is expected for any discrete variable optimization problem having 10-29 discrete design variables (40-116 linked discrete variables). Among the three methods, BBMDSS was the most efficient but it did not give the best solution in all the cases (for Problem 3, a better solution was obtained with GADSS). The SADSS approach was more efficient than the GADSS approach, except for Problem 3; however, the final solution was always worst than that with the GADSS or BBMDSS. It is difficult to draw very general conclusions from these experiments; a larger set of test problems needs to be used. One point is clear that, if the problem can be formulated with continuous and differentiable functions, then the BBMDSS method is the best choice; however, this is not always possible.

7 Concluding remarks

Optimization of engineering systems having linked discrete design variables is discussed in this paper. Such problems in structural engineering are encountered when one considers design of steel frames using standard sections. Here specification of a section number or a property from the available sections dictates the use of all the remaining properties for that section. For such problems, most of the discrete variable optimization methods presented in the literature cannot be readily used. For example, the branch and bound method discussed in engineering optimization literature is not capable of handling this type of problems because its subproblem solver normally requires continuity and differentiability of the problem functions. In the present work, it was used by defining one section property as the primary design variable for each member group and then relating other section properties to it via empirical relationships.

It is important to note that the search process with the genetic algorithms and simulated annealing can be accelerated considerably with the use of parallel processing and reanalysis of the modified structure. In both the methods, many designs need to be generated and evaluated. With the parallel processing capabilities, many of the designs can be analysed at the same time reducing the total wall-clock time required to solve the problem. Also, many new alternative designs that need to be analysed are only slightly different from a baseline design. Therefore, it is possible to explore the use of re-analysis methods to determine response of the modified structure. This may reduce the total computational effort needed to solve the problem. Both of these capabilities will allow design optimization of larger structures. Thus research work needs to continue in order to develop and evaluate better methods for optimization of problems having linked discrete variables with a focus on design of steel frameworks.

Acknowledgement

The authors would like to acknowledge partial support for this research under the project "Design Sensitivity Analysis and Optimization of Nonlinear Structures", from the NSF of USA, Grant No. CMS-93-01580.

References

Al-Saadoun, S.S.; Arora, J.S. 1989: Interactive design optimization of framed structures. *J. Comp. Civil Eng., ASCE* **3**, 60-74

Amir, H.M.; Hasegawa, T. 1989: Nonlinear mixed-discrete structural optimization. *J. Struct. Eng., ASCE* **115**, 626-646

Arora, J.S. 1989: *Introduction to optimum design.* New York: McGraw Hill

Arora, J.S.; Huang, M.W. 1996: Discrete structural optimization with commercially available sections: a review. *J. Struct. Earthquake Eng., JSCE* **13**, 93–110

Arora, J.S.; Huang, M.W.; Hsieh, C.C. 1994: Methods for optimization of nonlinear problems with discrete variables: a review. *Struct. Optim.* **8**, 69-85.

Balling, R.J. 1991: Optimal steel frame design by simulated annealing. *J. Struct. Eng., ASCE* **117**, 1780-1795

Balling, R.J.; Fonseca, F. 1989: Discrete optimization of 3D steel frames. In: *Computer Utilization in Structural Engineering*, pp. 458-467. New York: ASCE

Barthelemy, J.-F.M.; Haftka, R.T. 1993: Approximation concepts for optimum structural design: a review. *Struct. Optim.* **5**, 129-144

Table 13. AISC sections for 200-bar truss

Group No.	GADSS	SADSS	BBMDSS IT = 1	BBMDSS IT = 2	BBMDSS IT = 3	BBMDSS IT = 4	BBMDSS IT = 5
1	W 6×15	W 6×15	W 6×15	W 6×15	W 6×15	W 6×15	
2	W 8×10	W 12×14	W 6×15	W 6×15	W 6×15	W 6×15	
3	W 6× 9	W 10×12	W 6×15	W 6×15	W 6×15	W 6×15	
4	W 12×16	W 10×12	W 6×15	W 6×15	W 6×15	W 6×15	
5	W 12×14	W 6× 9	W 6×15	W 6×15	W 6×15	W 6×15	
6	W 6×12	W 16×26	W 6×15	W 6×15	W 6×15	W 6×15	
7	W 6×15	W 12×26	W 6×15	W 6×15	W 6×15	W 6×15	
8	W 8×10	W 14×22	W 6×15	W 6×15	W 6×15	W 6×15	
9	W 8×10	W 8×10	W 6×15	W 6×15	W 6×15	W 6×15	
10	W 4×13	W 8×10	W 6×15	W 6×15	W 6×15	W 6×15	
11	W 6× 9	W 14×22	W 6×15	W 6×15	W 6×15	W 6×15	
12	W 8×18	W 14×22	W 6×15	W 6×15	W 6×15	W 6×15	
13	W 4×13	W 12×14	W 6×15	W 6×15	W 6×15	W 6×15	
14	W 6×12	W 14×22	W 6×15	W 6×15	W 6×15	W 6×15	Same as
15	W 8×15	W 12×16	W 6×15	W 6×15	W 6×15	W 6×15	IT = 4
16	W 8×13	W 4×13	W 6×15	W 6×15	W 6×15	W 6×15	
17	W 6×25	W 16×26	W 6×15	W 6×15	W 6×15	W 6×15	
18	W 8×21	W 5×19	W 8×28	W 8×28	W 8×28	W 8×28	
19	W 5×19	W 10×12	W 6×15	W 6×15	W 6×15	W 6×15	
20	W 10×26	W 10×22	W 8×35	W 8×31	W 8×28	W 8×28	
21	W 8×10	W 8×15	W 6×15	W 6×15	W 6×15	W 6×15	
22	W 6×25	W 12×26	W 6×15	W 6×15	W 6×15	W 6×15	
23	W 18×35	W 12×35	W 8×40	W 8×40	W 8×40	W 8×40	
24	W 6×12	W 6×16	W 8×24	W 8×24	W 8×24	W 8×24	
25	W 21×50	W 18×35	W 8×40	W 8×40	W 8×40	W 8×40	
26	W 8×10	W 12×16	W 8×28	W 6×15	W 6×16	W 6×15	
27	W 6×25	W 8×18	W 6×15	W 6×15	W 6×15	W 6×15	
28	W 10×54	W 12×53	W 5×19	W 8×48	W 6×25	W 6×25	
29	W 8×48	W 16×57	W 12×79	W 10×49	W 12×72	W 12×65	
Cost	12.7437	15.2540	14.3384	14.2712	13.7880	13.6797	13.6797

Cameron, G.E.; Xu, L.; Grierson, D.E. 1991: Discrete optimal design of 3D frameworks. In: Ural, O.; Wang, T-L. (eds.) *Proc. 10th Electronic Computation Conf.*, pp. 181-188. New York: ASCE

Cella, A.; Logher, R.D. 1971: Automated optimum design from discrete components. *J. Struct. Eng., ASCE* **97**, 175-189

Chan, C.-M. 1992: An optimality criteria algorithm for tall steel building design using commercial standard sections. *Struct. Optim.* **5**, 26-29

Chan, C.-M.; Grierson, D.E.; Sherbourne, A.N. 1995: Automatic optimal design of tall steel building frameworks. *J. Struct. Eng., ASCE.* **121**, 838-847

Elwakeil, O.; Arora, J.S. 1995: Methods for finding feasible points in constrained optimization *AIAA J.* **33**, 1715-1719

Geoffrion, A.M. 1967: Integer programming by implicit enumeration and Balas' method. *Soc. Indus. & Appl. Math. Rev.* **9**, 178-190

Glankwahmdee, A.; Liebman, J.S.; Hogg, G.L. 1979: Unconstrained discrete nonlinear programming. *Eng. Optim.* **4**, 95-107

Gomory, R.E. 1960: An algorithm for the mixed integer problem. *Report No. P-1885.* Santa Monica: The Rand Corporation

Grierson, D.E.; Cameron, G.E. 1989: Microcomputer-based optimization of steel structures in professional practice. *Microcomputers in Civil Engineering* **4**, 289-296

Grierson, D.E.; Lee, W.H. 1984: Optimal synthesis of steel frameworks using standard sections. *J. Struct. Mech.* **12**, 335-370

Grierson, D.E.; Lee, W.H. 1986: Optimal synthesis of frameworks under elastic and plastic performance constraints using discrete sections. *J. Struct. Mech.* **14**, 401-420

Hager, K.; Balling, R.J. 1988: New approach for discrete structural optimization. *J. Struct. Eng., ASCE* **114**, 1120-1134

Haug, E.J.; Arora, J.S. 1979: *Applied Optimal Design.* New York: Wiley-Interscience, John Wiley and Sons

Hua, H.M. 1983: Optimization of structures of discrete-sized elements. *Comp. Struct.* **17**, 327-333

Huang, M.W. 1995: *Algorithms for mixed continuous-discrete variable problems in structural optimization.* Ph.D. Dissertation, Civil and Environmental Engineering, The University of Iowa

Huang, M.W.; Arora, J.S. 1995: Engineering optimization with discrete variables. In: *Proc. 36th AIAA SDM Conf.* (held in New Orleans, LA), pp. 1475-1485. Washington D.C.: AIAA

Huang, M.W.; Arora, J.S. 1996: A self-scaling implicit SQP method for large scale structural optimization. *Int. J. Num. Meth. Eng.* **39**, 1933-1953

Huang, M.W.; Arora, J.S. 1997: Optimal design with discrete variables: some numerical experiments. *Int. J. Num. Meth. Eng.* **40**, 165-188

Lai, Y-S.; Achenbach, J.D. 1973: Direct search optimization method. *J. Struct. Eng., ASCE* **98**, 19-31

Liebman, J.S.; Khachaturian, N.; Chanaratna, V. 1981: Discrete structural optimization *J. Struct. Eng., ASCE* **107**, 2177-2197

May, S.A.; Balling, R.J. 1991: Strategies which permit multiple discrete section properties per member in 3D frameworks. In:

Ural, O.; Wang, T-L. (eds.) *Proc. 10th Electronic Computation Conf.*, pp. 189-196. New York: ASCE

May, S.A.; Balling, R.J. 1992: A filtered simulated annealing strategy for discrete optimization of 3D steel frameworks. *Struct. Optim.* **4**, 142-148.

Reinschmidt, K.F. 1971: Discrete structural optimization. *J. Struct. Eng., ASCE* **97**, 133-156

Toakley, R. 1968: Optimum design using available sections. *J. Struct. Eng., ASCE* **94**, 1219-1241.

Received Oct. 28, 1996