Optimization of Large Steel Truss Structures Using Standard Cross Sections

R. SOEGIARSO and H. ADELI

ABSTRACT

A multi-constraint optimality criteria discrete optimization algorithm is presented for minimum weight design of large steel truss structures subjected to stress, displacement, and buckling constraints. The structures are subjected to actual constraints of AISC Allowable Stress Design (ASD) and Load Resistance Factor Design (LRFD) specifications. A design linking strategy is used to reduce the number of design variables. An efficient integer mapping strategy is presented for mapping the computed cross-sectional areas to the available standard wide flange (W) shapes. The algorithm has been applied to minimum weight design of two structures: a 52-story structure with 848 members and an 80-story structure with 5860 members. This research also sheds some light on the comparison of the AISC ASD and LRFD specifications. For the examples presented, designs based on the LRFD specifications resulted in weight savings of 5 to 9 percent.

INTRODUCTION

A number of papers have been published recently on the subject of optimization of large truss structures assuming continuous design variables (for example, Hsu and Adeli, 1991, Adeli and Cheng, 1993, Soegiarso and Adeli, 1994). This paper is concerned with optimization of large steel truss structures consisting of a few thousand members using standard cross sections such as wide flange (W) shapes included in the American Institute of Steel Construction (AISC) manuals (AISC, 1989, 1994). Recently Chan (1992) and Chan and Grierson (1993) presented a discrete optimization algorithm for minimum weight design of two dimensional steel frames subjected to drift constraints only.

In this paper, a multi-constraint optimality criteria discrete optimization algorithm is presented for minimum weight design of structures subjected to stress, displacement or multiple drift, and buckling constraints. The structures are subjected to actual constraints of the AISC Allowable Stress Design (ASD) and Load and Resistance Factor Design (LRFD) specifications (AISC, 1989, 1994). A design linking strategy is used to incorporate the fact that in practice the same shape is used for members with similar physical and loading conditions.

Cross-sectional areas of members are normally used as design variables. When buckling constraints are considered two additional difficulties are encountered. First, the radius of gyration of the cross-section appears in the buckling constraint. In order to avoid doubling the number of design variables, various researchers have tried to relate the radius of gyration to the cross-sectional area approximately (for example, Adeli and Balasubramanyam, 1988, John and Ramakrishnan, 1990, Chan, 1992). Second, buckling constraints are nonlinear and implicit functions of design variables. This can create convergence difficulties.

Our discrete optimization algorithm includes an efficient integer mapping strategy for mapping the computed cross-sectional areas to the available W shapes. This strategy includes the creation of a tree network and an integer search method.

The algorithm has been applied to minimum weight design of two highrise and super highrise truss structures: a 52-story structure with 848 members and an 80-story structure with 5,860 members

R. Soegiarso is Ph.D. candidate, Department of Civil Engineering, The Ohio State University, Columbus, OH 43210-1275, USA.

H. Adeli is professor, Department of Civil Engineering, The Ohio State University, Columbus, OH 43210-1275, USA.

AN OPTIMALITY CRITERIA APPROACH

In this section, we present an optimality criteria approach for optimization of steel structures subjected to stress, displacement, and buckling constraints based on the AISC ASD or LRFD specifications. The algorithm is general, but we limit the scope of the paper to axial-load space structures. The general structural optimization problem with design linking strategy can be stated in the following form: Find the set of design variables, A_i (cross-sectional areas), such that the weight of the structure

$$W = \sum_{i=1}^{N_d} \rho_i A_i \sum_{m=1}^{N_{mi}} L_{im} \qquad (1)$$

is minimized subject to the displacement, fabricational, stress and buckling constraints to be described shortly. In Equation 1,

N_d = the number of design variables (groups of members with identical cross sections)
L_{im} = the length of member m belonging to group i
ρ_i = the unit weight of members in group i,
N_{mi} = the number of members in group i, and
A_i = the cross-sectional area of members in group i.

The displacement and fabricational constraints are

$$\begin{cases} r_j^L \le u_{jk} \le r_j^U, j = 1, ..., N\ k = 1, ..., L \\ \text{or} \\ r_s^L \le u_{sk} \le r_s^U, s = 1, ..., N_s\ k = 1, ..., L \end{cases} \quad (2)$$

$$A_i^L \le A_i \le A_i^U, i = 1, ..., N_d \quad (3)$$

where

k = is the loading number
N = the total number of displacement degrees of freedom
N_s = the number of interstory drift constraints
L = the number of loading conditions
A_i^L = the lower bound on the cross-sectional area
A_i^U = the upper bound on cross-sectional area
u_{jk} = the displacement of the jth degree of freedom due to loading condition k,
r_j^L and r_j^U = the lower and upper bounds on the displacement of the jth degree of freedom
r_s^L and r_s^U = the lower and upper bounds on the interstory drift, and
u_{sk} = the maximum interstory drift at the sth floor.

A) Stress and buckling constraints based on the AISC ASD specifications

Using positive values for tensile stresses and negative values for compressive stresses, the stress constraints according to the AISC ASD specifications can be expressed as

$$-F_{am} \le \sigma_{mk} \le 0.6F_y \quad m = 1, 2, ..., N_m \quad (4)$$

where

σ_{mk} = the stress in member m due to the loading condition k
F_y = the yield stress of the steel
N_m = the total number of members in the structure equal to

$$N_m = \sum_{i=1}^{N_d} N_{mi} \quad (5)$$

and

F_{am} = is the allowable axial compressive stress given as a function of the slenderness ratio L_m / r_m, (AISC, 1989)

$$F_{am} = \begin{cases} \left\{ \dfrac{\left[1 - \dfrac{(L_m / r_m)^2}{C_c^2}\right] F_y}{\dfrac{5}{3} + \dfrac{3(L_m / r_m)}{8C_c} - \dfrac{(L_m / r_m)^3}{C_c^3}} \right\} & \text{for } L_m / r_m \le C_c \\ \dfrac{12\pi E}{23(L_m / r_m)^3} & \text{for } L_m / r_m > C_c \end{cases} \quad (6)$$

B) Stress and buckling constraints based on the AISC LRFD specifications

$$-F_{cm} \le \sigma_{mk} \le 0.9F_y \quad m = 1, 2, ..., N_m \quad (7)$$

where F_{cm} is the critical compressive strength given by (AISC, 1994)

$$F_{cm} = \begin{cases} (0.658^{\lambda_c^2})F_y & \text{for } \lambda_c = \dfrac{L_m}{r_m \pi}\sqrt{\dfrac{F_y}{E}} \le 1.5 \\ \left(\dfrac{0.877}{\lambda_c^2}\right)F_y & \text{for } \lambda_c = \dfrac{L_m}{r_m \pi}\sqrt{\dfrac{F_y}{E}} > 1.5 \end{cases} \quad (8)$$

Considering only the active constraints, the Lagrangian function for displacement constraints is defined as

$$L(\lambda_{jk}, A_i) = \sum_{i=1}^{N_d} \rho_i A_i \sum_{m=1}^{N_{mi}} L_{im} + \sum_{j=1}^{N_{ac}} \sum_{k=1}^{L} \lambda_{jk}(u_{jk} - r_j) \quad (9)$$

where

λ_{jk} = the positive Lagrange multiplier for displacement constraint associated with the jth constrained degree of freedom and the kth loading case, and
N_{ac} = the number of active displacement constraints.

A displacement constraint is called active when it is in the neighborhood of the limiting value within a given tolerance.

In order to obtain the optimality condition, we differentiate the Lagrangian function with respect to design variable A_i.

$$\rho_i L_i + \sum_{j=1}^{N_{ac}} \sum_{k=1}^{L} \lambda_{jk} \frac{\partial u_{jk}}{\partial A_i} = 0 \quad (10)$$

Employing the principle of virtual load, the gradient of the jth displacement degree of freedom under the kth loading condition with respect to design variables A_i can be expressed as

$$g_{ijk} = \frac{\partial u_{jk}}{\partial A_i} = \frac{1}{A_i} \sum_{m=1}^{N_{mi}} -v_{imj}^T K_{im} u_{imk} \quad (11)$$

where

v_{imj} = the 6×1 vector of virtual displacements for member m belonging to group i due to the application of a unit load in the direction of the jth degree of freedom (three displacements for each node of the member)

K_{im} = the 6×6 stiffness matrix of member m belonging to group i, and

u_{imk} = the 6×1 displacement vector of member m belonging to group i due to the loading condition k.

After substituting the displacement constraint gradient, Equation 11, into Equation 10, rearranging terms and considering only the jth active displacement constraint under the kth loading condition, we find the Lagrange multipliers (Khot and Berke, 1987):

$$\lambda_{jk} = \frac{W}{u_{jk}} \tag{12}$$

Substituting Equations 11 and 12 into Equation 10 and rearranging the terms we obtain

$$(A_i)_{p+1} = (A_i)_p \left\{ \left[\left(\sum_{c=1}^{N_{ac}} \left(\frac{W}{u_{jk}} \right)_c \right) \left(\frac{\sum_{m=1}^{N_{mi}} (v_{imj}^T)_c K_{im} u_{imk}}{\rho_i A_i \sum_{m=1}^{N_{mi}} L_{mi}} \right) \right]^{\zeta} \right\}_p ,$$

$$i = 1, 2, ..., N_d \tag{13}$$

where

p = the iteration number, and

$\zeta = \frac{i}{n}$ determines the step size.

In this work a value of 0.5 is initially used for ζ in each iteration. If the weight of the structure is increased at the end of the iteration, a new value is used for ζ equal to one half of the previous value. If the weight is decreased, the same ζ = 0.5 is used in the following iteration. A lower limit of 0.01 is used for ζ. When ζ is reduced to this value, in the next iteration ζ is again initialized to 0.5. The idea behind a variable ζ coefficient is to dampen the convergence oscillations.

For stress constraints, the cross-sectional areas are updated from the stress ratio relationship in the following form:

A) For the case of AISC ASD specifications

$$(A_m)_{p+1} = (A_m)_p \left[\frac{\sigma_{mk}}{F_{am} \text{ or } 0.60F_y} \right] \tag{14}$$

B) For the case of AISC LRFD specifications

$$(A_m)_{p+1} = (A_m)_p \left[\frac{\sigma_{mk}}{F_{cm} \text{ or } 0.90F_y} \right] \tag{15}$$

In the multi-constraint optimization process, we first determine whether the most critical constraint (the most violated constraint) is a displacement constraint or a stress constraint. If a displacement constraint is the critical constraint each member is first resized according to Equation 13. Then, the maximum ratio of the calculated displacement to the allowable displacement for various constraint degrees of freedom (SFS) is found. If this ratio is not equal to one within a given tolerance then the design variables are scaled by this ratio. After the resizing and scaling, those members whose stress constraints are still violated are scaled again according to Equation 14 or Equation 15.

If a stress constraint is the critical constraint all the members are scaled according to Equation 14 or Equation 15. In the case of highrise and super highrise building structures the maximum interstory drifts are often the critical constraints. Thus, a resizing of the members according to Equation 13 is performed in each iteration followed by two aforementioned scaling operations.

For large truss structures with slender members the allowable compressive stress (the nominal compressive strength in the case of design on the basis of the LRFD specification) is much smaller than the allowable tensile stress (the nominal tensile strength in the case of design on the basis of the LRFD specifications).

Furthermore, the constraints for compressive strength are nonlinear functions of the radii of gyration and implicit functions of design variables (cross-sectional areas). Scaling the members under compression may create a convergence problem. When the number of members whose compression constraints are violated is large, an "overshooting" problem may deteriorate the convergence performance of the algorithm.

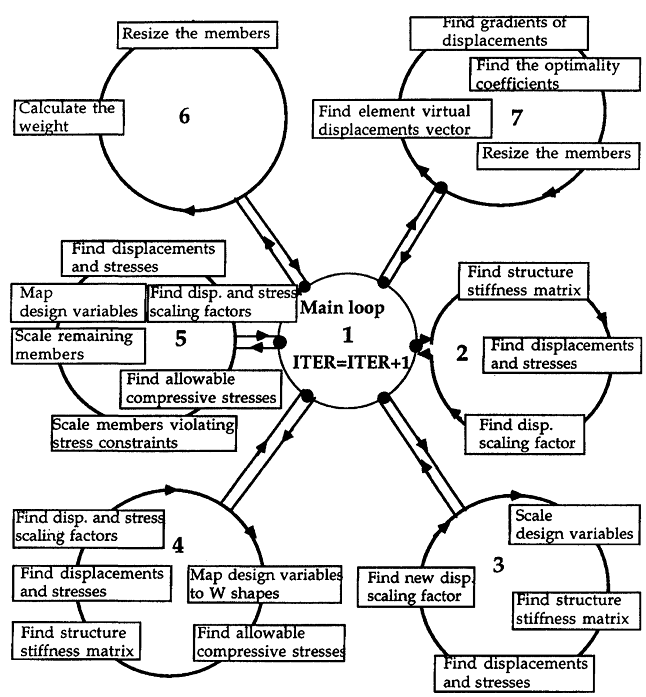

Fig. 1. Multi-constraint optimality criteria discrete optimization.

That is after scaling of the members, they become substantially stiffer resulting in a lateral drift substantially less than the allowable value. This can create a loop with a very slow convergence before we continue to calculate the Lagrange multipliers corresponding to active displacement constraints. To circumvent this problem, the scaling factor for the most violated displacement constraint (SFD) in the iteration is adjusted by a factor of γ whose value depends on the number of members whose compressive stress constraints are violated (N_c). The reason for introducing this parameter is to reduce the number of required structural analyses which is of paramount importance for the efficiency of the algorithm for optimization of large truss structures. We found that by using $\gamma = SFD(0.9)(1 - N_c / N_m)$, the required number of structural analyses is reduced substantially.

There are seven loops in our multi-constraint optimality criteria approach for discrete optimization of large truss structures, as shown schematically in Figure 1. The first loop is the main loop where the number of optimization iterations is controlled. In the second loop the maximum scaling factor is obtained from the initial values for the cross-sections. In the third loop the design variables are estimated by satisfying the displacement constraints only, assuming continuous variables, and a new scaling factor is computed.

In the fourth loop the continuous design variables are mapped to the available discrete standard cross-sections from the AISC W sections database following a strategy to be discussed in the next section. The maximum displacement scaling factor (SFD) and scaling factors for those members whose compressive stress constraints are violated (SFS's) are obtained. In the fifth loop all the members whose compressive stress constraints are violated are scaled by SFS's and the remaining members are scaled by γ. Then, all the members are mapped again to the discrete standard cross-sections from the AISC W sections database. This process is repeated in loop 5 until all compression members satisfy the stress and buckling constraints.

The sixth loop is to calculate the weight of the structure and resize the members according to Equation 13. If the weight of the structure in the current iteration is less than the weight in the previous iteration or the step size parameter ζ is less than 0.01, the optimality coefficients are calculated and the members are resized in the loop 7. Otherwise, the resizing procedure in loop 6 is repeated using a value of $\zeta = \zeta / 2$.

MAPPING TO STANDARD CROSS SECTIONS

In practical design of steel structures, only a finite number of shapes are available such as those given in the AISC manuals (AISC, 1986 & 1989). We create a database containing the properties of the commonly used wide flange (W) shapes from these manuals in the ascending order of the cross-sectional areas. In loops 4 and 5 of the discrete optimization algorithm (Figure 1) a W shape is selected for each group of the members. For a large truss structure with a few thousand members (and a couple of hundreds different types of design variables, the number of searches to select the right W shapes from the database can reach tens of thousands. Thus, an efficient strategy needs to be devised to map the computed cross-sectional areas to the existing W shapes from the database.

In this research, an integer search method is developed to map the computed cross-sectional areas (A_i) to the areas of actual W shapes in the database (A_i). The integer search method is similar to the interval halving method (Adeli and Al-Rijleh, 1987). In the interval halving method, logical (if-then) operations are used to find the search path. In the integer search method, we use division operations to find the search path. The CPU time required for a division operation is a fraction of the CPU time required for a logical operation. Hence, the integer search method developed in this research is more efficient than the interval halving method.

Let us consider a 3-layer tree network with eight cross-sectional areas as target output as shown in Figure 2a. Each node in this network has an upper and a lower branch. The kth node in the jth layer is assigned an area $A_{k,j}$ equal to the largest cross-sectional area of the all the "child" nodes in the upper branch.

In the interval halving method, the search is performed as follows: If the computed cross-sectional area is less than or equal to $A_{1,0}$, the search path follows the upper branch, otherwise it follows the lower branch. If the computed cross-sectional area is less than $A_{1,0}$ then the next step is to find out whether the computed cross-sectional area is less than or equal to $A_{1,1}$. If true, the search path follows the upper branch, otherwise it follows the lower branch. If the computed cross-sectional area is greater than $A_{1,1}$ then the search path follows the lower branch. If the computed cross-sectional area is less than or equal to $A_{2,2}$ the target output is $A_{3,3}$; otherwise, it is $A_{4,3}$.

In our integer search method, we assign a weight $W_{k,j}$ to the kth node in the jth layer of the tree network (layers are numbered from left to right, and nodes in each layer are numbered from top to bottom). These weights are stored in a two dimensional array as shown in Figure 2b. The search is performed by finding the address (row number) of the weight in the successive layer that can lead to the right output. The address of the weight in the next layer is determined by the integer value resulting from division of the computed cross-sectional area by the weight of the current node. For instance, the address of the weight of the node in layer one on the solution path is determined by dividing the computed cross-sectional area by $W_{1,0}$. The integer value of this division plus one equals to the address of the weight in layer one. If, for instance, the result of this division plus one is one, then the corresponding weight $W_{1,1}$ is stored in the first row and the second column of the array W. The next step is to find the address of the weight of the node on the search path in layer 2. This is obtained by dividing the computed cross-sectional area by $W_{1,1}$. If, for instance, the result of this division plus one is two, then the corresponding weight in layer two is $W_{2,2}$ and is stored in the second row and third column of the array W. If the result of the division is, say, 3, then the weight $W_{2,2}$ is stored in the third row and the third column of the array W. Finally, the computed cross-sectional area is divided by $W_{2,2}$. The result of this division determines the address of the target output. If the result of this division plus one is three, then the output is A_3. If the result of the division plus one is four, then the output is A_4. The problem now is how to determine the weight of each node and how to store them in a two dimensional array that can lead to the right target output.

In Figure 2a the cross-sectional area of the first node of each layer j is assigned as the weight of that node, that is, $W_{1,j} = A_{1,j}$. If the computed cross-sectional area is less than $W_{1,j}$ then the search path follows the upper branch. Otherwise, the search path follows the lower branch. Let us consider the weight of the 1st node in the 1st layer, $W_{1,1}$. If the computed cross-sectional area (A_i) is less than $W_{1,1}$ then the search path follows the upper branch. The address of the weight $W_{1,2}$ in the next layer is

$$\left(1 + \left\lfloor \frac{A_i}{W_{1,j}} \right\rfloor\right) = 1$$

where the notation $\lfloor \ \rfloor$ indicates the integer of the real number inside the semi-brackets. If the computed cross-sectional area (A_i) is larger than $W_{1,1}$ then the search path follows the lower branch. The address of the weight $W_{2,2}$ in the next layer is

$$\left(1 + \left\lfloor \frac{A_i}{W_{1,j}} \right\rfloor\right) \geq 2$$

Thus, the number of addresses for the weight $W_{2,2}$ may be more than one. This number depends on the largest cross-sectional area of all the "child" nodes in the lower branch, that is $A_{4,3}$. Let us assume

$$\left(1 + \left\lfloor \frac{A_{4,3}}{W_{1,j}} \right\rfloor\right) = 3$$

then $W_{2,2}$ is stored in both rows 2 and 3 and column 3 of the array W as shown in Figure 2b. Thus, 3 is the maximum address number for $W_{2,2}$. The maximum address number for weight $W_{k,j}$ is called $L_{k,j}$.

Now, let us find the weights of other nodes. For instance, let us find the weight of the 2nd node of layer 1, $W_{2,1}$. From Figure 2a, the path from $W_{2,1}$ follows the upper branch if the computed cross-sectional area is greater than $A_{4,3}$ and less than or equal to $A_{6,3}$. Therefore, these two areas are important in determining the weight $W_{2,1}$. When the computed cross-sectional area is slightly greater than $A_{4,3}$ and is divided by $W_{2,1}$, the result of this division plus one has to be equal to the address of $W_{3,2}$. When the computed cross-sectional area is

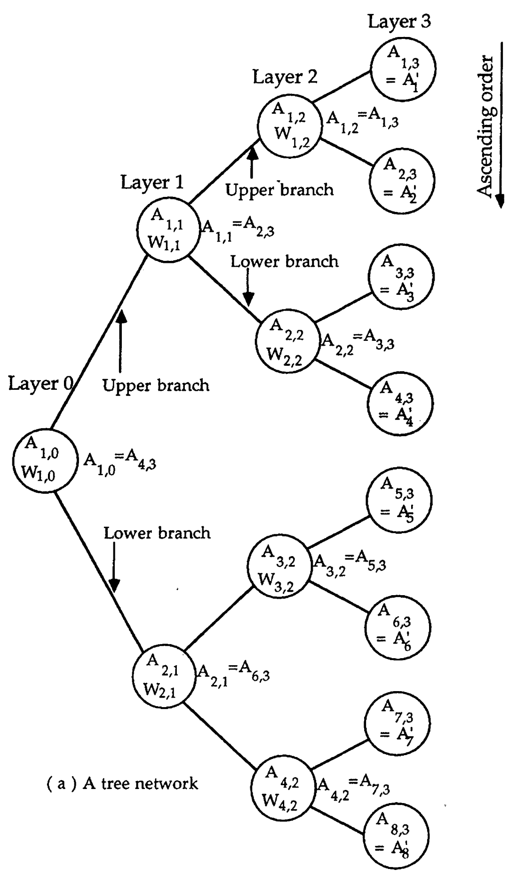

Fig. 2a. A three-layer tree network.

Column number →

Row number ↓

	1	2	3	4
	Weight ($W_{k,j}$)			Output
	Layer 0	Layer 1	Layer 2	Layer 3
1	$W_{1,0}$	$W_{1,1}$	$W_{1,2}$	$A_{1,3}$
2		$W_{2,1}$	$W_{2,2}$	$A_{2,3}$
3			$W_{2,2}$	$A_{3,3}$
4			$W_{3,2}$	$A_{4,3}$
5			$W_{4,2}$	$A_{5,3}$
6				$A_{6,3}$
7				$A_{7,3}$
8				$A_{8,3}$

Fig. 2b. Storage scheme for the weights and outputs.

slightly greater than $A_{6,3}$ and is divided by $W_{2,1}$, the result of this division plus one has to be equal to the address of $W_{4,2}$. In order to satisfy these two conditions, the weight $W_{2,1}$ is determined in three steps as follows: First, $A_{4,3}$ is divided by the maximum address number of $W_{2,2}(L_{2,2})$ which is 3 in our example. The result of this division is called s. The weight of $W_{2,1}$ has to be less than s. Second, we compute the integer

$$I = 1 + \left\lfloor \frac{A_{6,3}}{s} \right\rfloor$$

Finally, the weight is computed as $W_{2,1} = A_{6,3} / I$. The maximum address number of the weight $W_{3,2}$ in the next layer is

$$L_{32} = \left(1 + \left\lfloor \frac{A_{6,3}}{W_{2,1}} \right\rfloor \right)$$

Now, we generalize the concepts presented in the previous paragraphs. Let us assume 2^n W shape are available in the database where n is an integer. A tree network with n layers is created for the W shapes as shown in Figure 3. Each node is assigned an area from the W shape database according to the following equations:

$$A_{k,j} = A_i \quad i = 2^{n-j-1} + (k-1) \times 2^{n-j}, k = 1,2^j, j = 1, ..., n \qquad (16)$$

The next step is to assign a weight $W_{k,j}$ to the kth node in the jth layer of the tree network and find the maximum address number of the weights in the next layer ($L_{k,j+1}$). Each weight is stored in an array as shown in Figure 2b. For instance, if $L_{2,2}$ is equal to 3, then $W_{2,2}$ is stored in rows 2 and 3 and column 3 of array W.

The weights ($W_{k,j}$) and the maximum address numbers of the weights in the successive layers ($L_{k,j+1}$) are determined in three steps. The first step is to assign the weights of the first node of layers 1 to $n-1$ and find the maximum address number of the weights in the next layer as follows:

$$W_{1,j} = A_{1,j} \quad j = 1, ..., n-1 \qquad (17)$$

$$L_{1,j+1} = 1 \quad j = 1, ..., n-1 \qquad (18)$$

$$L_{2,j+1} = \left\lfloor \frac{A_{4,j+2}}{W_{1,j}} \right\rfloor + 1 \quad j = 1, ..., n-1 \qquad (19)$$

The second step is to compute the weights of the nodes in layers 1 to $n-2$ and the maximum address numbers of the weights in the successive layers 2 to $n-1$. The weights are determined by the following rules:

$$s = \frac{A_{4k-4,j+2}}{L_{2k-2,j+1}} \quad k = 2,2^j \quad j = 1, ..., n-2 \qquad (20)$$

$$I = \left\lfloor \frac{A_{k,j}}{s} \right\rfloor \quad k = 2,2^j \quad j = 1, ..., n-2 \qquad (21)$$

$$W_{k,j} = \frac{A_{k,j}}{I+1} \quad k = 2,2^j \quad j = 1, ..., n-2 \qquad (22)$$

$$L_{2k-1,j+1} = \left\lfloor \frac{A_{4k-2,j+2}}{W_{k,j}} \right\rfloor + 1 \quad k = 2,2^j \quad j = 1, ..., n-2 \qquad (23)$$

$$L_{2k,j+1} = \left\lfloor \frac{A_{4k,j+2}}{W_{k,j}} \right\rfloor + 1 \quad k = 2,2^j \quad j = 1, ..., n-2 \qquad (24)$$

The third step is to compute the weights of the nodes in layer $n-1$ and the maximum address numbers of the outputs in layer n.

$$s = \frac{A_{2k-2,j+1}}{L_{2k-2,j+1}} \quad k = 2,2^j \quad j = 1, ..., n-1 \qquad (25)$$

$$I = \left\lfloor \frac{A_{k,j}}{s} \right\rfloor \quad k = 2,2^j \quad j = 1, ..., n-1 \qquad (26)$$

$$W_{k,j} = \frac{A_{k,j}}{I+1} \quad k = 2,2^j \quad j = 1, ..., n-1 \qquad (27)$$

$$L_{2k-1,j+1} = \left\lfloor \frac{A_{2k-1,j+1}}{W_{k,j}} \right\rfloor + 1 \quad k = 2,2^j \quad j = 1, ..., n-1 \qquad (28)$$

$$L_{2k,j+1} = \frac{A_{2k,j+1}}{W_{k,j}} + 1 \quad k = 2,2^j \quad j = 1, ..., n-1 \qquad (29)$$

Then the mapping of the input layer to the output layer is done as follows:

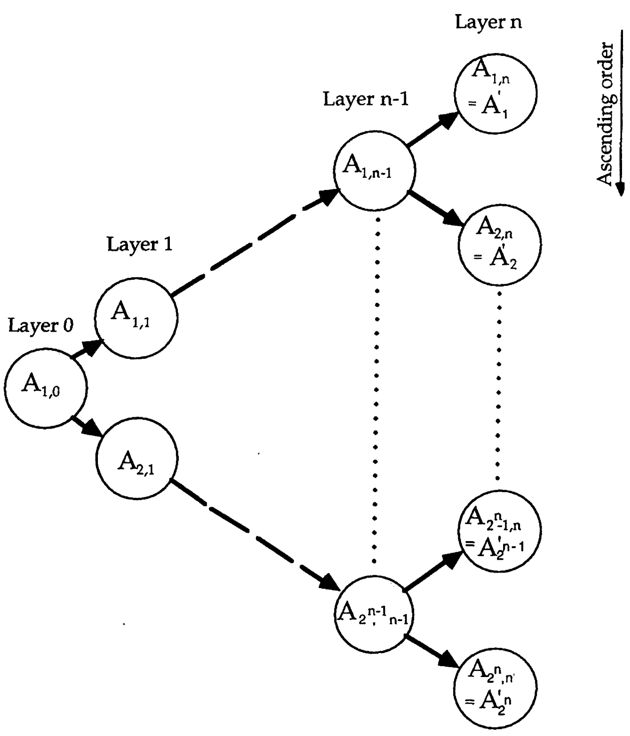

Fig. 3. A tree network for mapping to standard sections

$$I_0 = 0 \tag{30}$$

$$I_j = \left\lfloor \frac{A_i}{W_{I_{j-1}+1,\, j-1}} \right\rfloor j = 1, \ldots, n \tag{31}$$

$$A_i = A_{I_n} \tag{32}$$

We compared the integer search method presented in this article with the interval halving approach (Adeli and Al-Rijleh, 1987) on the Cray YMP8/864 and found the new approach to be about 30 times faster than the interval halving method.

APPLICATIONS

We present two examples in this section.

Example 1 52-story highrise building

The 848-member space truss structure shown in Figures 4a to 4c is created to model the exterior envelope structure of a 52-story highrise building (mega structure). Fifty-four design variables are used to represent the cross-sectional areas of fifty-four groups of members in this structure, as identified in Figure 4b. The groups are organized as follows: Each four floors is divided into four groups: a group of outer column members, a group of inner column members, a group of diagonal members and a group of horizontal members.

The loading on the structure consists of horizontal loads acting on the exterior nodes of the space structure at every four floors (14.63 m or 48 feet height). The horizontal loads in the Y direction at each node on the sides AB and CD are obtained from the Uniform Building Code (UBC, 1994) wind loading using the equation

$$d_p = C_e C_q q_s I$$

where

d_p = the design wind pressure
C_e = the combined height, exposure, and gust factor coefficient
C_q = the pressure coefficient
q_s = the wind stagnation pressure, and
I = the importance factor.

The value of C_q for inward face is 0.8 and for the leeward face is 0.5. Assuming a basic wind speed of 70 mph (113 km/h), the value of q_s is 0.6 kPa (12.6 psf) and the importance factor is assumed to be one. The values of C_e are taken from the UBC (1994) assuming exposure C (generally open area). The lower and upper bounds of the cross-sectional areas in this example

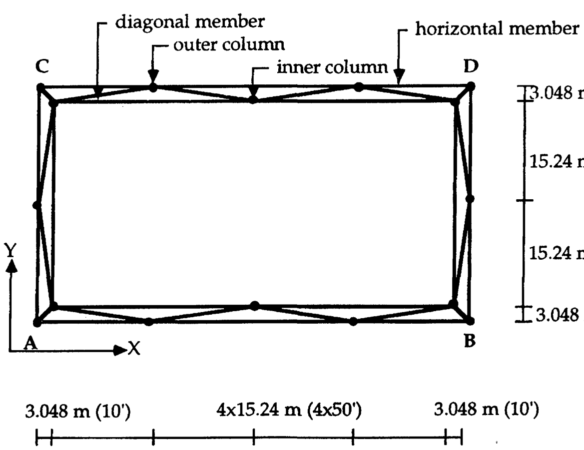

Fig. 4a. Plan of 52-story highrise building structure.

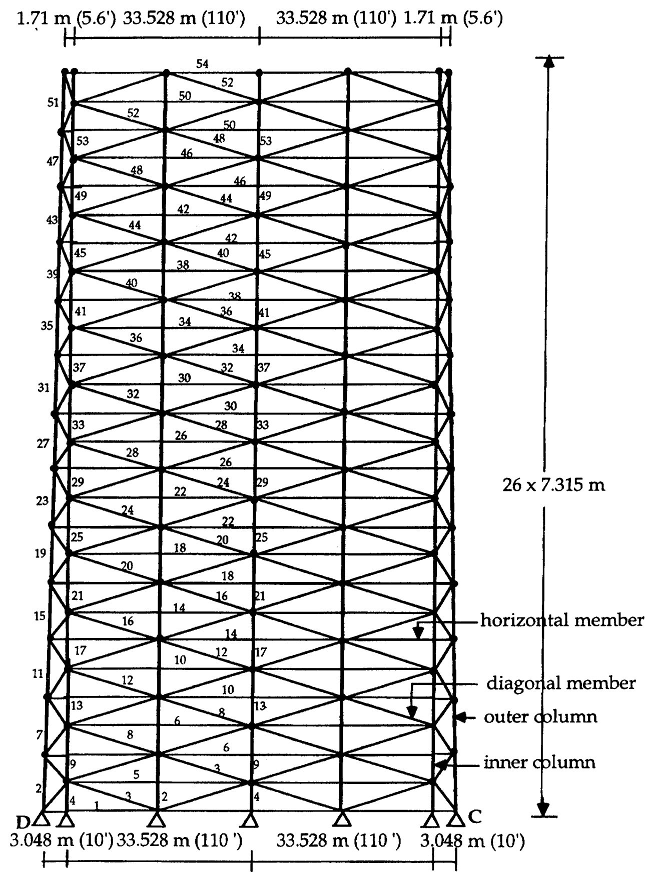

Fig. 4b. Front view of the 52-story highrise building structure.

are 6 cm² and 1600 cm². This structure is designed according to the AISC ASD specifications (AISC, 1989). The material is assumed to be steel with modulus of elasticity of 199.9×10^3 MPa (29,000 ksi) and the unit weight of material 0.077 N/cm³ (0.284 lb/in³).

Two cases of displacement constraints are considered. In the first case, the displacement constraints are specified at the top of the structure as ±78.1 cm (28.8 in.) in the Y-direction for the nodes on the top floor only (equal to $0.004H$, where H is the height of the structure). In the second case, interstory drifts are limited to 5.85 cm (2.3 in.). Figure 5 shows the convergence histories for the cases of continuous variables optimization and discrete optimization using standard W shapes presented in this article. When the displacement constraints are given only for the nodes on the top floor, minimum weight of 107.2 MN (24,108 kips) is found assuming continuous variables. When W shapes are chosen on the basis of rounding up these values, and after performing additional optimization (scaling procedure) iteratively in order to satisfy the design constraints, a minimum weight of 123.3 MN (27,724 kips) is found. The discrete optimization algorithm presented in this article yielded a minimum weight of 112.0 MN (25,186 kips) directly after 8 iterations. When the interstory drift constraints are imposed, a minimum weight of 117.6 MN (26,460 kips) is obtained which is about 5 percent higher than the weight when the displacement constraints are given for the nodes on the top floor only.

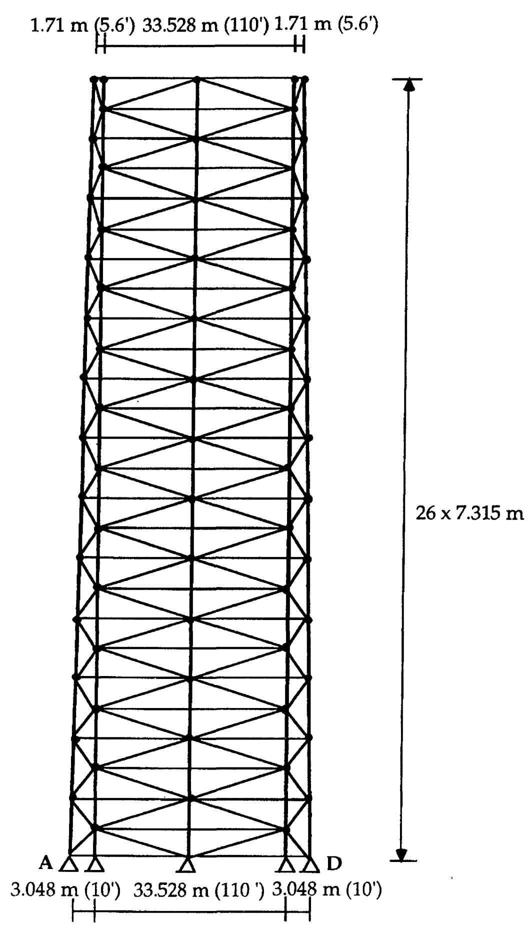

Fig. 4c. Side view of the 52-story highrise building structure.

Example 2 80-story super highrise building structure

The 80-story super highrise building structure shown in Figure 6 has 1,160 nodes. This structure consists of two different sections as shown in Figures 6a to 6c. We have used two slightly different models of the space structure in order to study the effect of the transition zone between the two sections in the overall efficiency of the structure. The model shown in Figure 6c has a gradual transition zone (Example 2a). We have also considered another modified version of this structure where all the inclined members of the transition zone are removed (Example 2b), not shown in Figure 6. Example 2a shown in Figure 6c has 160 additional inclined members going from the 41st floor to the 48th floor. It has 5,860 members grouped into 231 design variables. Example 2b has 5,700 members grouped into 227 design variables. The groups are organized as follows: for section one, each two floors is divided into seven groups i.e., a group of outer

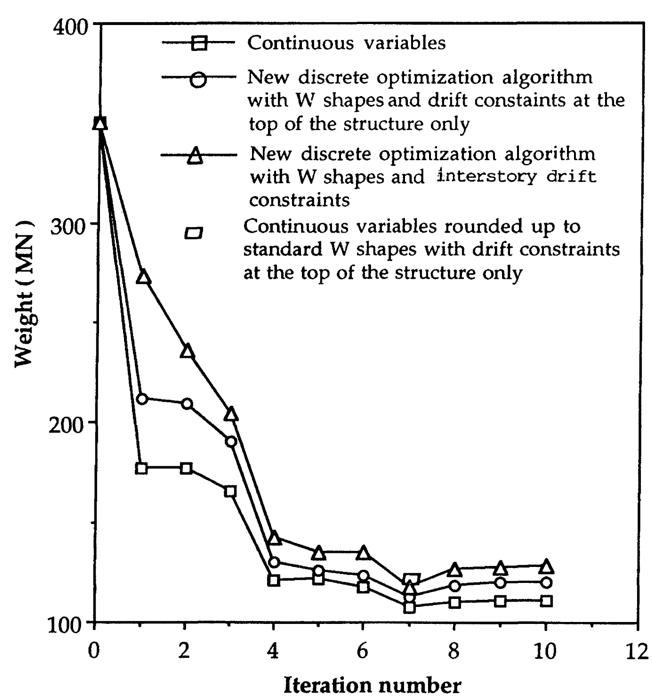

Fig. 5. Convergence histories for the 52-story highrise building structure.

column members, a group of inner column members, a group of outer vertical diagonal members, a group of inner vertical diagonal members, a group of outer horizontal members, a group of inner horizontal members and a group of horizontal diagonal members. For section two, each two floors is divided into four groups that is, column members, vertical diagonal members, horizontal members and horizontal diagonal members.

The loading on section one consists of dead load (D) of 68.9 kN (15.5 kips) at each inner node and 10.14 kN (2.28 kips) at each outer node, live load (L) of 61.4 kN (13.80 kips) at each inner node and 12.54 kN (2.82 kips) at each outer node and roof live load (L_r) of 61.4 kN (13.80 kips) at each inner node and 12.54 kN (2.82 kips) at each outer node of the top level of the structure. The loading on section two consists of vertical dead loads of 58.7 kN (13.2 kips) at each node, live loads of 51.2 kN (11.5 kips) at each node and roof live loads of 51.2 kN (11.5 kips) at each node of the top level of the structure. This structure is subjected to horizontal wind loads in the Y-direction acting on each node on the sides AB and DC similar to Example 1.

This structure is subjected to both AISC ASD and LRFD specifications, material properties are the same as the previous example. For the ASD specifications, only one case of loading is considered, that is, $D + L + W$. For the LRFD specifications, three cases of loadings are considered $1.4D$, $1.2D + 1.6L + 0.5L_r$ and $1.2D + 1.3W + 0.5L + 0.5L_r$. The lower and upper bounds for the cross-sectional areas of this example are 6.0 cm^2 and 3000 cm^2.

Two cases of displacement constraints are considered for the ASD specifications. In the first case, the displacement constraints are specified at the top of the structure as ±118.3 cm (46.6 in.) in the Y-direction for the nodes on the top floor only (equal to $0.004H$, where H is the height of the structure). In the second case, interstory drifts are limited to 1.46 cm (0.576 in.) (equal to $0.004h$, where h is the story height). For the LRFD specifications, the displacement constraints are specified at the top of the structure as ±153.8 cm (60.6 in.) (equal to $0.0052H$), and interstory drift is limited to 1.90 cm (0.749 in.). Since the drift is primarily due to the wind loading, the ASD drift limitation of $0.004H$ (or $0.004h$) is multiplied by 1.30 (the load factor for wind loading in the AISC LRFD specifications) to obtain the LRFD drift limitation.

Figure 7 shows the convergence history for two different models of the structure with 227 and 231 design variables. For the ASD specifications, a minimum weight of 50.72 MN

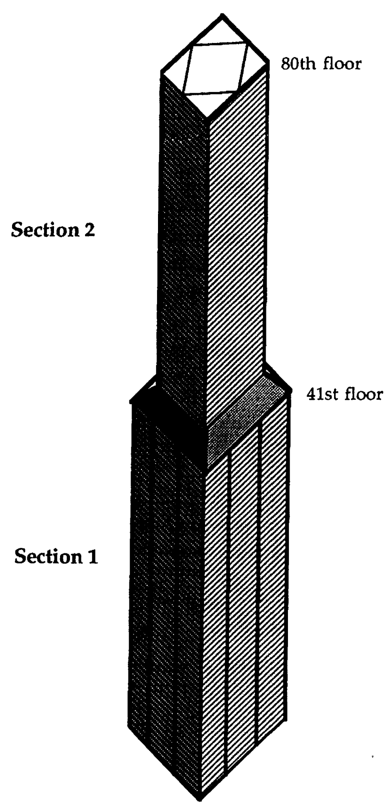

Fig. 6a. Perspective view of 80-story super highrise building structure.

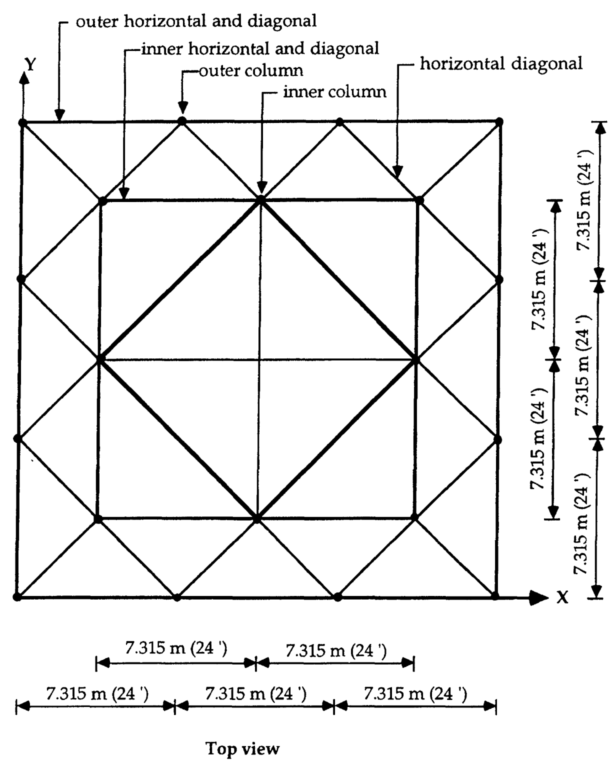

Fig. 6b. Plan of the 80-story super highrise building structure.

(11,403 kips) is found for Example 2b (with 227 design variables) when drift is limited at the top of the structure only. For the same model, but including the interstory drift constraints, a minimum weight of 54.81 MN (12,331 kips) is found which is about 8 percent higher than the previous value. A minimum weight of 43.53 MN (9,787 kips) is obtained for Example 2a in Figure 6c (with 231 design variables). Using the LRFD specifications and limiting the drift at the top of the structure only, a minimum weight of 48.23 MN (10,842 kips) is found for Example 2b and a minimum weight of 40.96 MN (9,208 kips) is obtained for Example 2a.

FINAL COMMENTS

The multi-constraint discrete optimization algorithm presented in this paper has been implemented in FORTRAN on a Cray YMP/864 supercomputer. The code has been fully vectorized using the techniques described in Soegiarso and Adeli (1994). The efficiently vectorized code performs 16–19 times faster than the nonvectorized code.

The optimization convergence histories for large truss structures presented in Figures 4 and 6 demonstrate consistently good performance and fast convergence. We also compared the integer search method developed in this research with the interval-halving approach (Adeli and Al-Rijleh, 1987). We found the former to be about 30 times faster than the latter. This is primarily due to the fact that the main operation is our integer search method requires much less time than the logical operation required in the interval halving approach.

We note that in the examples presented in this article, designs based on the LRFD specifications result in weight savings compared with designs based on the ASD specifications in the range of 5 to 9 percent.

We also studied the effect of the inclined members in the transition zone of a super highrise structure when the floor plan is reduced substantially. In Example 2b this transition is sudden versus gradual transition in Example 2a. Despite the fact that the number of design variables in Example 2a is increased to 231 from 227 for Example 2b its minimum weight is about 14 percent (15 percent) less than the minimum weight for Example 2a based on the AISC ASD specifications (based on the AISC LRFD specifications).

In this article, we presented an effective and robust method for automated design and optimization of large steel truss structures subjected to the actual AISC ASD and LRFD codes. Optimization of large structures can result in substantial structural efficiency in design of steel structures thus making steel structures more competitive in the market place. In a

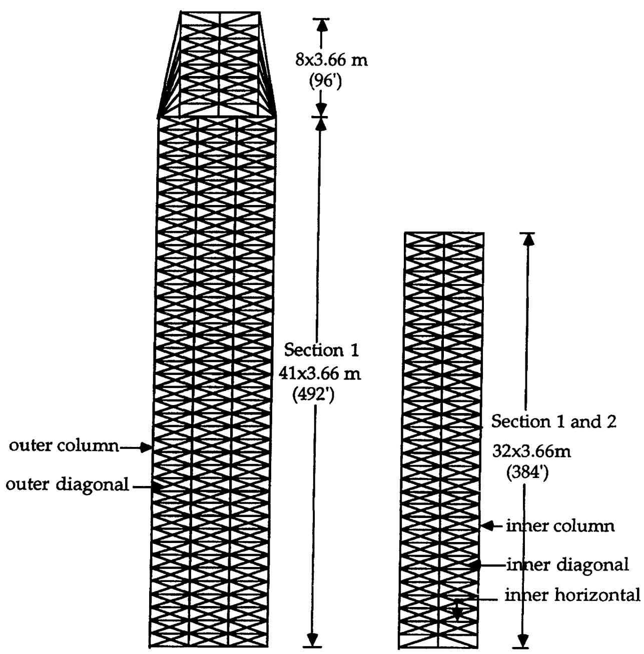

Fig. 6c. Elevation for Example 2a (5,780 members and 231 design variables).

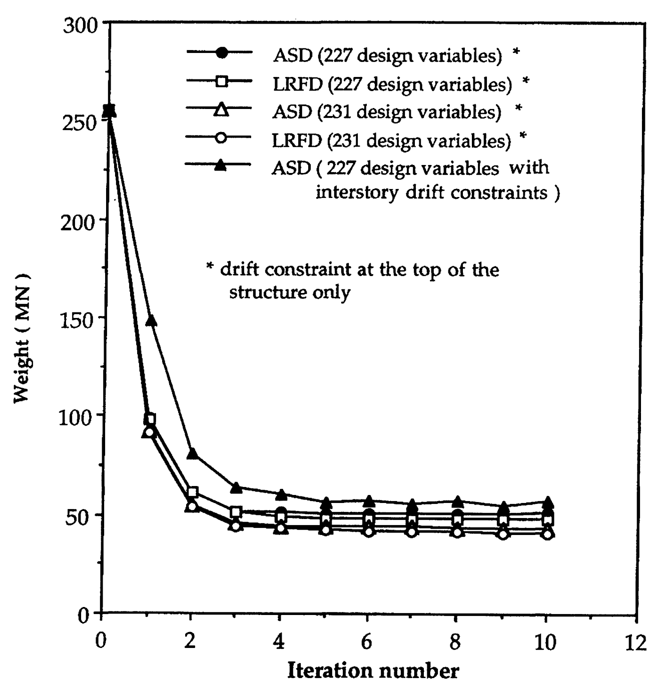

Fig. 7. Convergence histories for the 80-story super highrise building structure.

follow-up paper, we will present the extension of this work to optimization of large space frame steel structures.

ACKNOWLEDGMENT

This research is based upon work sponsored by the National Science Foundation under Grant No. MSS-9222114, American Iron and Steel Institute, and American Institute of Steel Construction which is gratefully acknowledged. Computer time was provided by the Ohio Supercomputer Center.

APPENDIX I. REFERENCES

Adeli, H., Ed., *Advances in Design Optimization*, Chapman and Hall, London, 1994.

Adeli, H. and Al-Rijleh, M. M., "A knowledge based Expert System for Design of Roof Trusses", *Microcomputers in Civil Engineering*, Vol. 2, No.1, 1987, pp. 179–195.

Adeli, H. and Balasubramanyam, K.V., *Expert Systems for Structural Design—A New Generation*, Prentice-Hall, Englewood-Cliffs, NJ, 1988.

Adeli, H. and Cheng, N. T., "Concurrent Genetic Algorithms for Optimization of Large Structures", ASCE *Journal of Aerospace Engineering*, Vol. 7, No. 9, 1994, pp. 276–296.

Adeli, H. and Kamal, O., "Efficient Optimization of Space Trusses, *Computers & Structures*, Vol. 24., No. 3, 1986, pp. 501–511.

Adeli, H. and Kamal, O., "Concurrent Optimization of Large Structures Part I—Algorithm", ASCE *Journal of Aerospace Engineering*, Vol. 5, No. 1, 1992a, pp. 79–90.

Adeli, H. and Kamal, O., "Concurrent Optimization of Large Structures Part II—Applications", ASCE *Journal of Aerospace Fngineering*, Vol. 5, No. 1, 1992b, pp. 91–110.

American Institute of Steel Construction, *Manual of Steel Construction, Allowable Stress Design*, Chicago, Illinois, 1989.

American Institute of Steel Construction, *Manual of Steel Construction, Load and Resistance Factor Design*, Chicago, Illinois, 1994.

Arora, J. S., *Introduction to Optimum Design*, MacGraw Hill, New York, 1989.

Chan, C. M., "An Optimality Criteria Algorithm for Tall Steel Building Design Using Commercial Standard Sections," *Structural Optimization*, Vol. 5, 1992, pp. 26–29.

Chan, C. M. and Grierson, D. E., "An Efficient Resizing Technique for the Design of Tall Steel Buildings Subject to Multiple Drift Constraints", *The Structural Design of Tall Buildings*, Vol. 2, 1993, pp. 17–32.

John, K. V. and Ramakrishnan C. V., "Discrete Optimal Design of Trusses with Stresses and Frequency Constraints", *Journal of Eng. Computing*, Vol. 7, 1990, pp. 37–47.

Hsu, H. L. and Adeli, H., "A Microtasking Algorithm for Optimization of Structures", *International Journal of Supercomputer Applications*, Vol. 5, No. 2., 1991, pp. 81–90.

Khot, N. S., and Berke, L., "Structural Optimization Using Optimality Criteria Methods", Atrek, E., Gallagher, R. H., Ragsdell K. M., and Zienkiewicz O. C., Eds., *New Directions in Optimum Structural Design*, John Wiley, New York, 1987, pp. 47–74.

Soegiarso, R. and Adeli, H., "Impact of Vectorization on Optimization of Large Structures", *Structural Optimization*, Vol. 7, No. 1, 1994, pp. 117–125.

UBC, *Uniform Building Code*, Vol. 2, International Conference of Building Officials, Whittier, California, 1994.

Venkayya, V. B., "Design of Optimum Structures," *Computers and Structures*, Vol. 1, 1971, pp. 265–309.

APPENDIX II. NOTATION

The following symbols are used in this paper

A_i = cross-sectional area of member i

A_i^L = lower bound for the cross-sectional area A_i

A_i^u = upper bound for the cross-sectional area A_i

$A_{k,j}$ = cross-sectional area at node k in layer j of the tree network

A_i = output cross-sectional area of node i from W shapes database

F_y = specific yield stress

F_{am} = allowable compressive stress from the AISC ASD specifications

F_{cm} = critical compressive strength from the AISC LRFD specifications

g_{ijk} = gradient of the jth displacement degree of freedom under the kth loading condition

k = number of loading condition

K_{im} = stiffness matrix of member m belonging to group i

L = number of loading conditions

L_{im} = length of member m belonging to group i

n = total number of layers in the tree network

N = number of degrees of freedom

N_{ac} = number of active displacement constraints

N_c = number of elements whose stress constraints are violated

N_d = number of design variables

N_m = total number of members in the structure

N_{mi} = number of members belonging to group i

N_s = the number of interstory drift constraints

r_i = governing radius gyration for member i

r_j^L = lower bound for displacement constraint at the jth degree of freedom

r_j^U = upper bound for displacement constraint at the jth degree of freedom

r_s^L = the upper bound on the interstory drift at the sth floor

r_s^U = upper bound on the interstory drift at the sth floor

u_{sk} = the maximum interstory drift on the sth floor

u_{imk} = displacement vector of member m belonging to group i due to the loading condition k
u_{jk} = displacement of the jth degree of freedom due to loading condition k
v_{imj} = virtual displacement vector of member m belonging to group i due to the application of a unit load in the direction of the jth degree of freedom
$W_{k,j}$ = weight of the kth node in layer j of the tree network
W = weight of the structure
λ_{jk} = positive Lagrange multiplier for displacement constraint associated with the jth constrained degree of freedom and the kth loading case
ζ = step size
ρ_i = unit weight of member i
σ_{mk} = stress in member m due to the loading condition k