Available online at www.sciencedirect.com

ScienceDirect

Engineering Structures 30 (2008) 197–205

ENGINEERING STRUCTURES

www.elsevier.com/locate/engstruct

Optimal load and resistance factor design of geometrically nonlinear steel space frames via tabu search and genetic algorithm

S.O. Degertekin[a], M.P. Saka[b], M.S. Hayalioglu[a,*]

[a] *Department of Civil Engineering, Dicle University, 21280, Diyarbakir, Turkey*
[b] *Department of Engineering Sciences, Middle East Technical University, 06531, Ankara, Turkey*

Received 7 August 2006; received in revised form 16 February 2007; accepted 22 March 2007
Available online 30 April 2007

Abstract

In this paper, algorithms are presented for the optimum design of geometrically nonlinear steel space frames using tabu search and genetic algorithm. Tabu search utilizes the features of short-term memory facility (tabu list) and aspiration criteria. Genetic algorithm employs reproduction, crossover and mutation operators. The design algorithms obtain minimum weight frames by selecting suitable sections from a standard set of steel sections such as American Institute of Steel Construction (AISC) wide-flange (W) shapes. Stress constraints of AISC Load and Resistance Factor Design (LRFD) specification, maximum drift (lateral displacement) and interstorey drift constraints, size constraints for columns were imposed on frames. The algorithms were applied to the optimum design of three space frame structures. The designs obtained using tabu search were compared to those where genetic algorithm was considered. The comparisons showed that the former algorithm resulted in lighter structures.

Keywords: Optimum design; Tabu search; Genetic algorithm; Steel space frames; Load and resistance factor design; Nonlinear analysis

1. Introduction

A large number of techniques and algorithms have been developed for the optimum design of structural systems in the last four decades. Most of the algorithms deal with continuous design variables and use mathematical programming techniques or optimality criteria approaches. However, the design variables are discrete in most practical design problems. This is due to the availability of standard sizes and their limitations for construction and manufacturing reasons.

A number of algorithms were reported for the optimum design of discrete structural systems [1–3]. Mathematical programming techniques were employed in all these algorithms.

A few articles deal with the optimum design of structures subjected to actual design constraints of code specifications [4–7]. Mathematical programming and optimality criteria methods with continuous design variables were used in all these articles.

Genetic algorithms (GAs), which are applications of biological principles into computational algorithms, have been used to solve optimum structural design problems in recent years. They apply the principle of survival of the fittest to the optimization of structures. They are also able to deal with discrete optimum design problems and do not need derivatives of functions, unlike classical optimization. However, the procedure for the genetic algorithm is time consuming and the optimum solutions may not be global ones, but they are feasible both mathematically and practically. Genetic algorithms have been employed to solve many structural optimization problems. They were used for the optimum design of planar and space trusses [8–10]. Optimum designs of planar and space frames were performed using genetic algorithms [11–15]. Genetic algorithms were also employed to obtain optimum design of semi-rigid steel frames under the actual constraints of design codes [16–18].

Tabu search (TS) is a combinatorial optimization method which is also suitable for discrete design variables. TS is a

* Corresponding author. Tel.: +90 412 248 8402; fax: +90 412 248 8405.
E-mail address: hsedat@dicle.edu.tr (M.S. Hayalioglu).

0141-0296/$ - see front matter
doi:10.1016/j.engstruct.2007.03.014

heuristic technique that prevents the procedure from getting trapped at local optimal solutions, and hence keeps searching for global optima. TS was developed by Glover [19–21] for solving combinatorial optimization problems. The probability of becoming entrapped into a local optima is prevented because TS uses artificial memory facility and records information about recent search moves and employs tabu list to forbid certain moves. However, the optimum solutions obtained by GA may not be global ones. The major difference between GA and TS is that the latter has an artificial memory which prevents the algorithm turning back to the old designs, whereas the former do not guarantee this situation. In addition, TS algorithm considers each design variable in the current design independently when it generates a new neighbourhood design. On the other hand, GA has to consider a design as a whole due to gene structure. Therefore TS has more flexibility to search and find global optima than GA. Although TS is comparatively simple, it has been applied to many different fields of engineering and technology successfully. For a broad information and various applications of TS, see Glover and Laguna [22,23]. The applications of TS to the structural optimization were only about the optimal design of planar and space trusses which behave linear elastically [24–30]. Any sophisticated design codes for the stress and other constraints are not used in those studies. Only simple stress and displacement constraints are imposed on the trusses.

The contribution of the present study is to show the applicability of TS to a sophisticated design code such as AISC–LRFD [31] which considers limited state conditions and second-order effects (geometrical nonlinearity) for the member strength together with multiple load combinations. Vertical and lateral loads were also taken from actual standards and codes [32,33]. Displacement and size constraints were also adopted in the optimal design of frames. Discrete design variables selected from the standard set of AISC wide-flange (W) shapes were also used. A TS algorithm was developed which is similar to the one described in [25] but somewhat different from that regarding terminating criteria.

Moreover, a genetic algorithm was also developed and optimum designs of three steel space frames were performed using the TS algorithm and the GA. The results obtained from two algorithms were compared to each other.

2. Tabu search

TS is an optimization method which finds optimum solution by neighbourhood search in the solution space. A constrained optimization problem consists of constraints to be satisfied and an objective function whose minimum value is searched. Objective function is composed of design variables. Design variables are selected from a list of discrete variables that each of them is represented by a sequence number in that list.

First an initial design is generated randomly. A variable of this design is also selected randomly and various designs are obtained by changing only that variable in the range of a predetermined neighbourhood depth. For example, if the neighbourhood depth is determined as ± 2, four different designs are obtained by exchanging the selected variable with two upper and lower variables in the sequence of the list. Let us consider an initial design with three design variables: 20, 45, 60. These numbers represent the sequence numbers in a determined list of discrete variables. Let the second variable (no. 45) be selected randomly for perturbation. Four different neighbourhood designs obtained for the neighbourhood depth ± 2 are as follows: (20, 46, 60); (20, 47, 60); (20, 44, 60); (20, 43, 60). The best of the four designs is found (the best design is the one with the lowest objective function value). Assume that the best design of the four is the last one (20, 43, 60). Meanwhile, the move (design variable) which determines the best design is recorded in a one-dimensional list called "tabu list". (43-rd variable is recorded in tabu list). The other design variables of the best design are also checked whether they are in the tabu list or not. This design is replaced with the current design even if a design variable of it is not in the tabu list and the process continues starting with the new current design. The other design variables are also selected randomly and the same process is applied to each of them. An iteration is completed when all design variables are considered. For example, if the third variable is selected randomly the new designs become: (20, 43, 61); (20, 43, 62); (20, 43, 59); (20, 43, 58). Let us now assume the first one (20, 43, 61) is the best design. The 61-st variable is then put into the tabu list as its second element. The process starts with (20, 43, 61) again. The first variable is perturbed lastly and the new designs are obtained as (21, 43, 61); (22, 43, 61); (19, 43, 61); (18, 43, 61). Assume that the third design is the best one. The 19-th variable is also recorded in the tabu list thirdly. Thus, the first iteration is completed. The second iteration starts with the current design (19, 43, 61). The best of the neighbourhood designs is recorded in a list with single member if it satisfies all the constraints. This list is called "aspiration list". The aspiration list is updated throughout the iterations when a better feasible design is encountered. During the search process, even if all the variables of a best neighbourhood design are in the tabu list, its tabu status is temporarily ignored providing that it is a better design than the one in the aspiration list and satisfies all the constraints. These three conditions are called "aspiration criteria". This design is accepted as new current design and also put into the aspiration list. This design is rejected if it does not satisfy the aspiration criteria. Tabu list is a one-dimensional array whose size is kept constant during the search process. For this reason, when the tabu list is filled the oldest move at the beginning of the list is dropped and a new move is put into the end of the list.

In the present work, two terminating criteria were adopted for TS. The first one stops the optimization process when a predetermined total number of iterations is performed. The second criterion stops the process before reaching the maximum iteration number, if the same design remains in the aspiration list during a definite number of iterations. This criterion saves on computing time which is quite long due to the feature of nonlinear analysis as well as multiple loading of LRFD. When the process stops by one of the aforementioned criteria, the design in the aspiration list at the end of the last iteration is accepted as the optimum design.