# Chapter 2

# Literature Review

## 2.1 Structural Design Optimization

In structural design optimization, an unsafe structure can be obtained if the design is of too low weight. Therefoie, the objective of the structural design optimization is to obtain the most economical design which satisfies all the constraints. To solve optimization problems, many methods in the field of structural optimization have been created. Some examples are as follows.

The old-school technique, gradient-based search, uses the concept of searching the maximum or minimum point of a function in calculus. This method starts solving the optimization problem with the evaluation of the stresses and displacements of the structure. This calculation can be done by first assuming the size of the members. After that, by considering the stress and displacement calculations of the previous structure, each design variables are disturbed in order to obtain a new and better structure. These steps are repeated until a satisfactory result is obtained. However, the first step of assuming the size of member is one of the most important steps. The result obtained might be of low quality if the assumed starting structure is too far from the optimum point. Therefore, experiences are required in order to effectively use this technique.

Additionally, the gradient-based optimization algorithm also has other drawbacks. The solution obtained from this method can be trapped at a local optimum point. Moreover, the method itself is actually suitable for problems with continuous design variables. The solution obtained from optimization with continuous variables is hardly practical in real world construction since, in most cases, there are always only discrete sections available in the market. Therefore, at present, this method is not popular in solving structural optimization problems and hence other methods are being applied to structural optimization problems instead.

Another technique, Complex-Simplex Procedure, was proposed to solve discrete frame optimization by Fu and Levey (1978). This technique combines the complex method and simplex method together. The complex method is used to search the optimum and the simplex method is used to evaluate the feasibility. The design of structural frame is formulated as an optimization problem of linear function subject to linear constraints. However, in their study, the influence of axial loads was neglected and the deflection constraint is not considered in the optimization procedure. Another method, Optimality Criterion Method, was used to solve frame structures with members having general cross-sectional relationship by Sadek (1992). This design procedure has been developed for stress and displacement-constrained structural problems. The method seems to be versatile and valid to various types of structure.

Direct Search Methods was used to solve multi-bay multi-storey steel frames by Thevendran et al. (1992). In their study, the numerical optimization techniques are used to solve the nonlinear constrained minimization problems. The standard steel I-sections are used as available sections in their study.

Evolution Strategies and Neural Networks were also used to solve structural optimization problems by Papadrakakis et al. (1998). Both shape and sizing structural optimization problems are investigated in their study. Later on, the Adaptive Evolution Strategy was proposed and used to solve large-scale sizing optimization problems by Lagaros (2002).

Another algorithm, Simulated Annealing, was used to solve layout optimization of trusses by Hasançebi and Erbatur (2002). Their study aims to obtain the optimum size, shape and topology of simply supported truss structures under a set of particular constraints.

Even though there are many algorithms for solving structural optimization problems, there is one well-known technique, which many researchers use with a great degree of success, called the Genetic Algorithms (GAs). GAs have been used to solve many types of structural optimization problems such as shape and topology design optimization of trusses (Rajan, 1995), sizing optimization of truss or frame problems (Rajeev and Krishnamoorthy, 1992; Wu and Chow, 1995; Galante, 1996; Camp et al., 1998 and Nanakorn and Meesomklin, 2001). The technique is now very popular and hence selected as a benchmark algorithm in this study.

The genetic algorithms (GAs) are inspired by the basic mechanisms of natural evolution. Genetic algorithms solve problems by simulating the natural evolution processes or, in other words, by the reproduction of solutions. They are also efficient for searching for the solution from the entire search space or global-searches. GAs employ the Darwinian survival-of-the-fittest theory to yield the better characters among the old population and performs a random information exchange to create a better-quality offspring. Simple genetic algorithms use only three genetic operators namely, reproduction, crossover and mutation. The mechanism of genetic algorithms is simple. However, they also have some drawbacks. For example, GAs always have to work with the binary strings instead of real parameter sets. This is inconvenience in some real-world problems. The ACO algorithm does not have this drawback. For more information related to GAs, see, for example, Goldberg (1989), Rajeev and Krishnamoorthy (1992), Rajan (1995) and Nanakorn and Meesomklin (2001).

Genetic Algorithms might have been one of the best tools for researchers in the field of optimization for the last decade. However, many new methods, including the proposed ACO algorithm, still have chances to show their capabilities in solving structural design optimization problems.

It can be seen that there are many types of problem, many types of algorithm and also many conditions in solving structural design optimization problems. However, the discrete structural design optimization with stress and/or displacement constraints is considered in this research. This is because the stress and/or displacement constraints are generally considered in the real optimization problems.

## 2.2 Ant Colony Optimization

The ACO algorithm is also inspired by a natural process. The basic idea of the algorithm comes from the foraging behavior of real ants. Real ants can find the shortest path to a food source by using the pheromone trail-laying and pheromone trail-following mechanisms. The ACO technique solves problems by simulating this natural behavior of ants while they find their shortest path to the food source. Artificial ants in the algorithm will repeatedly walk on different paths which are considered as different solutions. The path or solution with high quality will receive more pheromone by artificial ants. In the next round of walking, artificial ants will search for better solutions by looking for the path that has high level of pheromone. This process can be done repeatedly until the satisfactory result is obtained. Since the ACO technique utilizes information from many search points at the same time, there is less chance for the search to be trapped in any local optima points. Hence, the ACO technique can be considered as a global-search algorithm.

Recently, the ACO technique is becoming popular among researchers in the field of heuristic optimization (see, for example, a survey in Dorigo et al., 1999). The problem that seems to fit the technique naturally is the traveling salesman problem (Dorigo and Gambardella, 1997). The traveling salesman problem (TSP) is the problem of finding the shortest closed tour which visits all the cities in a given set. Because of similarities between the TSP and food foraging behavior of ant colonies, the ACO is hence suitable for solving this problem.

The ACO technique has been applied to other various types of problem, such as the real-time control of intelligent network traffic (Jennings et al., 1999), frequency assignment problem (Maniezzo and Carbonaro, 2000), optimization problems for designing and scheduling of batch plants (Jayaraman et al., 2000), quadratic assignment problem (Talbi et al., 2001), the just-in-time sequencing problem (McMullen, 2001), two-machine flowshop scheduling problem (T'kindt et al., 2002), weapon-target assignment problem (Lee et al., 2002), scheduling continuous casting of aluminum (Gravel et al., 2002), optimization of the keyboard arrangement (Eggers et al., 2003), solving the mesh-partitioning problem (Korošec et al., 2004), etc. Nevertheless, the application of the technique in the field of civil engineering is still rare (see, for example, Abbaspour et al., 2001).