Chapter 5

Optimal Structural Elements Sizing Using Neural Network and Adaptive Differential Algorithm

Wonsiri Punurai
Mahidol University, Thailand

Nantiwat Pholdee
Khon Kaen University, Thailand

ABSTRACT

Engineering design is one of many disciplines involving optimization problems. This chapter will introduce differential evolution (DE), as well as educate readers on an efficient adaptive differential evolution algorithm (ADEA). The ADEA is considered an effective method for complex structural optimization problems such as truss structures. First, the chapter will provide an overview on the DE and ADEA concepts. Using illustrative examples, the chapter will provide the reader with a methodology used to perform 2D and 3D truss sizing optimization. The remainder of the chapter will discuss the computational aspect of neural network assisted adaptive differential algorithm for truss optimization design. Future work can show the benefits of this adaptation to accelerate the convergence speed while using fewer structural analyses than those required by other methods.

INTRODUCTION

Optimization is one of three specific core concepts in the engineering design process. By applying appropriate design constraints and criteria, results found in the optimization process help the designer deliver valuable preliminary conclusions. These conclusions lead to positive solutions while avoiding repetitive evaluation of a trial and error design.
Over the past several decades, many optimization techniques have been developed (Dieu Ngoc & Peter, 2013; Erol & Eksin, 2006; Geem, Kim, & Loganathan, 2001; Hindriyanto Dwi & Hui-Ming, 2014; Hossein & Fatemeh, 2016; Imran, Pandian, Balbir Singh Mahinder & Abdullah-Al-Wadud, 2016; Jia, Wang, Cai, & Jin, 2013; Karaboga & Basturk, 2007; Kaveh & Talatahari, 2010; Khatib & Fleming, 1998; Rao, Savsani, & Vakharia, 2011; Rashedi, Nezamabadi-pour, & Saryazdi, 2009; Socha & Dorigo, 2008; Storn & Price, 1997; Sotirios, Katherine& John, 2016; Tan & Zhu, 2010; Teh & Rangaiah, 2003; Venter & Sobiesczanski-Sobieski, 2003; Yang & Deb, 2010; Yang & Gandomi, 2012). Mathematical programming (MP) and optimality criteria (OC) are two of the earliest methods. Some of these classical methods rely on gradient information (i.e., derivative of the objective function, with respect to design variables). These have potential drawbacks, including trapping in local optimum and requiring high computational effort. Metaheuristic and evolutionary algorithms are more recent and efficient. These latter approaches are experience-based optimization (sometimes called population-based). The methods have no requirement for a differentiable or continuous objective function. Additionally, they make little or no assumptions about the problem being optimized and have global search capability of feasible candidate solutions. The best-known names of these methods include: genetic algorithm (GA) (Holland, 1975); DE (Storn & Price, 1997); particle swarm optimization (PSO) (Venter & Sobiesczanski-Sobieski, 2003); ant colony optimization (ACO) (Socha & Dorigo, 2008); tabu search (TS) (Teh & Rangaiah, 2003); and simulated annealing (SA) (Hwang, 1988).

The GA was introduced by Holland in the 1970s (Holland, 1975) and further applied by Goldberg in the 1980s (Goldberg, 1983; Goldberg & Holland, 1988; Goldberg, Korb, & Deb, 1989). The GA principle is based on Darwin’s theory of evolution. In the GA scheme, an initial set of possible solutions is randomly created in the form of binary strings. Each string represents one feasible solution to the problem being solved. The solution strings are converted into decimal equivalents and evaluated on their fitness through the inverse of the cost function. Next, the natural random processes of crossover, mutation, and selection take place to form the next generation of solution strings. These operations repeat; the algorithm is geared toward maximizing the fitness function parameter. The algorithm is stopped when the average fitness of the population exceeds some fraction of the best fit in the population.

The DE algorithm is similar to the GA’s approach. The algorithms are based on the principle of natural selection. They were first developed by Storn and Price (1997) to minimize nonlinear and nondifferentiable continuous space functions with real-valued parameters. DE works with a population of individuals considered candidates of solution in the feasible domain. Each candidate solution is real numbers; the population is seen as a real matrix. The population is developed iteratively using evolutionary operators of selection, mutation, and crossover. In the DE method, the difference between two vectors yields a difference vector which is then used with a scaling factor to guide the search space. The DE application can be found in many research studies, including neural network learning (Bas, 2016; Chauhan, Ravi, & Chandra, 2009; Ionen, Kamarainen, & Lampinen, 2003; Magoulas, Plagianakos, & Vrahatis, 2004; Slowik & Bialko, 2008; Wang, Zeng, & Chen, 2015), digital signal and image processing (Aslantas, 2009; Baştürk & Güney, 2009; Bazi et al., 2014; Zhong & Zhang, 2012), and design applications of truss structures and laminated composite plates (Bureerat & Pholdee, 2016; Ho-Huu, Do-Thi, Dang-Trung, Vo-Duy, & Nguyen-Thoi, 2016; Kitayama, Arakawa, & Yamazaki, 2011; Le-Anh, Nguyen-Thoi, Ho-Huu, Dang-Trung, & Bui-Xuan, 2015; Pholdee & Bureerat, 2013; Punurai, Nantayatron & Pholdee, 2013; Roque & Martins, 2015; Wu & Tseng, 2010).
Related Content

Towards HIPAA-Compliant Healthcare Systems in Cloud Computing
[www.igi-global.com/article/towards-hipaa-compliant-healthcare-systems/72873?camid=4v1a](www.igi-global.com/article/towards-hipaa-compliant-healthcare-systems/72873?camid=4v1a)

Pure and Hybrid Metaheuristics for the Response Time Variability Problem
[www.igi-global.com/chapter/pure-hybrid-metaheuristics-response-time/69889?camid=4v1a](www.igi-global.com/chapter/pure-hybrid-metaheuristics-response-time/69889?camid=4v1a)

A Partial Optimization Approach for Privacy Preserving Frequent Itemset Mining
[www.igi-global.com/chapter/partial-optimization-approach-privacy-preserving/65149?camid=4v1a](www.igi-global.com/chapter/partial-optimization-approach-privacy-preserving/65149?camid=4v1a)

A Smart Structural Algorithm (SSA) Based on Infeasible Region to Solve Mixed Integer Problems
[www.igi-global.com/article/a-smart-structural-algorithm-ssa-based-on-infeasible-region-to-solve-mixed-integer-problems/171461?camid=4v1a](www.igi-global.com/article/a-smart-structural-algorithm-ssa-based-on-infeasible-region-to-solve-mixed-integer-problems/171461?camid=4v1a)