An Improved Genetic Local Search Algorithm for Defect Reconstruction from MFL Signals

W. Han and P. Que
Department of Information Measurement Technology and Instruments, Shanghai Jiaotong University, Shanghai, China
Received November 7, 2005

Abstract—This paper presents an improved genetic local algorithm by incorporating the simulated-annealing technique into the perturbation process of the genetic local search algorithm and proposes an improved-genetic-local-search-algorithm-based inverse algorithm for two-dimensional defect reconstruction from the magnetic-flux-leakage signals. In the algorithm, a radial-basis-function neural network is utilized as a forward model, and the improved genetic local search algorithm is used to solve the optimization problem in the inverse problem. Experiments are presented to compare the proposed inverse algorithm with both the canonical-genetic-algorithm-based inverse algorithm and the genetic-local-search-algorithm-based inverse algorithm. The results demonstrate that the proposed inverse algorithm is more accurate and robust to the noise.

INTRODUCTION

The magnetic-flux-leakage (MFL) method has established itself as the most widely used in-line inspection technique for the evaluation of gas-and-oil pipelines. One important challenge in MFL NDE is the determination of the defect parameters such as the length, width, or defect shape on the basis of the information contained in the measured signals. Iterative methods are commonly used approaches for solution of inverse problems (see [1] and references therein). These methods involve solving a well-behaved forward problem in a feedback loop. Traditionally, numerical models such as the finite-element model (FEM) have been used to represent the forward process. However, iterative methods using the numerical-based forward models are computationally expensive. Neural networks are utilized for solving inverse problems in NDE [1–4] and used to represent the forward process in iterative methods [1]. Huang et al. [3] described the use of a wavelet-basis-function neural network to predict three-dimensional defect profiles. Ramuhalli et al. [4] used two neural networks in feedback configurations. In this method, a forward network is used as the forward model, and an inverse network is used to predict the profile for given measured MFL signals. Ramuhalli et al. [1] proposed a neural-network-based iterative inversion algorithm using neural networks as the forward model. In the algorithm, the problem of defect profile reconstruction from MFL signals is formulated as an optimization problem, where the defect profile is updated using a combination of gradient descent and simulated annealing to minimize the error between the measured signal and the model-predicted signal. Li et al. presented a genetic local search algorithm (GLSA) for reconstructing the profiles of 3D defects from eddy-current NDE signals [5].

This paper presents an improved genetic local search algorithm (IGLSA) by incorporation of the simulated-annealing technique into the perturbation process of the GLSA and proposes an IGLSA-based inverse algorithm for 2D defect reconstruction from MFL signals. In the algorithm, a radial-basis-function neural network (RBFNN) is utilized as the forward model, and the IGLSA is used to solve the optimization problem in the inverse problem. Experiments are presented to show the performance of the IGLSA-based inverse algorithm and the GLSA-based inverse algorithm, respectively. The results demonstrate that IGLSA-based inverse algorithm is more accurate and robust to noise.

REVIEW OF CANONICAL GENETIC ALGORITHM

The genetic algorithm is a search mechanism based on the principle of natural selection and population genetics and has been widely applied in a number of different areas [6, 7]. Instead of a point-by-point search, the genetic algorithm offers a parallel search of the solution space rather than a single region. Hence, the
genetic algorithm can find a near-global optimal solution and avoid the local minimum solution possibly encountered in the gradient-descent-based optimization approach. A canonical genetic algorithm (CGA) usually begins with a randomly generated set of potential solutions, called the initial population. Each member in the population represents a possible solution of the problem. A fitness function is used as a measure of the closeness of each member in the population to the global optimum solution. Then, members from the population are selected to produce new members, called descendents or children, by applying stochastic operators to the selected members. The selection mechanism often favors the highly fit members in such a way that the members more close to the global optimal solution are assigned higher probabilities for producing children. The most common genetic operators are crossover and mutation. A crossover operator combines the features presented in parents to produce descendents. Crossover methods include single-point crossover, two-point crossover, and uniform crossover. A mutation operator slightly perturbs a selected member in a random manner. Typical mutation methods include Gaussian mutation and uniform mutation. A new generation is then formed using the descendents to replace part of or all the members in the current population. Crossover operations ensure that the new population inherits highly fit features, while mutation operations may add previously unexploited features into the population. It is hoped that, in doing so, the population would very likely drift to a global or near-global solution after a number of generations. This search process is called genetic evolution. Genetic-algorithm-based optimization approaches involve consideration of five key issues: parameter representation, measure of the fitness of potential solutions, an initial population, genetic operators to update the population, and a termination criterion. The CGA can be summarized as follows:

(i) Randomly build an initial population.
(ii) Create a sequence of new populations, or generations. At each step, the algorithm uses the individuals in the current generation to create the next generation. To create the new generation, the algorithm performs the following steps:
   (a) Score each member of the current population by computing its fitness value.
   (b) Scale the raw fitness scores to convert them into a more usable of values.
   (c) Select parents on the basis of their fitness by a roulette-wheel selection method.
   (d) Produce children from the parents. Children are produced by performing uniform crossover and by performing Gaussian mutation.
   (e) Replace the current population with the children to form the next generation.
(iii) The algorithm stops when the stopping criterion is met. The average fitness value of the population is used as the stopping criterion. The algorithm stops when the average fitness value of the population is greater than a predefined threshold. Otherwise, return to step (ii).

AN IMPROVED GENETIC LOCAL SEARCH ALGORITHM

The simple canonical genetic algorithm may be trapped by local optima or converge prematurely during the solution of the optimum of large-scale combinatorial optimization problems. The premature convergence can be prevented by preservation of the population diversity during the search. Various methods have been developed to prevent the simple CGA to rapidly concentrate their population to a single point in the search space [5, 8, 9]. The GLSA [5] casts the simulated-annealing method into the frame of the CGA to prevent the premature convergence. The GLSA utilizes the probabilistic acceptance test technique of the simulated annealing, and its basic idea is briefly reviewed in the following. The simulated-annealing method is a common local search technique. It first randomly selects a member from the current population. Then, the selected member is perturbed in its neighborhood. The new member is accepted probabilistically according to the threshold probability defined by

\[ p_t = p_0 e^{\frac{k \Delta F}{T}}, \]  

where \( k \) is the generation number, \( \Delta F \) is the increase of the fitness after the perturbation, and \( p_0 \) and \( T \) are positive constants. If \( \text{Rand}(0, 1) \leq p_t \), the perturbed member is accepted and added to the current population to replace a selected member. Otherwise, it is discarded. \( \text{Rand}(0, 1) \) generates a random real number between 0 and 1.

At first, \( k = 0 \), then \( p_t = p_0 \). \( p_0 \) is the initial threshold probability. If \( \Delta F > 0 \), the perturbed member is closer to the global optima and \( p_t > p_0 \) according to (1). If \( \Delta F < 0 \), \( p_t < p_0 \). As a result, perturbations that increase the fitness will have much higher probabilities of being accepted than those that decrease the fit-