Acta Automatica Sinica
Vol. 33, No. 1 January, 2007
DOI: 10.1360/aas-007-0091

Hybrid Simplex-improved Genetic Algorithm for Global Numerical Optimization

REN Zi-Wu[1] SAN Ye[1] CHEN Jun-Feng[2]

Abstract In this paper, a hybrid simplex-improved genetic algorithm (HSIGA) which combines simplex method (SM) and genetic algorithm (GA) is proposed to solve global numerical optimization problems. In this hybrid algorithm some improved genetic mechanisms, for example, non-linear ranking selection, competition and selection among several crossover offspring, adaptive change of mutation scaling and stage evolution, are adopted; and new population is produced through three approaches, i.e. elitist strategy, modified simplex strategy and improved genetic algorithm (IGA) strategy. Numerical experiments are included to demonstrate effectiveness of the proposed algorithm.

Key words Genetic algorithm, simplex method, competition and selection,mutation scaling

1 Introduction

Genetic algorithm (GA) is a stochastic and parallel search technique based on the mechanics of natural selection, genetics and evolution, which was first developed by Holland in 1970s[1]. In recent years, GA has been widely applied to different areas such as fuzzy systems, neural networks, *etc.*[2] Although GA has become one of the popular methods to address some global optimization problems, the major problem of GA is that it may be trapped in the local optima of the objective function when the problem dimension is high and there are numerous local optima. This degradation in efficiency is apparent especially in applications where the parameters being optimized are highly correlated[3]. In order to overcome these flaws and improve the GA′s optimization efficiency, recent research works have been generally focused on two aspects. One is improvements upon the mechanism of the algorithm, such as modification of genetic operators, or the use of niche technique[4], *etc*; the other is combination of GA with other optimization methods, such as BFGS methods[5], simulated annealing (SA), *etc.*

In this paper, a hybrid simplex-improved genetic algorithm (HSIGA) is proposed to solve global numerical optimization problems. In this hybrid algorithm some improved genetic mechanisms are adopted, such as non-linear ranking selection, competition and selection among several crossover offspring, adaptive change of mutation scaling and adaptive stage evolution mechanism, to form an improved genetic algorithm (IGA). For further performance enhancement, the IGA algorithm is combined with the simplex method (SM) and the new population is generated through three approaches, *i.e.* elitist strategy, simplex strategy and IGA strategy. We investigate the effectiveness of this proposed algorithm by solving 10 test functions with high dimensions.

2 Hybrid simplex-improved genetic algorithm (HSIGA) for numerical optimization

In this paper, the following minimization problem with fixed boundaries is considered

$$\begin{aligned} &\text{minimize} && f(\boldsymbol{x}) = f(x_1, x_2, \cdots, x_n) \\ &\text{subject to} && x_i^{\min} \leqslant x_i \leqslant x_i^{\max} \quad (i = 1, 2, \cdots, n) \end{aligned} \tag{1}$$

where $\boldsymbol{x} = (x_1, x_2, \cdots, x_n)$ is a variable vector in $\boldsymbol{R}^n$, $f(\boldsymbol{x})$ denotes the objective function, and $x_i^{\min}$ and $x_i^{\max}$ represent the lower and the upper bounds of x_i such that the meaningful range of x_i is $[x_i^{\min}, x_i^{\max}]$.

2.1 Improved genetic algorithm (IGA)

For the minimization problem like (1), we adopt real-code GA and firstly introduce IGA.

1) **Non-linear ranking selection operator**

In order to select some excellent chromosomes from the parent generation, non-linear ranking selection is adopted in this paper, which maps chromosome′s serial number in the queue to an expected selection probability.

With the population $pop = \{\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots \boldsymbol{x}_i, \cdots, \boldsymbol{x}_P\}$ of P chromosomes, we distribute the probability to each chromosome from the best to the worst by a non-linear function, so the selection probability of chromosome $\boldsymbol{x}_i$ is

$$\begin{cases} p(\boldsymbol{x}_i) = q'(1-q)^{i-1} \\ q' = \dfrac{q}{1-(1-q)^P} \end{cases} \tag{2}$$

where q is the selection probability of the best chromosome, i is the serial number of the chromosome.

After the selection probability of each chromosome is determined, the roulette wheel selection is adopted to select the excellent chromosome. Ranking selection need neither use the individual′s fitness nor transform the fitness scaling, which can prevent the premature convergence or stagnation phenomenon to a certain extent.

2) **Competition and selection among several crossover offspring**

In the natural evolution, parents often reproduce more than two offspring after crossover operation, and competition phenomenon between offspring of the same parents also always occurs. Illumined by this idea, competition and selection of the excellent among the several offspring is employed in the crossover operator. Different from the crossover operation of the simple genetic algorithm (SGA), four chromosomes are created firstly from the parents $\boldsymbol{x}_s = [x_1^s, x_2^s, \cdots, x_n^s]$ and $\boldsymbol{x}_t = [x_1^t, x_2^t, \cdots, x_n^t]$

Received June 17, 2005; in revised form April 26, 2006
Supported by National Natural Science Foundation of P. R. China (60474069)
1. Control & Simulation centre, Harbin Institute of Technology, Harbin 150080, P.R.China 2. College of Computer & Information Engineering, Hohai University, Changzhou 213022, P.R. China

$(s \neq t)$ according to the following formulae[2].

$$\boldsymbol{y}_1 = [y_1^1, y_2^1, \cdots, y_n^1] = \frac{\boldsymbol{x}_s + \boldsymbol{x}_t}{2} \tag{3}$$

$$\boldsymbol{y}_2 = [y_1^2, y_2^2, \cdots, y_n^2] = \boldsymbol{x}_{\max}(1-\omega) + \max(\boldsymbol{x}_s, \boldsymbol{x}_t)\omega \tag{4}$$

$$\boldsymbol{y}_3 = [y_1^3, y_2^3, \cdots, y_n^3] = \boldsymbol{x}_{\min}(1-\omega) + \min(\boldsymbol{x}_s, \boldsymbol{x}_t)\omega \tag{5}$$

$$\boldsymbol{y}_4 = [y_1^4, y_2^4, \cdots, y_n^4] = \frac{(\boldsymbol{x}_{\max} + \boldsymbol{x}_{\min})(1-\omega) + (\boldsymbol{x}_s + \boldsymbol{x}_t)\omega}{2} \tag{6}$$

$$\boldsymbol{x}_{\max} = [x_1^{\max}, x_2^{\max}, \cdots, x_n^{\max}] \tag{7}$$

$$\boldsymbol{x}_{\min} = [x_1^{\min}, x_2^{\min}, \cdots, x_n^{\min}] \tag{8}$$

where $\omega \in [0, 1]$ is a weight, $\max(\boldsymbol{x}_s, \boldsymbol{x}_t)$is the vector with each element obtained by taking the maximum among the corresponding element of $\boldsymbol{x}_s$ and $\boldsymbol{x}_t$. For example, $\max([1\ \ -2\ \ 3], [2\ \ 3\ \ 1]) = [2\ \ 3\ \ 3]$. Similarly, $\min(\boldsymbol{x}_s, \boldsymbol{x}_t)$ denotes a vector obtained by taking the minimum value. Among these 4 chromosomes, the two with superior fitness value are taken as the offspring of the crossover operation. It can be seen that the potential offspring can spread over the domain, and this crossover operator is superior to the single arithmetic crossover or heuristic crossover.

3) **Adaptive change of mutation scaling**

Different from the uniform mutation, boundary mutation, *etc*, a mutation operator with adaptive change of mutation scaling is employed. According to "the punctuated equilibrium theory"[6] in the evolution field, species evolution always appears in many and different directions at previous stage while the evolution tends to be conservative at later stage. Therefore, a larger mutation range is employed during previous stage to keep the population diversified while the mutation range will be shrunken gradually during later stage to focus on local search.

Supposing that the mutation scaling is μ $(0 \leqslant \mu \leqslant 1)$, the element x_k $(x_k \in [x_k^{\min}, x_k^{\max}])$ selected in the chromosome $(x_1, x_2, \cdots, x_k, \cdots, x_n)$ is to be mutated with a certain mutation probability Pm. Then this original value x_k will be replaced by the mutated value x_k^{new} chosen from the range

$$x_k^{new} \in \Big\{\max(x_k - \mu\tfrac{x_k^{\max} - x_k^{\min}}{2}, x_k^{\min}),\ \min(x_k + \mu\tfrac{x_k^{\max} - x_k^{\min}}{2}, x_k^{\max})\Big\} \tag{9}$$

with a uniform probability.Based on the concept that the mutation scaling μ is decreasing gradually during the process, a monotonously decreasing function of the mutation scaling μ is built

$$\mu(\tau) = 1 - r^{(1-\frac{\tau}{T})^b} \tag{10}$$

where T is the number of generations, τ is the current iteration, and the weight $r \in [0, 1]$. From the formula (10) it can be seen that for a small value of weight r, the mutation scaling μ is near to one at the beginning of the optimization, and the mutation scaling μ will be decreasing down to zero as the run progresses.

4) **Adaptive strategy of stage evolution**

During the process of evolution, the diversity of population is descending. When the diversity decreases to a certain level, the algorithm searching is over[1]. Generally, at a previous stage larger crossover and mutation probability can work obviously, while at a later stage the crossover and mutation probability had better be smaller since the algorithm has entered into the local searching process. For the selection operator, it is a good idea to choose smaller selection pressure at the beginning, and adopt larger selection pressure later to promote local searching.

Based on this idea, a model based on stage evolution is developed. We divide the whole process into 3 stages:

First stage:	$\tau \in [0, T_1]$	$T_1 = \alpha T$
Second stage:	$\tau \in (T_1, T_2]$	$T_2 = (1-\alpha)T$
Third stage:	$\tau \in (T_2, T]$	

where T and τ have been defined as above, generally parameter α is equal to 0.382. Then we choose three different best individual′s selection probability q, crossover probability Pc and mutation probability Pm for each stage.

2.2 Hybrid simplex-improved genetic algorithm (HSIGA)

In order to improve the local fine tuning capability of GA and quicken the convergence rate, we combine the IGA with the simplex method (SM) to form a hybrid optimization algorithm[7]. The detailed process is as follows.

All chromosomes in the current generation are arranged from the best to the worst firstly, and new population in the next generation is generated through the following three approaches.

1) Elitist strategy: The first N top-ranking chromosomes (elites) are reproduced directly into the next generation so that these elites can not be destroyed by the 3 operations of the GA or other operations.

2) Modified simplex strategy: In this HSIGA algorithm, the S $(S > N)$ top members in population produce S-N new chromosomes through the modified simplex method. In modified simplex method, the new generated chromosome is generated by reflecting $\boldsymbol{x}_j$ over the centroid $\boldsymbol{x}_c$ of the remaining points as follows.

$$\boldsymbol{x}_j^{new} = \boldsymbol{x}_c + \alpha(\boldsymbol{x}_c - \boldsymbol{x}_j) \quad (j = N+1, \cdots, S) \tag{11}$$

where the centroid is equal to $\boldsymbol{x}_c = (\boldsymbol{x}_1 + \boldsymbol{x}_2 + \cdots + \boldsymbol{x}_N)/N$, α is a random value.

3) Improved genetic algorithm (IGA) strategy: The remaining P-S children (where P is the population size) in the new generation are created through the IGA acting on the whole population.

Fig.1 depicts the architecture of this HSIGA algorithm. We can refer to the hybrid degree (S/P) by using the percentage of population to which the modified simplex operator is applied. From it we can see that the hybrid algorithm will become a real-code IGA when the hybrid degree (S/P) is zero; while the hybrid degree (S/P) is equal to 100%, the algorithm will turn into the modified simplex method. Generally S is around 20 percent of the size P.

The new population in HSIGA is produced through these 3 strategies with the following advantages.

1) In GA the coding of elitists would be changed or destroyed after the genetic operation, and the produced offspring may be less fitness than their parents. Elitist strategy is an effective approach to avoid the damage of the elitists, which is necessary to enhance the capacity of the algorithm.

2) Some novel genetic operations are used in IGA, such as the crossover and the mutation operator. The idea of these operations is mainly from the nature. The genetic mechanisms try to mimic the maturing phenomenon in nature, which makes the individuals more suitable to the environment, and enhances the optimization performance.

3) GA is global search procedure. It is less likely to be trapped in local optima, but the convergence rate will slow down and the computational cost is high at later stage. SM