An Ant Colony Genetic Algorithm

Shao Xiaowei and **Shao Changsheng** and **Zhao Changan**[1]

Abstract In this paper, a hybrid genetic algorithm called *ant colony genetic algorithm* is presented. The initial population of the method is generated from the each subspace which is divided from the feasible solution space of the optimization problem evenly. And every subspace is marked by initial pheromone. During the genetic operation, selection operator is under the effect of subspace's pheromone remaining. Because the initial population of points are scattered uniformly over the feasible solution space, so that the algorithm can evenly scan the feasible solution space once to located good points for further exploration in subsequent operator. In addition, the effect of pheromone of each subspace in the selection can improve convergence speed of the algorithm.

1 INTRODUCTION

Genetic algorithms (GAs) are randomized search and optimization techniques guided by the principles of evolution and natural genetics, have a large amount of implicit parallelism. To improve convergence speed and deal with premature convergence, many Hybrid Genetic Algorithms (HGA) have been presented.

On the basis of the fine optimizations carried out in a real ant colony. Coloni A and Dorigo M presented the ant colony system (ACS) in 1991[1]. The key features of an ant colony system include distributed computation, positive feedback, and constructive greedy heuristic. The algorithm has emerged as a new heuristic to solve many stochastic combinatorial optimization problems.

In this paper, a Hybrid Genetic Algorithm called *Ant Colony Genetic Algorithm* (ACGA) is presented. In particular, we propose the following two enhancements.

1) Generate the initial population from the subspace which is divided from the feasible solution space of the optimization problem evenly. So as the algorithm iterates and improves the population of points, some points may move closer to the global optimum.

2) Mark each subspace with pheromone. During the genetic operation, make the pheromone remaining to be an affection factor of selection operator, so that the population can converge into a subspace which contains the global optimum faster in the further evolution.

2 ANT CONLONY GENETIC ALGORITHM

Problem Definition

Maximize $f(x)$ Subject to $l \leqslant x \leqslant u$

Where $x = \{x_1, x_2, \ldots, x_n\}$, $x \in R^n$, $f(x)$ is the objective function, $l = \{l_1, l_2, \ldots, l_n\}$ and $u = \{u_1, u_2, \ldots, u_n\}$ define the feasible solution space. We denote the domain of x_i by $[l_i, u_i]$, and the feasible solution space by $[l, u]$. n is the dimension of the problem.

In ACGA, first, the solution space is divided into subspace evenly. And then initial population is generated from each subspace which is marked by initial pheromone. Finally use the extended genetic operators on the population until the termination conditions. The details are given in the following algorithm.

Setp 1: Divide the feasible solution space into subspace evenly.

Under the constraint of the dimension of problem n and the domain of each x_i we divide the feasible solution into subspace evenly. We use a two dimension example to introduce the process of ACGA.

Let $x = \{x_1, x_2\}$, the domain are $[l_1, u_1]$ and $[l_2, u_2]$. We quantize each variable into a finite number of values evenly. In particular, we quantize the domain $[l_1, u_1]$ and $[l_2, u_2]$.into M and N part distinguish. So the feasible solution space is divided into M×N subspace E_{ij}, $(i=1,2,\ldots,M; j=1,2,\ldots,N)$. and the length of each subspace is $D_{1L} = \frac{u_1 - l_1}{M}$ 和 $D_{2L} = \frac{u_2 - l_2}{N}$, the left and right boundary of each subspace are x_{1iL}、x_{2jL} and x_{1iR}、x_{2jR}, where $x_{1iL}=l_1+(i-1)D_{1L}$, $x_{2iL}=l_2+(j-1)D_{2L}$; $x_{1iR}=l_1+iD_{1L}$, $x_{2iR}=l_2+jD_{2L}$。

Setp 2: Generate initial population and mark each subspace with initial pheromone

1. Generate initial population from each subspace： In subspace E_{ij},

generate both the values of x_{1i} and x_{2j} from left and right boundary of $[x_{1iL} \quad x_{1iR}]$ and $[x_{2jL} \; x_{2jR}]$ distinguishingly, and the individual $A_{ij}(1) = x_{1i}, x_{2j}$, where 1 denotes the individual is in the initial population. So the initial population can be expressed as $\{A_{ij}(1)\}$, $(i=1,2,\ldots,M; j=1,2,\ldots,N)$, and the

[1] Harbin Institute of Technology, Harbin, China. Email: xw_shao@hit.edu.cn

population scale is $M \times N$.

2. Mark subspace with initial pheromone: each subspace E_{ij} is marked by the initial pheromone Ph_{ij}. Ph_{ij} of E_{ij} is a necessary condition of Ant Colony Genetic Algorithm operation and it is defined by fitness $f(A_{ij}(1))$ of the individual $A_{ij}(1)$ which is generated from the subspace E_{ij}. if $f(A_{ij}(1)) > 0$, $Ph_{ij}(1) = C_1 f(A_{ij}(1))$, if $f(A_{ij}(1)) < 0$, $Ph_{ij}(1) = \frac{C_3}{C_2 - f(A_{ij}(1))}$, C_1, C_2, C_3 are positive constants according to the problem.

Setp 3: Ant colony genetic operation to the population.

1. Selection operator: Suppose in k generation, population size is $M \times N$, the fitness of individual l is $f(A_l(k))$, and pheromone remaining of subspace which individual l is stay in is $Ph_l(k-1)$. Probability of individual l which is selected is

$$P_l(k) = \frac{ph_l^{\alpha}(k-1) f^{\beta}(A_l(k))}{\sum_{l=1}^{M \times N} ph_l^{\alpha}(k-1) f^{\beta}(A_l(k))} \qquad (1)$$

where α is a parameter which determines the relative importance of pheromone remaining versus the individual selection, and β is a parameter which determines the relative importance of fitness versus individual selection.

2. Crossover operator: crossover operator can be selected from the operators which are in common use.

3. Mutation operator: mutation operator can be selected from the operators which are in common use.

4.Update pheromone: Pheromone updating is intended to allocate much more pheromone to the subspace which contents better point. The pheromone updating formula is meant to simulate the change in the amount of pheromone due to both the addition of new pheromone deposited by ants on the visited subspace and to pheromone evaporation. Pheromone is updated in every subspace according to

$$Ph_{ij}(k) = (1-\rho) Ph_{ij}(k-1) + Ph_{ij}'(k) \qquad (2)$$

Where $Ph_{ij}(k)$ is pheromone of subspace ij in generation k, $Ph_{ij}(k-1)$ is pheromone of subspace ij in generation k-1, $Ph_{ij}'(k)$ is pheromone remaining after ant colony genetic operation of subspace ij in generation k; $0<\rho<1$ is a pheromone decay parameter. Suppose A_{ijmax} is an individual which has the maximum fitness in subspace ij before generation k, and $A_{ijmax}(k)$ is an individual which has the maximum fitness after genetic algorithm operation in subspace ij and generation k. if $f(A_{ijmax}(k)) > f(A_{ijmax})$ then $A_{ijmax} = A_{ijmax}(k)$, else $A_{ijmax} = A_{ijmax}$, when $f(A_{ijmax}) > 0$, $Ph_{ij}'(k) = C_1 f(A_{ijmax})$, when $f(A_{ijmax}) < 0$, $Ph_{ij}'(k) = \frac{C_3}{C_2 - f(A_{ijmax})}$, C_1, C_2, C_3 are positive constants according to the problem. In generation k, if there's no individual in subspace ij, A_{ijmax} should keep invariant.

As the population evolutes heterogamously, the individual A_{ijmax} which has maximum fitness in subspace ij is remained. It is not necessary to compare all of individuals in the subspace, but to compare the individuals which are generated in current generation.

Cooperate with the initial pheromone $Ph_{ij}(1)$ is generated in step 2 and formula (2), pheromone of each subspace is accumulating and updating.

Step 4: Termination

If the generation or individual fits for the termination conditions, the algorithm should be stopped, else go back to step 3.

3 SIMULATION RESULTS AND COMMENT

Here are two examples to certify the ability of ACGA, one has been used by Michalewicz in [2], another is Schaffer's F_6.

Example 1 $\quad f(x_1, x_2) = 21.5 + x_1 \sin(4\pi x_1) + x_2 \sin(20\pi x_2)$

$$-3.0 \leqslant x_1 \leqslant 12.1, \quad 4.1 \leqslant x_2 \leqslant 5.8,$$

Example 2 $\quad f(x,y) = 0.5 - \frac{\sin^2 \sqrt{x^2+y^2} - 0.5}{(1 + 0.001(x^2+y^2))^2}$

$$-4<x, \ y<4$$

Table 1 Simulation parameters

	Example 1		Example 2	
Parameters	SGA	ACGA	SGA	ACGA
Code	Binary	Binary	Binary	Binary
CP	**5**	**5**	**5**	**5**
Scale	**50**	**50**	**64**	**64**
M		**10**		**8**
N		**5**		**8**
P_c	**0.4**	**0.4**	**0.4**	**0.4**
P_m	**0.1**	**0.1**	**0.1**	**0.08**
α		**0.1**		**0.1**
β		**2**		**2**
ρ		**0.2**		**0.1**
C_1		**1**		**1**
IT	**500**	**500**	**500**	**500**

CP: Calculating precision (bit after point); IT: Iteration times

Statistical solutions of 100 times examples simulation are given in table 2.

Table 2 Simulation results of examples

	Example 1		Example 2	
	◇	▽	◇	▽
ACGA	**141**	**7**	**153**	**11**
SGA	**207**	**21**	**245**	**34**

◇Convergence generations ▽Premature convergence times.

4 CONCLUSIONS

In this paper, a hybrid genetic algorithm ACGA based on pheromone remaining of ant colony and genetic operation is presented. Initial population is divided and marked by pheromone. Selection operator is under the effect of subspace's pheromone remaining. Because the initial population of points that are scattered uniformly, so the algorithm can evenly scan the feasible solution space. The effect of pheromone of each subspace in the selection can improve the speed of convergence rate of algorithm. The simulation shows the improvement in convergence speed and global searching ability of ACGA.

Reference

[1] Coloni A, Dorigo M, Maniezzo V. Distributed Optimization by Ant Colonies. Proc. 1st European Conf. Artificial life. Pans, France:Elsevier,1991.

[2] Michalewicz, z., Genetic Algorithm+Data Structure=Evolution Programs, 2nd ed., Springer -Verlag, Newyork,1994..