2-2007

# Quality of service routing strategy using supervised genetic algorithm

Zhaoxia WANG
*Singapore Management University*, zxwang@smu.edu.sg

Yugeng SUN

Zhiyong WANG

Huayu SHEN

**Transactions of Tianjin University**
**ISSN 1006-4982 pp048-052**
**Vol. 13 No. 1 Feb. 2007**

# Quality of Service Routing Strategy Using Supervised Genetic Algorithm*

WANG Zhaoxia (王兆霞)[1,2], SUN Yugeng (孙雨耕)[1], WANG Zhiyong (王志勇)[2], SHEN Huayu (沈花玉)[2]

(1. School of Electrical Engineering and Automation, Tianjin University, Tianjin 300072, China;
2. Tianjin Key Laboratory of Film Electronic and Communication Device, Tianjin University of Technology, Tianjin 300191, China)

**Abstract**: A supervised genetic algorithm (SGA) is proposed to solve the quality of service (QoS) routing problems in computer networks. The supervised rules of intelligent concept are introduced into genetic algorithms (GAs) to solve the constraint optimization problem. One of the main characteristics of SGA is its searching space can be limited in feasible regions rather than infeasible regions. The superiority of SGA to other GAs lies in that some supervised search rules in which the information comes from the problems are incorporated into SGA. The simulation results show that SGA improves the ability of searching an optimum solution and accelerates the convergent process up to 20 times.

**Keywords**: supervised genetic algorithm; supervised search rules; QoS routing

Genetic algorithms (GAs) can deal with complex optimization problems better than traditional optimization techniques[1,2]. The use of GAs for optimization tasks has become popular with the constant development of new algorithms, theoretical achievement and novel applications[3,4]. GAs aim at understanding computational systems and developing more robust and efficient ones for solving complex real-world problems[4]. Some researchers solved routing problem with improved genetic algorithms which were proved to be feasible[2,4]. The networks require the route to support the quality of service (QoS) requests. To support the requirement of extensive QoS, routing algorithm needs the complicated matrix to characterize the network with indexes, such as delay, bandwidth, packet loss rate, and cost[5—10]. Thus, the routing problem based on QoS can be converted to the optimum-searching problem that satisfies several objective functions simultaneously, and it is proved to be NP-complete problem[5,6].

GA is suitable for solving NP problems. However, GA is not able to bias efficiently the search towards the feasible region in the constrained search spaces. In real-world problems, it is difficult to design operators that avoid constraints entirely while still being effective in locating useful feasible solution[2].

The purpose of this paper is to present an approach in which most time is spent on searching in feasible regions rather than infeasible regions. The constraints are transformed into the search rules. The unique characteristic of the proposed supervised genetic algorithm (SGA) that differs from other GAs is that there are some techniques which can be built into SGA, and guarantee the SGA to search in feasible space. Here, SGA arises from the investigation of real-world QoS routing

Accepted date: 2006-06-29.
WANG Zhaoxia, born in 1972, female, Dr, associate Prof.
E-mail: zhxwang@tju.edu.cn.
* Supported by China Postdoctoral Foundation (No. 2005037529), Doctoral Foundation of Education Ministry of China (No. 2003005607) and Tianjin High Education Science Development Foundation (No. 20041325).

problem, which will be described in detail in Sections 2 and 3, and in Section 4 we analyze the feasibility of the proposed SGA and compare it with other GAs.

# 1 QoS routing problem

QoS routing is composed of cost, bandwidth, packet loss rate, and delay[5-7]. The QoS routing problem focuses on finding a set of routes satisfying these factors.

We model the topology network structure with undirected graph $G = (V, E)$, where $V$ is the set of network nodes and $E$ is the set of bidirectional links in the graph. The aim of routing selection is to search for the optimal path while satisfying the QoS requirement between the source and the end nodes. Associated with each edge $l_{ij} \in E$, $B(l_{ij})$ is the bandwidth capacity, $D(l_{ij})$ the delay, $W(l_{ij})$ the cost, and $E(l_{ij})$ the error rate.

The routing request $q$ of a single destination consists of source node, destination node and QoS request which includes bandwidth capacity $B_u$, error rate $L_u$, cost $W_u$, and delay $D_u$. In general, the error rate is almost 0 and the delay is small when the signals transmit in the optical fibers. We should consider the error rate and the delay of the nodes. The following conditions must be satisfied.

Delay produced by the middle nodes from source to destination nodes is

$$D_u = \sum_{l_{ij} \in E_u} D(l_{ij}) \quad (j \text{ is middle nodes, including destination node}) \tag{1}$$

The total error rate from end to end must satisfy the following formula:

$$(1 - L_u) = \prod_{l_{ij} \in E_u} (1 - L(l_{ij})) \quad (j \text{ is middle nodes, including destination node}) \tag{2}$$

The cost from end to end is

$$W_u = \sum_{l_{ij} \in E_u} W(l_{ij}) \tag{3}$$

where $l_{ij} \in E_u$, $E_u$ is the selected route, $E_u \subset E$; $D(l_{ij})$ is the handling delay of node $j$; $L(l_{ij})$ is the loss rate of node $j$; $W(l_{ij})$ is the cost of link $l_{ij}$.

# 2 Routing optimization based on SGA

## 2.1 Coding and decoding mechanisms of SGA

The genetic representation will be crucial to the success of GAs[4]. The simplified binary coding details were described in Ref. [9]. Assuming that the network has $N$ nodes and $\boldsymbol{L} = (l_{ij})_{N\times N}$ represents the topology network matrix. Except diagonal elements, each element corresponds to the likely existing link $l_{ij}$, evidently, $l_{ij} = l_{ji}$. The element $l_{ij}$ located in the matrix corresponds to the link $l_{ij}$ from node $i$ to $j$.

$$l_{ij} = \begin{cases} 1, & \text{if } l_{ij} \text{ exists in the topology network;} \\ 0, & \text{otherwise.} \end{cases} \tag{4}$$

The routing problem can be described by $N$-dimensional binary routing evolution matrix, $\boldsymbol{A} = (v_{ij})_{N\times N}$, and each element of the evolution matrix corresponds to the link $v_{ij}$ from node $i$ to $j$, located on row $i$ and column $j$.

$$v_{ij} = \begin{cases} 1, & \text{if the optimal route includes link from node } i \text{ to } j; \\ 0, & \text{otherwise.} \end{cases} \tag{5}$$

Similarly, the link elements that are not located in the selected route are set to 0. We set the values of the diagonal elements as 0, thus the value of $v_{ij}$ presents whether the link $l_{ij}$ from $i$ to $j$ is selected.

The decoding of this genetic algorithm maps the $N$-dimensional matrix to the representation of binary array. Therefore, the optimal state in the genetic evaluation routing matrix represents an optimal path from the initial source node to the final determination node. For example, there are 6 nodes in the topology of optical fiber communication networks; the matrix $\boldsymbol{A}$ can represent a route from $a_1$ to $a_6$.

The route is $a_1 \to a_2 \to a_4 \to a_3 \to a_5 \to a_6$.

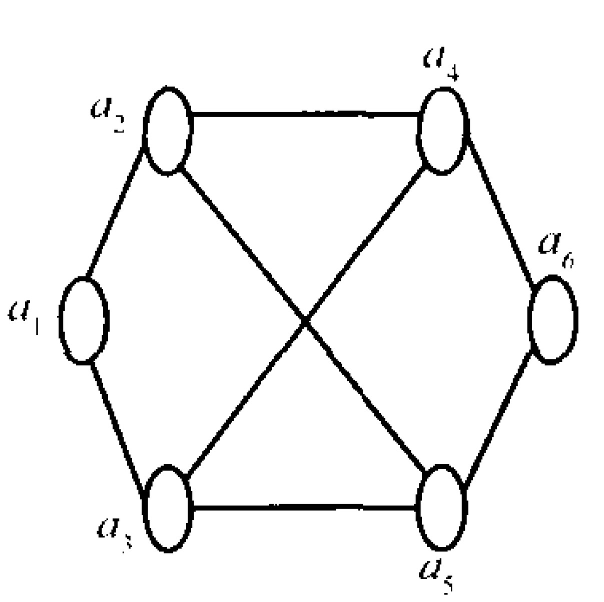
$$\boldsymbol{L} = (l_{ij})_{N\times N} = \begin{bmatrix} 0 & l_{12} & \cdots & l_{1N} \\ l_{21} & 0 & \cdots & l_{2N} \\ \cdots & \cdots & 0 & \cdots \\ l_{N1} & l_{N2} & \cdots & 0 \end{bmatrix}$$

$$\boldsymbol{A} = \begin{array}{c} a_1 \\ a_2 \\ a_3 \\ a_4 \\ a_5 \\ a_6 \end{array} \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

It is obvious that each route can be represented by a matrix. The binary string representing a route is the concatenation of rows (or columns) in the matrix. Evidently, only if it includes link $l_{ij}$ in the network topology structure, i. e. $l_{ij} = 1$, the element

$a_{ij}$ in the evaluation routing matrix may take 1.

Because the elements in the link matrix represent the real link from node $i$ to $j$, the inexistence of links are always set to constant 0. Values of the diagonal elements are always 0. Fig. 1 gives the example of the direct encoding scheme of this routing problem. In general, the matrix of the network topology and the route matrix are sparse. The useful information is stored in the non-zero elements.




(a) Network topology

$$\begin{pmatrix} 0 & 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 \end{pmatrix}$$

(b) Network topology matrix

$$\begin{pmatrix} 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

(c) Upper triangle matrix

$$\begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

(d) Route matrix

**Fig. 1 Network topology structure and its representation with matrix**

Since the topology network structure is represented by the undirected graph $G = (V, E)$, we only need to represent the upper triangle of the matrix which is given by Fig. 1 (c) to reduce the chromosome length. The element $l_{ij}$ indicates the presence or absence of a connection from node $i$ to $j$. Once the topology network structure is confirmed, and the 0's will be determined. We can reduce the length of the chromosome more by getting rid of the 0's. Similarly, the routes can be incorporated into such representation. Fig. 1(d) shows a route from node 1 to 6, where the element indicates the presence or absence of a selected connection from node 1 to 6 in the route.

## 2.2 Establishment of evaluation function

The evaluation function is as follows [9]:

$$E_1 = \frac{F}{2}\left(\sum_{i=1}^{N}\sum_{j=1}^{N} w_{ij}v_{ij} - W_u\right)^2 + \frac{G}{2}\left[\sum_{i=1}^{N}\sum_{j=1}^{N} t_{ij}v_{ij} - (D_u - d_0)\right]^2 + \frac{H}{2}\left[\prod_{i=1}^{N}\prod_{j=1}^{N}(1 - v_{ij}\mu_{ij}) - (1 - L_u)\right]^2 \quad (6)$$

The selected fitness function is

$$E = \frac{1}{E_1 + 1} \quad (7)$$

where $F$, $G$ and $H$ denote the weight coefficients of three constraints; $t_{ij}$, $\mu_{ij}$ and $w_{ij}$ denote the delay, signal error rate and costs, respectively, which can be obtained by measuring network as the vital index of routing; $D_u$, $L_u$, $W_u$ and $d_0$ are designated by QoS.

## 2.3 Supervised rules and supervised mutation operator

A good search operator should increase the probability of finding a global optimum[4]. No-free-lunch theorems[10] also indicate the importance of incorporating problem-specific knowledge into the behavior of the algorithm. The supervised rules and supervised mutation operator are used as search operators to ensure searching in feasible space.

From the above coding methods, the constraints for the QoS problems are described by the following inequation or equation:

$$\begin{aligned} g_1 &= 1 - \sum_{i=1}^{N} a_{ij} \geqslant 0 \\ &\quad i = 1,2,\cdots, N;\ j = 1,2,\cdots,N \\ g_2 &= 1 - \sum_{j=1}^{N} a_{ij} \geqslant 0 \\ &\quad i = 1,2,\cdots, N;\ j = 1,2,\cdots,N \\ g_3 &= \sum_{i=2}^{N-1} a_{ij} - \sum_{i=2}^{N-1} a_{ji} = 0 \\ &\quad i = 2,\cdots, N-1;\ j = 2,\cdots,N-1 \end{aligned} \quad (8)$$

It is easy to transform the above constraints into rules for supervising search operator and mutation operator. The new concepts of SGA search methods are as follows.

**Definition 1** (Search rule) The nodes (except source and destination nodes) must be the inceptive node of the next link, if it is the arriving node.

**Definition 2** (Preference rule) The destination node and the source node have the priority to be selected over other nodes when a node connects many nodes among which there is a destination node or source node.

**Definition 3** (Supervised mutation operator and regeneration) An integrated chromosome can be divided into 2 parts from the select position in which the bit element value must be 1, as called supervised mutation operator, different from the traditional mutation operator which randomly flips some of the bits in a chromosome. Traditional mutation can occur at each bit position in a string with some probability. However, the supervised mutation operator must occur at those positions where the bit value must be 1. Then the selected part will regenerate the rest part by obeying the search rule and preference rule. We call this process regeneration operator.

### 2.4 Realization of SGA

Performing specific routing algorithm is as follows.

(1) Present route delay matrix $\{t_{ij}\}$, cost matrix $\{w_{ij}\}$ and loss rate matrix $\{\mu_{ij}\}$ as the information of delay, cost and loss rate of current link, respectively. Translate current matrices to three items.

(2) Generate the initial population matrices $\boldsymbol{P}_0(\boldsymbol{P}_t,\ t = 0)$ and translate current matrices to three items according to the above rules and operators. By using fitness function, the fitness of all solutions in the colony is evaluated and the best one is labeled.

(3) Create offspring population $\boldsymbol{P}_{t+1}$ from parent population $\boldsymbol{P}_t$ according to the rules and operators of SGA. The fitness of all solutions in the colony is evaluated and the best one is labeled compared with the former if the former generation exists, then choose the better one as the best one of current generation.

(4) Rule of stopping testing. If it is satisfied, stop; otherwise, $t = t + 1$, turn to (3).

## 3 Simulation analysis of SGA

Fig. 2 and Fig. 3 show two examples of the network system topology [8, 9]. One consists of 6 nodes and 8 edges, and the other consists of 13 nodes and several edges connecting the nodes. Each of nodes and edges is denoted by parameters. The elements between brackets are error rate and delay, respectively. Cost marks are given on the side of edges.

**Example 1** Employing the network topology structure as shown in Fig. 2. Perform routing from the source node $a_1$ to the destination node $a_6$. Population dimension is set to 10, and coefficient $F = G = H = 100$. By employing the SGA proposed above, we can obtain the global optimal solution as early as in the first generation.

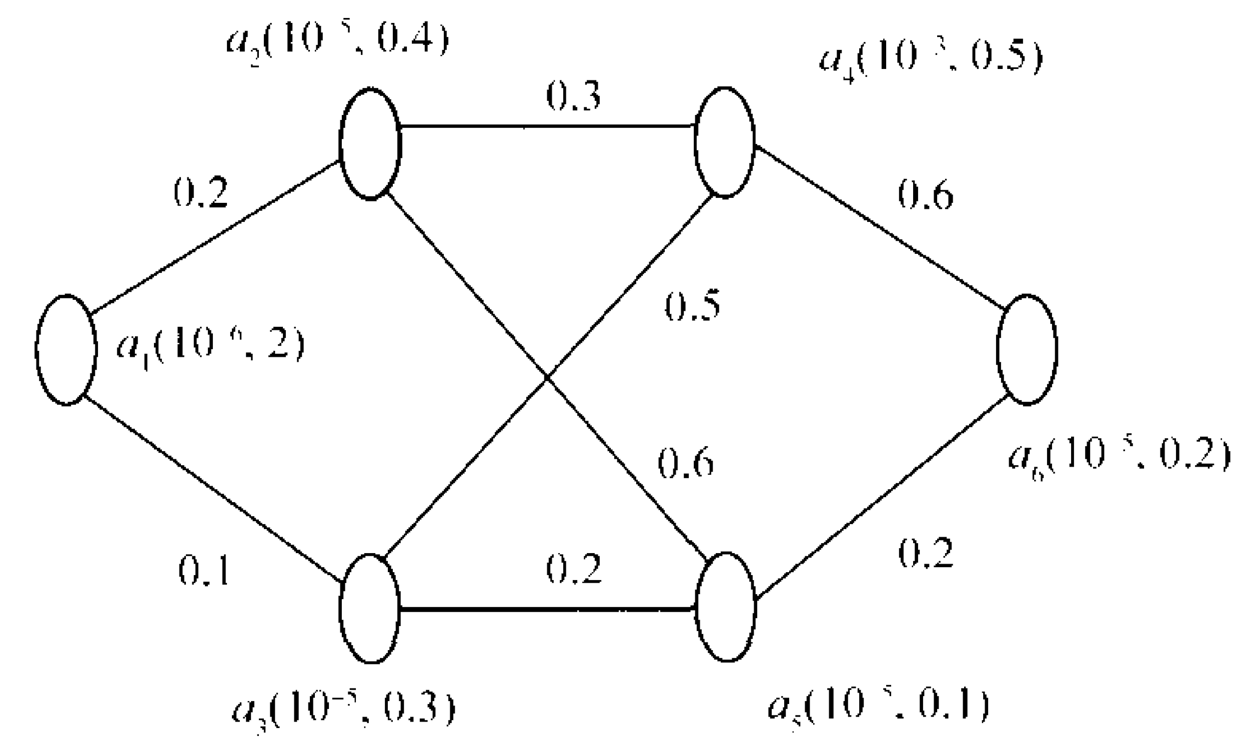


**Fig. 2** Topology structure and parameters of Example 1

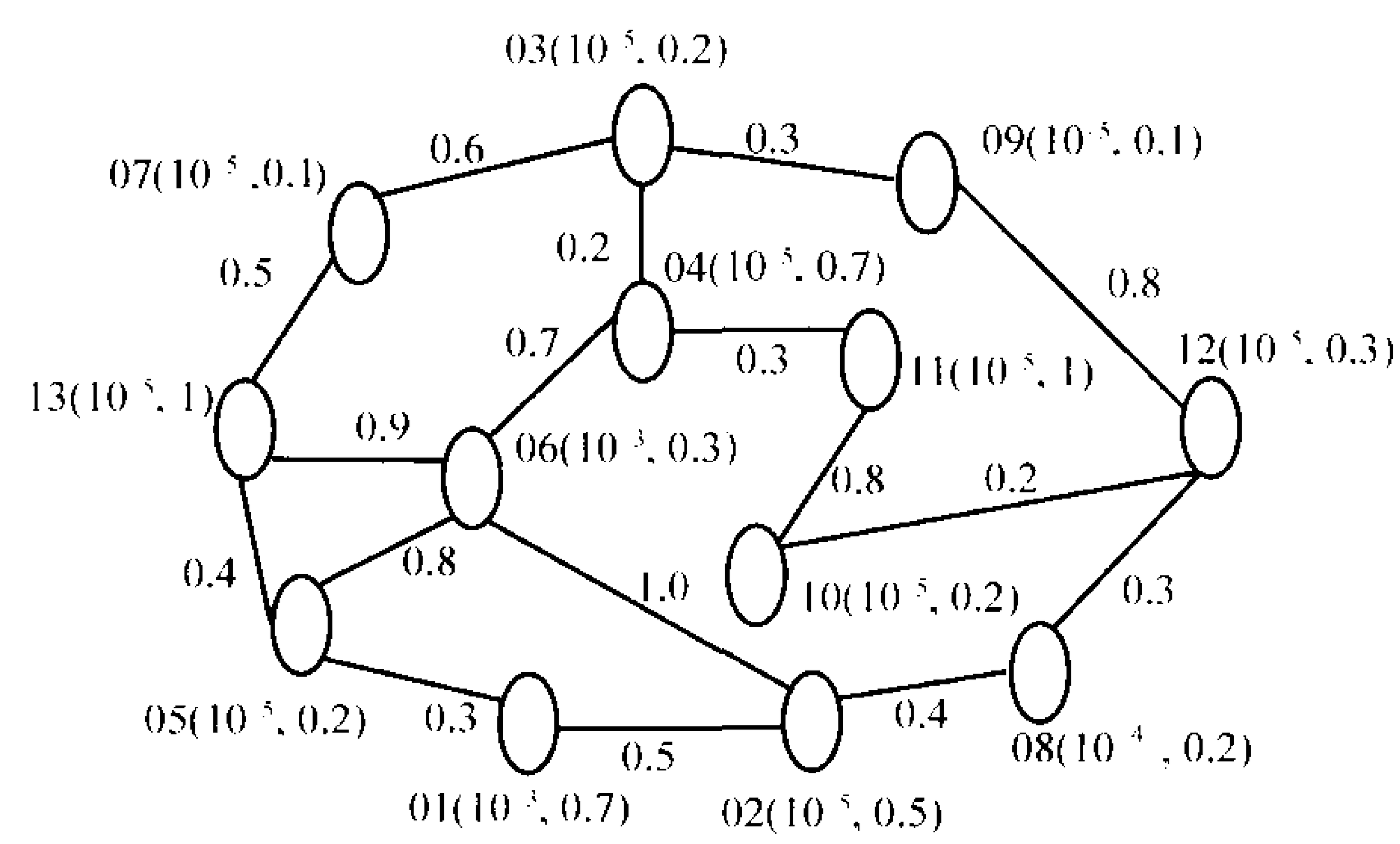


**Fig. 3** Topology structure and parameters of Example 2

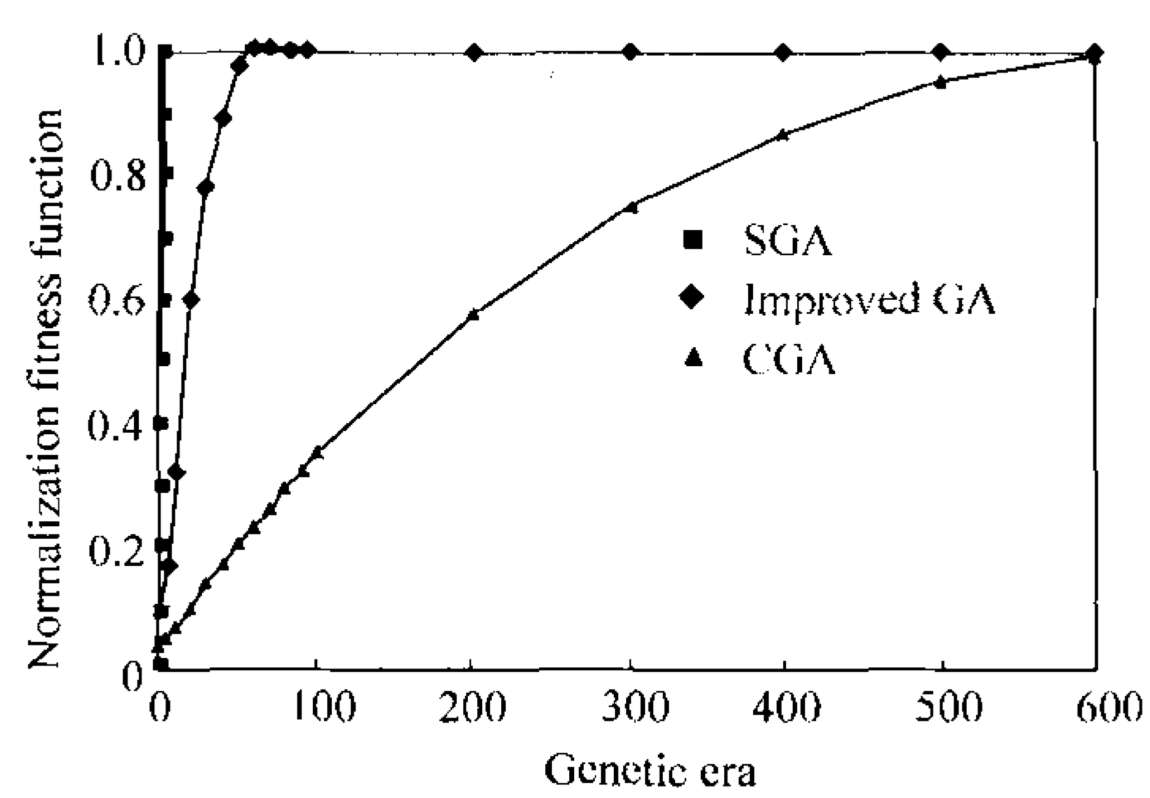
**Example 2** Employing the network topology structure as shown in Fig. 3. Perform routing from node 01 to 10, and the size of the population is set to 40. We can obtain the optimal path, and through several simulations the mean number of generations to gain optimal solution is 3.

Tab. 1 and Fig. 4 show experimental results obtained with SGA. The data clearly indicates that the genetic eras of SGA are less than those of other GAs[8, 9].

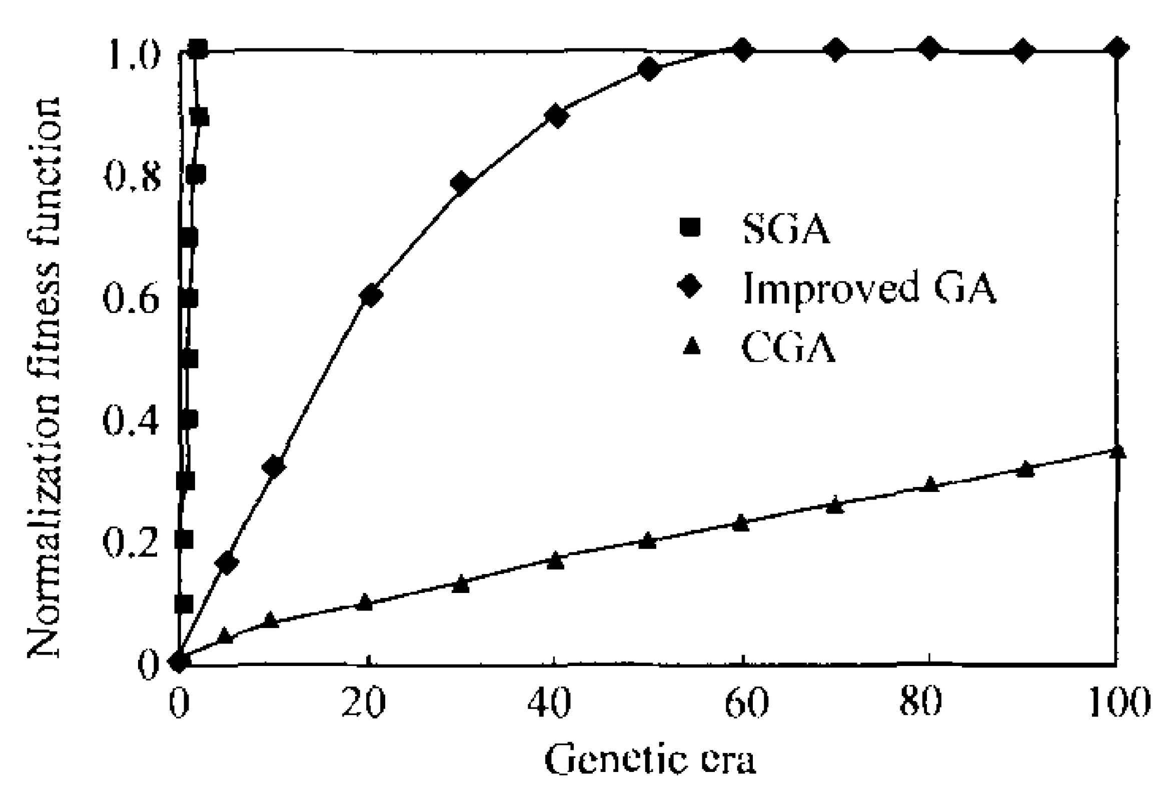
**Tab. 1 Simulation results of SGA and other GAs**

| Example | Population | Mean genetic era | | |
|---|---|---|---|---|
| | | SGA | Improved GA[9] | CGA |
| 1 | 10 | 1 | 4 | 15 |
| 2 | 40 | 3 | 62 | 627 |

The results presented in Fig. 4 illustrate the benefits of SGA compared with CGA and other improved GAs because SGA increases the convergent speed greatly. The results of simulation validate that SGA can find the optimal path from initial source node to final determination node in a single run. This algorithm can search out the optimal routing under limited time for the problem. To take an in-depth analysis, the simulation of SGA can also be employed in a large-scale network, which contains more than 100 nodes.




(a) Whole 600 genetic era




(b) First 100 genetic era

**Fig. 4 Normalization fitness function *vs.* generation**

# 4 Conclusions

In this paper, the supervised rules of intelligent concept are introduced into GA. We propose the SGA by incorporating the routing problem-specific knowledge into the behavior of GA and using the information from problems to supervise the searching process. SGA diminishes decision space and searches in feasible regions rather than in infeasible regions. The proposed method can solve the routing problem in communication networks with several QoS constraints. The results of simulation verify its superiority over other GAs . This algorithm is so independent that it does not rely on any specific network which ensures it extensive applicability in network systems.

**Acknowledgments**

The authors would like to thank Prof. Yao Xin and members of the CERCIA in University of Birmingham for useful discussions.

## References

[1] Yao Xin. *Evolutionary Computation: Theory and Applications* [M]. Singapore: World Scientific Publishing Corporation, 1999.

[2] Sarker R, Mohammadian M, Yao X. *Evolutionary Optimization* [M]. Boston: Kluwer Academic Publishers, 2002.

[3] Yao Xin, Liu Yong. A new evolutionary system for evolving artificial neural networks [J]. *IEEE Transactions on Neural Networks*, 1997, 8(3): 694—713.

[4] Yao Xin. Evolutionary Computation [C]. In: *Evolutionary Optimization*. Edited by Sarker R, Mohammadian M and Yao X. New York: Kluwer Academic Publishers, 2002. 27—53.

[5] Wang Zhengying, Shi Bingxin, Zhao Erdun. Bandwidth-delay-constrained least-cost multicast routing based on heuristic genetic algorithm [J]. *Computer Communications*, 2001, 24(7/8): 685—692.

[6] Chen Shigang, Nahrstedt K. An overview of quality of service routing for next-generation high-speed networks: Problems and solutions [J]. *IEEE Network*, 1998, 12(6): 64—79.

[7] Cui Xunxue, Lin Chuang, Wei Yaya. A multiobjective model for QoS multicast routing based on genetic algorithm [C]. In: *Proceedings of the* 2003 *International Conference on Computer Networks and Mobile Computing* (*ICCNMC*03). Shanghai: IEEE Computer Society, 2003. 49—53.

[8] Fei Xiang, Luo Junzhou, Wu Jieyi. QoS routing based on genetic algorithm [J]. *Computer Communications*, 1999, 22(15/16): 1392—1399.

[9] Wang Zhaoxia, Chen Zengqiang, Yuan Zhuzhi. QoS routing optimization strategy using genetic algorithm in optical fiber communication networks [J]. *Journal of Computer Science and Technology*, 2004, 19(2): 213—217.

[10] David H Wolpert, William G Macready. No free lunch theorems for optimization [J]. *IEEE Transactions on Evolutionary Computation*, 1997, 1(1): 67—82.