

# 基于组织进化的 ABC 支持型单播路由机制

王兴伟 孙永健 蒋定德 黄敏

(东北大学信息科学与工程学院 沈阳 110819)

**摘要** 提出一种支持总最佳连接 ABC(Always Best Connected)的服务质量 QoS(Quality of Service)单播路由机制。该机制采用区间形式刻画用户 QoS 需求和网络链路参数;引入偏好序列,以满足用户对不同类型网络的偏爱;引入概率密度函数、满意度函数和评价函数,以精确测量网络链路参数值和准确表达用户 QoS 需求;引入成本、价格、费用和博弈机制,兼顾用户和网络提供方利益;基于组织进化算法 OEA(Organizational Evolutionary Algorithm),寻找使各方效用达到或接近 Nash 均衡下 Pareto 最优的 QoS 单播路径。仿真结果表明,该机制是可行和有效的。

**关键词** QoS,单播,路由,ABC,组织进化

**中图分类号** TP393 **文献标识码** A

## Organizational Evolution-based ABC Supported Unicast Routing Scheme

WANG Xing-wei SUN Yong-jian JIANG Ding-de HUANG Min

(College of Information Science and Engineering, Northeastern University, Shenyang 110819, China)

**Abstract** An ABC (Always Best Connected) supported QoS (Quality of Service) unicast routing scheme was proposed. In the proposed scheme, intervals were used to describe the user QoS requirements and network link parameters; preference sequences were introduced to reflect users' preferences to different types of networks; probability density functions, satisfaction functions and evaluation functions were adopted to overcome difficulties on accurately measuring network link parameter values and exactly expressing on user QoS requirements; cost, price, billing and gaming were taken to deal with profits of both the user and the network provider; finally, OEA (Organizational Evolutionary Algorithm) was used to find the specific QoS unicast path with Pareto optimum under Nash Equilibrium among all parties' utilities achieved or approached. Simulation results show that the proposed scheme is both feasible and effective.

**Keywords** QoS (Quality of Service), Unicast, Routing, ABC (Always Best Connected), Organization evolution

## 1 引言

下一代互联网(Next Generation Internet, NGI)是由地面网与空天网、固定网与移动网等融合而成的一体化网络,在通信端到端路径上每一跳都可能存在多种不同类型的链路供选择,从而可以在通信开始时和进行期间实现用户对 NGI 的总最佳连接(Always Best Connected, ABC)<sup>[1]</sup>。用户在任何时间、任何地点都可以通过当前最佳方式使用 NGI,而且每当有更好的连接方式出现时都可以自适应透明切换,实现服务质量(Quality of Service, QoS)无缝漫游<sup>[2]</sup>。QoS 路由需要支持 ABC<sup>[3]</sup>。

NGI 的异构性和动态性、终端和网络的移动性、信息传递延迟的不可避免及其不确定性等都导致路由选择时所依赖的网络状态信息难以精确表达。用户 QoS 需求受主观因素影响较大,难以准确刻画,需要体现一定柔性。ABC 意味着用户应使用最佳连接,而“最佳”本身就是一个模糊概念,依赖很

多因素。此外,“利己”不应导致“排他”,特别是在网络运营商业化的环境下,ABC 不是用户单方面的事,需要兼顾网络提供方与用户利益,实现各方共赢<sup>[4]</sup>。受两个或两个以上加性或乘性 QoS 参数约束的单播路由问题是 NP 完全的<sup>[5]</sup>,考虑上述因素的 ABC 支持型 QoS 路由问题更加复杂,需要综合运用图论、概率论、微观经济学以及启发式或智能优化算法等求解。

已经提出了多种 QoS 单播路由机制。Shariff 等使用路径可行性预测模型,提出了一种延迟受限单播路由算法<sup>[6]</sup>。Koundinya 等基于 Dijkstra 算法,提出了一种给定最优条件下的多 QoS 参数单播路由算法<sup>[7]</sup>。Resende 等利用广义模糊约束模型求解 QoS 单播路由问题,考虑了 QoS 约束和网络状态的不确定性<sup>[8]</sup>。Dai 等提出了一种基于向量变换的多约束 QoS 路由算法<sup>[9]</sup>。Hu 等人针对 Internet 中存在大量不确定信息及其对 QoS 的影响,提出了考虑延迟和带宽约束的 QoS 单播路由算法<sup>[10]</sup>。Leung 等提出了基于凹约束的 QoS 路由

到稿日期:2010-11-21 返修日期:2011-02-26 本文受国家自然科学基金项目(61070162,71071028,60802023,70931001),高等学校博士学科点专项科研基金资助课题(20100042110025),中央高校基本科研业务费专项资金(N090504003, N090504006)资助。

王兴伟(1968—),男,博士,教授,博士生导师,CCF 高级会员,主要研究方向为新一代互联网关键技术, E-mail: wangxw@mail.neu.edu.cn; 孙永健(1986—),男,硕士生,主要研究方向为网络路由与疏导机制; 蒋定德(1974—),男,博士,副教授,主要研究方向为网络路由与流量预测技术; 黄敏(1968—),女,博士,教授,博士生导师,主要研究方向为优化方法及其应用。

算法<sup>[11]</sup>。Yang 等研究了异构网络环境下的端到端路由,提出了一种 QoS 感知型路由算法<sup>[12]</sup>。Song 等研究了多约束 QoS 路由问题,提出了 6 种启发式路由算法<sup>[13]</sup>。Bashandy 等提出了一种广义 QoS 路由算法<sup>[14]</sup>。Khadivi 等提出了一种混合量度多约束 QoS 路由算法<sup>[15]</sup>。Wang 等提出了多种基于模糊数学和微观经济学知识的智能 QoS 路由算法<sup>[16,17]</sup>。

目前尚没有从支持 ABC 的角度深入研究如何在网络状态信息与用户 QoS 需求不精确的情况下实现支持网络提供方与用户利益共赢的 QoS 单播路由机制。本文基于组织进化算法(Organizational Evolutionary Algorithm, OEA)<sup>[18]</sup>,提出了一种 ABC 支持型 QoS 单播路由机制来解决上述问题。仿真结果表明,本机制具有较好的性能。

## 2 模型设计

### 2.1 网络模型与路由请求

已知图  $G(V, E)$ , 其中  $V$  是节点集合,  $E$  是边集合。 $\forall v_i, v_j \in V(i, j=1, \dots, |V|)$ , 其间可能存在多条边。简单起见, 本文把节点参数归并到边参数中。对  $v_i$  与  $v_j$  之间的第  $k$  条边  $e_{ij}^k$  考虑如下 6 个参数: 可用带宽  $b_{ij}^k \in [b_{ij}^{kL}, b_{ij}^{kH}]$ , 延迟  $d_{ij}^k \in [d_{ij}^{kL}, d_{ij}^{kH}]$ , 延迟抖动  $j_{ij}^k \in [j_{ij}^{kL}, j_{ij}^{kH}]$ , 出错率  $l_{ij}^k \in [l_{ij}^{kL}, l_{ij}^{kH}]$ , 单位带宽成本  $c_{ij}^k$  和售价  $pr_{ij}^k$ 。QoS 单播路由请求采用八元组  $\langle v_s, v_d, \Delta_B, \Delta_D, \Delta_J, \Delta_L, \Delta_P, P_U \rangle$  表示, 其元素依次表示源节点、目的节点、用户带宽需求区间  $[B_L, B_H]$ 、延迟需求区间  $[D_L, D_H]$ 、延迟抖动需求区间  $[J_L, J_H]$ 、出错率需求区间  $[L_L, L_H]$ 、用户网络偏好序列  $\{\vartheta_1, \vartheta_2, \dots, \vartheta_M\}$  ( $\vartheta$  是网络类型编号, 按偏好程度从高到低排列,  $M$  是可供用户选择的不同类型网络数,  $1 \leq i \leq M$ ) 和愿付费用上限。采用区间是为了精确测量网络链路参数值和准确表达用户 QoS 需求。

### 2.2 边参数刻画与用户评价

本文采用高斯分布<sup>[19]</sup>刻画边上各参数。

$e_{ij}^k$  向用户提供带宽  $b_{ij}^{k*} \in [b_{ij}^{k*L}, b_{ij}^{k*H}] \subseteq [b_{ij}^{kL}, b_{ij}^{kH}]$  的概率密度  $p_{ij}^{k*}$ 、用户对  $b_{ij}^{k*}$  的满意度  $S_{ij}^{k*}$  和评价函数  $E_{ij}^{k*}$  分别定义如下:

$$p_{ij}^{k*} = \frac{1}{\sqrt{2\pi} \cdot (\sigma_{ij}^{k*})^2} e^{-\frac{(b_{ij}^{k*} - \mu_{ij}^{k*})^2}{2 \cdot (\sigma_{ij}^{k*})^2}} \quad (1)$$

$$S_{ij}^{k*} = \begin{cases} 0, & b_{ij}^{k*} < B_L \\ \epsilon, & b_{ij}^{k*} = B_L \\ e^{-\frac{(B_H - b_{ij}^{k*})^2}{(b_{ij}^{k*} - B_L)^2}}, & B_L < b_{ij}^{k*} < B_H \\ 1, & b_{ij}^{k*} \geq B_H \end{cases} \quad (2)$$

$$E_{ij}^{k*} = \frac{\int_{b_{ij}^{k*L}}^{b_{ij}^{k*H}} p_{ij}^{k*} \cdot S_{ij}^{k*} d(b_{ij}^{k*})}{\int_{b_{ij}^{k*L}}^{b_{ij}^{k*H}} p_{ij}^{k*} d(b_{ij}^{k*})} \quad (3)$$

式中,  $\epsilon$  是远小于 1 的正纯小数。式(1)表示  $b_{ij}^{k*}$  服从均值为  $\mu_{ij}^{k*} = \frac{b_{ij}^{kL} + b_{ij}^{kH}}{2}$  和方差为  $(\sigma_{ij}^{k*})^2$  的高斯分布。 $b_{ij}^{k*}$  越大, 用户对其越满意, 评价越高; 若  $b_{ij}^{k*}$  取精确值, 则  $E_{ij}^{k*} = S_{ij}^{k*}$ 。

$e_{ij}^k$  的延迟等于  $d_{ij}^{k*} \in [d_{ij}^{k*L}, d_{ij}^{k*H}] \subseteq [d_{ij}^{kL}, d_{ij}^{kH}]$  的概率密度  $p_{ij}^{k*}$ 、延迟抖动等于  $j_{ij}^{k*} \in [j_{ij}^{k*L}, j_{ij}^{k*H}] \subseteq [j_{ij}^{kL}, j_{ij}^{kH}]$  的概率密度  $p_{ij}^{k*}$ 、出错率等于  $l_{ij}^{k*} \in [l_{ij}^{k*L}, l_{ij}^{k*H}] \subseteq [l_{ij}^{kL}, l_{ij}^{kH}]$  的概率密度  $p_{ij}^{k*}$  的定义均与  $p_{ij}^{k*}$  类似, 只需分别用  $d_{ij}^{k*}$ 、 $j_{ij}^{k*}$  和  $l_{ij}^{k*}$  和  $(\sigma_{ij}^{k*})^2$ ,

$$j_{ij}^{k*}, \mu_{ij}^{k*} = \frac{j_{ij}^{kL} + j_{ij}^{kH}}{2} \text{ 和 } (\sigma_{ij}^{k*})^2, l_{ij}^{k*}, \mu_{ij}^{k*} = \frac{l_{ij}^{kL} + l_{ij}^{kH}}{2} \text{ 和 } (\sigma_{ij}^{k*})^2 \text{ 替代式}$$

(1) 中的  $b_{ij}^{k*}$ 、 $\mu_{ij}^{k*}$  和  $(\sigma_{ij}^{k*})^2$  即可。

用户对在  $e_{ij}^k$  上实际经历延迟  $d_{ij}^{k*}$  的满意度  $S_{ij}^{k*}$  定义如下:

$$S_{ij}^{k*} = \begin{cases} 1, & d_{ij}^{k*} \leq D_L \\ 1 - e^{-\frac{(D_H - d_{ij}^{k*})^2}{(d_{ij}^{k*} - D_L)^2}}, & D_L < d_{ij}^{k*} < D_H \\ \epsilon, & d_{ij}^{k*} = D_H \\ 0, & d_{ij}^{k*} > D_H \end{cases} \quad (4)$$

显然,  $d_{ij}^{k*}$  越大, 用户对其越不满意。

用户对  $j_{ij}^{k*}$  和  $l_{ij}^{k*}$  的满意度  $S_{ij}^{j*}$  和  $S_{ij}^{l*}$  的定义与式(4)类似, 只需用  $j_{ij}^{k*}$ 、 $J_H$  和  $J_L$ 、 $l_{ij}^{k*}$ 、 $L_H$  和  $L_L$  分别替代式(4)中的  $d_{ij}^{k*}$ 、 $D_H$  和  $D_L$  即可。

用户对  $d_{ij}^{k*}$ 、 $j_{ij}^{k*}$  和  $l_{ij}^{k*}$  的评价函数  $E_{ij}^{d*}$ 、 $E_{ij}^{j*}$  和  $E_{ij}^{l*}$  的定义与  $E_{ij}^{k*}$  类似, 只需分别用  $d_{ij}^{k*}$ 、 $d_{ij}^{j*}$ 、 $d_{ij}^{l*}$  和  $S_{ij}^{d*}$ 、 $S_{ij}^{j*}$ 、 $S_{ij}^{l*}$ 、 $p_{ij}^{d*}$  和  $S_{ij}^{d*}$ 、 $l_{ij}^{k*}$ 、 $l_{ij}^{j*}$ 、 $l_{ij}^{l*}$  和  $S_{ij}^{l*}$  替代式(1)中的  $b_{ij}^{k*}$ 、 $b_{ij}^{j*}$ 、 $b_{ij}^{l*}$  和  $S_{ij}^{k*}$  即可。

用户对  $e_{ij}^k$  的综合评价函数定义如下:

$$E_{ij}^{k*} = a_B \cdot E_{ij}^{b*} + a_D \cdot E_{ij}^{d*} + a_J \cdot E_{ij}^{j*} + a_L \cdot E_{ij}^{l*} \quad (5)$$

式中,  $a_B, a_D, a_J, a_L \in [0, 1]$  是权重系数, 表示带宽、延迟、延迟抖动和出错率对用户 QoS 需求的相对重要性, 取值根据应用类型而定, 其和为 1。

### 2.3 博弈分析

在本文中, 用户和网络提供方在边上博弈。用户有  $n$  种策略, 表示为  $\langle b_{ij}^{k*1}, \dots, b_{ij}^{k*n} \rangle$ , 对应用户可以在  $e_{ij}^k$  上使用的  $n$  种带宽。网络提供方有  $m$  种策略, 表示为  $\langle pr_{ij}^{k*1}, \dots, pr_{ij}^{k*m} \rangle$ , 对在  $e_{ij}^k$  上的  $m$  种单位带宽售价。在策略对  $\langle b_{ij}^{k*x}, pr_{ij}^{k*y} \rangle$  ( $x=1, \dots, n, y=1, \dots, m$ ) 下, 用户因使用  $e_{ij}^k$  的  $b_{ij}^{k*x}$  带宽而支付的费用  $PR_{ij}^{k*xy}$  和网络提供方因向用户提供  $b_{ij}^{k*x}$  带宽而承担的成本  $CT_{ij}^{k*xy}$  分别计算如下:

$$PR_{ij}^{k*xy} = pr_{ij}^{k*y} \cdot b_{ij}^{k*x} \quad (6)$$

$$CT_{ij}^{k*xy} = c_{ij}^{k*} \cdot b_{ij}^{k*x} \quad (7)$$

用户效用定义如下:

$$uu_{ij}^{k*xy} = \vartheta_t \cdot E_{ij}^{k*} \cdot \frac{P_U - PR_{ij}^{k*xy}}{P_U} \quad (8)$$

$$\vartheta_t = \frac{1}{t^2} \quad (9)$$

式中,  $\vartheta_t$  表示用户对  $e_{ij}^k$  所属网络类型的偏爱程度,  $t$  是  $e_{ij}^k$  所属网络类型在  $\vartheta$  中的序号,  $t=1, \dots, M$ 。显然, 用户对边的综合评价越高, 需付费用越低, 网络类型越符合其偏爱, 其效用越大。

网络提供方效用定义如下:

$$nu_{ij}^{k*xy} = \frac{PR_{ij}^{k*xy} - CT_{ij}^{k*xy}}{CT_{ij}^{k*xy}} \quad (10)$$

显然, 网络提供方从用户收取的费用越高, 承担的成本越低, 其效用越大。

用户和网络提供方在路径  $PT_{sd}$  上的效用分别计算如下:

$$UU_{PT_{sd}} = \sum_{e_{ij}^k \in PT_{sd}} uu_{ij}^{k*xy} \quad (11)$$

$$NU_{PT_{sd}} = \sum_{e_{ij}^k \in PT_{sd}} nu_{ij}^{k*xy} \quad (12)$$

效用对  $\langle uu_{ij}^{k*xy}, nu_{ij}^{k*xy} \rangle$  在  $e_{ij}^k$  上的 Pareto 优势定义如下:

$$PD_{ij}^{k*xy} = \frac{1}{\beta_U \cdot \frac{1}{uu_{ij}^{k*xy}} + \beta_N \cdot \frac{1}{nu_{ij}^{k*xy}}} \quad (13)$$

式中,  $\beta_U$  和  $\beta_N$  是对用户和网络提供方的倾斜权值,  $0 \leq \beta_U, \beta_N \leq 1, \beta_U + \beta_N = 1$ , 两者相等时表示各方平等。各方效用越大且越均衡,  $PD_{ij}^{ky}$  值越大, 对应的策略对越能使各方效用 Pareto 最优。

定义用户和网络提供方在  $e_{ij}^k$  上的效用矩阵为  $[\langle u_{ij}^{ky}, nu_{ij}^{ky} \rangle]_{n \times m}$ , 矩阵的  $n$  行与  $m$  列分别对应用户和网络提供方的  $n$  种与  $m$  种策略。博弈的目的是通过确定最佳策略对使各方效用在边上达到或接近 Nash 均衡下的 Pareto 最优。若存在  $\langle u_{ij}^{ky*}, nu_{ij}^{ky*} \rangle$  满足式(14), 则达到 Nash 均衡<sup>[20]</sup>,  $x, x^* = 1, \dots, n, y, y^* = 1, \dots, m$ 。  $\langle u_{ij}^{ky*}, nu_{ij}^{ky*} \rangle$  是用户和网络提供方最佳策略对, 若有多个或没有, 则比较 Pareto 优势, 选择与  $PD_{ij}^{ky}$  最大值对应的策略对作为最佳策略对(若还有多个, 则任选其一)。

$$\begin{cases} u_{ij}^{ky*} y^* \geq u_{ij}^{ky} y \\ nu_{ij}^{ky*} y^* \geq nu_{ij}^{ky} y \end{cases} \quad (14)$$

## 2.4 数学模型

本文模式的目标是在满足用户 QoS 需求的前提下使各方效用在所选路径上达到或接近 Nash 均衡下的 Pareto 最优, 数学描述如下:

$$\text{Maximize } \{UU_{PT_{sd}}\} \quad (15)$$

$$\text{Maximize } \{NU_{PT_{sd}}\} \quad (16)$$

$$\text{Maximize } \{UU_{PT_{sd}} + NU_{PT_{sd}}\} \quad (17)$$

$$\text{s. t. } B_{PT_{sd}} \geq B_L \quad (18)$$

$$D_{PT_{sd}} \leq D_H \quad (19)$$

$$J_{PT_{sd}} \leq J_H \quad (20)$$

$$L_{PT_{sd}} \leq L_H \quad (21)$$

$$EP_{PT_{sd}} \leq P_U \quad (22)$$

式中,  $B_{PT_{sd}}, D_{PT_{sd}}, J_{PT_{sd}}, L_{PT_{sd}}$  和  $EP_{PT_{sd}}$  分别是  $PT_{sd}$  的可用带宽、延迟、延迟抖动、出错率和用户需付费用, 分别计算如下:

$$B_{PT_{sd}} = \min\{b_{ij}^k \mid e_{ij}^k \in PT_{sd}\} \quad (23)$$

$$D_{PT_{sd}} = \sum_{e_{ij}^k \in PT_{sd}} d_{ij}^k \quad (24)$$

$$J_{PT_{sd}} = \sum_{e_{ij}^k \in PT_{sd}} j_{ij}^k \quad (25)$$

$$L_{PT_{sd}} = 1 - \prod_{e_{ij}^k \in PT_{sd}} (1 - l_{ij}^k) \quad (26)$$

$$EP_{PT_{sd}} = \sum_{e_{ij}^k \in PT_{sd}} EP_{ij}^{ky} \quad (27)$$

## 3 算法设计

OEA 基于经济学中的“组织”概念。种群由组织构成; 不同组织的交集为空; 组织表示一组个体, 个体称为组织的成员, 适值最大的成员称为领导, 如果在组织中有多个成员具有最大适值, 则随机选择其一作为领导, 领导的适值即为组织的适值。组织形成之后, 主要是扩大规模和增强实力, 彼此竞争, 实力弱的组织被吞并; 在有些情况下, 多个组织会相互合作, 以增强各自实力; 有的组织在发展过程中由于变得过于庞大而分裂成多个小组。进化操作作用于组织而非个体, 引入 3 个组织进化算子——分裂、吞并和合作来引导种群进化。OEA 不仅全局收敛性好, 搜索机制高效, 不易陷入局部最优, 而且对参数不敏感, 解质量和计算复杂度优于其他常用进化算法<sup>[18]</sup>。本文基于 OEA 设计 ABC 支持型 QoS 单播路由机

制, 描述如下。

### 3.1 解表达、初始解生成与适值函数

在本文中, 每个个体都对应一个问题解即 QoS 单播路径。对  $E$  中每条边  $e_{ij}^k$  都赋予唯一的编号  $no_{ij}^k, 1 \leq no_{ij}^k \leq |E|$ 。解采用向量表示形式, 按从  $v_s$  到  $v_d$  的方向由组成  $PT_{sd}$  的边的编号依次作为向量元素, 记作  $\langle no_{s_1}^{k_1}, no_{s_1 i_2}^{k_2}, \dots, no_{i_{w-2}^{k_{w-2}} i_{w-1}}^{k_{w-1}}, no_{i_{w-1}^{k_{w-1}} d} \rangle, v_{s_1}, \dots, v_{i_{w-1}}$  是  $PT_{sd}$  上从  $v_s$  到  $v_d$  的中间节点序列。采用如下随机路径算法生成初始解:

步骤 1 当前访问节点  $N_c = v_s$ , 上次访问节点  $N_p = \text{Null}$ , 将  $N_p$  压栈; 标记  $G(V, E)$  中所有节点状态均为“未访问”。

步骤 2 若  $N_c = \text{Null}$ , 则失败结束; 若  $N_c \neq v_d$ , 则转步骤 3, 否则转步骤 5。

步骤 3 置  $N_c$  状态为“已访问”。若  $N_c$  的所有相邻节点都处于“已访问”状态, 则弹出栈顶元素给  $N_p, N_c = N_p$ , 转步骤 2。

步骤 4 从  $N_c$  的相邻节点中任选一处于“未访问”状态的节点  $N_n$ ; 从  $N_c$  与  $N_n$  之间的边中任选一条作为从  $v_s$  到  $v_d$  路径的组成边, 将其编号作为解向量的元素; 将  $N_c$  压栈,  $N_c = N_n$ , 转步骤 2。

步骤 5 若路径满足式(18)一式(22), 则把其作为初始解, 成功结束; 否则, 失败结束。

$PT_{sd}$  的适值函数定义如下:

$$FT_{PT_{sd}} = \sum_{e_{ij}^k \in PT_{sd}} NS_{ij}^k \cdot PD_{ij}^{ky} \quad (28)$$

$$NS_{ij}^k = \begin{cases} 1, & \text{Nash 均衡} \\ < 1, & \text{非 Nash 均衡} \end{cases} \quad (29)$$

式中,  $NS_{ij}^k$  是 Nash 均衡调节系数。显然, 适值越大, 路径越好。

### 3.2 组织分裂

一个组织  $OG$  满足如下条件时进行分裂:

$$(|OG| > MX_w) \vee (|OG| \leq MX_w) \wedge (RD < \frac{|OG|}{N_{OG}}) \quad (30)$$

式中,  $|OG|$  是组织成员数,  $N_{OG}$  是初始种群的组数,  $MX_w$  是允许的组织最大成员数,  $RD$  是介于 0 和 1 之间的随机小数。分裂时, 从  $OG$  中随机选取  $o$  个成员形成组织  $OG_1$ , 其余成员形成组织  $OG_2$ , 在种群中以  $OG_1$  和  $OG_2$  取代  $OG, \frac{|OG|}{3} \leq o \leq \frac{|OG|}{2}$ 。

### 3.3 组织吞并

对  $OG_1 = \{x_1, \dots, x_m\}$  及其领导  $x^* = \{no^{*11}, \dots, no^{*1Q}\}, OG_2 = \{y_1, \dots, y_n\}$  及其领导  $y^* = \{no^{*21}, \dots, no^{*2A}\}, 1 \leq m, n \leq |E|$ , 若  $FT_{x^*} - FT_{y^*} > \delta$  即  $OG_1$  比  $OG_2$  足够强势, 则  $OG_1$  吞并  $OG_2$ , 产生  $OG_3 = \{z_1, \dots, z_m, z_{m+1}, \dots, z_{m+n}\}$ , 并在种群中以  $OG_3$  取代  $OG_1$  和  $OG_2, \delta$  是依据经验设定的组织势力差异阈值。

$$z_r = \begin{cases} x_i, & r=1, \dots, m \\ xy_j, & ((FT_{xy_j} \geq FT_{y_j}) \vee ((FT_{xy_j} < FT_{y_j}) \wedge (RD_j < e^{(FT_{xy_j} - FT_{y_j})})) \wedge (r=m+j)) \\ y_j, & \text{其他} \end{cases} \quad (31)$$

式中,  $RD_l$  是介于 0 和 1 之间的随机小数。对每个  $r=m+j$ , 随机产生介于 0 和 1 之间的小数  $RD_r$ , 若  $RD_r > OT$ , 则按式 (32) 产生与  $z_r$  对应的  $xy_j$ , 否则按式 (33) 产生,  $OT$  是根据经验预先设定的阈值。记  $y_j = \{no^{j1}, \dots, no^{jR}\}$ ,  $xy_j = \{no^{jv1}, \dots, no^{jvR}\}$ ,  $R = \max\{Q, J\}$ ,  $1 \leq Q, J, R \leq |E|$ , 则:

$$no^{jv_l} = \begin{cases} (\lfloor no^{*u} + RD_l \cdot (no^{*u} - no^{*l}) \rfloor \bmod |E|) + 1, & l=1, \dots, \min\{Q, J\} \\ (\lfloor no^{*u} + RD_l \cdot no^{*u} \rfloor \bmod |E|) + 1, & (Q > J) \wedge (l = (J+1), \dots, R) \\ (\lfloor RD_l \cdot no^{*l} \rfloor \bmod |E|) + 1, & (Q < J) \wedge (l = (Q+1), \dots, R) \end{cases} \quad (32)$$

$$no^{jv_l} = \begin{cases} \lceil (RD_l \cdot |E|) \rceil, & RD_l > \frac{1}{Q} \\ no^{*l}, & \text{其他} \end{cases} \quad (33)$$

### 3.4 组织合作

对  $OG_1$  及其领导  $x^*$ 、 $OG_2$  及其领导  $y^*$ , 若  $RD > OC$ , 则按式 (34) 和式 (35) 合作产生新个体  $xy_1$  和  $xy_2$ , 按式 (38) 和式 (39) 产生  $OG_3$  和  $OG_4$ , 否则按式 (36) 和式 (37) 产生  $OG_3$  和  $OG_4$ ; 在种群中以  $OG_3$  和  $OG_4$  取代  $OG_1$  和  $OG_2$ 。这里,  $RD$  是介于 0 和 1 之间的随机小数,  $OC$  是根据经验预先设定的阈值。

记  $xy_1 = \{no^{jv11}, \dots, no^{jv1R}\}$ ,  $xy_2 = \{no^{jv21}, \dots, no^{jv2R}\}$ ,  $R = \max\{Q, A\}$ ,  $1 \leq Q, A, R \leq |E|$ , 则:

$$no^{jv_{1l}} = \begin{cases} (\lfloor RD_l \cdot no^{*u} + (1-RD_l) \cdot no^{*2l} \rfloor \bmod |E|) + 1, & l=1, \dots, \min\{Q, A\} \\ no^{*u}, & (Q > A) \wedge (l = (A+1), \dots, R) \\ no^{*2l}, & (Q < A) \wedge (l = (Q+1), \dots, R) \end{cases} \quad (34)$$

$$no^{jv_{2l}} = \begin{cases} (\lfloor (1-RD_l) \cdot no^{*u} + RD_l \cdot no^{*2l} \rfloor \bmod |E|) + 1, & l=1, \dots, \min\{Q, A\} \\ no^{*u}, & (Q > A) \wedge (l = (A+1), \dots, R) \\ no^{*2l}, & (Q < A) \wedge (l = (Q+1), \dots, R) \end{cases} \quad (35)$$

$$xy_1 = \{no^{*11}, \dots, no^{*1i_1-1}, no^{*2i_1}, \dots, no^{*2i_2}, no^{*1(i_2+1)}, \dots, no^{*1Q}\} \quad (36)$$

$$xy_2 = \{no^{*21}, \dots, no^{*2j_1-1}, no^{*1j_1}, \dots, no^{*1j_2}, no^{*2(j_2+1)}, \dots, no^{*2A}\} \quad (37)$$

$$OG_3 = \begin{cases} x_1, \dots, x_{i-1}, xy_1, x_{i+1}, \dots, x_m, & (\exists x_i \in OG_1) \wedge (FT_{x_i} < FT_{xy_1}) \\ x_1, \dots, x_m, & \text{其他} \end{cases} \quad (38)$$

$$OG_4 = \begin{cases} y_1, \dots, y_{j-1}, xy_2, y_{j+1}, \dots, y_n, & (\exists y_j \in OG_2) \wedge (FT_{y_j} < FT_{xy_2}) \\ y_1, \dots, y_n, & \text{其他} \end{cases} \quad (39)$$

### 3.5 算法描述

基于组织进化的 ABC 支持型单播路由算法描述如下。

步骤 1 设最大进化代数数为  $EX$ , 组织最大成员数为  $OX$ ; 进化代数计数器  $ex=0$ ; 按 3.1 节生成具有  $ON$  个组织且每个组织只有 1 个成员的种群  $P_{ex}$ , 种群最大适应值记为  $FT^*$ , 与  $FT^*$  对应的领导记为  $L^*$  (若有多个, 则任选其一), 精英解集

$ES = \{\text{适值为 } L^* \text{ 的领导}\}$ 。

步骤 2 若  $ex < EX$ , 则转步骤 3。否则, 若  $|ES| = 1$ , 则返回其中的领导作为问题解; 若  $|ES| > 1$ , 则任选一领导作为问题解, 路由成功结束。

步骤 3 对  $P_{ex}$  中的每个组织, 按 3.2 节执行分裂操作。

步骤 4 若  $P_{ex}$  中的组织数大于 1, 则转步骤 5, 否则转步骤 6。

步骤 5 从  $P_{ex}$  中随机选取两个组织对其随机执行吞并或合作操作, 转步骤 4。

步骤 6 记当前种群最大适应值为  $FT_c^*$ , 与  $FT_c^*$  对应的领导记为  $L_c^*$ ; 若  $FT_c^* > FT^*$ , 则  $ES = \{\text{适值为 } L_c^* \text{ 的领导}\}$ ,  $FT^* = FT_c^*$ ; 若  $FT_c^*$  和  $FT^*$  相等, 则  $ES = ES \cup \{\text{适值为 } L_c^* \text{ 的领导}\}$ ; 从  $ES$  中任选其一作为  $L^*$ ;  $ex = ex + 1$ , 转步骤 2。

本算法时间复杂度是  $O(EX \cdot ON \cdot |E|)$ 。

## 4 性能评价

基于 NS2 (Network Simulator 2)<sup>[21]</sup> 仿真实现了上述路由机制。仿真时, 在路径每一跳上都设置 3 种不同类型的网络链路供选择, 在多个实际和虚拟网络拓扑上仿真运行本文机制和基于 Dijkstra 算法<sup>[22]</sup> 的单播路由机制 (以下分别简称 O 机制和 D 机制)。

图 1—图 4 是两种机制在 CERNET2 拓扑 (拓扑 1)、CERNET 拓扑 (拓扑 2)、NSFNET 拓扑 (拓扑 3) 和使用 Waxman2 模型<sup>[23]</sup> 生成的 30 节点网络拓扑 (拓扑 4) 上的性能比较结果。这里, QoS 路由由请求成功率系指成功完成 QoS 路由选择的请求数占 QoS 路由请求总数的百分比, Nash 均衡下 Pareto 最优比系指达到 Nash 均衡下 Pareto 最优的解占找到的解总数的百分比。从图中可以看出, 本文机制的性能更好。

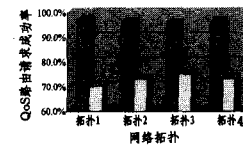


图 1 QoS 路由请求成功率

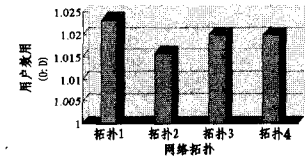


图 2 用户效用

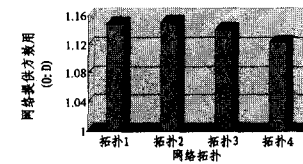


图 3 网络提供方效用

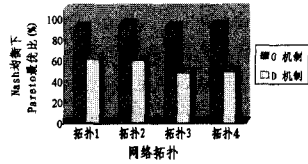


图 4 Nash 均衡下 Pareto 最优比

**结束语** 本文提出并基于 NS2 仿真实现了一种基于组织进化的 ABC 支持型 QoS 单播路由机制。在不精确网络链路状态和用户 QoS 需求下, 通过博弈, 寻找使用户和网络提供方效用达到或接近 Nash 均衡下 Pareto 最优的 QoS 单播路径。仿真结果表明, 本文机制是可行和有效的, 在 QoS 路由请求成功率、用户效用、网络提供方效用、Nash 均衡下 Pareto 最优比等性能指标上均表现良好。算法与模型的实用化、原型系统的开发、同其他智能 QoS 单播路由算法的比较以及对组播的扩展是今后研究的重点。

## 参考文献

[1] Gustafsson E, Jonsson A. Always best connected [J]. IEEE

- [2] Zahariadis T B, Vaxeankis K, Tsantilas C, et al. Global roaming in next-generation networks[J]. IEEE Communications Magazine, 2002, 40(2): 145-151
- [3] Fodor G, Eriksson A, Tuoriniemi A. Providing quality of service in always best connected networks[J]. IEEE Communications Magazine, 2003, 41(7): 154-163
- [4] 全贤唐, 张健. 经济博弈论[M]. 北京: 机械工业出版社, 2003: 1-9
- [5] Wang Z, Crowcroft J. Quality of service routing for supporting multimedia applications[J]. IEEE Journal on Selected Areas in Communications, 1996, 14(7): 1288-1294
- [6] Shariff A, Woodward M. A distributed algorithm for unicast QoS-routing using path feasibility prediction[C]// Proceedings of HPSR'06. Piscataway: IEEE Computer Society, 2006: 235-240
- [7] Koundinya K, Negi A, Sastry V. Unicast routing algorithm with multiple quality-of-service parameters[C]// Proceedings of IS-PAN'04. Piscataway: IEEE Computer Society, 2004: 520-525
- [8] Resende R, Lavelha A, Yamakami A, et al. A fuzzy algorithm to solve the problem of QoS unicast routing in IP networks[C]// Proceedings of ITS'06. Piscataway: IEEE Computer Society, 2006: 856-861
- [9] Dai F, Liu A. A multi-constrained quality of service routing algorithm based on vector converting [C] // Proceedings of WiCom'09. Piscataway: IEEE, 2009: 1-4
- [10] Hu Z, Zhou L, Gui Z. Delay-bandwidth constrained QoS unicast routing algorithms considering uncertain information[C]// Proceedings of ICC'05. Piscataway: IEEE Inc, 2005: 229-232
- [11] Leung K, Lui K, Leung K, et al. Quality-of-service routing with two concave constraints [C] // Proceedings of ICC'08. Piscataway: IEEE Inc, 2008: 5746-5750
- [12] Yang K, Wu Y, Chen H. QoS-aware routing in emerging heterogeneous wireless networks[J]. IEEE Communications Magazine, 2007, 45(2): 74-80
- [13] Song M, Sahni S. Approximation algorithms for multi-constrained quality-of-service routing [J]. IEEE Transactions on Computers, 2006, 55(5): 603-617
- [14] Bashandy A, Chong E, Ghafoor A. Generalized quality-of-service routing with resource allocation[J]. IEEE Journal on Selected Areas in Communications, 2005, 23(2): 450-463
- [15] Khadivi P, Samavi S, Todd T. Multi-constraint QoS routing using a new single mixed metrics[J]. Journal of Network and Computer Applications, 2008, 31(4): 656-676
- [16] 王兴伟, 秦培玉, 黄敏. 基于人工鱼群的 ABC 支持型 QoS 单播路由机制[J]. 计算机学报, 2010, 33(4): 718-725
- [17] 王兴伟, 侯美佳, 黄敏, 等. 一种基于博弈论方法的模糊 QoS 单播路由机制[J]. 计算机学报, 2007, 30(1): 10-17
- [18] 刘静, 钟伟才, 刘芳, 等. 组织进化数值优化算法[J]. 计算机学报, 2004, 27(2): 157-167
- [19] 肖德琴, 冯健昭, 周权, 等. 基于高斯分布的传感器网络信誉模型[J]. 通信学报, 2008, 29(3): 47-53
- [20] 施锡铨. 博弈论[M]. 上海: 上海财经大学出版社, 2000: 11-81
- [21] 徐雷鸣, 庞博, 赵耀. NS 与网络模拟[M]. 北京: 人民邮电出版社, 2003: 1-9
- [22] Dijkstra E W. A note on two problems in connection with graph [J]. Numerical Mathematics, 1959, 1(5): 269-271
- [23] Waxman B M. Routing of multipoint connections [J]. IEEE Journal on Selected Areas in Communications, 1988, 6(9): 1617-1622

(上接第 33 页)

- [22] Gottlob G, Leone N, Scarcello F. Robbers, marshals, and guards: game theoretic and logical characterizations of hypertree width[J]. J. Comput. System Sci., 2003, 66: 775-808
- [23] Grohe M, Marx D. Constraint Solving via Fractional Edge Covers[C]//SODA '06. Miami, FL, January 2006: 289-298
- [24] Freuder E C. A Sufficient Condition for Backtrack-bounded Search[J]. Artificial Intelligence, 1985, 32(4): 755-761
- [25] Montanari U. Networks of constraints: Fundamental properties and applications to picture processing[J]. Information Sciences, 1974, 7: 95-132
- [26] Mackworth A K. Consistency in Networks of Relations[J]. J. Artificial Intelligence, 1977, 8(1): 99-118
- [27] Freuder E. Synthesizing constraint expressions[J]. Communications of the ACM, 1978, 21(11): 958-966
- [28] Pearson J, Jeavons P G. A Survey of Tractable Constraint Satisfaction Problems[R]. CSD-TR-97-15. London: Royal Holloway, University of London, 1997
- [29] [http://en.wikipedia.org/wiki/Join-tree\\_clustering](http://en.wikipedia.org/wiki/Join-tree_clustering)
- [30] Dechter R, Pearl J. Tree clustering for constraint networks[J]. Artificial Intelligence, 1989, 38: 353-366
- [31] Gottlob G, Leone N, Scarcello F. Hypertree Decomposition and Tractable Queries[C]//Proc. 18th ACM SIGACT SIGMOD SIGART Symposium on Principles of Database Systems (PODS-99). Philadelphia, PA, 1999: 21-32
- [32] Gottlob G, Leone N, Scarcello F. On Tractable Queries and Constraints[C]//Proc. 12th International Conference on Database and Expert Systems Applications (DEXA 2001). Florence, Italy, 1999
- [33] Tarjan R, Yannakakis M. Simple linear-time algorithms to test chordality of graphs, test acyclicity of hyper graphs, and selectively reduce acyclic hyper graphs[J]. SIAM Journal on Computing, 1984, 13(3): 566-579
- [34] Gottlob G, Hutle M, Wotawa F. Combing Hypertree, Bicomplex, and Hinge Decomposition. [C]//Proc. of the 15 th. ECAI. Lyon, France, 2002: 161-165
- [35] Aho A V, Hopcroft J E, Ullman J D. The Design and Analysis of Computer Algorithms[M]. Addison-Wesley Publishing Company, 1974
- [36] Gompert J M, Choueiry B Y. A Decomposition Technique for CSPs Using Maximal Independent Sets and Its Integration with Local Search[C]// International FLAIRS Conference (FLAIRS'05). 2008
- [37] Boppana R, Halldórsson M M. Approximating maximum independent sets by excluding subgraphs[J]. BIT, 1992, 32(2): 180-196
- [38] <http://www.nada.kth.se/~viggo/wwwcompendium/node33.html>
- [39] Gottlob G, Samer M. A backtracking-based algorithm for hypertree decomposition[C]//ACM J. Exp. Algor., 2008, 13: 19
- [40] Gottlob G, Leone N, Scarcello F. Hypertree decompositions and tractable queries[J]. Journal of Computer and System Sciences, 2002, 64(3): 579-627
- [41] Gottlob G, Leone N, Scarcello F. Robbers, marshals, and guards: Game theoretic and logical characterizations of hypertree width[J]. J. Comput. Syst. Sci., 2003, 66(4): 775-808
- [42] Marx D. Approximating fractional hypertree width[C]// Proceedings of the Nineteenth Annual ACM-SIAM Symposium on Discrete Algorithms. 2009: 902-911