

NGI 中一种基于食物链算法的柔性 QoS 组播路由算法^{*}

王军伟^{1,2} 王兴伟¹ 黄敏¹ 刘彭程¹

(东北大学信息科学与工程学院 沈阳 110004)¹ (东北大学秦皇岛分校 秦皇岛 066004)²

摘要 针对下一代互联网 (Next Generation Internet, NGI) 难以精确测量和用户服务质量 (Quality of Service, QoS) 需求难以完全表达的特点, 设计了一种基于食物链算法 (Food Chain Algorithm, FCA) 的柔性 QoS 组播路由算法。给出了 QoS 组播路由问题模型及其数学描述, 针对 NGI 中 QoS 参数信息不精确和用户需求柔性的特点, 通过博弈分析确定用户和网络方在边上的效用能否达到 Nash 均衡, 基于模糊数学的相关知识并结合 FCA 的寻优能力, 找出在给定条件下用户效用、网络方效用和满足用户 QoS 需求的可信度同时达到最大的组播路由树。对算法进行了仿真实现与性能评价, 结果表明, 它是可行和有效的。

关键词 NGI, 柔性服务质量, 组播路由, 食物链算法, 博弈论, Nash 均衡

A Food Chain Algorithm Based Flexible QoS Multicast Routing Scheme in NGI

WANG Jun-Wei^{1,2} WANG Xing-Wei¹ HUANG Min¹ LIU Peng-Cheng¹

(School of Information Science and Engineering, Northeastern University, Shenyang 110004)¹

(Northeastern University at Qinhuangdao, Qinhuangdao 066004)²

Abstract Taking difficulty on the exact measurement of NGI (Next Generation Internet) status and the complete expression of the user QoS (Quality of Service) requirement into account, a flexible QoS multicast routing scheme based on FCA (Food Chain Algorithm) is presented. In this paper, the corresponding model and its mathematical description are introduced. Under the inaccurate network status information and the flexible user QoS requirement, whether Nash equilibrium between the user utility and the network provider utility can be achieved on the candidate edge is determined by gaming analysis. Combing knowledge of fuzzy mathematics and optimum searching ability of FCA, the proposed algorithm tries to find the multicast tree with the user utility, the network provider utility and the confidence degree on meeting with the user QoS requirement maximized. Simulation results have shown that the proposed algorithm is both feasible and effective.

Keywords NGI (Next Generation Internet), Flexible QoS (Quality of Service), Multicast routing, FCA (Food Chain algorithm), Game theory, Nash equilibrium

1 引言

下一代互联网 (Next Generation Internet, NGI) 的拓扑易变和网络参数的不确定性迫切需要设计合理的组播路由算法。已经证明, 基于多个约束条件的服务质量 (Quality of Service, QoS) 路由问题是 NP 问题^[1], 研究者对此类问题提出了很多解决方法。文[2]首先设定链路和目的节点的 QoS 级别, 把链路费用定义为 QoS 级别的线性函数, 然后用启发式算法得到费用最小的组播树。在考虑延迟和延迟抖动的基础上, 文[3]以最小化网络负载和带宽消耗为目标, 提出了一种可扩展的组播路由算法。文[4]通过引入概率模型来解决在非精确状态信息下带宽和延迟受限的路径选择问题。文[5]对网络组播服务提出了一种通用模型结构, 设计了无环路组播路由协议。文[6,7]分别将遗传 (Genetic Algorithm, GA) 和粒子群等智能优化算法引入 QoS 组播路由计算中, 取得了较好结果。

上述路由算法大都假设网络中的节点已获得网络的精确 QoS 参数信息, 然而在实际的网络环境中, 节点所能获得的网络状态并非完全精确, 文[4]虽然引入了参数的不精确性, 但

文章中考虑的 QoS 参数较少。另外, 算法都是基于刚性 QoS 约束的组播路由算法, 而组播用户的需求往往是弹性的, QoS 参数属于某个区间都能满足用户要求。为此, 本文设计了一种基于食物链算法 (Food Chain Algorithm, FCA)^[8] 的柔性 QoS 组播路由算法, 通过引入模糊数学知识适应 QoS 参数不精确的需要, 基于 FCA 进行优化操作, 在满足各用户柔性 QoS 约束的前提下寻求用户效用、网络方效用和满足 QoS 约束的可信度同时达到最大的最优 (或近优) 组播路由树。

2 数学模型

2.1 问题模型

已知图 $G(V, E)$, V 是节点集, E 是边集。 $\forall v \in V$, 考虑如下参数: 延迟、延迟抖动和丢包率; $\forall e \in E$, 考虑如下参数: 可用带宽、延迟和出错率。简单起见, 本文把节点参数归并到边参数中。这样, 边 e 上的参数变为: 可用带宽 $bw(e) \in [bw_{min}, bw_{max}]$ 、延迟 $dl(e) \in [dl_{min}, dl_{max}]$ 、延迟抖动 $jt(e) \in [jt_{min}, jt_{max}]$ 和出错率 $ls(e) \in [ls_{min}, ls_{max}]$ 。采用区间形式表示各 QoS 参数是为了适应参数信息难以精确表达的需要。

为支持柔性 QoS, 定义用户 QoS 请求为 $(s, M, \Delta_{bw}, \Delta_{Dl},$

^{*} 新世纪优秀人才支持计划资助项目; 国家发改委 CNGI 示范工程资助项目 (CNGI-04-13-2T, CNGI-04-6-2T 和 CNGI-04-15-7A); 国家自然科学基金资助项目 (60473089)。王军伟 博士生, 主要研究方向: 下一代互联网、QoS 路由算法; 王兴伟 教授, 博士生导师, 主要研究方向: 下一代互联网、IP/DWDM 光因特网及移动因特网; 黄敏 教授, 博士生导师, 主要研究方向: 企业风险管理、优化模型算法等; 刘彭程 硕士生。

$\Delta_{J_i}, \Delta_{L_s}, \Delta_C$ 。其中, $s \in V$ 是源节点, $M \subseteq V$ 是组播目的节点集, 其所包含节点个数为 $|M|$; $\Delta_{Bw} = (\Delta_{Bw_1}, \Delta_{Bw_2}, \dots, \Delta_{Bw_{|M|}})$ 为用户的带宽约束向量, 其中, $\Delta_{Bw_k} = [\Delta_{Bw_k}^{low}, \Delta_{Bw_k}^{high}]$ ($k=1, 2, \dots, |M|$) 表示第 k 个用户的带宽约束区间。类似地, $\Delta_{Dl} = (\Delta_{Dl_1}, \Delta_{Dl_2}, \dots, \Delta_{Dl_{|M|}})$, $\Delta_{Jt} = (\Delta_{Jt_1}, \Delta_{Jt_2}, \dots, \Delta_{Jt_{|M|}})$ 和 $\Delta_{Ls} = (\Delta_{Ls_1}, \Delta_{Ls_2}, \dots, \Delta_{Ls_{|M|}})$ 分别为用户端到端延迟、延迟抖动和出错率约束向量, $\Delta_{Dl_k} = [\Delta_{Dl_k}^{low}, \Delta_{Dl_k}^{high}]$, $\Delta_{Jt_k} = [\Delta_{Jt_k}^{low}, \Delta_{Jt_k}^{high}]$ 和 $\Delta_{Ls_k} = [\Delta_{Ls_k}^{low}, \Delta_{Ls_k}^{high}]$ 分别表示第 k 个用户的延迟、延迟抖动和出错率约束区间。 $\Delta_C = (C_1, C_2, \dots, C_{|M|})$ 为费用约束向量, 其中, C_k 表示第 k 个用户所愿支付的费用上限。要求构造支持 s 与 M 间通信的一棵 QoS 组播路由树 $T(s, M)$ 。

2.2 不精确参数处理

给定不精确参数 σ 对应不精确区间为 $[\sigma_L, \sigma_H]$, 设其对应一个三角模糊数 $A(\sigma) = (a, c)$, 其隶属函数定义为^[9]:

$$\mu_A(x) = \begin{cases} 1 - \frac{|x-a|}{c} & x \in (a-c, a+c) \\ \epsilon & x = a-c \vee x = a+c \\ 0 & \text{其他} \end{cases} \quad (1)$$

其中, $a = \frac{\sigma_L + \sigma_H}{2}$, $c = \frac{\sigma_H - \sigma_L}{2}$, ϵ 为一远小于 1 的正常数。

对于 QoS 约束区间 Δ , 设其对应一个模糊集 $F(\Delta) = (a, c)$, 其隶属函数定义为:

$$\mu_F(x) = \begin{cases} 1 & x \in [a-c, a+c] \\ 0 & \text{其他} \end{cases} \quad (2)$$

其中, $a = \frac{\Delta^{low} + \Delta^{high}}{2}$, $c = \frac{\Delta^{high} - \Delta^{low}}{2}$ 。

由模糊数学的理论可知, 两个模糊集 A_1 和 A_2 的贴进度为:

$$T(A_1, A_2) = (\bigvee_{x \geq 0} (\mu_{A_1}(x) \wedge \mu_{A_2}(x))) \wedge (\bigwedge_{x \geq 0} (\mu_{A_1}(x) \vee \mu_{A_2}(x))) \quad (3)$$

将 A_1 小于 A_2 的可信度 $R(A_1, A_2)$ 定义为:

$$R(A_1, A_2) = \begin{cases} 0.5 & a_1 = a_2 \\ 1 - 0.5T(A_1, A_2) & a_1 < a_2 \\ 0.5T(A_1, A_2) & a_1 > a_2 \end{cases} \quad (4)$$

令 $P(s, m_k)$ 表示在 $T(s, M)$ 上从 s 到 $m_k \in M$ ($k=1, 2, \dots, |M|$) 的通信路径, 其带宽满足约束的可信度定义为:

$$R_{Bw}(m_k) = \text{Min}\{R(A(F(\Delta_{Bw_k}), Bw(e))) \mid \forall e \in P(s, m_k)\} \quad (5)$$

令 $dl(m_k)_{\min} = \sum_{e \in P(s, m_k)} dl_{\min}$ 和 $dl(m_k)_{\max} = \sum_{e \in P(s, m_k)} dl_{\max}$ 分别表示 $P(s, m_k)$ 的最小和最大延迟, 则 $P(s, m_k)$ 延迟对应的模糊数为:

$$A(dl(m_k)) = (\frac{dl(m_k)_{\min} + dl(m_k)_{\max}}{2}, \frac{dl(m_k)_{\max} - dl(m_k)_{\min}}{2}) \quad (6)$$

$P(s, m_k)$ 延迟满足约束的可信度定义为:

$$R_{Dl}(m_k) = R(A(dl(m_k)), F(\Delta_{Dl_k})) \quad (7)$$

同理, $P(s, m_k)$ 延迟抖动满足约束的可信度定义为:

$$R_{Jt}(m_k) = R(A(jt(m_k)), F(\Delta_{Jt_k})) \quad (8)$$

令 $ls(m_k)_{\min} = 1 - \prod_{e \in P(s, m_k)} (1 - ls_{\min})$ 和 $ls(m_k)_{\max} = 1 - \prod_{e \in P(s, m_k)} (1 - ls_{\max})$ 分别表示 $P(s, m_k)$ 的最小和最大出错率, 则 $P(s, m_k)$ 出错率对应的模糊数为:

$$A(ls(m_k)) = (\frac{ls(m_k)_{\min} + ls(m_k)_{\max}}{2}, \frac{ls(m_k)_{\max} - ls(m_k)_{\min}}{2}) \quad (9)$$

$P(s, m_k)$ 出错率满足约束的可信度定义为:

$$R_{Ls}(m_k) = R(A(ls(m_k)), F(\Delta_{Ls_k})) \quad (10)$$

$P(s, m_k)$ 满足 QoS 约束的可信度定义为:

$$R(m_k) = \text{Min}\{R_l(m_k) \mid \forall l \in \{Bw, Dl, Jt, Ls\}\} \quad (11)$$

$T(s, m_k)$ 满足 QoS 约束的可信度定义为:

$$R(T) = \text{Min}\{R(m_k) \mid \forall m_k \in M\} \quad (12)$$

2.3 用户 QoS 满意度

2.3.1 隶属函数

采用模糊数学的三分法为每个 QoS 参数 σ 设置三个模糊集 $F_l(\sigma)$, $F_m(\sigma)$ 和 $F_h(\sigma)$, 分别对应着用户对当前链路满意度低 L 、中等 M 和高 H 。

对于边 e , 其带宽满意度低 ($I=Bw$), 或者其延迟、延迟抖动及出错率 ($I=Dl, Jt, Ls$) 满意度高对应的隶属函数为:

$$\mu_l(x) = 1 - \Phi\left(\frac{x-I_1}{\beta_l}\right) \quad (13)$$

对于边 e , 其带宽、延迟、延迟抖动或者出错率 ($I=Bw, Dl, Jt, Ls$) 满意度中等对应的隶属函数为:

$$\mu_m(x) = \Phi\left(\frac{x-I_1}{\beta_l}\right) - \Phi\left(\frac{x-I_2}{\beta_l}\right) \quad (14)$$

对于边 e , 其带宽满意度高 ($I=Bw$), 或者其延迟、延迟抖动及出错率 ($I=Dl, Jt, Ls$) 满意度低对应的隶属函数为:

$$\mu_h(x) = \Phi\left(\frac{x-I_2}{\beta_l}\right) \quad (15)$$

其中, $\Phi(x) = \int_{-\infty}^x \frac{1}{\sqrt{2\pi}} e^{-\frac{t^2}{2}} dt$, β_l ($I=Bw, Dl, Jt, Ls$) 为正常数。

对于给定边 e 保证的带宽约束区间 Δ_{Bw_i} , 用户对其满意度低、中等和高的隶属度分别为: $T(F(\Delta_{Bw_i}), F_l(Bw))$, $T(F(\Delta_{Bw_i}), F_m(Bw))$ 和 $T(F(\Delta_{Bw_i}), F_h(Bw))$ 。类似地, 可以定义出给定其他 QoS 参数保证区间所对应用户满意度低、中等和高的隶属度。

2.3.2 模糊规则

定义 IF-THEN 模糊规则为:

表 1 模糊规则

规则	Bw	Dl	Jt	Ls	模糊结果
1	L	L	L	L	EL
2	L	L	L	M	EL
3	L	L	L	H	VL
4	L	L	M	M	VL
5	L	L	M	H	L
6	L	L	H	H	M
7	L	M	M	M	L
8	L	M	M	H	M
9	L	M	H	H	H
10	L	H	H	H	VH
11	M	M	M	M	M
12	M	M	M	H	H
13	M	M	H	H	VH
14	M	H	H	H	EH
15	H	H	H	H	EH

在表 1 中, EL, VL, L, M, H, VH 和 EH 分别表示模糊结果的隶属度为: 特低、很低、低、中等、高、很高和特高。例如, 规则 2 的含义是: 在 4 种参数中有三个对应用户满意度为低而另一个对应用户的满意度中等情况下, 其模糊结果为 EL 。其它规则意义与其类似。

模糊规则中存在两条规则对应同一个模糊结果的情况,

此时的模糊结果定义为其对应各种参数组合结果的最大值。

2.3.3 用户 QoS 满意度计算

各模糊结果的中心点定义为： $el=0, vl=16, l=33, m=50, h=66, vh=83, eh=100$ 。

则用户 k 对 e 的满意度定义为：

$$Q_k(e) = \frac{el * EL + vl * VL + l * Li + m * Me + h * Hi + vh * VH + eh * EH}{100 * (EL + VL + L + M + H + VH + EH)} \quad (16)$$

2.4 博弈分析

采用分级 QoS 付费方式。将边 e 保证的 QoS 分为 m 个级别, $p_i(e) (i=1, 2, \dots, m)$ 分别表示 e 为用户提供第 i 级 QoS 保证时用户所需支付的费用。

在 e 上, 对用户与网络方进行博弈。每个用户的策略集是 m 个 QoS 级别: Q_1, Q_2, \dots, Q_m 以及放弃当前链路 Q_{m+1} 。网络方的策略集 $\{NS_1, NS_2\}$ 表示网络方愿意和不愿意提供当前的链路。定义 e 上的效用矩阵 $[a_{ij}, b_{ij}]_{(m+1) \times 2}$, 矩阵元素 (a_{ij}, b_{ij}) 表示: 在策略对 (Q_i, NS_j) 下, 用户和网络方的效用分别为 a_{ij} 和 b_{ij} 。博弈的目的是计算哪个策略对能使用户和网络方的效用都达到“最优”。

在 (Q_i, NS_j) 下, e 上第 k 个用户的效用 a_{ij} 为:

$$UU_k^i = \begin{cases} (Q_k(e) - \omega) * \frac{C_k}{p_i(e)} & i \neq m+1 \wedge j=1 \\ \omega & i \neq m+1 \wedge j=2 \\ -\mu_1 (Q_k(e) - \omega) & i = m+1 \wedge j=1 \\ 0 & i = m+1 \wedge j=2 \end{cases} \quad (17)$$

其中, μ_1 为惩罚因子, 取值大于 1, 表示一个用户拒绝网络方受到的影响, ω 为用户期望的满意度值。

网络方的效用 b_{ij} 为:

$$UN_k^j = \begin{cases} \frac{p_i(e) - C_e(Q_i) - f(i)}{C_e(Q_i)} & i \neq m+1 \wedge j=1 \\ -\mu_2 \frac{p_i(e) - C_e(Q_i)}{C_e(Q_i)} & i \neq m+1 \wedge j=2 \\ 0 & i = m+1 \wedge j=1 \\ 0 & i = m+1 \wedge j=2 \end{cases} \quad (18)$$

其中, $C_e(Q_i)$ 表示在 e 上提供第 i 级 QoS 保证的成本费用; f

$(i) = K * \frac{\Delta bw_i(e)}{bw_{max} - bw_{min}}$, 表示为用户提供第 i 级 QoS 保证时由于对其他用户影响或阻塞其它用户而造成的损失费用, $\Delta bw_i(e)$ 表示为用户提供第 i 级 QoS 保证所需要分配的带宽区间; μ_2 为惩罚因子, 取值大于 1, 表示网络方拒绝一个用户受到的影响; K 为常数。

用上述方法计算出 e 的效用矩阵中的每一个元素值, 比较矩阵中的所有元素值, 根据 Nash 均衡的定义^[10], 效用矩阵中达到 Nash 均衡的效用对满足:

$$\begin{cases} a_{i,j} \geq a_{ij} & i=1, 2, \dots, m \\ b_{i,j} \geq b_{i,j} & j=1, 2 \end{cases} \quad (19)$$

如果找到了这样的元素, 则它所对应的策略对 (Q_i, NS_j) 就是 e 上的 Nash 均衡策略。但是, 纯策略 Nash 均衡不一定存在, 而且还可能存在多个 Nash 均衡。所以, 当两个元素都是或都不是 Nash 均衡时, 还要比较它们的 Pareto 优势。

$$pa_{ij} = \frac{1}{(\alpha * \frac{1}{a_{ij}} + \beta * \frac{1}{b_{ij}})} \quad (20)$$

则 e 上最优策略下的 Pareto 值为:

$$pa(e) = \max\{pa_{ij} | i \in 1, 2, \dots, m+1 \wedge j=1, 2\} \quad (21)$$

其所对应的策略对就是要在该链路上采用的用户和网络方策略。

2.5 数学模型

本算法的优化目标为: 在满足所有用户的各种端到端 QoS 约束的前提下, 使用户的端到端效用、网络方效用和满足用户 QoS 约束的可信度同时达到最大。

定义第 k 个用户在 $P(s, m_k)$ 上的费用为:

$$c(m_k) = \sum_{v \in P} p_i(e) \quad (22)$$

定义第 k 个用户在 $P(s, m_k)$ 上的效用为:

$$TU(P(s, m_k)) = \frac{\sum_{v \in P} UU_k^i}{Jp} \quad (23)$$

其中, Jp 是 $P(s, m_k)$ 的跳数。

定义 $T(s, M)$ 上的用户效用为:

$$TU(T) = \frac{\sum_{i=1}^{|M|} TU(p(s, m_i))}{|M|} \quad (24)$$

定义 $T(s, M)$ 上的网络方效用为:

$$TN(T) = \frac{\sum_{v \in T} UN_k^j}{Ne(T)} \quad (25)$$

其中, $Ne(T)$ 表示树 T 上的边的数目。

算法的数学模型描述如下:

$$\text{Max}\{R(T)\} \quad (26)$$

$$\text{Max}\{TU(T)\} \quad (27)$$

$$\text{Max}\{(TN(T))\} \quad (28)$$

$$s. t. \quad \{bw_{max} | v \in P(s, m_k)\} \geq \Delta_{bw_k}^{hu} \quad \forall m_k \in M \quad (29)$$

$$dl(m_k)_{min} \leq \Delta_{D_k}^{hgh} \quad \forall m_k \in M \quad (30)$$

$$jt(m_k)_{min} \leq \Delta_{J_k}^{hgh} \quad \forall m_k \in M \quad (31)$$

$$ls(m_k)_{min} \leq \Delta_{L_k}^{hgh} \quad \forall m_k \in M \quad (32)$$

$$c(m_k) \leq C_k \quad \forall m_k \in M \quad (33)$$

3 算法实现

3.1 人工生命定义

将图 $G(V, E)$ 划分为 Y 个不相交且各自连通的子图, 分别对应为 Y 种不同生物群体: Cr_1, Cr_2, \dots, Cr_Y , 两子图间的双边表示多种传递食物途径。人工生命基本特征定义如下:

h_i^0 : Cr_i 初始能量水平;

h_i : Cr_i 当前能量水平, 对应 Cr_i 中能量最大的个体的能量水平;

$g_i = \log_{1+\eta}(\frac{h_i}{h_i^0})$: Cr_i 的当前级别, η 为常数;

类似地:

h_{ij}^0 : Cr_i 中第 j 个个体的初始能量水平;

h_{ij} : Cr_i 中第 j 个个体的当前能量水平。

从目的节点对应的生命(以下简称为目的生命)开始, 向源节点对应的生命(以下简称为源生命)传递食物, 当源生命得到来自所有目的生命的食物时, 从源生命到所有目的生命的一组食物链构成一个解。

当一个生物群体消化完食物后, 其排出的废物成为其邻域内多个相邻生物群体沿不同的边竞争的食物。定义竞争到食物的群体中, 得到食物的生命为当前群体的捕食生命, 排放废物的生命为当前群体的排放生命。

食物在食物链上传递过程中其所含能量逐渐减少, 使用

生态效率 $\rho_n (0 < \rho_n < 1)$ 表示第 $n+1$ 个得到食物的生物群体得到食物的能量占第 n 个得到食物的群体得到食物的能量比值。因此,在一条食物链上,第 i 个群体所得到食物的能量为:

$$h_i = h_0^0 * \prod_{j=0}^{i-1} \rho_j \quad (34)$$

本文所设计的食物链算法主要包括群体之间的食物竞争规则,群体内部的食物共享规则和能量累积规则三个部分。随着时间的推移,获得食物较多的群体将存活下来;同样地,在每个生物群体内部,由于长期分享食物一些生命也得以存活,这样,存活下来的生命构成 $|M|$ 条食物链,其所对应的组播树即为问题的最终解。

3.2 竞争规则

当 Cr_i 消化完食物后排出废物,设 Cr_j 有 n 种获得此食物的途径(即有 n 条边与 Cr_i 相连), $(i, j = 1, 2, \dots, Y)$, Cr_j 通过 $e_r (r = 1, 2, \dots, n)$ 获得食物的概率定义为:

$$gf_{ij}^r = \frac{h_j * pa(e_r)}{\sum_{j \in \delta \wedge j \neq i} \sum_{r=1}^n h_j * pa(e_r)} \quad (35)$$

其中, δ 为 Cr_i 的相邻群体集合。

对于竞争 Cr_i 产生的食物的两个群体 j 和 k , Cr_j 按以下概率杀死 Cr_k :

$$kill_{jk} = \begin{cases} 1 - \frac{1}{1 + (\frac{g_i}{g_k} - a)^2} & \frac{g_i}{g_k} \geq a \\ \frac{1}{a} < \frac{g_i}{g_k} < a \\ \frac{1}{1 + (\frac{g_k}{g_j} - a)^2} - 1 & 0 < \frac{g_i}{g_k} \leq \frac{1}{a} \end{cases} \quad (36)$$

其中, a 为常数,且 $a > 1$, 控制算法的收敛速度。当 $kill_{jk}$ 为负数时表示 Cr_j 被 Cr_k 杀死的概率为 $-kill_{jk}$ 。当一个群体被杀死时,群体中所有的个体全部消亡。

3.3 共享规则

当生物群体内某生命得到食物后,其它生命可以向其发送共享食物请求,捕食生命将食物传递到要求共享食物的生命,传递过程中经过的生命共享这一食物。群体内部传递路径的选择规则和群体之间竞争食物的规则相同。在生物群体内部,共享食物的原则是平均分配捕食生命所得到的食物能量。

Cr_i 中第 j 个个体通过共享所获得的能量与 Cr_i 中捕食生物所获得的能量 h_{ij}^0 的关系为:

$$h_{ij}^1 = \frac{h_{ij}^0}{m} \quad (37)$$

其中, m 为共享食物的生命个数。

3.4 积累规则

当且仅当源生命得到食物构成一条食物链时,该链上的生命才可以累加它所得到的食物的能量。能量积累到一定程度,生物个体将成长。

Cr_i 中第 j 个个体的第 k 次成长能量水平与初始能量的关系为:

$$h_{ij}^k = (1 + \rho)^k h_{ij}^0 \quad (38)$$

Cr_i 中个体 j 在一个时间周期内的能量消耗定义为:

$$h_{ij}^c = \lambda_i h_{ij} \quad (39)$$

Cr_i 中第 j 个个体的消亡能量水平为:

$$h_{ij}^d = (1 - \tau) h_{ij}^0 \quad (40)$$

其中, ρ, τ 为常数,用于控制算法的收敛速度; λ_i 为 Cr_i 的能量消耗系数。如果生物个体 j 的能量水平低于其消亡能量水平时,则该生物消亡。当一个群体中所有个体全部消亡时,定义该生物群体消亡。

3.5 Pareto 最优解集

将解看成是一个向量,它有多个分量。对于两个解 u 和 v , 定义如下四种比较规则用来比较哪个解的质量更优:

- ① $u > v: ui \geq vi, \exists ui \neq vi, i = 1, 2, 3, \dots$;
- ② $u = v: ui = vi, i = 1, 2, 3, \dots$;
- ③ $u < v: ui \leq vi, \exists ui \neq vi, i = 1, 2, 3, \dots$;
- ④ 其他情况称为 u 和 v 等价。

本文中,一个解 X 具有如下分量, $x_1 = R(T), x_2 = TU(T), x_3 = TN(T)$, 分别反映优化目标式(26)~(28)。

初始化 Pareto 最优解集为空,算法中产生的解都与库中的所有解比较。①如果库为空,则直接将该解放入库中;②如果该解的质量优于库中某个解,则把该解加入库中同时删除库中的原有解;③如果该解的质量与库中所有解都等价,则同样把该解加入库中;④否则不修改库。

3.6 算法描述

步骤 1: 初始化网络拓扑信息,进行生物群体划分,为每个生命设置其初始能量, Pareto 最优解集为空,组播目的节点数为 J ;

步骤 2: 在每个目的生命处置 I 个食物, $i = 0$;

步骤 3: $i = i + 1$, 如果 $i > I$, 转步骤 9; 否则, $j = 0$, 转步骤 4;

步骤 4: $j = j + 1$, 如果 $j > J$, 转步骤 7; 否则置当前群体为第 j 个目的生命所在群体;

步骤 5: 如果源生命在当前群体内,则它向当前群体的捕食生命发出共享食物请求,捕食生命按式(37)将食物共享给源生命,转步骤 4; 否则,转步骤 6;

步骤 6: 相邻群体依式(35)竞争当前群体所排出的食物,当前群体的捕食生命按式(37)将食物共享排放生命,置胜出群体为当前群体,转步骤 5;

步骤 7: 对于当前从源生命到所有目的生命的食物链构成的解 T , 判断是否满足式(29)~(33)的要求,若不满足要求,转步骤 3; 否则转步骤 8。

步骤 8: 依 3.5 节的描述更新 Pareto 最优解集, T 上生命按式(34)积累能量,按式(39)消耗所有生命的能量,根据式(38)和(40)判断各生命的成长和消亡,转步骤 3。

步骤 9: 对 Pareto 最优解集中的解进行归一化处理,输出归一化结果最大的解作为最优组播树,算法结束;如果 Pareto 最优解集为空,则无法找到可行解,与用户进行协商后重新运行算法。

4 仿真实现与结果分析

基于 NS2(Network Simulator 2)平台进行了仿真实现,在多个实际与虚拟的网络拓扑(如 NSFNET、CERNET 和 CERNET2)上运行本文设计的算法(以下简称 F 算法)、文[11]中基于 GA 算法的 QoS 组播路由算法(以下简称 G 算法)和基于 Prim 的组播路由算法(以下简称 P 算法),进行仿真实现与性能评价。仿真时,主要参数取值如下: $\epsilon = 0.0005$, $m = 4$, $\omega = 0.75$, $\mu_1 = \mu_2 = 5$, $K = 20$, $\alpha = 0.5$, $\beta = 0.5$, $\eta = 1$, $\rho = 1$, $\tau = 0.95$, $a = 6$ 。

(下转第 57 页)

值空间和收敛性。有许多文章对解决寻找全局最优解和提高收敛速度之间的矛盾进行了研究,可以借鉴这些文章提出的方法对 IACSMR 算法进行改良。

结论 针对网络电视、视频点播类型的大规模实时流媒体应用,本文提出一个两层应用层组播框架结构 ALMF,并对其中 MSN 覆盖网络的构建和维护进行了研究,提出了一个 MSN 覆盖网络路由由组播模型 MOSSMRM 和相应的基于智能蚁群的路由算法 IACSMR。仿真实验表明,IACSMR 算法是一个有效的应用层组播路由算法。我们下一步的努力方向是基于 ALMF 框架构建一个具体的组播应用,在该应用中实现 MOSSMRM 模型和 IACSMR 算法并在真实的互联网环境中进行测试。

参考文献

- 1 Yeo C K, Lee B S. A survey of application level multicast techniques[J]. Computer Communications,2004,1547~1568
- 2 Zhang Beichuan, Sugih Jamin. Host Multicast: A Framework for Delivering Multicast to End Users[C]. In: Proceedings of IEEE

- INFOCOM, 2002. 1366~1375
- 3 Pendakaris D, Shi S. ALMI: an application level multicast infrastructure[C]. In: Anderson T, ed. The 3rd USENIX Symposium on Internet Technologies and Systems. San Francisco, CA, USA: USENIX Association, 2001. 49~60
- 4 Chu Yang-hua, Rao S G. A Case for End System Multicast[J]. Selected Areas in Communications, 2002, 20(8): 1456~1471
- 5 Luo Chong, Li Jiang. DigiMetro - An Application-Level Multicast System for Multi-party Video Conferencing[C]. In: IEEE Globecom, 2004
- 6 Shi S, Turner J. Dimension Server Access Bandwidth and Multicast Routing in Overlay Networks[C]. In: 11th International Workshop on Network and Operating System Support for Digital Audio and Video(NOSSDAV01), June 2001
- 7 Shi S, Turner J. Routing in Overlay Multicast networks[C]. In: Proceedings of IEEE INFOCOM, 2002
- 8 潘耘,余镇危,等. 求解应用层组播路由问题的遗传算法[J]. 小型微型计算机系统, 2005, 26(1): 55~58
- 9 Hosseini M, Georganas N D. Design of a multi-sender 3D video-conferencing application over an end system multicast protocol [C]. In: Proceedings of the Eleventh ACM International Conference on Multimedia, Berkeley, CA, USA, November 2003
- 10 吴启迪,汪毓. 智能蚁群算法及应用[M]. 上海科技教育出版社, 2004. 4
- 11 Bauer F, Varma A. Degree-Constrained Multicasting in Point-to-Point Networks[C]. In: IEEE INFOCOM'95, 369~376

(上接第 33 页)

在一段时间内成功的路由请求占总组播请求的比率,称为请求成功率。三种算法请求成功率的比较结果如图 1 所示。可以看出,运行 F 算法和 G 算法得到的组播路由树对网络状态不精确有着更好的适应性, F 算法的请求成功率大于另外两种算法。P 算法由于没有网络状态不精确问题而体现出较差的性能。

三种算法所求得的组播树满足用户 QoS 约束的可信度的比较结果如图 2。仿真结果表明, F 算法的满足用户 QoS 约束的可信度明显优于另外两种算法,并且随着网络拓扑的复杂化,优化目标值没有明显的波动,说明算法的扩展性很好。G 算法仅仅考虑了 QoS 中延迟一个参数,而 P 算法没有考虑 QoS 参数,所以结果不理想。

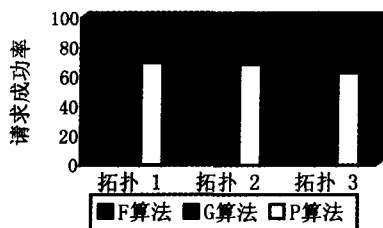


图 1 请求成功率

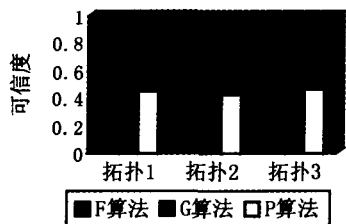


图 2 满足用户 QoS 约束的可信度

以拓扑 2 为例并以 P 算法为基准,比较三种算法的用户效用和网络方效用,结果如表 2 所示。相较而言, F 算法能更好地保证用户和网络方效用的最大化,这是因为该算法引入了博弈理论和动态 Pareto 最优解集,更好的保证了多个优化目标的同时最大化。

结束语 本文在充分考虑了 NGI 网络状态不精确和用

户柔性 QoS 需求的前提下,设计了一种基于 FCA 的 QoS 组播路由算法。引入模糊数学的理论适应网络状态不精确的需要,采用用户效用和网络方效用的博弈建立选路原则,利用 FCA 的寻优能力,找出在给定条件下用户效用、网络方效用和满足用户 QoS 需求可信度同时达到最大的组播路由树。仿真结果证明了该算法的可行性和有效性。

表 2 用户效用和网络方效用的比较

组播节点比例	F 算法 : G 算法 : P 算法	
	用户效用	网络方效用
16.2%	1.206 : 1.206 : 1	1.183 : 1.183 : 1
32.4%	1.303 : 1.225 : 1	1.211 : 1.15 : 1
48.6%	1.215 : 1.18 : 1	1.253 : 1.126 : 1
64.8%	1.253 : 1.197 : 1	1.149 : 1.117 : 1
81.0%	1.174 : 1.129 : 1	1.24 : 1.203 : 1
97.3%	1.28 : 1.209 : 1	1.177 : 1.114 : 1

参考文献

- 1 Wang Zheng, Jon C. Quality of Service for Supporting Multimedia application. IEEE journal on selected areas in communication, 1996, 14(7): 1228~1234
- 2 Charikar M, Naor J, Schieber B. Resource Optimization in QoS Multicast Routing of Real-Time Multimedia. IEEE/ACM transaction on networking, 2004, 12(2): 340~348
- 3 Moussaoui O, Ksentini A, Naimi M, Gueroui A. Multicast Tree Construction with QoS Guaranties. Springer LNCS 3754, 2005. 96~108
- 4 Korkmaz T, Krunz M. Bandwidth-Delay Constrained Path Selection Under Inaccurate State Information. IEEE/ACM transaction on networking, 2003, 11(3): 384~398
- 5 Zappala D. Alternate Path Routing for Multicast. IEEE/ACM transaction on networking, 2004, 12(1): 30~43
- 6 koyama A, Nishie T, Arai J, Barolli L. A New Quality of Service Multicast Routing Protocol Based on Genetic Algorithm. IEEE ICPADS'05, 2005. 1~6
- 7 Wang Junwei, Wang Xingwei, Huang Min. A Hybrid Intelligent QoS Multicast Routing Algorithm in NGI. IEEE PDCAT', 2005. 723~727
- 8 喻海飞,汪定伟. 食物链算法及其在供应链中的应用. 系统仿真学报, 2005, 17(5): 1195~1199
- 9 张品, 李民乐, 王晟. 模糊参数下多播 QoS 路由及分解. 计算机学报, 2006, 29(2): 279~285
- 10 谢识予. 经济博弈论. 上海: 复旦大学出版社, 2002
- 11 Ping C, Tianling D. A fuzzy genetic algorithm for QoS multicast routing. Computer communications, 2003, 26: 506~512