

基于自适应变异蚁群算法的 QoS 路由算法

古明家, 宣士斌, 廉侃超, 李永胜

(广西民族大学数学与计算机科学学院, 南宁 530006)

摘要: 多约束 QoS 单播路由问题是 NP 完全问题, 针对基本蚁群算法在解决该问题时易于陷入局部最优、收敛速度慢的缺点, 提出自适应变异蚁群算法对该问题进行求解。该算法采取自适应变异方法, 引入二次蚁群搜索机制, 减少了算法陷入局部极值的可能性, 提高了算法的寻优能力和收敛速度。仿真实验结果验证了该算法的可行性和有效性。

关键词: 多约束 QoS; 单播路由; 自适应变异; 蚁群算法

QoS Routing Algorithm Based on Adaptive Mutated Ant Colony Algorithm

GU Ming-jia, XUAN Shi-bin, LIAN Kan-chao, LI Yong-sheng

(College of Mathematics and Computer Science, Guangxi University for Nationalities, Nanning 530006)

【Abstract】 The multiple constrained QoS unicast routing problem is a NP-complete problem. Ant colony algorithm is easy to fall in local best, and its convergent speed is slow in solving multiple constrained QoS unicast routing problems. An adaptive mutated ant colony algorithm is proposed in this paper to solve multiple constrained QoS unicast routing problem. The algorithm adopts adaptive mutated manner, and introduces twice ant colony algorithm search mechanism, reduces the possibility to fall in local best. It improves the search ability and convergence speed. The feasibility and effectiveness of the algorithm are validated by series of simulated results.

【Key words】 multiple constrained QoS; unicast routing; adaptive mutated; ant colony algorithm

1 概述

随着 Internet 和多媒体技术的发展, 网络多媒体业务迅速增长, 对 QoS(网络服务质量)提出了更高的要求, 如何充分利用网络资源来满足日益多样化的 QoS 要求已成为人们关注和研究的热点。文献[1]证明了具有 2 个及以上约束条件的 QoS 路由问题是 NP 完全问题。近几年来, 国内外许多研究人员相继提出了许多 QoS 路由算法, 如蚁群算法、遗传算法和模拟退火算法等, 并取得了一定的成果。

蚁群算法是文献[2]提出的一种优化方法, 它以蚂蚁觅食行为为基础, 用蚂蚁的行走路线表示待求解问题的可行解, 不依赖于具体问题的数学描述, 具有很强的全局优化能力, 已广泛应用于组合优化中的 NP 完全问题。但蚁群算法本身也存在一些缺点, 如收敛速度慢、易陷入局部最优。许多学者提出了一些改进策略, 并应用于 QoS 路由优化上, 取得了一定的成果^[3-5]。本文分析了基本蚁群算法在求解 QoS 单播路由时存在的问题, 提出了基于自适应变异蚁群算法来进行求解。

2 QoS单播路由问题描述

QoS 单播路由问题就是要找到一条满足多个 QoS 约束, 如带宽、延时、丢包率等, 同时优化配置网络资源的路径。本文只讨论包含带宽和延时约束的 QoS 单播路由, 其网络拓扑结构用一个无向赋权图 $G=(V, E)$ 来表示, 其中, V 是网络节点的集合; E 是所有链路的集合。每条链路 E_i 上包含 2 个元素, 即 $E_i=(bandwidth, delay)$, 分别表示链路 E_i 上的带宽和延时。

对任何一个 QoS 路由请求均有 $Path=(V_s, V_d, Bandwidth, Delay)$, 其中, V_s 表示源节点; V_d 表示目的节点; $Bandwidth$ 表示最小传输带宽; $Delay$ 表示最大传输延时。

QoS 单播路由优化问题可以表述为^[6]: 在网络中寻找一条满足 $Path$ 要求的传输路径 $P=(V_s, \dots, V_d)$ 。其中:

(1) $bandwidth_j \leq Bandwidth, \forall j \in P$, 满足带宽要求。

(2) $\sum_{j \in P} delay_j \leq Delay, \forall j \in P$, 满足延时要求。

(3) $\min(Resources(P))$, 满足资源消耗最小。

$$Resources(P) = A \cdot R(P) + B \cdot \delta^2 \quad (1)$$

其中, A 和 B 为正实系数。 $R(P)=H(P) \cdot Bandwidth \cdot D(P)$; $H(P)$ 为路径 P 的跳数; $D(P)$ 为路径 P 的延时总和; δ^2 为路径 P 利用率的方差。

$$H(P) = length(P) - 1, \quad D(P) = \sum_{j \in P} Delays$$

$$U_{i,j} = \frac{Bandwidth}{bandwidth_{i,j}}, \quad \delta^2 = \sum_{i=1}^n \sum_{j=1}^n \left(U_{i,j} - \frac{\sum_{i=1}^n \sum_{j=1}^n U_{i,j}}{H(P)} \right)^2$$

3 基本蚁群算法分析

基本蚁群算法在求解 QoS 单播路由问题时虽然取得了满意的结果, 但也存在一些不足之处: (1) 依据状态转移规则来

基金项目: 国家民委科研基金资助项目(07GX04)

作者简介: 古明家(1980-), 男, 硕士研究生, 主研方向: 智能计算及应用; 宣士斌, 教授; 廉侃超、李永胜, 硕士研究生

收稿日期: 2009-04-06 **E-mail:** gmj19801102@163.com

选择下一节点时, 容易造成每次选择结果相同, 陷入局部极值。(2)每一周期蚂蚁搜索结束后, 按照信息素更新规则对所有蚂蚁走过的链路进行信息素更新, 会对下一周期的蚂蚁选择下一节点时造成误导作用, 而减少了未被访问的链路被访问的可能性, 也会陷入局部极值。

4 自适应变异蚁群算法

针对上述基本蚁群算法的分析, 本文提出了自适应变异蚁群算法, 改进策略如下:

(1)增加节点使用计数器 $NodeSum(1, N)$, 计数器中每个节点的初始值都为 0, 每经过一个节点 j 1 次, 该节点的计数值就增加 1。

(2)在 t 时刻蚂蚁由当前节点 i 选择下一节点 s 时, 将上文定义的节点使用计数器考虑进去, 在一定概率下, 让蚂蚁选择节点计数值最小的节点, 使未被访问的节点有被访问的可能性, 在一定程度上有利于跳出局部极值。具体选择如下:

$$s = \begin{cases} \max(\frac{[\tau_{ij}(t)]^\alpha \cdot [bandwidth_{ij}(t)]^\beta \cdot [delay_{ij}(t)]^\gamma}{\sum_{\mu \in allowed_k} [\tau_{i\mu}(t)]^\alpha \cdot [bandwidth_{i\mu}(t)]^\beta \cdot [delay_{i\mu}(t)]^\gamma}) \max(\cdot) > rand \\ \min(NodeSum(1, \mu), \mu \in allowed_k & \text{others} \end{cases} \quad (2)$$

其中, $\tau_{ij}(t)$, $bandwidth_{ij}(t)$, $delay_{ij}(t)$ 分别表示 t 时刻链路 (i, j) 上的信息素强度、带宽和延时; 参数 α 表示信息素的重要程度; 参数 β 和 γ 分别表示带宽和延时的重要程度; $allowed_k$ 表示 t 时刻蚂蚁可以选择的节点集, $NodeSum(1, \mu)$ 表示节点 μ 走过的次数。

(3)在进行下一周期搜索前, 对上一周期搜索到的符合 QoS 约束的路径进行自适应变异和二次蚁群算法搜索。具体策略如下: 假设上一周期的某条合格路径为 $Path = (1, 4, 12, 7, 20, 9)$, 该路径长度为 $length(Path) = 6$, 则其变异长度定义为 $L = round(length(Path)/3)$, 即变异长度为接近总长度的 $1/3$ 的整数, 在本例中 $L(Path) = 2$, 即变异长度为 2。 $mutated$ 为变异的次数, 固定为一常数, 每次根据变异长度 $L(Path)$ 随机产生 2 个变异点 R_1 和 R_2 , $R_2 = R_1 + L$, 然后按上述蚂蚁爬行规则, 执行二次蚁群算法搜索 R_1 和 R_2 之间的路径。如果变异后搜索到的路径优于当前的最优路径, 则将变异后的路径替换当前的最优路径, 如果变异后搜索到的路径优于变异前的路径, 则将变异后的路径替换变异前的路径; 然后进行下一次变异, 如此重复, 直到执行完 $mutated$ 次变异。

(4)在进行下一周期搜索前, 仅对上一周期和变异后的最优路径上的节点的信息素进行按下式(3)更新, 其他节点的信息素不更新。

$$\tau_{ij}^{k+1}(t+1) = (1 - \omega) \cdot \tau_{ij}^k(t) + \omega \cdot \Delta \tau_{ij} \quad (3)$$

$$\Delta \tau_{ij} = \begin{cases} Q / L_{best} & (i, j) \in Path_{best} \\ 0 & \text{others} \end{cases}$$

其中, $0 < \omega < 1$; L_{best} 为当前最佳路由长度; $Path_{best}$ 为当前最佳路由, Q 为固定常数。

上述的改进策略既保证了在下一周期搜索时倾向于部分较优解附近, 提高算法的收敛速度, 又保证了使蚂蚁倾向于未搜索过的链路, 减少算法陷入局部极值的可能性, 增强了算法的全局寻优能力。

利用本文算法求解 QoS 单播路由的步骤如下:

Step1 设置请求路由源节点 V_s , 目的节点 V_d , 带宽约束 $Bandwidth$, 延时约束 $Delay$, 删除不满足带宽约束的链路,

得到新的网络拓扑图。

Step2 设置蚂蚁种群数量 m , 最大迭代次数 $DiedaiNum$, 变异次数 $MutatedNum$, 初始化各链路信息素强度 τ , 节点使用计数器 $NodeSum$, 参数 $A, B, \alpha, \beta, \gamma, \omega, Q$, 迭代次数 $l = 0$, 最优路径记录 $Path_{best}$ 。

Step3 迭代次数 $l = l + 1$, 在源点创建 m 只蚂蚁分组, 初始化禁忌表, 并将源点加入禁忌表, $t = 0$, 节点使用计数器 $NodeSum(1, V_s) = NodeSum(1, V_s) + m$ 。

Step4 $t = t + 1$, 对每只蚂蚁分组 k 根据式(2), 由当前节点 V_i 选择下一节点 V_j , 然后把节点 V_j 加入禁忌表, 更新节点使用计数器 $NodeSum(1, V_j) = NodeSum(1, V_j) + 1$ 。

Step5 重复执行 Step 4 直到蚂蚁死亡或蚂蚁完成搜索。

Step6 记录完成从 V_s 到 V_d 搜索的蚂蚁分组 k , 并标记该路径为初始解路径, 计算资源函数 $Cost(Path_k)$, 路径总延时 $\sum_{i \in Path_k} delay_i$, 并与当前最优路径 $Path_{best}$ 比较, 如果优于当前最优路径, 则用此路径更新最优路径 $Path_{best} = Path_k$ 。

Step7 对标记为初始解的路径 $Path_k$ 进行自适应变异, 根据上述的方法确定变异长度 L , 随机产生每次变异的节点位置 $R_1, R_2, R_2 = R_1 + L$, 按 Step 4, Step 5 执行二次蚂蚁搜索, 寻找以 R_1 为源节点, R_2 为目的节点的路径, 记录变异后的路径 $Path_{k_mutated}$ 。如果变异后的路径 $Path_{k_mutated}$ 优于当前最优路径 $Path_{best}$, 则 $Path_{best} = Path_{k_mutated}$, 然后进行下一次变异, 如此重复, 直到执行完 $mutated$ 次变异。

Step8 根据式(3)对变异后的当前最优路径 $Path_{best}$ 更新链路信息素。

Step9 转 Step3, 进入迭代循环。

Step10 $DiedaiNum$ 次迭代结束后, 输出当前的最优路径 $Path_{best}$, 算法结束。

5 仿真实验及其比较

本文算法用图 1 所示的网络模型进行仿真, 链路属性用二元组(带宽, 延时)描述。

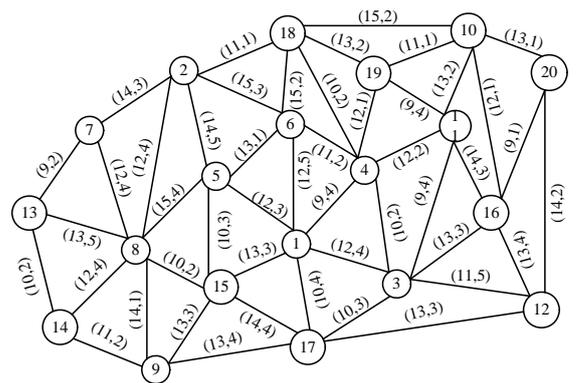


图 1 20 个节点随机生成的网络模型

假设现有一个 QoS 路由请求 $Path = (1, 20, 8, 30)$, 分别运用本文算法、基本蚁群算法和文献[5]的算法处理 $Path$ 。

本文算法、基本蚁群算法和文献[5]的参数设置为: $V_s = 1, V_d = 20, Bandwidth = 8, Delay = 30, m = 10, DiedaiNum = 20, MutatedNum = 10, \tau_0 = 1, NodeSum_0 = 0, A = 1, B = 500, \alpha = 1, \beta = 2, \gamma = 1, \omega = 0.3, Q = 2, \omega_{min} = 0.3, \omega_{max} = 0.8$ 。

3 种算法运行结果比较如表 1 和图 2 所示。

表 1 3种算法运行结果比较

算法	最优路径	延时/s	资源
本文算法	1-3-16-20	8	212.765 9
基本蚁群算法	1-3-11-16-12-20	17	714.285 7
文献[5]算法	1-3-16-12-20	13	416.666 7

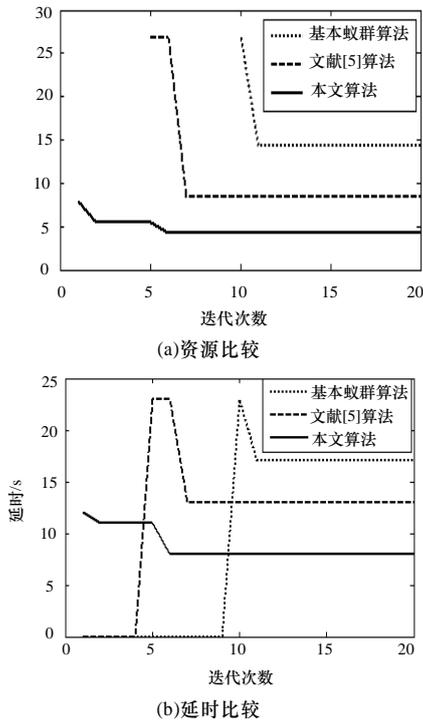


图 2 3种算法仿真结果比较

编辑 金胡考

(上接第 208 页)

图 4 为 α -OPRA 与其他降维方法分别对 3D 空间中 SWISSROLL 流形结构进行 2D 降维后的比较, 其中 PCA 属于线性降维, 损失了高维空间中的非线性相关性, 效果最差; SLLE、LLE、KPCA 和 LAPLACIAN 属于非线性降维, 但效果都不如 α -OPRA。

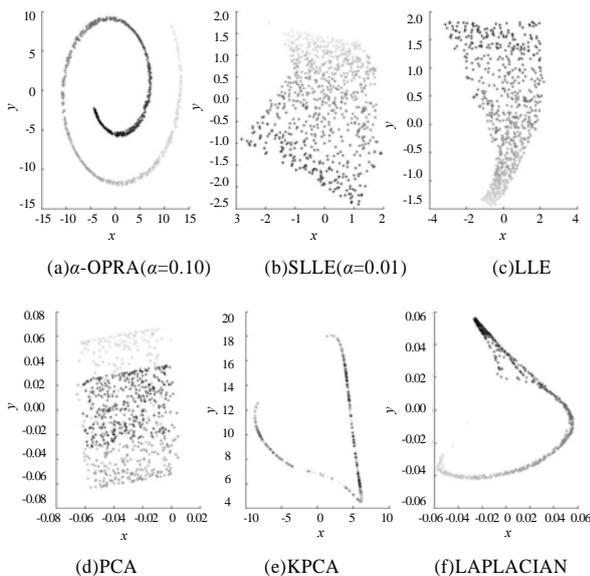


图 4 α -OPRA 与其他几种降维方式的比较

5 结束语

OPRA 是一种相对较新的降维方法, 既有 PCA 正交投影

可见, 在同条件下利用本文算法比其他 2 种算法找到最优解的收敛速度快, 而且收敛结果也比其他 2 种算法的要好, 从而验证了本文算法在解决多约束 QoS 单播路由优化的可行性、有效性、收敛性。

6 结束语

本文提出了自适应变异蚁群算法的 QoS 路由算法, 实验结果表明, 利用此算法能有效、快速地实现多约束 QoS 单播路由优化, 下一步将应用此算法去解决组播路由问题。

参考文献

- [1] Wang Zheng, Crowcroft J. Quality-of-Service Routing for Supporting Multimedia Applications[J]. IEEE Journal of Selected Areas in Communications, 1996, 14(7): 1228-1234.
- [2] Dorigo M, Maniezzo V, Colomi A. The Ant System: Optimization by a Colony of Cooperating Agents[J]. IEEE Transactions on Systems, Man, and Cybernetics, 1996, 26(1): 1-13.
- [3] 封国剑, 范俊锋, 高随祥. 基于改进蚁群优化算法的分布式多播路由算法[J]. 计算机工程, 2006, 32(3): 146-148.
- [4] 刘震宇, 陈哲强, 王 蔚. 基于免疫-蚂蚁算法的多约束 QoS 多播路由优化[J]. 计算机工程, 2007, 33(9): 201-203.
- [5] 邵志伟. 蚁群算法研究及其在 QoS 路由中的应用[D]. 郑州: 郑州大学, 2007.
- [6] 李元臣, 刘维群. 基于遗传算法的 QoS 路由优化算法[J]. 微计算机信息, 2007, 23(4/3): 129-131.

编辑 金胡考

的优点, 又保持了空间非线性结构。 α -OPRA 主要是将 SLLE 的思想引入 OPRA 中。同传统的 OPRA 相比, α -OPRA 使高维到低维的映射在保留某些流形结构的同时, 可进一步分离不同类别的流形。其核心思想是, 通过加入额外的参数 α 来控制监督的程度, 通过对 SWISSROLL 空间流形 3D 离散点进行 2D 降维。实验结果同时表明, α 取值较大时, 将导致空间流形结构的丧失; α 取 0.10, K 取 12 时效果最好; α -OPRA 与其他降维方法相比, 具有更好的降维效果。

参考文献

- [1] Jolliffe I T. Principal Component Analysis[M]. New York, USA: Springer-Verlag, 1989.
- [2] Roweis S T, Saul L K. Nonlinear Dimensionality Reduction by Locally Linear Embedding[J]. Science, 2000, (290): 2323-2326.
- [3] Kokiopoulou E, Saad Y. Face Recognition Using OPRA-faces[C]//Proc. of IEEE Int'l. Conf. on Machine Learning and Application. NY, USA: [s. n.], 2005: 15-17.
- [4] Qing Xiangyun, Wang Xingyu. Face Recognition Using Laplacian+OPRA-faces[C]//Proc. of IEEE Conf. on Intelligent Control and Automation. Dalian, China: [s. n.], 2006: 10013-10016.
- [5] Zhao Qijun, Zhang David, Lu Hongtao. Supervised LLE in ICA Space for Facial Expression Recognition[C]//Proc. of International Conference on Neural Networks and Brain. Shanghai, China: [s. n.], 2005: 1970-1975.

编辑 索书志

