

基于蚁群算法的移动自适应网 QoS 路由算法

林本强, 唐依珠

(福州大学 数学与计算机科学学院, 福建 福州 350108)

摘要:在移动自适应网中,由于无线环境的不稳定性及终端的移动性造成网络拓扑结构不断变化,使网络中端对端的服务质量随时间不断变化。为移动自适应网提供服务质量保证的关键是移动自适应网的 QoS 路由问题。提出了一种适用于移动自适应网的 QoS 路由算法。该算法将蚁群算法与移动自适应网的特点相结合,可以高效地在该网络中寻找满足延迟、延迟抖动、带宽、包丢失率、节点跳数和费用等多项指标的 QoS 路由,从而显著地提高移动自适应网的 QoS 路由性能。

关键词:蚁群算法;移动自适应网;QoS 路由

中图分类号:TP393

文献标识码:A

文章编号:1673-629X(2009)06-0009-04

Ant Colony Algorithm Based Ad Hoc Network QoS Routing Algorithm

LIN Ben-qiang, TANG Yi-zhu

(College of Mathematics and Computer Science, Fuzhou University, Fuzhou 350108, China)

Abstract: In Ad Hoc networks, because of the instability of the wireless environment and the ambulation of the terminal, the topology of the network is changing and the end-to-end network service quality transforms all the time. QoS routing problem is the key for supplying service quality assurance in Ad Hoc networks. Presents a QoS routing algorithm which can be applied in Ad Hoc networks. The algorithm combines the ant colony algorithm together with the characteristics of Ad Hoc networks and it can efficiently look for the QoS routing which satisfies the delay, delay-jitter, bandwidth, packet-loss, hop, cost in Ad Hoc networks. Consequently the Ad Hoc networks's QoS routing performance significantly improves.

Key words: ant colony algorithm; Ad Hoc networks; QoS routing

0 引言

移动自适应网^[1]是指由多个独立的具有路由功能的移动节点组成的采用分布式管理的多跳网络。该网络中在传输范围内的节点间可以直接通信;在彼此传输范围外的节点通信时,则可动态地搜索路由,数据包借助其他节点转发,以多跳方式传递至最终的目的节点。在采用无线网卡构建的移动自适应网中,由于无线信道环境的快速变化及终端的移动性造成的网络拓扑结构不断变化使网络中端对端的服务质量随时间的不断变化^[2]。为移动自适应网提供服务质量保证的关键问题是移动自适应网的 QoS 路由问题。QoS 路由^[3]即根据条件利用网络的可用资源,在移动自适应网中建立满足 QoS 要求的传输路径。

1 蚁群算法应用于移动自适应网路由的优越性

移动自适应网是分布式的网络,而蚁群算法^[4,5]可以在任何节点放出单独进行路径搜索的蚂蚁正好适应此特点。蚁群算法的正反馈和快速收敛性则可以满足快速适应网络拓扑变化的要求,算法的正反馈保证了优良的信息得以保存,随机性保证全局优化。蚁群算法中的蚂蚁结构简单,网络中传输的报文就比较短,占用的网络带宽资源较小;信息素只是一个数值,存储信息素并不需要占用较多的存储空间。这些特性完全适合于系统资源少的移动自适应网。尽管蚁群算法在每次循环都是通过调节信息素来对下一次循环产生影响,但是每次循环中的信息素调节和路径选择都是独立完成的。这样在蚁群算法中就更容易引入其他算法,可以结合其他算法的优点改善蚁群算法的性能。

2 移动自适应网 QoS 路由模型

移动自适应网模型^[6]用加权图 $G = (V, E)$ 表

收稿日期:2008-09-27;修回日期:2009-02-06

作者简介:林本强(1982-),男,福建福州人,硕士研究生,研究方向为嵌入式系统设计、智能技术;唐依珠,副教授,研究方向为嵌入式系统。

示, V 是网络节点的集合, E 是节点间双向链路集合。对任意链路 $e \in E$, 用四元组 $(\text{delay}(e), \text{bandwidth}(e), \text{delay_jitter}(e), \text{cost}(e))$ 表示 QoS 其特征值, 分别为延迟函数、带宽函数、延迟抖动函数和费用函数。对任意节点 $n \in V$, 用四元组 $(\text{delay}(n), \text{packet_loss}(n), \text{delay_jitter}(n), \text{cost}(n))$ 表示其 QoS 特征值, 分别为延迟函数、包丢失率函数、延迟抖动函数和费用函数。

$p(s, d)$ 为源节点 s 到目标节点 d 的一条路径, 则:

$$\text{delay}(p(s, d)) = \sum_{e \in p(s, d)} \text{delay}(e) + \sum_{n \in p(s, d)} \text{delay}(n)$$

$$\text{bandwidth}(p(s, d)) = \min\{\text{bandwidth}(e)\}$$

$$\text{delay_jitter}(p(s, d)) = \sum_{e \in p(s, d)} \text{delay_jitter}(e) +$$

$$\sum_{n \in p(s, d)} \text{delay_jitter}(n)$$

$$\text{packet_loss}(p(s, d)) = 1 - \prod_{e \in p(s, d)} (1 - \text{packet_loss}(e))$$

$$\text{cost}(p(s, d)) = \sum_{e \in p(s, d)} \text{cost}(e) + \sum_{n \in p(s, d)} \text{cost}(n)$$

QoS 路由是寻找到满足以下约束条件下使 $\text{cost}(p(s, d))$ 尽可能小的 $p(s, d)$:

(1) 延迟约束 $\text{delay}(p(s, d)) \leq D$, D 为最大时延;

(2) 带宽约束 $\text{bandwidth}(p(s, d)) \geq B$, B 为瓶颈带宽;

(3) 延迟抖动约束 $\text{delay_jitter}(p(s, d)) \leq DJ$, DJ 为最大时延抖动;

(4) 包丢失率约束 $\text{packet_loss}(p(s, d)) \leq PL$, PL 为最大包丢失率;

(5) 节点最大允许跳数约束 $\text{hop}(p(s, d)) \leq h_{\text{top}}$, h_{top} 为最大允许跳数。

3 移动自适应网 QoS 路由实现规则

将移动自适应网的源节点和目的节点分别当作“蚁巢”和“食物”, 数据包视为蚂蚁。在本算法中移动自适应网 QoS 路由实现基于以下规则^[4,7]:

规则 1: 状态转移规则。

蚂蚁 k 在节点 i 选择节点 j 为下一跳节点的概率为:

$$p_{ij}^k = \frac{(\tau_{ij} + \zeta)^{\alpha} \eta_{ij}^{\beta} \lambda_{ij}^{\delta}}{\sum_{j \in \text{allow}(d_k)} (\tau_{is} + \zeta)^{\alpha} \eta_{ij}^{\beta} \lambda_{is}^{\delta}}$$

式中: $j \in \text{allow}(d_k)$ 。蚂蚁 k 在节点 i 从 QoS 要求等方面出发, 下一步允许选择的节点并且被选择的节点不在蚂蚁 k 当前的禁忌表 tabu_k 中。 α 为信息素启发因子; β 为费用期望启发式因子; δ 为稳定性期望启发式因子; 参数 ζ 反映了蚂蚁对少量信息素的敏感性。 α 越大

蚂蚁越倾向于选择其它蚂蚁走过的路径; ζ 越大蚂蚁对少量信息素的敏感性越低, 以避免寻找优化路径时陷入局部最优。 τ_{ij} 表示从节点 i 到节点 j 链路上的信息素; $\eta_{ij} = 1/\text{cost}_{ij}$ 为费用启发函数, 即节点倾向于选择费用最小的链路; $\lambda_{ij} = \frac{R_i - d_{ij}}{V}$ 为稳定性启发函数, 其中 R_i 是节点 i 的传输覆盖范围, d_{ij} 是节点 i 和节点 j 之间的距离, V 是节点移动的平均速度。

规则 2: 信息素的局部更新规则。

蚂蚁 k 对其所经过路径 (i, j) 上的信息素根据下式进行局部更新:

$$\tau_{ij}(t + \Delta t) = (1 - \rho_1) \tau_{ij}(t) + \Delta \tau_{ij}^k(t)$$

设蚂蚁 k 在到达节点 i 时共有 R_k 只蚂蚁经过 i 点, 其中有 r_k 只蚂蚁选择节点 j 作为下一跳, 则:

$$\Delta \tau_{ij}^k(t) =$$

$$\begin{cases} Q & \text{蚂蚁 } k \text{ 从 } i \text{ 移动到 } j \text{ 且 } R_k = 0 \\ Q(1 - \frac{r_k}{R_k}) & \text{蚂蚁 } k \text{ 从 } i \text{ 移动到 } j \text{ 且 } R_k > 0 \end{cases}$$

式中: $0 < \rho_1 < 1$ 为局部信息素挥发系数, 常量 Q 是信息量。

采用此方法进行局部信息素更新提高了算法搜索的全局性, 同时避免了蚂蚁集中在前一次最好路径的有限相邻区域内搜索。

规则 3: 信息素的全局更新规则。

所有蚂蚁完成了一次循环后要要对网络各条链路上的信息素作全局更新。计算满足约束条件路由 QoS 的蚂蚁 k 搜索到路由 QoS 的评价目标函数值 L_k :

$$L_k = \frac{1}{\text{cost}(p(s, d))} (Af_d + Bf_{dj} + Cf_{pl} + Df_{bw} + Ef_{hp})$$

$$f_d = \frac{D - \text{delay}(p_k(s, d))}{D}$$

$$f_{dj} = \frac{DJ - \text{delay_jitter}(p_k(s, d))}{DJ}$$

$$f_{pl} = \frac{PL - \text{packet_loss}(p_k(s, d))}{PL}$$

$$f_{bw} = \frac{\text{bandwidth}(p(s, d)) - B}{\text{bandwidth}(p(s, d))}$$

$$f_{hp} = \frac{h_{\text{top}} - \text{hop}(p(s, d))}{h_{\text{top}}}$$

其中: A, B, C, D, E 分别为 $f_d, f_{dj}, f_{pl}, f_{bw}, f_{hp}$ 的加权系数, 分别表示延时、延时抖动、包丢失率、带宽、跳数在路由的 QoS 评价目标函数中所占的比重。

在所有找到满足 QoS 约束条件路由的蚂蚁中, 搜索到路由的 QoS 评价目标函数值 L_k 最大的蚂蚁搜索到的路由为最优路径, 其目标函数值为 L_{best} ; 在所有找到满足 QoS 约束条件路由的蚂蚁中, 搜索到路由的

QoS 评价目标函数值 L_k 最小的蚂蚁搜索到的路由为最差路径,其目标函数值为 L_{worst} 。

对网络链路 (i, j) 上的信息素按以下公式进行全局更新:

$$\tau_{ij}(t + \Delta t) = (1 - \rho_2)\tau_{ij}(t) + \Delta\tau_{ij}(t)$$

其中: ρ_2 为全局信息素挥发系数,上式第一部分表示的是这一代蚂蚁走完所有信息挥发;第二部分表示这一代蚂蚁对信息素的修改。第二部分计算方式如下:

$$\Delta\tau_{ij}(t) = \sum_{k=1}^m \Delta\tau_{ij}^k(t)$$

如果蚂蚁 k 没有找到满足 QoS 约束条件的路由,则 $\Delta\tau_{ij}^k(t) = 0$;

如果蚂蚁 k 找到满足 QoS 约束条件的路由,则按下式计算:

$$\Delta\tau_{ij}^k(t) = \begin{cases} aL_k Q & \text{蚂蚁 } k \text{ 从 } i \text{ 移动到 } j \text{ 而蚂蚁 } k \text{ 走过的路径既不是最优路径也不是最差路径} \\ aL_{\text{best}} Q + \frac{Q_1}{L_{\text{best}}} & \text{蚂蚁 } k \text{ 从 } i \text{ 移动到 } j \text{ 而蚂蚁 } k \text{ 走过的路径是最优路径} \\ aL_{\text{worst}} Q - \frac{Q_2}{L_{\text{worst}}} & \text{蚂蚁 } k \text{ 从 } i \text{ 移动到 } j \text{ 而蚂蚁 } k \text{ 走过的路径是最差路径} \\ 0 & \text{蚂蚁 } k \text{ 没有走过该路径} \end{cases}$$

其中: a 为满足 QoS 要求的激励系数, Q_1 是用来奖励最优路径的信息素量, Q_2 是用于惩罚最差路径的信息素量。

4 基于蚁群算法的移动自适应网路由过程

4.1 路由算法中的蚂蚁

1) 消息传递蚂蚁。

消息传递蚂蚁结构简单,包括广播蚂蚁和路径维护蚂蚁。

2) 路由信息蚂蚁。

路由信息蚂蚁^[8]分为源节点到目的节点的前向蚂蚁和目的节点返回源节点的后向蚂蚁。当前向蚂蚁到达目的节点,目的节点将产生后向蚂蚁而前向蚂蚁消亡。在反方向上,后向蚂蚁采取与前向蚂蚁相同的路径。路由信息蚂蚁主要完成对节点路由表的更新,链路信息素的加强,记录路径信息,根据路径要求确定最优路径,完成路由发现的功能。

4.2 路由建立的过程

路由的建立过程^[9]即发现路由寻找最优路径的过程,主要完成源节点到目的节点的通信以发现一条或多条可行的路径。具体步骤如下^[3,4]:

(1) 精简网络拓扑:首先删除不满足 QoS 约束条

件的节点以及与之相连的链路,再删除不满足 QoS 约束要求的链路,得到一个新的网络拓扑。基于此网络拓扑图进行搜索。令循环次数 $N_c = 0$,设置最大循环次数 N_{max} 。确定源节点和目的节点。设置网络中的每条链路的信息素初始值,初始化节点的路由信息表,给出每个节点和链路的 QoS 特征值以及约束条件。

(2) 在路由请求的节点(即源节点)设置 m 只前向蚂蚁(前向蚂蚁的数目根据网络的情况和需求确定),为每只蚂蚁初始化禁忌表 tabu_k 用来记录已经走过的节点。

(3) 前向蚂蚁重复应用状态转移规则(规则 1)选择路径并根据信息素的局部更新规则(规则 2)进行信息素局部更新,将经过节点的地址添加到该蚂蚁的禁忌表 tabu_k 中。

(4) 如果前向蚂蚁到达目的节点后,由目的节点产生一只后向蚂蚁。前向蚂蚁中的禁忌表中记录了前向蚂蚁由源节点到目的节点的路由信息,前向蚂蚁将这些信息复制给后向蚂蚁,然后前向蚂蚁消亡。后向蚂蚁采取与前向蚂蚁相同的路径从目的节点返回。后向蚂蚁记录返回路径上的各项 QoS 指标。若后向蚂蚁返回路径上的节点出现失效,则该后向蚂蚁消亡。

(5) 后向蚂蚁返回源节点。如果后向蚂蚁记录的返回路径上的各项 QoS 指标满足 QoS 约束条件,则按规则(3)计算该后向蚂蚁 k (对应源节点发出的前向蚂蚁 k)对应的路由 QoS 评价目标函数值 L_k 。更新返回路径各节点的路由信息表。对于没有找到满足 QoS 约束条件路由的蚂蚁对应的路由 QoS 评价目标函数值 $L_k = 0$ 。

(6) 所有蚂蚁重复执行步骤(3) ~ (5)。

(7) 按照信息素的全局更新规则(规则 3)对网络上的信息素进行全局更新。

(8) 若 $N_c < N_{\text{max}}$,删除所有后向蚂蚁, $N_c = N_c + 1$ 跳转到步骤(2),顺序执行步骤(2) ~ (7)。

(9) 若 $N_c = N_{\text{max}}$,循环结束。取得在第 N_{max} 次循环中路由 QoS 评价目标函数值 L_k 最大的后向蚂蚁 k ,该后向蚂蚁 k 的禁忌表中就记录了由源节点到目的节点的符合 QoS 要求的最佳路由。删除所有后向蚂蚁。在找到符合 QoS 要求的最佳路由的同时在该移动自适应网中也建立起了符合 QoS 要求的路由网,当最佳路由失效时,可以迅速建立起符合 QoS 要求的次优路由。

4.3 路由信息的维护过程

移动自适应网节点的灵活性和拓扑结构的快速变化性很容易导致源节点到目的节点的路由失效。路由信息的维护过程^[10]主要负责监控网络路由信息是否有效,如果路由信息失效,则重新寻找一条可以从源节

点到目的节点的路径。

每个节点周期性地发送广播蚂蚁,广播蚂蚁不留下信息素,相当于传统路由算法中的 hello 报文。当网络拓扑发生变化时,由于离开网络的节点将无法把广播蚂蚁返回,则发出广播蚂蚁的节点认为该节点已经离开此路径。这时此节点将发送路径维护蚂蚁来通知活动节点的前序列节点查找该节点的路由表以寻找另一条次优路径传输数据。如果该级前序列节点都找不到目的节点的路径,则继续从上一级前序列节点查找。如果上述过程未找到到达目的节点的路径,则源节点再次发起一个路由请求。由于维护过程采用了简单高效的消息传递蚂蚁进行操作,在网络快速变化的情况下仍能够取得较好的路由效果。

5 结束语

在分析移动自适应网特点的基础上,提出了一种基于蚁群算法的路由算法。该算法结合了先应式路由和按需式路由的两方面优点,具有快速的收敛性,整个的路由过程中具有自适应性、自治性、交互性等特点,路由维护简单,占用网络资源小,能显著提高移动自适应网性能,具有很好的发展前景。

参考文献:

- [1] David R, Ignas G N. Ad Hoc networking in future wireless communications[J]. Computer Communications, 2003, 26(1):

36-40.

- [2] 鲍晶. 无线 Ad Hoc 网络技术探讨[J]. 计算机技术与发展, 2006, 16(7): 201-203.
- [3] Kone V, Nandi S. QoS Constrained Adaptive Routing Protocol For Mobile Adhoc Networks[C]//In: Proceedings of the 9th Information Technology International Conference(ICIT'06). Mumbai, India: IEEE Computer Society Press, 2006: 40-45.
- [4] 段海滨. 蚁群算法原理及其应用[M]. 北京: 科学出版社, 2005.
- [5] Kwang M S, Weng H S. Ant colony optimization for routing and load - balancing: survey and new directions[J]. IEEE Transactions on Systems, Man and Cybernetics: Part A, 2003, 33(5): 560-572.
- [6] 孙艳歌, 刘明, 许芷岩. Ad Hoc 网络中基于双向收敛蚁群算法的 QoS 路由算法[J]. 微电子学与计算机, 2006, 23(10): 1-3.
- [7] Abuzanat H, Trouillet B, Toguyeni A. Routing Fairness Model for QoS Optimization in Wireless Network[C]//In: Proceedings of the Second International Conference on Sensor Technologies and Applications(SENSORCOMM 2008). Cap Esterel, France: IEEE Computer Society Press, 2008: 776-781.
- [8] 潘达儒, 袁艳波. 一种基于 AntNet 改进的 QoS 路由算法[J]. 小型微型计算机系统, 2006, 27(7): 1169-1174.
- [9] 刘卫彪, 张修如, 朱光辉. 一种基于节点位置信息的 Ad Hoc 网络路由算法[J]. 计算机技术与发展, 2007, 17(10): 158-161.
- [10] 王合义, 丁建立, 唐万生. 基于蚁群优化的路由算法[J]. 计算机应用, 2008, 28(1): 7-8.

(上接第 8 页)

渐接近 Min-Min 算法,甚至有可能超过 Min-Min 算法^[9]。文中所讨论的这些算法各有各的特点,没有绝对的最优或最差,它们适用于各种网格系统环境和系统需求。

参考文献:

- [1] 罗秉安, 张立臣. 网格技术及其应用[J]. 微机发展(现更名: 计算机技术与发展), 2002, 12(6): 2-4.
- [2] 秦金磊, 朱有产, 李玉凯. 基于网格计算的关键技术研究[J]. 计算机技术与发展, 2006, 16(11): 71-73.
- [3] 桂小林. 网格技术导论[M]. 北京: 北京邮电大学出版社, 2005.
- [4] 崔玉宝, 李建义, 薛桂香. 一种改进的启发式网格任务调度算法[J]. 微型电脑应用, 2006(5): 6-7.
- [5] Maheswaran M, Ali S, Siegel H J, et al. Dynamic matching and scheduling of a class of independent tasks onto heterogeneous computing systems [C]//8th Heterogeneous Computing

Workshop. Puerto Rico: IEEE Press, 1999: 30-44.

- [6] Maheswaran M, Ali S, Siegel H J, et al. A Comparison of Dynamic Strategies for Mapping a Class of Independent Tasks onto Heterogeneous Computing Systems[R]. School of Electrical and Computer Engineering, Purdue University, in preparation, 1999.
- [7] Wu Min You, Shu Wei, Zhang Hong. Segmented Min-Min: A Static Mapping Algorithm for Meta-tasks on Heterogeneous Computing Systems [EB/OL]. [2004-07-10]. <http://csdl2.computer.org/dl/proceedings/hcw/2000/0556/00/05560375.pdf>.
- [8] Hamscher V, Schwiigelshohn U, Streit A. Evaluation of job-scheduling strategies for grid computing[C]//Proceedings of 7th Int'l Conf on High Performance Computing. Berlin, Heidelberg: Springer-Verlag, 2000: 191-202.
- [9] 陈宇寒. 网格计算技术研究[J]. 计算机技术与发展, 2008, 18(5): 52-54.