

基于蚁群算法的 Ad Hoc 网络 QoS 组播路由研究

梁淑萍, 毛 力, 马亦先

(江南大学 物联网工程学院, 江苏 无锡 214122)

摘 要: 在解决 Ad Hoc 网络 QoS 组播路由问题上, 针对蚁群算法缺点, 提出了一种融合粒子群优化思想的改进蚁群算法. 该算法融合 PSO 思想以加速蚁群算法在路由发现及维护时的收敛速度. 仿真结果表明, 该算法具有较好的性能, 是解决 Ad Hoc 网络 QoS 组播路由问题的有效方法.

关键词: Ad Hoc 网络; QoS; 多播路由; 蚁群算法; 粒子群算法

中图分类号: TP393

文献标识码: A

文章编号: 1000-7180(2011)07-0164-04

Research on QoS Multicast Routing of Ad Hoc Network Based on Improved Ant Colony Algorithm

LIANG Shu-ping, MAO Li, MA Yi-xian

(IoT Engineering, Jiangnan University, Wuxi 214122, China)

Abstract: In allusion to the flaws of Ant Colony Algorithm for solving the QoS multicast routing problem of Ad Hoc network, an improved Ant Colony Algorithm based on Particle Swarm Optimization is proposed. This algorithm is applied in the establishment and maintenance of Ad Hoc network multicast routing process to accelerate the convergence rate of Ant Colony Algorithm. The simulation results show that this algorithm has better performance in solving the QoS multicast routing problem of Ad Hoc network.

Key words: Ad Hoc network; QoS; multicast routing; ant colony algorithm; particle swarm optimization

1 引言

Ad Hoc 网络是由一组具有路由和转发功能的移动节点自主构成的一个多跳的自治系统, 其网络中节点的移动使网络拓扑结构不断变化, 并且节点的带宽和能量有限, 因此, 为有线网络设计的 QoS 组播路由协议不再适用, 如何设计出满足多个 QoS 约束条件的组播路由协议已成为 Ad Hoc 网络研究的关键和难点之一.

Ad Hoc 网络的 QoS 组播路由问题^[1-2]是一个 NP 完全问题, 近年来蚁群优化(ACO)^[3]等算法在解决 Ad Hoc 组播路由问题上得到了越来越广泛的应用, 如 C. C. Shen 和 CHAIPORN JAIKAE0 提出的 MANSI^[4], Lee 和 Chang 提出的 ANMAS^[5]等均

为基于蚁群算法的 Ad Hoc 网络组播路由算法. 然而由于蚁群算法自身的缺陷, 使得利用它解决 Ad Hoc 组播路由问题时普遍存在初期信息素匮乏, 收敛速度慢等不足.

针对上述问题, 提出一种基于粒子群优化算法(PSO)^[6]的 Ad Hoc 网络 QoS 组播路由算法 PSOACO, 该算法充分利用 PSO 的快速性和全局收敛性以加速蚁群算法在路由发现及维护时的收敛速度. 仿真结果表明, 该算法性能良好.

2 基于 PSOACO 算法的 Ad Hoc 网络 QoS 组播路由算法

2.1 QoS 组播路由问题描述

Ad Hoc 网络拓扑变化频繁带来问题研究的复

收稿日期: 2010-11-29; **修回日期:** 2010-12-20

基金项目: 江南大学博士启动基金项目(1055211542080380); 农业部淡水鱼类遗传育种和养殖生物学重点开放实验室开放基金项目(BZ2009-07)

杂性,为了降低考虑问题的复杂程度,增加算法的可用性,本算法仅考虑实际网络中最重要的 2 个主要 QoS 参数,即带宽和时延.基于时延及带宽限制的 Ad Hoc 组播路由优化问题可表述为:给定 Ad Hoc 网络 $G = (V, E)$,其中 V 是所有网络节点组成的集合, E 是网络双向链路的集合,每一条边表示两节点的直达通信路径.对于给定的源节点 $s \in V$,目的节点 $M \in \{V - \{s\}\}$, $t \in M$, s 和 M 组成的组播树 $T(s, M)$ 存在下列关系:

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

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

$$\text{bandwidth}(P_T(s, t)) = \min\{\text{bandwidth}(e), e \in P_T(s, t)\}$$

其中, $P_T(s, t)$ 是组播树 $T(s, M)$ 上源点 s 到终点 t 的路由.

QoS 组播路由问题就是寻找满足下列 QoS 约束,且使 $\text{cost}(T(s, M))$ 最小的组播树:延迟约束: $\text{delay}(P_T(s, t)) \leq DL$;带宽约束: $\text{bandwidth}(P_T(s, t)) \geq BW$;其中, DL 、 BW 分别为业务对网络时延、带宽的约束限制.

2.2 算法设计

2.2.1 主要数据结构

(1) 前向蚂蚁的数据结构如下:

g	num	addr1	addr2	d	c	VisitedNodes	BW	DL
---	-----	-------	-------	---	---	--------------	----	----

其中 g 是组播组的标识; num 为蚂蚁的唯一标识符; $addr1$ 为组播源地址, $addr2$ 为组播接收者地址; d 为计算到目前为止的时延; c 为计算到目前为止的所需费用; $VisitedNodes$ 记录了前向蚂蚁所经过的节点; BW 表示对所选链路的带宽约束; DL 表示对所选链路的时延约束.

(2) 后向蚂蚁的数据结构不包含 BW 、 DL , 其它与前向蚂蚁类似.

(3) 粒子更新蚂蚁的结构如下:

g	num	addr1	addr2	F	path
---	-----	-------	-------	---	------

其中 g 是组播组的标识; num 为蚂蚁的唯一标识符; $addr1$ 为组播源节点地址; $addr2$ 为组播接收者地址集; F 是该蚂蚁对应的较优粒子的适应度值; $path$ 是组播路由.

(4) 组播路由表结构如下:

g	dest	next	pher
---	------	------	------

其中 g 是组播组的标识; $dest$ 是目的节点地址;

$next$ 是下一跳节点地址; $pher$ 是链路信息素值.

2.2.2 PSOACO 算法中 PSO 的设计

PSOACO 算法通过引入交换子和交换序等概念^[7],同时根据 Ad Hoc 网络 QoS 组播路由问题的特殊性,对基本 PSO 算法进行重新定义.

(1) 粒子的适应度函数 $F(x)$ 为:

$$F(x) = \begin{cases} \frac{\beta e^{f(x)}}{\text{cost}}, & \text{del} \leq DL \\ 0, & \text{否则} \end{cases} \quad (1)$$

$$f(x) = 1 - \frac{\text{del}}{DL} \quad (2)$$

cost 和 del 分别是组播树的费用和时延, DL 表示时延的约束限制, β 是调整系数且是正数.

(2) PSO 算法的基本操作定义如下:

- (位置, 位置): 结果为一组交换序列, 表示速度.

× (实数, 速度): 对实数 $c \in (0, 1)$, 假设速度 v 的长度为 k 个置换序列, 乘法操作是截取速度列表, 使得新的速度的长度等于 $c \times k$ (取整).

⊕ (ω , 速度; α, β , 加速度): 假设速度 v 的长度为 k , 加速度的长度为 l , 运算操作是: 从第 $k * \omega$ (取整) 位置开始, 把速度 v 的置换序列替换为加速度的前 $\alpha * l$ (取整) 个交换序列.

+ (位置, 速度): 速度的交换序列依次作用于粒子的某个位置, 结果得到一个新的位置.

$$V'_{id} = \omega V_{id} \oplus \alpha (P_{id} - X_{id}) \oplus \beta (P_{gd} - X_{id}) \quad (3)$$

其中 α, β ($\alpha, \beta \in [0, 1]$) 为随机数. $\alpha (P_{id} - X_{id})$ 表示基本交换序 ($P_{id} - X_{id}$) 中的所有交换子以概率 α 保留; 同理, $\beta (P_{gd} - X_{id})$ 表示基本交换序 ($P_{gd} - X_{id}$) 中的所有交换子以概率 β 保留. 可以看出, α 的值越大, ($P_{id} - X_{id}$) 保留的交换子就越多, P_{id} 的影响就越大; 同理, β 的值越大, ($P_{gd} - X_{id}$) 保留的交换子就越多, P_{gd} 的影响就越大.

PSOACO 算法采用节点序列的编码机制, 每个个体被编码为源节点到各个目标节点的节点序列. PSO 算法编码时使用的备选路径集由后向蚂蚁生成, PSO 编码时, 分别从每个目的节点的备选路径集中各选出一条路径组成一棵组播树, 将每棵树表示为粒子群中的一个粒子. 然后通过 PSO 算法得到一些较优的粒子, 将这些粒子按适应度函数 $F(x)$ 的值由大到小进行排列, 选取适应度函数值较大的前 M 个粒子转为粒子更新蚂蚁.

2.2.3 PSOACO 算法中 ACO 的设计

(1) 格网结构生成

① 在算法初始,组播源节点向网络中广播到各个目的节点的前向蚂蚁,前向蚂蚁 k 以组播接收者为目的地址在网络中进行寻路,并沿途记录下所走过的逆向路径.当前向蚂蚁 k 到达组播接收者后转化为后向蚂蚁 k ,在返回源节点的过程中,后向蚂蚁 k 依次对经过的链路进行信息素局部更新.

$$\tau_{ij}(t+1) = (1 - \rho_1) * \tau_{ij}(t) + \rho_1 * H(x) \quad (4)$$

$$H(x) = \frac{\lambda e^{h(x)}}{\text{cost}} \quad (5)$$

$$h(x) = 1 - \frac{\text{del}}{DL} \quad (6)$$

其中, $\rho_1 \in (0, 1)$ 是局部信息素挥发系数, cost 和 del 分别是路径的费用和时延, DL 表示时延的约束限制, λ 是调整系数且是正数.

② 利用后向蚂蚁分别得到从源节点到各个目的节点满足 QoS 约束条件的所有可能路径集合,作为 PSO 编码时使用的备选路径集.通过 PSO 算法得到一些粒子更新蚂蚁,粒子更新蚂蚁对网络进行全局信息素更新.

$$\tau_{ij}(t+1) = (1 - \rho) \tau_{ij}(t) + \rho * F(x) \quad (7)$$

其中, $\rho \in (0, 1)$ 是全局信息素挥发系数, $F(x)$ 为粒子更新蚂蚁所对应较优粒子的适应度函数值.

(2) 路径的选择

前向蚂蚁 k 根据式(8)选择节点 i 的下一跳节点 j .

$$p_{ij}^k(t) = \begin{cases} \frac{\tau_{ij}^\alpha(t) \eta_{ij}^\beta(t)}{\sum_{s \in \text{allowed}_k} \tau_{is}^\alpha(t) \eta_{is}^\beta(t)}, & j \in \text{allowed}_k \\ 0, & \text{否则} \end{cases} \quad (8)$$

其中, α 和 β 分别反映了第 k 只蚂蚁在运动过程中所积累的信息素与启发信息在蚂蚁选择路径中的重要程度, $\tau_{ij}(t)$ 表示在 t 时刻链路 (i, j) 上的信息素量, $\eta_{ij}(t)$ 为启发信息且 $\eta_{ij}(t) = 1/\text{cost}(i, j)$, 可使节点倾向于选择费用小的链路. allowed_k 表示蚂蚁 k 下一步允许转移的节点集合, 不包括蚂蚁 k 已访问的节点.

2.2.4 PSOACO 算法实现过程

PSOACO 算法的实现步骤如下:

步骤 1: 初始化网络. 给出 QoS 的约束条件, 删除不满足带宽要求的链路以及由此产生的孤立节点集. 若孤立节点集中存在目的节点, 则本次路由请求失败, 否则基于新的拓扑图进行路由搜索;

步骤 2: 源节点广播到各个目的节点的前向蚂蚁, 并将源节点加入 VisitedNodes 中; 中间节点判

断前向蚂蚁的时延是否已超出约束限制, 若已超则前向蚂蚁死亡; 前向蚂蚁 k 以组播接收者为目的地址在网络中进行寻路, 并沿途记录下所走过的逆向路径, 直至前向蚂蚁到达组播接收者;

步骤 3: 组播接收者收到前向蚂蚁后, 将其转为后向蚂蚁, 按照 VisitedNodes 中节点序列的反方向转至上一个节点, 根据式(4)进行局部信息素的更新. 重复此步骤, 直至后向蚂蚁到达源节点;

步骤 4: 源节点收到首个后向蚂蚁并等待 ΔT_c 时间后, 根据收到的所有后向蚂蚁携带的路径信息, 找到从源节点到各个目的节点满足 QoS 约束条件的所有可能路径, 并将其分别放入各个目的节点的备选路径集;

步骤 5: 运行 PSO 算法得到了一些较优的粒子, 通过发送对应的粒子更新蚂蚁, 按照式(7)对网络进行全局信息素更新;

步骤 6: 生成一个带有信息素浓度的格网组播结构, 组播路由建立过程完成;

步骤 7: 组播路由建立后, 组播源节点即可发送数据包. 由于算法在寻找路由过程中留有一定冗余路径, 故节点可根据式(8)来选择下一跳进行数据包传输.

在组播路由维护过程中, 每个节点通过周期性地广播 Hello 报文来维护一个邻节点表. 当数据包传输至某节点 V_i 处检测到有效路由的下一跳链路中断或包传输已超时, 则节点 V_i 立即产生异常通告信息通知源节点, 源节点对此异常通告进行计数.

另外源节点周期性地向外发送前向蚂蚁, 组播接收者收到前向蚂蚁后将其转为后向蚂蚁, 并在逆向路径上进行局部信息素的更新, 以此来达到对组播路由的动态维护. 当组播源节点收到各个组播接收者返回的后向蚂蚁后, 将所得有效路径分别放入 PSO 编码时使用的备选路径集. 各个备选路径集均最多保留 s 条不同路径, 使用先进先出法来删除或覆盖旧的路径记录, 以保留时间最新的备选路径. 若时间周期 ΔT 内异常通告数大于某一阈值, 则运行 PSO 算法, 通过粒子更新蚂蚁对格网结构进行信息素全局更新, 以加速组播路由修复过程.

3 实验结果及分析

仿真实验选择具有一定代表性的协议 MAODV^[8] 作为比较对象, 通过数据包的成功传输率和平均端到端传输延迟 2 个指标来比较算法的性能.

数据包传输率如图 1 所示. 随着节点移动速率的增加, PSOACO 与 MAODV 的包传输率均有所下降, 但 MAODV 下降得较 PSOACO 快. 原因是 PSOACO 在路由寻找过程中已选择了质量较好的组播路由进行传输, 并且由于有冗余路径信息, 使得组播数据由于链路失效而无法传输时, 可通过其他路径到达目的节点, 这就在一定程度上保证了数据包传输的成功率. 而 MAODV 采用的是转发树结构, 当某个中间节点断开时, 其连接的目的节点将无法收到数据, 被迫执行转发树的回复过程.

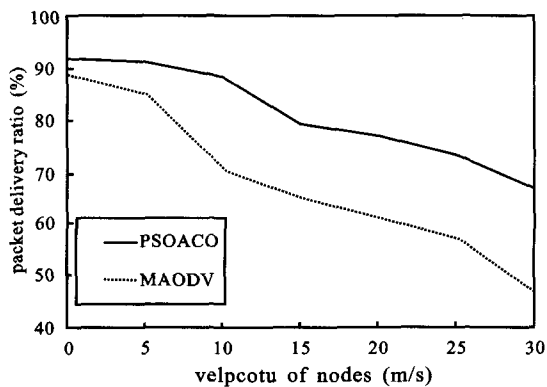


图 1 数据包成功传输率的比较

端对端时延随节点移动速率的变化情况如图 2 所示. 可以看出, PSOACO 组播路由算法在端对端时延方面具有较好的性能. 原因是 PSOACO 在寻找路由时已经对路由的时延做了评价, 数据在传输时会选择端对端时延相对较小的路径传输, 并且路径冗余信息的存在以及路由维护时 PSO 思想的引入, 也缩短了由于节点移动而修复路由的时间. 而 MAODV 中的源节点收到应答信号后不会立即启动组播路由实施数据传输, 在发送激活信令之前, 必须等待一个特定的时间以确认多个应答信令并选择最好的那条记录, 从而增加了寻路时间.

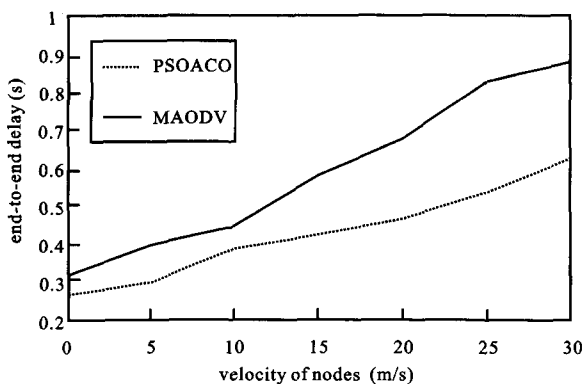


图 2 平均端到端延迟的比较

4 结束语

文中提出了一种融合粒子群优化思想的改进蚁群算法. 该算法融合 PSO 思想以加速蚁群算法在路由发现及维护时的收敛速度, 进一步提高算法对 Ad Hoc 网络环境的适应能力. 实验结果表明, 与 MAODV 路由协议相比, 该算法在数据包传输率及端对端时延方面具有更好的性能, 能较好适应 Ad Hoc 网络环境的变化.

参考文献:

- [1] 刘伟群, 李元臣. 基于蚂蚁算法的多约束 QoS 组播路由选择[J]. 微电子学与计算机, 2008, 25(7): 17-20.
- [2] 宋书强, 叶春明. 基于量子粒子群算法的 QoS 组播路由问题研究[J]. 微电子学与计算机, 2009, 26(4): 206-209.
- [3] 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): 23-31.
- [4] Shen C C, Jaikao C. Ad hoc multicast routing algorithm with swarm intelligence[J]. Mobile Networks and Applications, 2005, 10(1): 47-59.
- [5] Lee S, Chang H S. An ant system based multicasting in mobile ad hoc network[C]// The 2005 IEEE Congress on Evolutionary Computation. [s. l.]: IEEE, 2005: 1583-1588.
- [6] Kennedy J, Eberhart R C. Particle swarm optimization [C]// Proceedings of IEEE International Conference on Neural Networks. Piscataway, NJ: IEEE service center, 1995: 1942-1948.
- [7] Clerc M. Discrete particle swarm optimization illustrated by the traveling salesman problem[EB/OL]. [2007-04-20]. <http://www.mauriceclerc.net>.
- [8] 王楷, 肖诗松, 赵锦元. Ad Hoc 网络中基于粒子群优化的 QoS 多播路由研究[J]. 微电子学与计算机, 2006, 23(9): 41-43.

作者简介:

梁淑萍 女, (1987-), 硕士研究生. 研究方向为无线网络、智能算法.

毛力 男, (1967-), 副教授, 硕士生导师. 研究方向为智能算法、计算机网络.

马亦先 女, (1983-), 硕士研究生. 研究方向为无线网络、智能算法.