

## 【理论与探索】

## AODV 路由协议负载均衡的改进\*

刘 锐, 曾素华

(重庆邮电大学 计算机科学与技术学院, 重庆 400065)

**摘要:** 鉴于 Ad hoc 网络在军事和民用上已得到广泛应用, 而 AODV 是一种比较优秀的 Ad hoc 路由协议, 但是随着网络负荷的增大, 其网络性能下降很快, 提出一种根据节点的负载情况, 均衡 AODV 协议的负载的改进方法. 仿真试验表明, 改进后的协议性能更加稳定, 可扩展性更好.

**关键词:** AODV; 路由请求; 无理由路由回复; 负载均衡; 可扩展性

**中图分类号:** TN87

**文献标识码:** B

**文章编号:** 1006-0707(2008)06-0146-03

Ad hoc 网络是一种无线移动网络, 每个节点都可以自由移动. 因此, 网络的拓扑结构是动态变化的, 在 Ad hoc 网络中, 由于无线传输范围的限制, 网络中不能直接通信的 2 个节点间需要通过中间节点转发数据分组, 因此, 网络中的节点具有路由功能. 针对 Ad hoc 网络的特性, AODV 路由协议是一种比较优秀的协议, 它在一定的程度上有比其他协议较小的路由开销和更好的扩展性能. 但是这种路由协议在网络负荷较大的情况下, 其网络性能下降很快, 为此提出一种均衡负载的改进方法, 提高网络的吞吐率<sup>[1]</sup>.

## 1 负载均衡方法

### 1.1 负载均衡方法

这种方法的目的是阻止没有必要的路由负载, 以提高 AODV 路由协议的可扩展性和吞吐量. 它基于局部网络的负载水平, 决定是否抑制节点在这条路径上(例如包的转发)变为 1 个中间节点. 各个节点通过简单的放弃路由请求或路由答复放弃转发角色. 在轻负载的情况下, 每个节点遵循传统的反应式路由协议的规则, 各个节点都可以担当中间节点角色, 转发路由消息提供最短路径. 在局部网络负荷饱和的时候(当达到预先设定的值的时候), 节点在请求新的路由时, 拒绝担当中间节点的角色, 防止了不合需要的路由消息数据包<sup>[2-3]</sup>.

### 1.2 节点负载精确测量法

各个节点利用在 MAC 层测得的信道利用率参数来测量负载状态. 信道利用率是更加精确的负载状态指示器, 比平均队列大小更加准确; 而拥塞是会发生的, 即使平均队列大小(轻负荷的节点)是极其的小.

本负载均衡方法提供了一种暗示的调用允许进入机制. 每个节点在收到新的路由请求时, 预知网络状态饱和时将会拒绝转发路由请求. 节点在下列几种情况下拒绝变成新路径的一部分:

- 1) 这个信道的质量不适合新的路径, 由于高度冲突可能性或拥塞;
- 2) 新的路由请求加重了拥塞, 使得数据包丢包的几率增加;
- 3) 负载均衡路由方法能被用来代替使用普通的容易发生拥塞的最短路径法.

### 1.3 负载均衡算法

在算法中, 每个节点选择性的转发路由信息(例如路由请求、中间节点路由答复或无理由路由答复), 它是否连接到新的路径, 依靠局部负载水平( $L_i$ ): ① 每个节点繁殖路由消息, 当  $L_i$  是轻的时候; ② 每一个节点停止转发路由消息, 当  $L_i$  是饱和的时候.

负载水平测量: 建立在每个节点的 MAC 层可以一直监视共享媒介的事实之上的, 假定每个节点使用 802.11 DCF MAC 协议能够知道节点的带宽. Ad hoc 网络的信道状态能被分类为空闲、传输、接收和冲突. 在当前的 MAC 协议的设计, MAC 层保持监视信道的状态改变到决定当信道是空闲的时候. 使用这种观测, 在改进的协议中, 每个节点累计信道的前一个状态在空闲时间、传输时间、接收时间和冲突时间各自的持续时间, 无论何时发生改变, 到防止突然的改变数值, 每个时间数值和前一个数值是平滑的. (空闲时间 =  $\alpha$  \* 空闲时间 +  $(1 - \alpha)$  \* 前一次空闲时间). 其他, 周期(每次 Interval)信道利用率  $U_i$ , 传输概率  $TX_i$  和冲突累计时间  $T_c$  计算如下:

\* 收稿日期: 2008-10-25

作者简介: 刘锐(1973—)男, 四川泸州人, 讲师, 硕士研究生, 主要从事网络通信与管理研究.

$$U_i = 1 - \left( \frac{\text{idle\_time}}{\text{INTERVAL}} \right) \quad (1)$$

$$TX_i = \left( \frac{\text{trans\_time}}{\text{INTERVAL}} \right) \quad (2)$$

$$T_c = \left( \frac{\text{coll\_time}}{\text{INTERVAL}} \right) \times \left( \frac{\text{INTERVAL}}{\text{SLOT\_TIME}} \right) \quad (3)$$

用上面所定义的值给出了一种算法去决定负载水平. 在这里,  $\rho_{\max}$  是 802.11 DCF 协议的试验所得的最大吞吐量.  $\rho$  是预先设定的饱和量 ( $\rho_{\max}$ ). 在本研究中, 假定  $\rho_{\max}$  与  $\rho_{\text{single}}$  是相等的.  $\rho_{\text{single}}$  是 802.11 DCF 协议 1 个有 RTS/CTS 机制的节点的最大吞吐量.  $Q_i$  表示平均队列大小. 当前队列大小累加到总队列大小 (total queue size), 不论信道状态什么时候改变, 状态改变量 (status change cnt) 加 1. 用队列总大小 (total queue size) 除以变化总次数 (status change cnt) 来测量平均队列大小 ( $Q_i = \text{total queue size} / \text{tatus change cnt}$ ).  $L_i$  为负载水平.

饱和点的精确的测量是很重要的问题. 因此, 详细地描写上面列举的 2 种情况:

If ( $U_i > \rho$ ) or ( $Q_i > \text{queue\_predefine}$ )  
 Then  $L_i = \text{SATURATED}$ (饱和)  
 Else  $L_i = \text{MODERATE}$ (轻负荷)

1) 信道利用率 ( $U_i$ ) 大于预先设定的最大吞吐量, 当然这个最大吞吐量小于 IEEE 802.11 DCF 协议<sup>[4]</sup> 试验所得的最大吞吐量. 这种情况是简单易懂的, 因为信道的利用率就是局部网络的真正容量.

2) 节点的平均队列大小 ( $Q_i$ ) 大于预先定义值, 说明队列溢出即将来临, 那是网络饱和的症状.

## 2 AODV 中应用负载均衡算法

### 2.1 AODV 协议

AODV 是 DSDV 算法的改进, 它与 DSDV 的区别在于它是反应式路由协议. 为了找到通往目的节点的路由, 源端将广播 1 个路由请求分组, 邻居节点依次向周围节点广播此分组直到该分组被送到 1 个知道目的节点路由信息中间节点或目的节点本身. 1 个节点将丢弃重复收到的请求分组, 路由请求分组中的序列号用来防止路由环路, 并能判断中间节点是否响应了相应的路由请求. 当节点转发路由请求分组时, 它会将其上游节点的标志 ID 录入路由表, 从而构建 1 条从目的节点到源节点的反向路由. 当源端移动时, 它会重新发起路由发现算法; 如果中间节点移动, 那么与其相邻的节点会发现链路失效并向其上游节点发送链路失效消息 (RRER) 而且一直传到源节点, 而后源节点根据情况重新发起路由发现过程. AODV 的实现组合 DSR 和 DSDV 协议既具有 DSR 协议的路由发现和路由维护功能, 又使用了 DSDV 采用的逐跳路由、序列号和 Beacon 消息.

AODV 通过路由请求消息 (RREQ) 来进行路由发现. 路

由发现规程广播 1 个路由请求 (RREQ) 消息给它所有的邻节点, 并等待路由应答 (RREP) 消息. 若路由请求消息到达目的节点或者 1 个可以直接到达该目的节点的中间节点时, 该节点产生 1 个路由应答消息并发送回请求路由消息的源节点. 当路由应答消息到达源节点以后就建立了 1 条路由. 如果 1 个中间节点接收到 RREQ 消息, 但是它既不是目的节点又没有到目的节点的可达路由时, 该中间节点要转发该 RREQ 消息. 同时还要在它的路由表中建立 1 个到源节点的反向路由表项, 这是 1 个和发送前一个转发 RREQ 消息的节点只有一跳之隔的路由. 这样做的目的是建立 1 个从目的节点到源节点的反向路由, 为发送 RREP 消息提供了返回的路由. 当目的节点收到 RREQ 消息后, 它会生产 1 个 RREP 消息, 生产 RREP 消息的节点可能是目的节点本身, 或者是一个有到目的节点路由的中间节点. 如果 RREP 是目的节点本身产生的, 它从 RREQ 中获取相应的信息填入 RREP. 当 RREP 是中间节点产生时, 而且当 RREQ 的标准段是 G 时, 中间节点要产生一个无理由 RREP 消息 (Gratuitous RREPs), 沿着它已知道的路由发送到目的节点.

AODV 的路由维护是靠各个节点以固定的时间间隔发送 Hello 消息, 通知它的邻节点, 如果相邻的节点在一定的时间内没有收到则发送 RERR 到与此节点有相关路由的节点 (通过 Precursor Lists 来确定这些节点), 并更新路由信息.

### 2.2 AODV 的负载均衡改进

AODV 贮藏的是仅仅一跳的路由信息, 主要的开销是在路由请求和路由答复上. 因此路由协议修改为: 当一个新的、TTL (time to live) 大于“0”的 RREQ (路由请求消息) 进来时, 根据负载水平 ( $L_i$ ) 不同时分别作处理: ① 负载轻的时候, 节点转发 RREQ; ② 饱和的时候, 节点停止转发.

## 3 仿真环境和仿真结果

为了验证改进后 AODV 协议, 将利用 NS2 对改进后的协议进行仿真试验比较.

### 3.1 仿真模型

本研究将对进行负载均衡改进后的 AODV 协议 (enhanced AODV) 和传统的 AODV 进行仿真比较. 仿真模拟了 100 个点在 1 000 m × 1 000 m 情况下的仿真试验. 数据链路层和 MAC 层采用 802.11 规范, 带宽为 2 M. 包长为 512 bytes, 发送速度为 5 packets/s. 对于控制消息数量的比较, 节点最大移动速度为 5 m/s.

本研究将使用如下性能参数来对 AODV 和改进后的 AODV 协议 (enhanced AODV) 进行吞吐率、总的传输率、平均延迟、网络负载等性能的对比.

### 3.2 仿真结果及分析

从图 1~4 的仿真试验数据可以明显地看出, 当网络负载增大的时候, 进行了负载均衡改进的 AODV 协议工作性能更加稳定. 这使得改进后的 AODV 的扩展性能更好<sup>[5-8]</sup>.

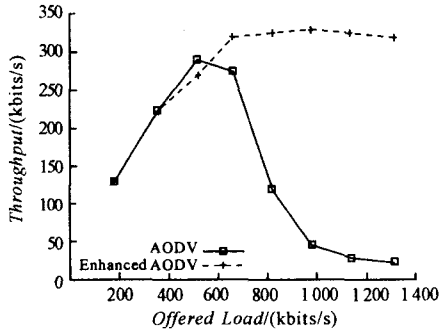


图1 吞吐率

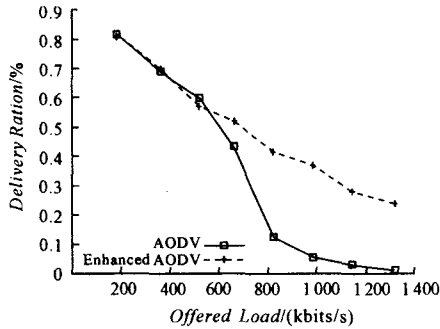


图2 总的传输率

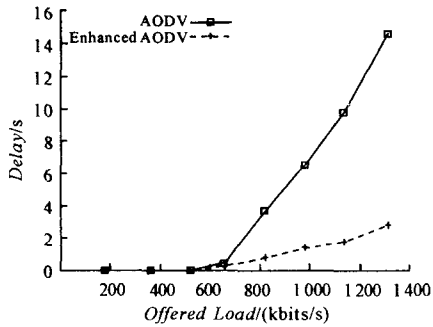


图3 平均延迟

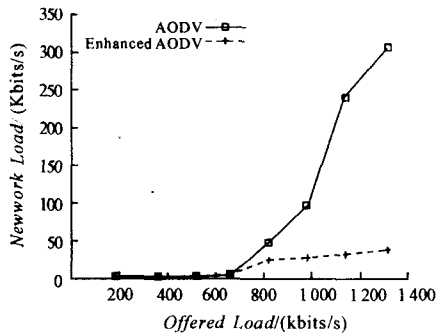


图4 网络负载

## 4 结束语

由于随着 Ad hoc 的扩展, AODV 路由协议因路由负荷加重而不可控制. 通过对 802.11DCF 协议的分析, 本研究提出了如何去对节点的饱和度进行测量, 并且根据测量数据来判断节点是否饱和, 进而决定节点是否要担负发送数据的功能. 通过对 AODV 负载均衡的改进, 仿真结果表明, 改进后的协议更稳定, 性能更好.

## 参考文献:

- [1] 王海涛, 郑少仁. 移动 Ad hoc 网络路由协议及性能比较[J]. 重庆邮电学院学报: 自然科学版, 2002(4): 73.
- [2] 王金龙, 王呈贵. Ad Hoc 移动无线网络[M]. 北京: 国防工业出版社, 2004: 81-83.
- [3] 方旭明. 移动 Ad Hoc 网络研究与发展现状[J]. 数据通信, 2003(4): 15.
- [4] 仲英济, 丁建英. 无线自组织网络在新闻采访系统中的应用研究[J]. 中国传媒科技, 2004(3): 41.
- [5] IEEE Computer Society LAN MAN Standards Committee. Wireless LAN Medium Access Protocol (MAC) and Physical Layer (PHY) Specification[S]. IEEE Std 802.11, 1997.
- [6] Ian D Chakeres, Elizabeth M. Belding-Royer. AODV Routing Protocol Implementation Design[EB/OL]. [2004-03-23]. <http://moment.cs.ucsb.edu/pub/wwan-chakeres-i.pdf>.
- [7] C Perkins, Nokia Research Center, E Belding-Royer. RFC3561, Ad hoc On-Demand Distance Vector (AODV) Routing[EB/OL]. [2008-01-23]. <http://www.knowsky.com/383622.html>.
- [8] Information Sciences Institute of USC. The Network Simulator ns-2[EB/OL]. [2008-01-23]. <http://www.isi.edu/nsnam/ns/>.