

基于传输控制协议的无线自组网主动队列管理建模

陈 亮^{1,2}, 张 宏¹

(1. 南京理工大学 计算机科学与技术学院, 江苏 南京 210094; 2. 南通纺织职业技术学院 信息系, 江苏 南通 226007)

摘要: 主动队列管理(AQM)通常研究队列控制器的设计. 作为被控对象, 传输控制协议(TCP)往往利用网络仿真器(NS)的仿真实现, 因此有必要研究无线自组网的TCP及AQM特性. 基于TCP窗口加性增-乘性减算法及排队原理, 推导了TCP窗口及队列的微分方程, 再基于比例积分AQM控制, 推导了拥塞丢弃概率的微分方程, 通过建立联立微分方程组, 提出了Ad Hoc网络TCP/AQM微分模型. 对比仿真显示, 新模型能较好地估计无线自组网的性能. 模型研究也表明, 网络跳数, 无线丢失和过小的队列成为AQM性能瓶颈, 队列信息则有助于TCP区分无线自组网的拥塞丢弃与无线丢失.

关键词: 无线自组网; 主动队列管理; 传输控制协议; 建模; 拥塞控制

中图分类号: TP393 **文献标识码:** A

Modeling active queue management based on transmission control protocol in Ad Hoc network

CHEN Liang^{1,2}, ZHANG Hong¹

(1. School of Computer Science and Technology, Nanjing University of Science and Technology, Nanjing Jiangsu 210094, China;

2. Information Department, Nantong Textile Vocational Technology College, Nantong Jiangsu 226007, China)

Abstract: The active queue management(AQM) is usually concerned with the queue controller design. As a controlled object, the transmission control protocol(TCP) is often realized by the simulation of Network Simulator(NS). Thus, it is necessary to study the characteristics of TCP and AQM in Ad Hoc network. Differential equations for the TCP window-size and queue-length are developed based on the TCP window additive-increase multiplicative-decrease algorithm and the queuing principle. Next, the differential equation of the congestion-loss probability is derived based on the proportional-integral AQM control. Then, the Ad Hoc network TCP/AQM differential model is proposed through the simultaneous differential equations. The comparison simulations show that the new model can estimate Ad Hoc network performance. The model research also shows that the number of hops, wireless loss and the very small queue become the bottleneck of the AQM performances. Furthermore, the queue information can help TCP discriminate between congestion loss and wireless loss in Ad Hoc network.

Key words: Ad Hoc network; active queue management; transmission control protocol; modeling; congestion control

1 引言(Introduction)

Ad Hoc网络是由移动终端组成的一个多跳自组织对等网络, 网络中的每个节点既是收发数据的主机, 也可能作为转发数据的路由器^[1]. 无线节点多对一的通信方式, 多跳拓扑及不稳定的无线链路, 都容易引起网络的局部或全局拥塞.

随着Ad Hoc网络规模越来越大, 结构日趋复杂, 仅仅依靠端到端的拥塞控制是不够的. 既然具有转发功能的中间节点是Ad Hoc网络拥塞直接的感受者, 考虑在Ad Hoc网络的中间节点引入主动队列管理(active queue management, AQM)机制, 通过动态检测并管理路由中的数据堆栈长度来预知可能产生的网络拥塞, 并及时通知源端, 使之能够提早采取措施, 从而尽量避免更严重的数据丢失, 实现对

网络拥塞的监测和预防. 近年来人们基于有线网络的研究^[2], 提出了一系列无线AQM算法. 文献[3]显示在有线网络中应用广泛的RED(randomly early detection)算法在Ad Hoc网络中具有不公平性, 文献[4]将有线网络的PID(proportional integral derivative) AQM应用于无线网络, 文献[5]根据无线信道特性设计了基于 H_∞ 方法的AQM, 文献[6]则设计了Ad Hoc网络的模糊AQM控制器. 但在目前的AQM研究中, 研究者大多侧重控制器设计, 传输控制协议(TCP)作为控制的对象, 在Ad Hoc网络中的特性往往进行了简化, 一般交由NS(network simulator)仿真完成. 研究Ad Hoc网络拥塞及AQM, 不能脱离TCP这一背景. 文献[7]就TCP/AQM有线模型做了很多基础性工作, 文献[8]虽然对TCP-Vegas的AQM进行了数

学建模,但主要是从控制理论的角度进行研究,文献[9]对有线网络的路由器缓存需求进行了深入分析,但没有涉及AQM机制的影响,文献[10]提出了一种基于小缓冲区的变速率控制协议,但其应用背景仍是有线路由器.总的来说,TCP/AQM研究和建模多以高性能路由器和低误码率信道为背景,而Ad Hoc网络中作为“路由器”的转发节点,其带宽非常有限,另外无线信道误码率也较高,因而有必要对Ad Hoc网络的TCP/AQM机制进行建模.

本文一方面对Ad Hoc网络TCP建模,首先考虑IEEE 802.11协议的分布式协调功能(DCF)机制,获得多跳拓扑的TCP链路容量特性,其次将Ad Hoc网络的分组丢失分为拥塞丢弃和无线丢失,根据TCP拥塞窗口规则,获得TCP窗口特性,再次根据Lindley原理,推导了瓶颈队列的特性;另一方面对AQM建模,基于比例积分(proportional integral, PI)控制理论,推导了AQM控制器的拥塞丢弃概率特性,由此组建了Ad Hoc网络的TCP/AQM微分模型;最后提供了仿真结果和性能分析.

2 Ad Hoc网络TCP/AQM建模(Modeling Ad Hoc network TCP/AQM)

变量定义:

$W(t)$ 为TCP拥塞窗口值(分组), t 为时间(s);

TCP窗口值 $=\min\{\text{TCP拥塞窗口值}, \text{TCP接受窗口值}\}$,由于网络瓶颈一般不在接收端,可认为接受窗口足够大且变化很小,即TCP窗口值等于TCP拥塞窗口值,也用 $W(t)$ 表示;

TCP分组长度 $L_{\text{TCP}}(\text{bit})$,本文取8000 bit;

$q(t)$ 为瞬时队列长度(分组); q_0 为理想队列长度(分组);

N 为TCP连接数; NH 为Ad Hoc网络跳数;

α 为常系数($0 < \alpha < 1$);

C 为理想链路容量(Mb/s), C_M 为单跳网络的最大链路容量(Mb/s), C_{ava} 为 NH 跳数网络的可用链路容量(Mb/s);

$R(t)$ 为往返时间(s),用下标 i 代表第 i 个TCP流,则其往返时间 $R_i(t) = D_i + q(t)/C_{\text{ava}}$,其中: D_i 为固定的传输延时, $q(t)/C_{\text{ava}}$ 为排队延时;

$p(t)$ 为拥塞分组丢弃概率, $p_{\text{wl}}(t)$ 为无线分组丢失概率, $p_t(t)$ 为总丢失概率;

$B_i(t) = W_i(t)/R_i(t)$,为第 i 个TCP流的发送速率(Mb/s);

$th = [1 - p(t) - p_{\text{wl}}(t)] \sum_{i=1}^N \frac{W_i(t)}{R_i(t)}$,为网络吞吐量(Mb/s);

变量的数学期望 $E[\phi]$ 简写为 $\bar{\phi}$.

拓扑结构:

无线节点 $S_1, S_2, \dots, S_N$ 通过中间节点 G 向目的节点发送数据,瓶颈节点 G 坐标(100, 125),节点 D_1, D_2, D_3 及 D_4 坐标分别为(200, 280), (350, 280), (500, 280)及(650, 280),其余节点都在坐标(0,0)至坐标(50, 50)的区域内.所有无线节点的数据传送速率 R_{DATA} 和基本速率 R_{control} 都为1 Mb/s,传输距离250 m,载波侦测范围550 m.瓶颈节点 G 接口队列缓冲区最大长度为100 packets,拓扑结构如图1所示.

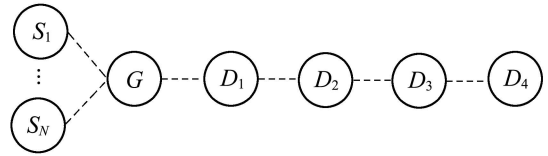


图1 Ad Hoc网络拓扑结构

Fig. 1 Ad Hoc network topology

2.1 TCP建模(Modeling TCP)

2.1.1 TCP链路容量特性(TCP link capacity property)

本文考虑802.11协议分布式协调功能(distributed coordination function, DCF)机制作为Ad Hoc网络的介质访问控制方式^[11]. DCF以载波侦听/冲突避免(CSMA/CA)为主,以请求发送/允许发送(RTS/CTS)消息交换机制为辅.在完整的DCF机制下,分组传送的周期为:首先等待 T_{DIFS} 和Back-Off(BO)平均耗费时间,之后发送RTS分组,等待 T_{SIFS} ,发送CTS分组,等待 T_{SIFS} ,此时才发送实际数据Data分组,再等待 T_{SIFS} ,由接收端送出ACK确认分组,完成了整个数据的传送过程,其原理如图2所示.



图2 DCF机制的数据传送过程

Fig. 2 Data transmission process in DCF mechanism

可见Ad Hoc网络在媒体访问层传送信息时存在无法避免的开销,此外上层的TCP确认机制也有一部分开销,这说明信道不是每时每刻都在传送数据,对Ad Hoc网络而言,其链路容量存在最大值.

TCP层的ACK确认分组只包含TCP/IP头,根据图2,考虑分组无碰撞的最理想状态,TCP及其ACK分组最小传输时间分别为

$$\begin{aligned} T_{\text{TCP.Date}} = & T_{\text{DIFS}} + T_{\text{BO}} + T_{\text{RTS}} + T_{\text{SIFS}} + T_{\text{CTS}} + \\ & T_{\text{SIFS}} + T_{\text{DATA}} + T_{\text{SIFS}} + T_{\text{MAC.ACK}} = \\ & T_{\text{DIFS}} + T_{\text{BO}} + (T_P + T_{\text{PHY}} + L_{\text{RTS}}/R_{\text{control}}) + \\ & T_{\text{SIFS}} + (T_P + T_{\text{PHY}} + L_{\text{CTS}}/R_{\text{control}}) + T_{\text{SIFS}} + \end{aligned}$$

$$\begin{aligned}
& [T_P + T_{PHY} + (L_{MAC} + L_{iph} + L_{TCP})/R_{DATA}] + \\
& T_{SIFS} + (T_P + T_{PHY} + L_{ACK}R_{control}), \\
& T_{TCP_ACK} = \\
& T_{DIFS} + T_{BO} + T_{RTS} + T_{SIFS} + T_{CTS} + \\
& T_{SIFS} + T_{ACK\ as\ Data} + T_{SIFS} + T_{MAC_ACK} = \\
& T_{DIFS} + T_{BO} + (T_P + T_{PHY} + L_{RTS}/R_{control}) + \\
& T_{SIFS} + (T_P + T_{PHY} + L_{CTS}/R_{control}) + T_{SIFS} + \\
& [T_P + T_{PHY} + (L_{MAC} + L_{iph})/R_{DATA}] + \\
& T_{SIFS} + (T_P + T_{PHY} + L_{ACK}/R_{control}).
\end{aligned}$$

则TCP流单跳的最大链路容量为

$$C_M = \frac{L_{TCP}}{T_{TCP_DATA} + T_{TCP_ACK}}.$$

其中: 传送Data分组的数据传送速率 R_{Data} Mb/s, 数值等于理想链路容量 C ; 传送RTS, CTS及ACK控制分组的基本速率 $R_{control}$ Mb/s.

当 $R_{Data} = 1$ Mb/s, $R_{control} = 1$ Mb/s及 $L_{TCP} = 8000$ bit时, 其余系数取802.11协议默认设置, 计算获得单跳网络的最大链路容量 $C_M = 0.65$ Mb/s.

当网络跳数为2, 即源节点 $S_1, S_2, S_3, \dots, S_N$ 通过中间节点 G 向目的节点 D_1 发送数据, 此时源节点与节点 G 能够感受到节点 D_1 的传输, 因而将与其共享单跳的最大链路容量. 以此类推, 在源节点与节点 G 能够感知的范围内, 跳数的增加意味着它们将与增加的节点共享信道容量. 由于 D_1, D_2 及 D_3 都在源节点与节点 G 的载波侦听范围内, 可用链路容量为

$$C_{ava} = C_M / NH, \quad NH = 1, 2, 3, 4.$$

而当跳数 NH 增加到5, 新增加的节点 D_4 已不在源节点及节点 G 的感知范围内, 5跳网络的可用链路容量将与4跳网络可用链路容量相等, 上式修正为

$$C_{ava} = C_M / \min\{NH, 4\}, \quad NH = 1, 2, 3, 4, \dots \quad (1)$$

2.1.2 TCP窗口特性(TCP window size property)

由TCP窗口特性, 当拥塞窗口超过设定的极限值时, 慢启动过程就停止了. 一旦拥塞窗口大于极限值后, 拥塞窗口变为线性增长, 即每经过一个 $R(t)$, 拥塞窗口才增长1个(加性增), 该过程称为拥塞避免, 如果没有分组丢失, 则拥塞避免阶段一直持续. 当收到3个重复ACK时, 则认为分组丢失, 拥塞窗口将被设定为现在拥塞窗口的一半(即系数 $\alpha = 1/2$, 乘性减). 随后, 源端又开始了新的慢启动.

由于有线网络中传输错误的概率很小, 如果出现分组丢失, 一般认为就是网络拥塞造成的, 因而TCP认为源端收到3个Dup ACK就是网络拥塞的标志. 但在Ad Hoc网络中, 由于干扰、噪声和冲突, 也会出

现分组丢失, 这是Ad Hoc网络TCP/AQM研究要考虑的因素. 由于无线自组网TCP流的丢包情况比较复杂, 本文考虑将丢失情况分为两类, 即拥塞造成的分组丢弃和无线信道造成的丢失, 记时间区间 $(t, t+R(t))$ 内拥塞丢弃分组次数为 $X(t)$, 无线信道丢失分组次数为 $Y(t)$, 则系统整体为 $[X(t)+Y(t)]$.

有线AQM研究显示^[12], 链路容量15 Mb/s且队长为200个分组时, 拥塞分组丢弃概率极小(低于1%), 而根据本文的NS仿真, 链路容量1 Mb/s而队长为50个分组时, $p(t)$ 不足10%. 无线丢失方面, 文献[13]以无线误码率 10^{-4} 为严重受干扰的信道, 无线误码率小于 10^{-6} 为干扰很小的信道, 无线误码率在 $10^{-6} \sim 10^{-5}$ 之间为信道干扰的一般状态. 由无线信道误码率BER与无线丢包率 $p_{wl}(t)$ 关系^[14], 有 $p_{wl}(t) = 1 - (1 - \text{BER})^{L_{TCP}}$.

本文取误码率 $\text{BER} = 6 \times 10^{-6}$, $L_{TCP} = 8000$ bit. 根据上式, 无线丢失概率 $p_{wl}(t) = 5\%$. 理论上无线丢失概率可达100%, 但对本文研究而言, $p_{wl}(t)$ 也不能过大, 否则会造成拥塞现象消失.

假定 $p_{wl}(t) = 5\%$ 而 $p(t) = 10\%$, 考虑最坏情况, 单纯叠加两种丢失, 则总丢失概率在15%左右. 在低带宽, 小队列的Ad Hoc网络, 其TCP窗口值 $W(t)$ 很小, 一般不到4个分组. 设总丢包次数 $[X(t)+Y(t)]$ 服从二项分布 $(W(t), p_t(t))$, 则有一般式

$$\begin{aligned}
& P\{X(t) + Y(t) = k\} = \\
& C_{W(t)}^k p_t^k(t) (1 - p_t(t))^{W(t)-k},
\end{aligned}$$

其中 $k = 0, 1, 2, \dots$. 当 $p_t(t) = 15\%$ 且 $W(t) = 4$, 有

$$\begin{aligned}
& P\{X(t)+Y(t)=0\} = C_4^0 0.15^0 (0.85)^{4-0} = 52\%, \\
& P\{X(t)+Y(t)=1\} = C_4^1 0.15^1 (0.85)^{4-1} = 37\%.
\end{aligned}$$

根据以上条件, 在一个 $R(t)$ 中, 拥塞及无线信道造成的分组丢失次数 $[X(t) + Y(t)]$ 是1或0的总概率达到89%, 而随着 $W(t)$ 减小到3, 这个概率可达94%. 在工程上, 可假设 $[X(t) + Y(t)]$ 为区间 $[0, 1]$ 之间的随机数, 假设的合理性也可从后文仿真中得到确认. 根据2.1.2节首段描述的TCP窗口特性, 忽略慢启动, TCP窗口值可写为

$$\begin{aligned}
& W(t + R(t)) = \\
& (W(t) + 1)[1 - (X(t) + Y(t))] + \\
& [1 + (1 - \alpha)W(t)](X(t) + Y(t)). \quad (2)
\end{aligned}$$

为了转化为微分形式, 式(2)化简为

$$\begin{aligned}
& W(t + R(t)) - W(t) = \\
& \frac{R(t)}{R(t)} - \alpha W(t)(X(t) + Y(t)). \quad (3)
\end{aligned}$$

考虑时间 $[0, t]$ 内拥塞造成的分组丢失次数 $M_1(t)$ 服从 $\lambda_1 = p(t)\bar{W}(t)/\bar{R}(t)$ 的泊松分布, 有

$$X(t) = M_1(t + R(t)) - M_1(t). \quad (4)$$

设时间 $[0, t]$ 内无线分组丢失次数 $M_2(t)$ 也服从参数为 $\lambda_2 = p_{wl}(t)W(t)/R(t)$ 泊松分布, 有

$$Y(t) = M_2(t + R(t)) - M_2(t). \quad (5)$$

将式(4)(5)代入式(3), 由TCP机制取 $\alpha=1/2$, 式(3)的微分形式为

$$dW(t) = \frac{1}{R(t)}dt - \frac{1}{2}W(t)d[M_1(t) + M_2(t)]. \quad (6)$$

对式(6)取数学期望, $R(t)$ 是 $q(t)$ 的函数, 考虑 $E[f(x)] \approx f(E[x])$, 对第 i 个TCP流, 有

$$\begin{aligned} dE[W_i(t)] &= \frac{dt}{R_i(E[q(t)])} - \\ &\frac{1}{2}E[W_i(t)](dE[M_1(t)] + dE[M_2(t)]). \end{aligned} \quad (7)$$

根据泊松过程性质 $E[M_j(t)] = \lambda_j t$, $j = 1, 2$, 有

$$\frac{d\bar{W}_i(t)}{dt} = \frac{1}{R_i(\bar{q}(t))} - \frac{1}{2}[p(t) + p_{wl}(t)] \frac{\bar{W}_i^2(t)}{R_i(\bar{q}(t))}. \quad (8)$$

2.1.3 TCP瓶颈队列特性(TCP bottleneck queue property)

考虑第 i 个TCP流在稳定状态下, 离散时刻的瓶颈队列分组到达数 $\{A(k), 0, 1, 2, \dots\}$, $a(k)$ 为离散间隔内分组到达数, 则 $a(k) = A(k) - A(k-1)$. 考虑拥塞丢弃及无线丢失对瓶颈队列的影响, 实际进入队列的分组数为 $[1 - p(k) - p_{wl}(k)]a(k)$. 由Lindley方程^[15], 并转化为连续时间上的函数, 有

$$\begin{aligned} q(t + \Delta t) &= \\ \max(q(t) + [1 - p(t) - p_{wl}(t)]a(t + \Delta t) - C_{ava}, 0), \end{aligned} \quad (9)$$

考虑 $a(t + \Delta t) \approx \bar{B}_i(t)\Delta t$, 则瓶颈队列的微分形式为

$$\begin{aligned} \frac{d\bar{q}(t)}{dt} &= \\ -C_{ava} + [1 - p(t) - p_{wl}(t)] \sum_{i=1}^N \frac{\bar{W}_i(t)}{R_i(\bar{q}(t))}. \end{aligned} \quad (10)$$

2.2 AQM控制器建模(Modeling AQM controller)

求解式(8)和式(10), 还需要拥塞丢弃概率的函数关系, AQM控制器设计的关键正是获得合理的拥塞丢弃概率. 本文目的不是设计AQM控制器, 因此选用PI控制, 建立其AQM微分模型. PI控制利用比例积分原理来调节拥塞丢弃概率, 其表达式为

$$p(t) = k_p[e(t) + \frac{1}{T_i} \int_0^t e(t)dt].$$

其中拥塞丢弃概率 $p(t)$ 作为控制器输出信号, 系统误差信号 $e(t) = q(t) - q_0$, $T_i = k_p/k_i$. 对上式求导, 据文献[16], 有 $\frac{d}{dt} \int_0^t e(t)dt = e(t)$, 化简取期望

$$\frac{dp(t)}{dt} = k_p \left[\frac{d\bar{q}(t)}{dt} + \frac{1}{T_i}(\bar{q}(t) - q_0) \right]. \quad (11)$$

2.3 Ad Hoc网络TCP/AQM微分模型(Ad Hoc network TCP/AQM model)

由以上推导, 联立式(1)(8)(10)及(11), Ad Hoc网络TCP/AQM微分模型(AHDM)为

$$\begin{cases} C_{ava} = C_M / \min\{NH, 4\}, \\ \dot{W}_i(t) = \frac{1}{R_i(q(t))} - \frac{W_i^2(t)}{2R_i(q(t))} [p(t) + p_{wl}(t)], \\ \dot{q}(t) = -C_{ava} + [1 - p(t) - p_{wl}(t)] \sum_{i=1}^N \frac{W_i(t)}{R_i(q(t))}, \\ \dot{p}(t) = k_p \left[\frac{dq(t)}{dt} + \frac{1}{T_i}(q(t) - q_0) \right]. \end{cases} \quad (12)$$

为书写简洁, 式(12)省略了 $W(t)$ 和 $q(t)$ 的数学期望上标. 使用Runge-Kutta (R-K)法解微分方程组(12), 可得 $W(t)$, $q(t)$ 及 $p(t)$ 的数值解, 而吞吐量可由定义计算获得.

3 仿真与研究(Simulation and research)

实验1 跳数对TCP性能的影响.

$$C = 1 \text{ Mb/s} = 125 \text{ packet/s} (L_{\text{TCP}} = 8000 \text{ bit}),$$

$$N = 20, p_{wl}(t) = 0\%,$$

仿真时间100 s. 根据式(1), AHDM模型与NS仿真的跳数与吞吐量关系见图3.

由图3可见, 当网络跳数增加, Ad Hoc网络TCP性能明显下降, 这是由于无线信道属性决定其一般采用CSMA/CA机制, 无线节点在自己的载波侦听范围内能够感知其他节点传输, 跳数的增加意味着它们将与增加的节点共享信道容量, 而当新增加的节点不在节点 G 的感知范围内, 吞吐量开始稳定于某一数值上. 另外, 由于仿真初期的路由和通信开销, 随着跳数的增加, NS仿真中吞吐量开始时间产生了较大的延迟.

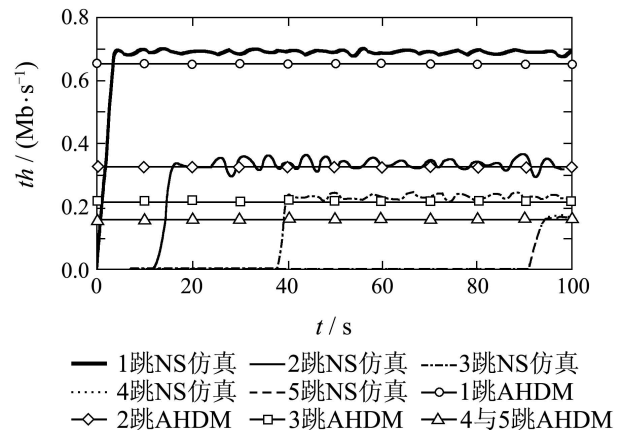


图3 跳数与吞吐量关系图

Fig. 3 Relation between the number of hops and throughput in AHDM and NS simulation

实验2 无线分组丢失对TCP性能的影响.

理想链路容量 $C = 1 \text{ Mb/s}$ ($L_{\text{TCP}} = 8000 \text{ bit}$), TCP连接数 $N = 20$, $NH = 2$, $q_0 = 50$, $p_{\text{wl}}(t) = 5\%$, $W(t)$ 初值为1, $q(t)$ 及 $p(t)$ 初值均为零. 根据Ziegler-Nichols方法, PI控制器系数整定为 $k_p = 0.2$ 及 $k_i = 0.02$, 以R-K法步长0.01求解式(12). 为了对照, 文献[7]有线网络DE(differential equation)模型的 $W(t)$ 方程增加无线丢失率 $p_{\text{wl}}(t)$ 项, 其RED也改为PI, 系数同AHDM模型. 瓶颈队列, TCP窗口, 总丢失概率及吞吐量仿真结果如图4至图7所示.

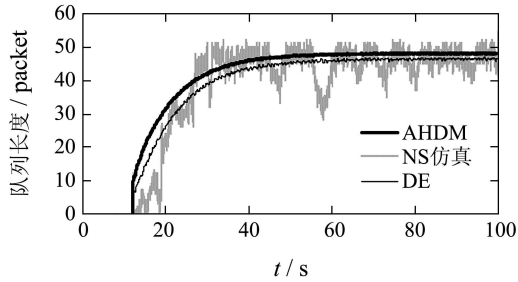


图4 $p_{\text{wl}}(t) = 5\%$ 时瓶颈队列长度曲线

Fig. 4 Bottleneck queue length when $p_{\text{wl}}(t) = 5\%$

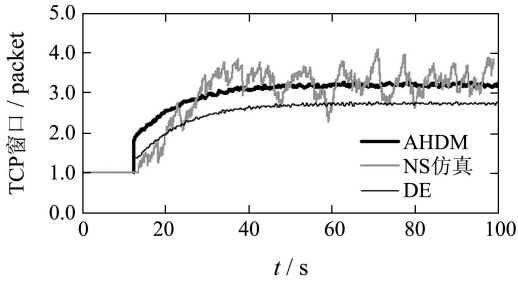


图5 $p_{\text{wl}}(t) = 5\%$ 时TCP窗口曲线

Fig. 5 TCP window size when $p_{\text{wl}}(t) = 5\%$

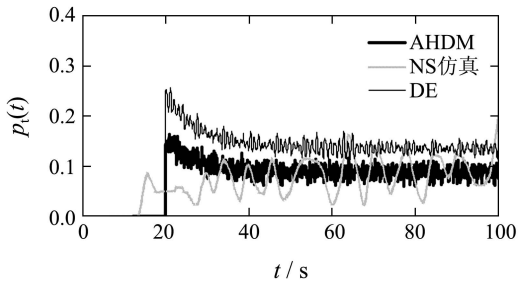


图6 $p_{\text{wl}}(t) = 5\%$ 时总丢失概率曲线

Fig. 6 Total loss probability when $p_{\text{wl}}(t) = 5\%$

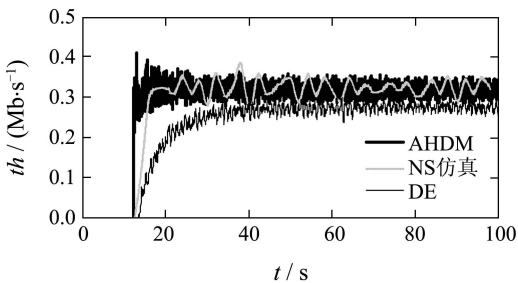


图7 $p_{\text{wl}}(t) = 5\%$ 时吞吐量曲线

Fig. 7 Throughput when $p_{\text{wl}}(t) = 5\%$

通过对比仿真, AHDM模型可以较好地估计Ad Hoc网络性能, 而有线的DE模型出现了较大的误差, 见图3~7. 有线高速路由器能很快处理排队的分组, 尽管有线AQM队列设定值一般在200分组左右^[17,18], 排队时延仍然可以忽略, 长队列使得拥塞丢弃概率很小^[12]; 又由于有线信道的可靠性, 分组丢失一般就是拥塞造成的. 可见DE模型的应用有两个前提: 1) 拥塞丢弃概率可以忽略; 2) 信道的高可靠性保证传输错误极少发生. 但在Ad Hoc网络中, 由于低可用链路容量, 小队列及不稳定的无线信道, 使得拥塞丢弃概率和传输错误都不能忽略, 因此DE模型应用于Ad Hoc网络时, 会错误地忽略排队时的分组丢失, 认为这些丢失的分组仍在队列中, 造成控制器过调, 使得有线DE模型的丢弃概率高于AHDM模型, 从而造成误差.

运用AHDM模型分析TCP/AQM机制对网络性能的影响, 网络跳数与网络性能关系, 见表1; 无线丢失概率与网络性能关系, 见表2; 理想队列与网络性能关系, 见表3; 由于式(12)的物理意义, 表数据均为平均值.

就网络跳数 NH 而言, 表1显示, 随着跳数 NH 从2增加3, 在吞吐量减小的情况下, TCP窗口出现增加, 这是由于 $R(t)$ 作为 $q(t)$ 与 C_{ava} 之比, 当跳数增加到3, 其 C_{ava} 减小, 从而使 $R(t)$ 变大, 增大的 $R(t)$ 意味着在此期间TCP发送机会的增加, 即TCP窗口增大. 需要说明的是, 此时TCP窗口的增大是以传送时间增大为代价的, 因此这种TCP窗口的增加对提高网络吞吐量没有帮助, 吞吐量仍取决于跳数. 当跳数为4时, 尽管吞吐量在 C_{ava} 附近, 但队列为9, 说明队列控制已经失效. 可见, 基于固定拓扑设计的经典PI队列控制较难适应拓扑灵活的Ad Hoc网络, 多跳成为Ad Hoc网络AQM性能的结构瓶颈.

表1 $p_{\text{wl}}(t) = 0$ 且 $q_0 = 50$ 时网络跳数与网络性能关系

Table 1 Relation between the number of hops and network performance ($p_{\text{wl}}(t) = 0$ and $q_0 = 50$)

NH	$W(t)/$ packet	$th/$ ($\text{Mb} \cdot \text{s}^{-1}$)	$q(t)/$ packet	$p_t(t)/$ %
2	3.5	0.325	50	6.21
3	4.5	0.220	41	4.47
4	3.2	0.165	9	5.83

就无线丢失概率 $p_{\text{wl}}(t)$ 而言, 表2说明, 当 $p_{\text{wl}}(t)$ 小于3%, 拥塞现象依然占主导, 队列控制器仍可将吞吐量保持在两跳网络可用链路容量附近; 而当 $p_{\text{wl}}(t)$ 超过3%, 无线丢失开始占主要地位, 其吞吐

量持续下降; 当无线丢失概率为10%时, 拥塞丢弃现象基本消失; 当无线丢失概率为0%, 5%及10%时, 总的丢失概率分别为6.2%, 8.8%及10%, 这说明无线丢失和拥塞丢弃在效应上并非单纯叠加或抵消, 无线丢失不仅削弱了拥塞现象, 而且在丢失份额中占更大比例, 拥塞现象的削弱也未对性能有改善作用. 可见, 由于TCP源的发送直接取决于分组丢失情况, 当拓扑结构一定时, 无线分组丢失成为Ad Hoc网络AQM性能的首要瓶颈.

另外表2也显示, 随着无线丢包率的持续增加, TCP窗口和吞吐量都在下降, 将队列维持在理想值变得越加困难, AQM控制器对队列的管理最终将会失效, 拥塞现象也将完全消失. 但无线丢包的存在, 并不意味着Ad Hoc网络的AQM没有意义. 对Ad Hoc网络而言, 当只有拥塞丢弃时, 降低TCP窗口的策略是合适的, 而随着无线分组丢失的加剧, TCP不但不应降低TCP窗口, 反而要加大发送窗口, 以弥补无线信道不佳产生的分组丢失, 但问题在于TCP机制无法区分拥塞丢弃和无线丢失. 由于拥塞现象会使得瓶颈节点的瞬时队列不断地超过设定的理想值; 而无线丢失现象则会使队列不断低于理想值. 因此, 当队列长时间低于理想值时, 说明无线丢失占主导地位, TCP发送窗口应不变或适当增加; 当队列一旦高于理想值, 说明拥塞丢弃仍然存在, TCP则按原机制降低发送窗口. 在Ad Hoc网络误码信道下, 通过监控瓶颈队列动态信息, AQM有助于TCP源端判断分组丢失原因, 而当TCP应对无线丢包而增加发送窗口后, AQM也对随后可能发生的拥塞起到控制作用.

表2 两跳网络 $q_0 = 50$ 时无线丢包率与网络性能关系

Table 2 Relation between wireless loss probability and network performance in 2-hop network ($q_0 = 50$)

$p_{wl}/$ %	$W(t)/$ packet	$th/$ (Mb · s ⁻¹)	$q(t)/$ packet	$p_t(t)/$ %
0	3.5	0.325	50	6.21
1	3.3	0.326	48	2.37
3	3.2	0.325	48	3.29
5	3.1	0.320	47	8.81
10	2.6	0.305	32	9.95

就理想队列 q_0 而言, 从表3看出, 无论控制器设定的理想队列多大, 其吞吐量都在2跳的可用链路容量附近, 理想队列的设定似乎对网络性能影响不大. 但值得注意的是, 当理想队列设定过低($q_0 = 10$), 对应的TCP窗口值仅有1.7, 由于TCP窗口的最小值为1, 该TCP窗口值是很小的, 侧面说明20个TCP流遇

到了严重的公平性问题, 即由于理想队列值过小, 很多TCP流一直无法成功发送数据. 随着理想队列设定值的增加, 丢失概率逐渐变小, TCP窗口逐渐增加, 所以 q_0 设置应较大一些, 但 q_0 也不宜过大, 由于 $R(t)$ 为 $q(t)$ 与 C_{ava} 之比, 在跳数一定的情况下, 长队列会导致大的 $R(t)$, 更重要的是, 过长的队列会使得分组长时间待在队列中, 因为TCP源端需要确认分组到达后才能继续发送新的分组, 为防止过长的等待导致传输死锁, TCP会采用重传计时器来终止一直没有确认的分组, 这一过程称为超时, 而重传计时器设定的重传时间(retransmission time-out, RTO)往往基于平滑的往返时间, RFC793推荐 $RTO = 2R(t)$. $R(t)$ 过大容易导致超时, 即一旦重传计时器等待的时间超过RTO, TCP会认为正在队列中的分组已经全部丢失, 重新传送这些分组, 反而造成更严重拥塞问题, 所以理想队列设定也不宜过大.

表3 两跳网络 $p_{wl}(t) = 0$ 时理想队列与网络性能关系

Table 3 Relation between reference queue length and network performance in 2-hop network ($p_{wl}(t) = 0$)

$q_0/$ packet	$W(t)/$ packet	$th/$ (Mb · s ⁻¹)	$q(t)/$ packet	$p_t(t)/$ %
10	1.7	0.324	8	10.1
30	2.5	0.325	26	6.91
50	3.5	0.325	50	6.21
70	4.2	0.326	63	3.88
100	5.1	0.327	89	2.78

4 结论(Conclusion)

本文根据Ad Hoc网络的TCP及队列管理特性, 建立了Ad Hoc网络TCP/AQM模型, 研究了多跳及误码信道下的TCP/AQM机制. 通过与NS仿真对比, 本模型可以较好地估计Ad Hoc网络的性能. 研究也表明, 拓扑跳数成为Ad Hoc网络性能的结构瓶颈, 拓扑固定时, 无线分组丢失是影响Ad Hoc网络性能的主要因素, 理想队列设定过小则会带来公平性问题.

参考文献(References):

- [1] DAYIN P, OHM S. A mesh-based QoS aware multicast routing protocol[C] // *First International Conference of Network-Based Information Systems*. Regensburg: Springer, 2007, 4658: 466 – 475.
- [2] 陆锦军, 王执铨. 基于粒子群优化的网络拥塞控制新算法[J]. 电子学报, 2007, 35(8): 1446 – 1451.
(LU Jinjun, WANG Zhiqun. A new network congestion control algorithm based on particle swarm optimization[J]. *Acta Electronica Sinica*, 2007, 35(8): 1446 – 1451.)
- [3] XU K X, GERLA M, QIL T, et al. TCP unfairness in Ad Hoc wireless networks and a neighborhood RED Solution[J]. *Wireless Networks*, 2005, 11(4): 383 – 399.

- [4] 续欣, 汤凯, 马刘非, 等. 无线误码信道上的拥塞控制策略[J]. 通信学报, 2004, 25(12): 8 – 13.
(XU Xin, TANG Kai, MA Yifei, et al. Congestion control scheme on wireless loss-prone link[J]. *Journal on Communications*, 2004, 25(12): 8 – 13.)
- [5] ZHENG F, NELSON J. An H_∞ approach to congestion control design for AQM routers supporting TCP flows in wireless access networks[J]. *Computer Networks*, 2007, 51(6): 1684 – 1704.
- [6] NATSHEH E, JANTAN A B, KHATUN S, et al. Intelligent reasoning approach for active queue management in wireless Ad Hoc networks[J]. *International Journal of Business Data Communications and Networking*, 2007, 3(1): 16 – 35.
- [7] MISRA V, GONG WB, TOWSLEY D. Fluid-based analysis of a network of AQM routers supporting TCP flows with an application to RED[J]. *ACM SIGCOMM Computer Communication Review*, 2000, 30(4): 151 – 160.
- [8] BIGDELI N, HAERI M. AQM controller design for networks supporting TCP vegas: a control theoretical approach[J]. *ISA Transactions*, 2008, 47(1): 143 – 155.
- [9] 李玉峰, 邱菡, 兰巨龙, 等. TCP流竞争拥塞及拥塞链路的缓存需求研究. 软件学报[J]. 2008, 19(6): 1499 – 1507.
(LI Yufeng, QIU Han, LAN Julong, et al. Study on TCP flow-competing congestion and buffer requirement of the congested links[J]. *Journal of Software*, 2008, 19(6): 1499 – 1507.)
- [10] KELLY F, RAINA G, THOMAS V. Stability and fairness of explicit congestion control with small buffers[J]. *ACM SIGCOMM Computer Communication Review*, 2008, 38(3): 53 – 62.
- [11] 雷磊, 许宗泽, 刘旭. 适用于ad hoc网络的DCF协议节点自适应休眠机制. 通信学报[J], 2009, 30(10): 14 – 23.
(LEI Lei, XU Zongze, LIU Xu. Node adaptive sleeping mechanism of DCF protocol in ad hoc networks[J]. *Journal on Communications*, 2009, 30(10): 14 – 23.)
- [12] 任丰原, 林闯, 任勇, 等. 大时滞网络中的拥塞控制算法. 软件学报[J]. 2003, 14(3): 503–511.
(REN Fengyuan, LIN Chuang, REN Yong, et al. Congestion control algorithm in large-delay networks[J]. *Journal of Software*, 2003, 14(3): 503 – 511.)
- [13] CHATZIMISIOS P, VITSAS V, BOUCOUVALAS A C. DIDD back-off scheme: an enhancement to IEEE 802.11 DCF under burst transmission errors[C] // *Proceedings of the 2006 IEEE Sarnoff Symposium*. New York: IEEE, 2006: 138 – 141.
- [14] KUMAR A, MANJUNATH D, KURI J. *Wireless Networking*[M]. Morgan: McGraw-Hill Pub, 2006: 31 – 32.
- [15] LINDLEY D V. The theory of queues with a single server[J]. *Mathematical Proceedings of the Cambridge Philosophical Society*, 1952, 48(2): 277 – 289.
- [16] RAMANA B V. *Higher Engineering Mathematics*[M]. Tata: Tata McGraw Hill Publishing Company, 2008: 75 – 76.
- [17] CHO H, FADALI S, LEE H. Adaptive neural queue management for TCP networks[J]. *Computers and Electrical Engineering*, 2008, 34(6): 447 – 469.
- [18] CHEN C K, KUO H H, YAN J J, et al. GA-based PID active queue management control design for a class of TCP communication networks[J]. *Expert Systems with Applications*, 2009, 36(2): 1903 – 1913.

作者简介:

陈亮 (1982—), 男, 博士研究生, 研究方向为无线自组网及拥塞控制, E-mail: njustchenliang@yahoo.com.cn;

张宏 (1956—), 男, 教授, 研究方向为信息安全理论与技术, E-mail: zhong@mail.njust.edu.cn.