

一种基于通信量趋势预测的 Ad-hoc 网络拥塞控制策略

王文涛 王 豪 朱容波 郭 峰 郑 芳
(中南民族大学计算机科学学院 武汉 430074)

摘 要 针对反应式 MANET 路由协议在较大通信量模式下出现的 RREQ 报文拥塞问题,提出一种新的机制。该机制根据分组投递率和数据包发送速率的拟合函数曲线设定不同的阈值响应路由请求,当平均队列长度超出最大阈值时,以一定概率随机丢弃 RREQ 报文,同时根据拟合函数公式来决定 Hello 包发送周期。仿真结果表明,新的机制在没有明显增加路由由发起频率的情况下,能够有效降低平均端到端延时和 Hello 包开销,提高了分组投递率。

关键词 拥塞控制,拟合函数,阈值,Ad-hoc 网络

中图分类号 TP393 **文献标识码** A **DOI** 10.11896/j.issn.1002-137X.2014.08.033

Congestion Strategy Based on Traffic Trend Forecasting for Ad-hoc Networks

WANG Wen-tao WANG Hao ZHU Rong-bo GUO Feng ZHENG Fang
(College of Computer Science, South-Central University for Nationalities, Wuhan 430074, China)

Abstract For the RREQ message congestion problems of reactive MANET routing protocols in high traffic mode, this paper proposed a new mechanism. The mechanism sets different threshold to response to the route request according to the fitting curve of packet delivery ratio and packet sending rate, and the RREQ is dropped with a certain probability randomly when the average queue length exceeds the maximum threshold. The Hello packet interval is also determined by the fitting function. The simulation results indicate that without obvious increase of routing discovery frequency, the new mechanism can reduce the average end-to-end delay and Hello packet overhead effectively, and thereby improves the packet delivery ratio.

Keywords Congestion control, Fitting function, Threshold, Ad-hoc networks

1 引言

移动无线自组网(Mobile Ad-hoc Network, MANET)是一种复杂的分布式网络系统,是没有中心实体、自组织、自愈的网络,无线移动节点可以自由而动态地自组织成任意临时性“Ad-hoc”网络拓扑。目前对 MANET 的研究主要集中在传输方式、组网方式和应用层上,其中组网方式当中的路由协议又是研究的热门领域。经典的反应式^[1,2]路由协议(AODV、AOMDV、DSR 和 DYMO 等)中,路由协议主要由两部分组成:路由发现和路由维护。和主动式路由协议相比,被动式路由协议采用了很多措施来减少网络负载,避免网络拥塞等问题,比如扩展环搜索、二次幂调整 RREQ(Route Request)报文重传时间、删除 RREQ 报文队列中的重复报文和使用周期性 Hello 消息来保持链路完整性。

在路由发现阶段,文献[3]提出了一种基于 802.11MAC 接入协议的无限状态拟生灭模型,该模型通过控制发包率来避免网络出现拥塞。文献[4]将随机早期检测(RED)机制加入到拥塞控制当中,根据 RREQ 报文队列的平均长度,以不同的方式响应路由请求,有效降低了平均延时和丢包率。文

献[5]提出了一种基于速率的端到端拥塞控制策略,以端到端流量作为其研究对象,尽量避免链路和网络拥塞。文献[6-10]利用信号强度信息预测链路稳定性和健壮性来减少链路失败的概率,采用信号强度预测链路失败的方式在控制电量消耗和提高分组投递率方面取得了比较明显的效果,但是信号强度量的加入在一定程度上增加了路由开销。文献[11]提出了一种新的稳定的拥塞控制协议,此协议是在 ECN 机制上扩展而来的,扩展的 ECN 机制用两种比特数据分别对应高低两种网络拥塞程度,这样源节点就能够获得更加精确的网络状态信息,并能根据网络状态信息准确地进行反馈响应。

在路由维护方面,文献[12]提出了一种基于小生境粒子群的多拓扑路由子层链路权值优化算法,算法根据网络流量的变化对拥塞程度做出评估,设置一个自适应的链路权值。文献[13]提出了一种 AODV-AH 算法,算法根据邻居节点单位时间内发送或者接收数据包的个数来判断邻居节点的活跃状况,从而决定 Hello 包周期。文献[14]提出了一种自适应的 Hello 包发送周期,讨论了不同拓扑变化程度状态下的 Hello 包发送周期。文献[15]提出了一种根据节点当前的移动速度动态调整 Hello 包发送频率的 TAP 算法,当网络拓扑

到稿日期:2013-10-05 返修日期:2014-01-03 本文受国家自然科学基金(61272497)资助。

王文涛(1967-),男,博士,副教授,主要研究方向为无线传感器网络、嵌入式系统、图像处理,E-mail:wangwt@mail.scuec.edu.cn;王 豪(1990-),男,硕士,主要研究方向为无线传感器网络;朱容波(1978-),男,博士,副教授,主要研究方向为无线网络;郭 峰(1990-),男,硕士,主要研究方向为无线传感器网络;郑 芳(1989-),女,硕士,主要研究方向为无线网络。

变化时,节点只需要周期性地取样路由表来计算当前的拥塞状态并得出相应的合适的 Hello 包发送频率。文献[16]提出了一种 Gilbert 模型,即用 Hello 包丢失率判断链路中断情况, Gilbert 模型从理论上给出了根据路由发现成功可能性设置 Hello 包参数的方法。文献[17]提出了一个增强型 Ad-hoc 按需适量路由协议(E-AODV),E-AODV 合并了扩展环搜索和路由包(比如 Hello 包)来减小网络负载。文献[18]提出一种基于队列长度和跳数的拥塞控制策略,建立了一种协同合作方式的路由机制。

对于通信量较大的情况下出现的 RREQ 报文拥塞以及自适应 Hello 包周期间隔问题,当前的文献研究还存在需改进之处。本文提出了一种新的反应式 MANET 路由协议拥塞控制机制,其主要分为两个部分:1)根据分组投递率和数据包发送速率的拟合曲线得到一个分组投递率和 RREQ 报文平均队列长度的近似拟合函数,根据拟合函数计算出最大门限值和最小门限值并以一定概率丢弃队列中的 RREQ 报文;2)根据平均队列长度动态控制 Hello 包发送周期和生命周期。仿真结果表明,在没有明显增加路由发起频率的情况下,新的机制降低了平均端到端延时和 Hello 包开销,提高了分组投递率和网络吞吐量。

2 反应式路由协议分析

反应式路由协议只在需要的时候发起路由查找过程,不需要周期性广播来维护路径,大大减少了路由开销。

2.1 路由发现过程

在路由初始化阶段,源节点发起对目的节点的 RREQ 报文请求,以初始环进行洪泛广播,中间节点在收到 RREQ 报文时,首先检查自己是否是目的节点,如果是目的节点,就沿前向路由返回一个 RREP 应答给源节点,如果不是目的节点,就继续进行转发,直到达到目的节点。如果在一个定时器内源节点无法收到一个有效的 RREP 回复,就会以初始环加上扩展环作为洪泛跳数继续进行路由查找,直到环数大小达到最终环的大小,源节点认为目的节点不可达。

2.2 路由维护过程

反应式路由协议的维护过程通常采用 Hello 消息包的方式,Hello 消息就是 TTL=1 的 RREP 消息,里面包含目的 IP 地址、目的序列号、跳数和生命周期。节点可以通过监听临近节点的 Hello 消息包来确定链接完整性。如果一个节点在一个有效期内没有收到 Hello 消息包,这个节点可以判定为与其他临近节点的链接中断,同时广播 RERR 错误消息包给临近节点,直到路由表项中包含此节点的源节点收到 RERR 后使这条链路失效。

2.3 拥塞问题

反应式路由协议和主动式路由协议相比,已经采取了很多措施来降低网络开销,避免网络拥塞问题的出现,例如采用扩展环搜索和二次幂发送超时 RREQ 报文,删除 RREQ 报文队列里的重复 RREQ 报文,采用周期性的 Hello 消息包维护链路完整性等。但是,我们在实验中发现:

(1)在通信量比较大的情况下,数据包在传输时就会发生阻塞,有一部分数据包就无法到达目的节点。此时,源节点没有收到目的节点应答,就会重新发起路由查找,这就会造成大量的 RREQ 报文在网络中传播,引起 RREQ 报文拥塞,在这种情况下,就会有更多的数据包无法到达目的节点,导致更多

重复的 RREQ 报文产生,RREQ 报文队列中就有许多重复的 RREQ 报文;

(2)大量的 RREQ 报文拥塞会造成一个假象,即源节点认为目的节点不可达。此时,源节点和中间节点就会认为这条路径不可达,如果出现这种情况,邻居节点再去定期发送 Hello 消息来检测链路的完整性反而没有意义。因此,为了使 Hello 消息随着拥塞程度的变化而变化,需要建立一个模型。

3 新的拥塞控制模型

模型一共分为两部分,分别解决上述两个问题。第一部分进行 RREQ 报文队列长度的控制:

(1)根据分组投递率和数据包发送速率的拟合曲线得到一个分组到达率和数据包发送速率的近似拟合函数,实验中我们统计了数据包发送速率和 RREQ 报文平均队列长度之间的关系,根据拟合函数计算出最大门限值和最小门限值,以此来决定 RREQ 报文平均队列长度以何种方式响应路由请求。

(2)在平均队列长度超过最大门限值时,以一定概率丢弃队列中的 RREQ 报文;

(3)根据平均队列长度动态变化扩展环搜索值。

第二部分进行 Hello 包周期控制:

根据平均队列长度动态控制 Hello 包发送周期和生命周期。

文献[4]将随机早期检测(RED)机制加入到拥塞控制当中,根据 RREQ 报文队列的平均长度,以不同的方式响应路由请求。文中设置了一个最大门限和最小门限来预测拥塞程度,文中最小门限 Min_th 设置为 $0.4 \times Q_lim$,最大门限 Max_th 设置为 $0.8 \times Q_lim$,同时设置一个类似于 RED 算法里面的随机丢失概率值 P_delay ,用来进行一定延时的转发。

当 $Q_avg > Max_th$ 时,RREQ 报文队列中有很多重复的 RREQ 报文,原文的做法是把所有新来的 RREQ 报文都丢弃掉,这种做法是不合适的,而且阈值的设定是固定值,这也是不合理的。重新建立一个 RREQ 报文拥塞控制的模型,当在链路中增加通信量时,我们统计了在不同数据包发送速率下的分组到达率,如图 1、图 2 所示。

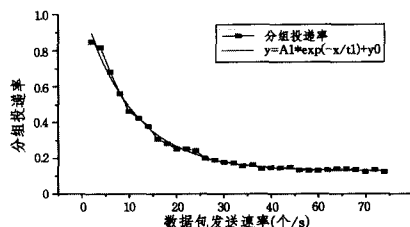


图1 数据包发送速率和分组投递率之间的关系

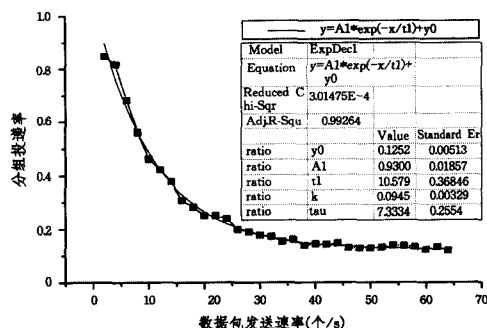


图2 数据包发送速率和分组投递率拟合曲线参数

从图 1、图 2 中可以看到,分组投递率和数据包发送速率存在着某种函数关系,我们假定此函数服从一个指数分布:

$$F(x) = \alpha e^{-x/\beta} + f(\lambda) \quad (1)$$

这里的 $f(\lambda)$ 可称作权值,它应该在一个很小的范围内波动,我们设计一个分布函数。假设一个参数值 λ , 这个分布函数应该满足以下条件:

(1) 在参数值 λ 较大时,函数值 $f(\lambda)$ 应该较小,在参数值 λ 接近无穷大时,函数值 $f(\lambda)$ 应该趋近于 0;

(2) 参数值 λ 由初始化逐渐增大时,函数值 $f(\lambda)$ 逐渐增大,在参数值 λ 到达某个值时,函数值 $f(\lambda)$ 达到最大,而后开始下降;

(3) 参数值 λ 初始化时,函数值 $f(\lambda)$ 不为 0。

因此,选取泊松分布作为 $f(\lambda)$ 值的计算公式:

$$f(\lambda) = \frac{\lambda^k}{k!} e^{-\lambda} \quad (0 \leq k \leq Q_{lim}, 0 \leq \lambda \leq Q_{lim}) \quad (2)$$

式中, λ 是 Q_{avg} , $f(\lambda)$ 是 Q_{now} , 通过泊松分布能够得到权值 $f(\lambda)$, $f(\lambda)$ 的取值范围是 0 到 1。通过拟合函数,我们可以得到一个近似于图 1 的非线性相关指数拟合函数。

由于数据包发送速率和 RREQ 队列平均长度是正线性相关的,因此,数据包发送速率反映了 RREQ 队列中的平均队列长度。我们统计了不同数据包发送速率情况下的平均队列长度 Q_{avg} , 如表 1 所列。根据不同的平均队列长度值和公式可以计算出数据包的分组投递率,对式(1)作泰勒公式展开可得:

$$F(x) = \alpha \left(\sum_{n=0}^{\infty} \frac{1}{n!} (-\beta)^n x^n \right) + f(\lambda) \\ = \alpha \left(-\frac{1}{\beta} x + \frac{1}{2! \beta^2} x^2 - \frac{1}{3! \beta^3} x^3 + \dots + \frac{\beta^n}{n!} x^n \right) + f(\lambda) \quad (3)$$

表 1 数据包发送速率和平均队列长度的关系

发送速率	平均队列长度
2	1
12	13
22	29
32	38
42	54
52	56
62	58

这里的平均队列长度在实际运行当中相当于一个典型的“增与消”的过程,因此怎样控制这个“增与消”的过程也就是本文要讨论的拥塞控制。把这个过程简化为一个排队模型,我们用 3 个参数来描述整个拥塞控制模型,即输入过程、处理过程和排队规则。

输入过程是用来描述请求数据包到来的序列。我们假设输入过程为随机输入,每个源节点在生成请求时,都具有相同的指数分布请求间隔时间。

处理过程主要是指转发和删除 RREQ 报文的操作。

排队规则指定拥塞报文的操作:1) 当拥塞的报文队列过长时,开始丢弃队列中的报文,这种规则称为“清除所有阻塞”(BCC);2) 不进行删除操作,报文一直处于等待状态,在这种情况下,一般会指定队列中的报文服务的次序,比如随机服务或者 FCFS 等。

令 $\{Q(t), t \geq 0\}$ 为 t 时刻的队列长度,则队列处于稳态的概率可由式(4)给出:

$$P_j = \begin{cases} \frac{1}{1 + \sum_{k=1}^{\infty} \left(\frac{\lambda_0 \lambda_1 \dots \lambda_{k-1}}{\mu_1 \mu_2 \dots \mu_k} \right)}, & j=0 \\ \frac{\lambda_0 \lambda_1 \dots \lambda_{j-1}}{\mu_1 \mu_2 \dots \mu_j} P_0, & j=1, 2, \dots \end{cases} \quad (4)$$

这个方程等效于

$$\lambda_j P_j = \mu_{j+1} P_{j+1} \quad j=0, 1, 2, \dots \quad (5)$$

其中

$$\sum_{j=0}^{\infty} P_j = 1 \quad (6)$$

$\lambda_j (j=0, 1, 2, \dots)$ 和 $\mu_j (j=0, 1, 2, \dots)$ 分别是队列长度增加的速率和减少的速率。在 $(t, t + \Delta t)$ 时间内从一个队列长度转移到另一个队列长度的概率为 $\lambda_j = \lambda (j=0, 1, 2, \dots)$; 引起这种转移发生的概率为 $\frac{\lambda}{\lambda + j\mu}$, 因此,在 $(t, t + \Delta t)$ 时间内发生 $E_j \rightarrow E_{j+1}$ 转移的概率为:

$$\left[(\lambda + j\mu) \Delta t + o(\Delta t) \right] \frac{\lambda}{\lambda + j\mu} + o(\Delta t) = \lambda \Delta t + o(\Delta t) \quad (\Delta t \rightarrow 0) \quad (7)$$

式中的 $\frac{\lambda}{\lambda + j\mu}$ 为处理完成时可能影响转移的概率,类似地,

$$P\{N(t + \Delta t) = j - 1 | N(t) = j\} = j\mu \Delta t + o(\Delta t) \quad (\Delta t \rightarrow 0) \quad (8)$$

在 $(t, t + \Delta t)$ 内无转移的概率为 $e^{-(\lambda + j\mu)\Delta t}$, 发生 1 个以上转移的概率为 $o(\Delta t) (\Delta t \rightarrow 0)$, 可得:

$$P\{N(t + \Delta t) = j | N(t) = j\} = e^{-(\lambda + j\mu)\Delta t} + o(\Delta t) \\ = 1 - (\lambda + j\mu) \Delta t + o(\Delta t) \quad (\Delta t \rightarrow 0) \quad (9)$$

在 $(t, t + \Delta t)$ 内发生多于 1 次转移的概率为 $o(\Delta t) (\Delta t \rightarrow 0)$, 可得:

$$P\{N(t + \Delta t) = k | N(t) = j\} = o(\Delta t) \quad (\Delta t \rightarrow 0, |k - j| \geq 2) \quad (10)$$

在 $(t, t + \Delta t)$ 时刻存在的队列长度应该遵循具有增速率 $\lambda_j = \lambda (j=0, 1, 2, \dots)$ 以及消速率 $\mu_j = j\mu (j=0, 1, 2, \dots)$ 的“增与消”过程的规则,于是可得:

$$P_j = \frac{(\lambda/\mu)^j}{j!} e^{-\lambda/\mu} \quad j=0, 1, 2, \dots \quad (11)$$

从平衡概率式(12)可以计算出瞬时的平均队列长度 a :

$$a = \sum_{j=1}^{\infty} j P_j + s \sum_{j=s}^{\infty} P_j \quad (12)$$

s 表示目前队列中有 s 个数据包。由于请求报文随机到达,只有当 $j < s$ (阻塞数据包被清除) 时影响状态改变,因此可得:

$$\lambda_j = \begin{cases} \lambda, & j=0, 1, \dots, s-1 \\ 0, & j=s \end{cases} \quad (13)$$

由于分组投递率和平均队列长度服从指数分布,当有 j 个数据包按照速率 μ_j 处理时,

$$\mu_j = j\mu \quad j=1, 2, \dots, s \quad (14)$$

由式(13)、式(14)结合式(3),可以得到平均队列长度超出阈值的概率:

$$P_j = \frac{(\lambda/\mu)^j}{\sum_{k=0}^{\infty} \frac{(\lambda/\mu)^k}{k!}} \quad j=0, 1, \dots, s \quad (15)$$

式(15)中出现的速率 λ 与 μ 通过 λ/μ 之比体现,这个公式意味着,如果我们想要达到某个分组投递率 $[\min, \max]$, 只需根据式(3)计算出 x 的值,在统计表中找出对应的 Q_{avg} ,

将此 Q_{avg} 和 Q_{lim} 的比值作为阈值,假设最小阈值和最大阈值分别是 Q_{lim} 和 Q_{max} 。在协议运行过程中,RREQ 队列长度也会逐渐增加,因此为了控制 RREQ 的队列长度,达到理想的分组投递率,我们将文献[4]中的算法加以改进,在 $Q_{avg} > Q_{max}$ 时,加入一个随即丢弃概率 P_{drop} 。

$$P_{drop} = P_i (\tau_1 + \tau_2) \quad (16)$$

其中

$$\begin{cases} \tau_1 = Q_{avg}/Q_{lim} \\ \tau_2 = Q_{now}/Q_{lim} \end{cases} \quad (17)$$

即当队列长度超出最大阈值时,以 P_{drop} 概率丢弃队列当中的包来控制 RREQ 队列长度,从而达到合适的队列长度。新的算法如下:

算法 1 基于随机丢弃概率的自适应队列长度算法

- (1)初始化,设置参数;
- (2)对 RREQ 预处理,选择丢弃或者建立反向路由;
- (3)设置 Q_{lim} ,读取 Q_{now} 并计算 Q_{avg} ;
- (4)if 自身是目的节点

更新路由信息并返回应答消息 RREP;

else if $Q_{avg} \leq Q_{max}$

if 自身有到目的节点活跃路由

返回应答消息 RREP;

更新路由消息并选择是否通知目的节点;

else if $Q_{avg} \leq Q_{min}$

更新广播消息并转发 RREQ 报文;

else

以延时 P_{delay} 转发 RREQ 报文;

else

以概率 P_{drop} 随机丢弃数据包。

实验中我们发现,在通信量比较大的情况下,即使采用上述随机丢弃的办法依然不能满足 RREQ 队列长度的有效控制,RREQ 队列长度仍会长时间保持比较大的状态,为此,我们从扩展环搜索部分入手。传统的扩展环搜索有 3 个参数:初始环 TTL_START、扩展环 TTL_INCREMENT、最终环 TTL_THRESHOLD。在传统的扩展环搜索中,这 3 个值都是定值,也就是在发送 RREQ 报文时,先以初始环值进行路由发现;如果没有发现可用路由,每次加上一个扩展环值进行搜索;如果在达到最终环值的情况下依然没有发现可用路由,源节点就会放弃到目的节点的路由发现过程,认为目的节点不可达。这个方法本来旨在控制 RREQ 在网络中的传播范围,从而减少网络拥塞和开销,但由于扩展环值是固定值,该方法并不能很好地适应动态变化的网络拥塞状况。因此,我们根据图 1 将扩展环设置为一个可变值,即

$$TTL_THRESHOLD = TTL_THRESHOLD * \lfloor ae^{-x/\beta} + f(\lambda) \rfloor \quad (18)$$

这里的 x 是指平均队列长度,当 RREQ 队列长度较大时,我们就减小扩展环搜索范围,以减少 RREQ 报文在网络中的扩散。

第二个问题是关于 Hello 包周期的计算。文献[13]中,Hello 包的周期是根据邻居节点的活跃情况来计算的,根据邻居节点单位时间内发送或者接收数据包的个数来判断邻居节点的活跃状况,从而决定 Hello 包周期。本文中,我们利用平均队列长度来计算 Hello 包的周期,当平均队列长度比较大时,RREQ 报文较长,网络拥塞状况比较严重,此时应该减

少 Hello 包周期间隔,以更好地维护链路的完整性;当平均队列长度比较小时,网络状况较好,此时应该减小 Hello 包周期间隔来减少不必要的链路维护。

Hello 周期间隔的计算同样采用式(3):

$$HELLO_INTERVAL = HELLO_INTERVAL * \lfloor ae^{-x/\beta} + f(\lambda) \rfloor \quad (19)$$

4 仿真实现与性能评价

采用 NS2 网络仿真软件进行仿真来评估网络性能,软件的本是 NS2.35。在协议的选择方面,选择 AODV 和 AOMDV 两个具有代表性的反应式路由协议。当然上述机制也可以用于其他采用洪泛请求方式发起路由,以 Hello 包维护链路完整性的路由协议。

(1)节点运动模型

节点数目为 50 个,拓扑范围为 1000m×300m 的矩形,节点移动暂停时间分别取 0s、50s、100s、200s、300s,无线传播模型采用 TwoRayGround 模型,仿真时间是 300s。

(2)通信流量模型

通信流量模型选择稳定比特流发送模型 CBR,最大连接数为 20,源节点、目的节点初始位置和节点初始运动时间随机产生,发送速率为 25packet/s,分组长度为 512Byte。

(3)性能评价

选取以下 3 个参数来评价网络性能:分组投递率、平均端到端延时和路由发起频率。

在上述通信流量模型下,通过实验可以得出 AODV 和 AOMDV 的分组投递率在 20%左右,因此我们假定在上述通信流量模型情况下的分组投递率取值范围为 15%~40%,由式(1)得到数据包发送速率为 12~38,通过表 1 可得对应的平均队列长度为 13~48,它们分别是阈值 Q_{min} 和 Q_{max} ,在不同的暂停时间下每组做 10 次重复试验取平均值,将结果同 AODV、AOMDV 和文献[4]中的结果做比较,得到图 3—图 5(AODV 是原始的按需距离矢量路由协议,AOMDV 是原始的按需多路径距离矢量路由协议,AODV-4 和 AOMDV-4 是根据文献[4]的方法改进的协议,AODV-DR 和 AOMDV-DR 是根据本文提出的方法改进的协议)。

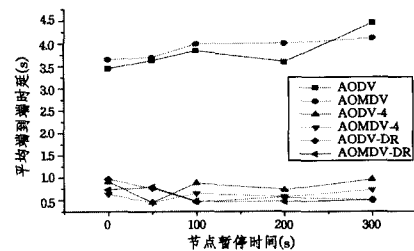


图3 平均端到端延时

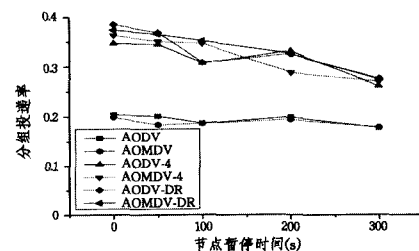


图4 分组投递率

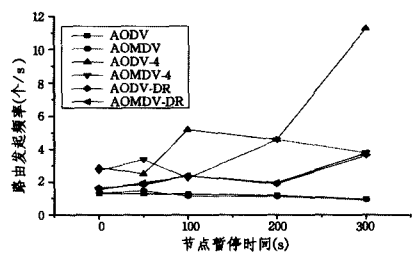


图5 路由发起频率

在 Hello 包的间隔控制方面,我们参考文献[13]中的参数设置,如表 2 所列。

表 2 仿真参数

Parameter	Value	Parameter	Value
Propagation model	Rayleigh fading	Mobility speed	Max 5m/s, 0sec pause time
Transmission power	16dBm	Mobility model	Random way point
Topology size	670×670m	Flows	ftp, Pareto
Data rate	11Mbps	Number of senders	10
Number of nodes	20	PHY/MAC	802.11(no RTS)

选取 Hello 包比率来和文献[13]比较,取 100 次试验的平均值,比较结果如图 6 所示(AODV 是原始的按需距离矢量路由协议, AODV-AH 是根据文献[13]中方法改进的协议, AODV-DR 是根据本文提出的方法进行改进的协议)。

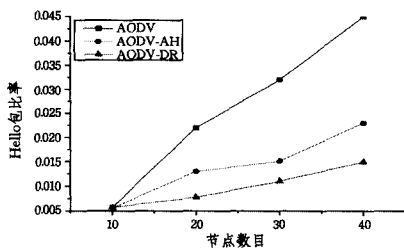


图 6 Hello 包开销

图 3 表明在平均端到端延时方面,文献[4]和我们提出的机制都大幅度减少了平均端到端延时,在暂停时间大于等于 100s 时,我们的机制拥有更小的平均端到端延时。这是因为在暂停时间较小时节点高速移动,网络拓扑变化比较剧烈,断裂的链路增多,源节点就要更频繁地发送 RREQ 请求报文来寻找目的节点,此时的 RREQ 报文队列严重阻塞。文献[4]中采取的办法是当报文队列大于阈值时,全部丢弃 RREQ 包。我们的机制是以一定概率删除队列中的 RREQ 包;因此我们的机制在节点移动速度很快时会增加少量的延时。

图 4 表明在提高分组投递率方面,我们的机制是十分有效的:根据拟合函数曲线预算出分组投递率,并把相应的平均队列长度设为阈值,实验结果得到的分组投递率在我们预测的范围之内,而且要高于文献[4]中的分组投递率。这为我们根据自己的分组投递率容忍值来预算出网络参数提供了一个方法,从而使得整个网络过程是可控的。

图 5 是路由发起频率,由于我们采用了阈值转发这种策略,路由发起频率会有所上升。文献[4]中的发起频率随着暂停时间的增加而急剧增加,而我们的机制使路由发起频率基本平稳,略大于原始协议。

图 6 是和文献[13]比较在不同节点数目下的 Hello 包负载,文献[13]中的 Hello 包负载大约是原始协议的 50%,我们

的机制当中 Hello 包负载大约是原始协议的 30%,有效降低了 Hello 包负载。

上述仿真结果及分析表明我们的机制在稍微增大路由发起频率的情况下,大幅度提高了分组投递率和延时性能,在平均端到端延时、分组投递率和路由发起频率 3 个性能上比文献[4]有所提高,在 Hello 包开销上比文献[13]有较大改善。因此对于反应式路由协议来说,我们的机制具有较好的性能。

结束语 本文针对反应式 MANET 路由协议在较大通信量的情况下出现的 RREQ 报文拥塞问题,首先拟合出分组投递率和数据包发送速率之间的函数曲线,然后统计出数据包发送速率和 RREQ 报文平均队列长度之间的关系,根据 RREQ 报文平均队列长度和拟合的函数曲线设定 RREQ 报文响应阈值、随机丢弃的概率值和 Hello 包发送周期。将改进后的机制在 AODV 和 AOMDV 协议上进行验证,仿真结果表明在分组投递率、平均端到端延时以及 Hello 包的开销方面,网络性能都有一定的改善,而且网络的分组投递率是可控的,这为我们通过 MANET 网络某一个指标的容忍值进行网络参数的理论计算提供了参考。

参考文献

- [1] Cai Xue-lian, Li Jian-dong, Zhu Li-na. Performance of Routing Protocols in Very Large Scale Wireless Sensor Networks[J]. Journal of Networks, 2012, 7(12): 1996-2002
- [2] Yun J, Comput D E S, et al. An Implementation of AODV Routing Protocol with Multi-Metrics[C]// International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery. Huangshan: IEEE, 2010: 429-432
- [3] 夏文洁, 李千目, 刘凤玉, 等. 基于拟生灭过程的多跳 Ad hoc 网络洪泛方式下拥塞控制及饱和条件研究[J]. 计算机科学, 2012, 39(4): 110-113
- [4] 张登银, 张保峰. 基于 RED 的 AODV 路由协议改进[J]. 南京邮电大学学报: 自然科学版, 2009, 29(1): 38-42
- [5] Zhang Li, Zou Jin. A Wireless Ad-hoc Network Congestion Control Algorithm Based on Game Theory[C]// International Conference on Future Computer Sciences and Application. Hongkong: IEEE, 2011: 137-141
- [6] Klemm F, Krishnamurthy V, Tripathi K. Improving TCP performance in ad hoc networks using signal strength based link management[J]. Ad hoc Networks, 2005, 2(3): 175-191
- [7] Chen Ching-wen, Weng Chuan-chi, Kuo Yu-chen. Signal strength based routing for power saving in mobile ad hoc networks [J]. Journal of Systems and Software, 2010, 8(83): 1373-1386
- [8] Biradar C, Manvi S. Neighbor supported reliable multipath multicast routing in MANETs[J]. Journal of Network and Computer Applications, 2012, 35(3): 1074-1085
- [9] Park G, Song C J. The residual battery capacity and signal strength based on power-aware routing protocol in mobile ad-hoc network[C]// International Conference on Computer Engineering and Applications. New York: ACM, 2007: 319-324
- [10] Sunita M, Usturge N. Study of congestion control using AODV and signal strength by avoiding link failure in MANET[C]// International Conference on Communication Information & Computing Technology. Mumbai: IEEE, 2012: 1-5
- [11] Liu Wei-rong, Wu Min, Peng Jun, et al. The improved ECN con-

gestion control in Ad Hoc network and simulation test[C]//International Conference on Electronic Commerce and Business Intelligence. Beijing; IEEE, 2009; 143-146

- [12] 陈铎龙, 孟相如, 梁霄, 等. 基于动态流量的多拓扑链路权值优化算法[J]. 计算机科学, 2013, 40(4): 86-90
- [13] Han S, Lee D. An adaptive Hello messaging scheme for neighbor discovery in on-demand MANET routing protocols[J]. IEEE Communications Letters, 2013, 17(5): 1040-1043
- [14] Gomez C, Cuevas A, Paradells J. AHR: a two-state adaptive mechanism for link connectivity maintenance in AODV[C]//International Symposium on Mobile Ad Hoc Networking & Computing. New York; ACM, 2006; 98-100
- [15] Ingelrest F, Mitton N, Simplot D. A turnover based adaptive

HELLO protocol for mobile Ad Hoc and sensor networks[C]//International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunications Systems. Istanbul; IEEE, 2007; 9-14

- [16] Razafindralambo P, Mitton N. Analysis of the impact of Hello protocol parameters over a wireless network self-organization[C]//International Workshop on Modeling Analysis and Simulation of Wireless and Mobile Systems. New York; IEEE, 2007; 46-53
- [17] Suman S, Balkrishan. Enhanced AODV for wireless networks [C]//Advance Computing Conference. Patiala; IEEE, 2010; 246-249
- [18] 周建存, 郝夏斐. LTE 中基于协作 AODV 协议的拥塞控制机制研究[J]. 计算机科学, 2012, 39(8): 122-125

(上接第 147 页)

值进行路由替换,一定程度上保护了节点能量,避免了关键节点能耗过大的现象,延长了网络生存期。

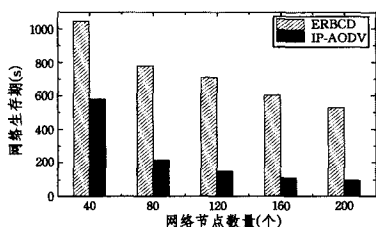


图 4 网络生存时间

(5) 平均分组能耗

图 5 为平均分组能耗,在每个场景中,ERBCD 算法平均分组能耗均低于 IP-AODV 算法,具有更高的能量效率。主要原因是:在 ERBCD 算法中,仅在网络初始时执行一次梯度扩散和梯度反馈任务,以构建“节点→网关”的上行路由和“网关→节点”的下行路由,节点充分利用先验式路径传输数据分组,无需执行建路操作,因而网络节点能耗主要集中于数据分组的传输;而 IP-AODV 算法中,节点能量除用于传输数据分组外,还主要消耗在路由建立阶段。随着网络节点数量增加,网络业务量增大,节点参与转发分组任务的比重加大,因此网络平均分组能耗增加,但 ERBCD 算法中平均分组能耗增长速度比 IP-AODV 要小很多。

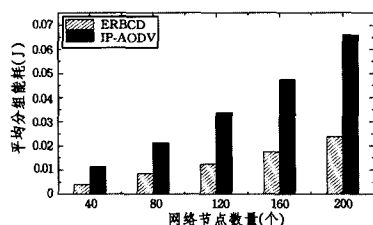


图 5 平均分组能耗

结束语 关于 Zigbee 网络多 PAN 组网策略,已经有了一些研究,但现有适用于多 PAN 网络的 IP-AODV 路由算法忽略了能量因素,缺少应对网络节点业务负载不均衡所导致的关键节点能量过早耗尽的机制。本文提出的 ERBCD 算法,采用梯度定向扩散过程中节点维护冗余路由策略,并使用梯度反馈机制构建网关至 PAN 内节点的下行路由,同时采用跨层协作感知方式,利用 MAC 层的 ACK 能量捎带方式实

时获取邻居节点能量信息,降低并均衡网络能耗,延长了网络生存期。

参考文献

- [1] ZigBee Document 053474r06-2004, ZigBee Specification Version 1.0[EB/OL]. <http://www.ZigBee.org>, 2004
- [2] Wu C-H, Liu H-S, Lee Y-F, et al. A gateway-based inter-PAN binding mechanism for ZigBee sensor networks[C]//The 37th Annual Conference on IEEE Industrial Electronics Society (IECON 2011). 2011; 3808-3813
- [3] 成小良, 邓志东. 基于 ZigBee 技术规范构建大规模无线传感器网络[J]. 通信学报, 2008, 29(11): 158-164
- [4] Lee C H. Inter ZigBee Personal Area Networks (PAN) Communication Mechanism; KR, 10-0587013-0000[P]. 2006
- [5] Heo J, Hong C S. The Optimum Path Selection Mechanism for Inter-PAN Communication Using ZigBee[C]//The 2008 IEEE International Conference on Information Networking (ICOIN 2008). Busan, Korea, 2008; 1-5
- [6] 钱志鸿, 张晓帆, 王义君, 等. 基于节点特性的 LR-WPAN 网络能量优化路由算法[J]. 通信学报, 2010, 31(10): 238-243
- [7] Ren Zhi, Fan Hai-bin, Cao Jian-ling. A Hybrid Routing Protocol for Inter-pan Communication in ZigBee Networks[C]//IEEE International Conference on Oxide Materials for Electronic Engineering (OMEE), 2012. IEEE, 2012; 363-366
- [8] 卿利, 朱清新, 王明文. 异构传感器网络的分布式能量有效成簇算法[J]. 软件学报, 2006, 17(3): 481-489
- [9] Heinzelman W R, Chandrakasan A P, Balakrishnan H. An application-specific protocol architecture for wireless microsensor networks [J]. IEEE Transactions Wireless Communications, 2002, 1(4): 660-670
- [10] 赵彬, 郭田德, 杨文国. 无线传感器网络能耗均衡路由模型及算法[J]. 软件学报, 2009, 20(11): 3023-3033
- [11] Ran P, Sun M-H, Zou Y-M. ZigBee Routing Selection Strategy Based on Data Services and Energy-balanced ZigBee Routing[C]//The 2006 IEEE Asia-Pacific Conference on Services Computing (APSCC'06). 2006; 400-404
- [12] IEEE 802. 11IEEE standard for information technology—telecommunications and information exchange between systems—local and metropolitan area networks specific requirements part 15. 4[S]. 2003