

## 基于无线网络编码的速率自适应协作 MAC 协议

姚玉坤,王 宇,朱丽青,吕盼成

(重庆邮电大学 移动通信技术重庆市重点实验室,重庆 400065)

**摘 要:**传统无线网络编码协议在单播传输模式中较少考虑速率自适应对网络性能的改善作用,而利用 RTS/CTS 握手信号进行信道估计的速率自适应机制则存在开销大、网络吞吐性能差等缺陷。为此,提出一种基于网络编码的速率自适应协作介质访问控制协议 RACNC。该协议使用网络编码模型和基于 ACK 帧的速率自适应机制,当源节点传输的原始数据帧未被成功接收时,选取中继节点对原始数据帧进行编码处理,然后转发编码数据帧,在保证数据帧可靠传输的同时减少其在网络上的传输次数。同时利用改进的 ACK 帧进行速率反馈,避免在估计信道质量时发送多余的 RTS/CTS 控制帧。仿真结果表明,与基于网络编码的 NCAC-WTC 协议相比,RACNC 协议可有效减小平均端到端时延,提高网络吞吐量。

**关键词:**无线网络;网络编码;协作通信;介质访问控制协议;速率自适应;网络吞吐量

**中文引用格式:**姚玉坤,王 宇,朱丽青,等. 基于无线网络编码的速率自适应协作 MAC 协议[J]. 计算机工程, 2017,43(9):87-93.

**英文引用格式:**YAO Yukun,WANG Yu,ZHU Liqing,et al. Rate-adaptive Cooperative MAC Protocol Based on Wireless Network Coding[J]. Computer Engineering,2017,43(9):87-93.

### Rate-adaptive Cooperative MAC Protocol Based on Wireless Network Coding

YAO Yukun,WANG Yu,ZHU Liqing,LÜ Pancheng

(Chongqing Key Lab of Mobile Communication Technology,

Chongqing University of Posts and Telecommunications,Chongqing 400065,China)

**[Abstract]** Traditional wireless network coding protocols seldom consider the improvement of rate adaption on network performance in unicast transmission mode, and using RTS/CTS handshaking signal to estimate channel will generate additional overhead and decrease the throughput capacity. Aiming at these problems, this paper proposes a Rate-adaptive Cooperative Medium Access Control (MAC) Protocol Based on Network Coding (RACNC). It uses network coding model and rate-adaptive mechanism based on ACK frame. When the original data frame transmitted by the source node is not received successfully, the selected relay node will encode the original data frame and forward the encoded data frame, so as to ensure reliable transmission of data frames and reduce the number of data frames transmitted over the network. Meanwhile, it uses the modified ACK frame for rate feedback instead of sending extra RTS/CTS control frames for channel quality estimation. Simulation results show that, compared with NCAC-WTC protocol based on network coding, RACNC protocol can effectively reduce the average end-to-end time delay and improve the network throughput.

**[Key words]** wireless network; network coding; cooperative communication; Medium Access Control (MAC) protocol; rate-adaptive; network throughput

**DOI:**10.3969/j.issn.1000-3428.2017.09.016

## 0 概述

网络编码<sup>[1]</sup>作为一种提高网络吞吐量的有效方式,允许中间节点存储和转发数据帧的同时对数据帧进行编码处理,增加单次传输数据帧的信息量,减少数据帧在网络上的传输次数。

速率自适应<sup>[2]</sup>是指在通信系统中,节点根据当前信道的状态信息,选择信道支持的最佳传输速率发送数据帧,可以提高网络系统的传输性能并降低传输能耗<sup>[3]</sup>。

NCCMAC<sup>[4]</sup>,Phoenix<sup>[5]</sup>,NCAC-MAC<sup>[6]</sup>等网络编码方案通过选取中继节点代替源节点对直传失败

**基金项目:**重庆市科委自然科学基金(cstc2012jjA40040);长江学者和创新团队发展计划项目(IRT1299)。

**作者简介:**姚玉坤(1964—),女,教授,主研方向为网络编码;王 宇、朱丽青、吕盼成,硕士研究生。

**收稿日期:**2016-10-19 **修回日期:**2016-12-02 **E-mail:**yaoyk@cqupt.edu.cn

(目的节点接收到被破坏的数据帧)的原始数据帧进行编码重传,从而有效减少数据帧的传输次数,但这些方案使用固定的传输速率发送数据帧,没有考虑速率自适应对网络性能的改善。

将网络编码技术和速率自适应技术相结合,在减少数据帧传输次数的同时,根据信道状态选择信道支持的最大传输速率发送数据帧,可进一步提高网络吞吐量。

ARF<sup>[7]</sup>是一个商用的实现多速率的介质访问控制(Medium Access Control,MAC)协议,通过对过去一段时间传输数据的统计估计出当前信道的状态,然后选择相应的速率发送数据帧,但是不能准确地估计出当前信道的状态信息,对于链路剧烈变化的情况不能及时做出反应。

RBAR<sup>[8]</sup>与 M-RBAR<sup>[9]</sup>是基于接收端信息反馈的速率自适应协议,利用 RTS/CTS 握手信号实现信道状态信息交换,在发送数据之前利用 RTS 帧对信道进行实时探测并用 CTS 帧反馈信道支持的最大传输速率。文献[10]提出了一种混合速率自适应算法 MRA,有效解决了当前基于 IEEE802.11 的速率自适应算法面临的在快速变化链路中的切换滞后问题及重复速率抖动问题。文献[11]针对 WLAN Mesh 网络提出了多信道速率自适应 MAC 协议,将不同传输速率的链路分配到不同的信道上,能有效提高网络的总吞吐量。但上述文献并没有考虑网络编码对网络性能的提高。

RAMPNC<sup>[12]</sup>是一种基于网络编码的速率自适应多播 MAC 协议。该协议采用网络编码组传输模型使源节点对多播数据帧首先进行网络编码处理再发送多播编码帧,在保证数据帧可靠传输的同时减少数据帧的传输次数;同时使用速率自适应技术,利用 RTS/CTS 握手信号实现信道信息的交换,使节点以信道支持的最大传输速率发送数据帧。但该协议在使用 RTS/CTS 握手信号进行信道估计时存在网络开销较大的问题。

本文在无线单跳网络传输模型中以 NCAC-WTC 协议<sup>[13]</sup>为基础,结合网络编码技术和速率自适应技术的优势,针对网络中的单播传输模式提出一种基于网络编码的速率自适应协作 MAC 协议(Rate-adaptive Cooperative MAC Protocol Based on Network Coding, RACNC)。

## 1 网络编码模型

考虑到无线信道的广播特性,当源节点直传原始数据帧失败时,可以利用中继节点先对原始数据帧进行网络编码,再传输编码数据帧,增加单次传输

数据帧的信息量。

RACNC 协议使用随机线性网络编码模型,对多个数据帧进行网络编码操作。为确保编码数据帧在目的节点有较高的解码概率,RACNC 协议仅对 2 个数据帧进行编码,即中继节点从缓存中选取一个在目的节点能被解码的概率最高的数据帧与监听到的源节点发送的原始数据帧进行随机线性网络编码。中继节点进行随机线性网络编码的编码系数从伽罗华域  $F(q)$  内随机选择。

随机线性网络编码的编码方式如下所述:

中继节点将源节点发送的原始数据帧  $P_1$  与从自身选取的参与编码的数据帧  $P_2$  进行随机线性网络编码组合,生成编码数据帧  $Y_1$  与  $Y_2$ ,如式(1)和式(2)所示,然后发送编码数据帧。

$$Y_1 = g_{11}P_1 + g_{12}P_2 \quad (1)$$

$$Y_2 = g_{21}P_1 + g_{22}P_2 \quad (2)$$

其中, $g_{11}, g_{12}$ 与  $g_{21}, g_{22}$ 为从伽罗华域  $F(q)$  中随机选取的编码系数。

考虑到无线网络存在信道衰落和高斯白噪声的干扰,目的节点收到的数据帧可能被破坏,若在目的节点采用线性网络编码的解码方式不能对编码帧进行成功解码。因此,RACNC 协议使用文献[11]所提出的自适应解码机制,使目的节点根据链路信息自适应地选择以下 2 种解码方法对编码数据帧进行解码,以提高对编码数据帧的解码效率和成功率。

1) 随机线性网络编码的解码方法。目的节点接收到编码数据帧后缓存到存储中,当接收到至少 2 个(参与编码的数据帧的个数)编码系数线性无关的编码数据帧时,利用高斯消元法从编码数据帧中解码出原始数据帧,如式(3)所示。

$$\begin{bmatrix} P_1 \\ P_2 \end{bmatrix} = \begin{bmatrix} g_{11} & g_{12} \\ g_{21} & g_{22} \end{bmatrix}^{-1} \begin{bmatrix} Y_1 \\ Y_2 \end{bmatrix} \quad (3)$$

其中, $Y_k (k=1,2)$ 表示目的节点接收到的编码数据帧; $P_k (k=1,2)$ 表示从编码数据帧中解码出的原始数据帧; $g_{kt} (k=1,2, t=1,2, i \geq j)$ 表示目的节点从编码数据帧中提取出的编码系数。

2) MIMO\_NC 解码方法<sup>[14]</sup>。此方法的主要优势是即使目的节点接收到被破坏的原始数据帧,只要接收的原始数据帧信噪比大于阈值  $A_{th}$ , MIMO\_NC 就可以通过最大似然估计方法从编码数据帧中解码出原始数据帧,但此解码方法的计算复杂度大于随机线性网络编码解码方法的计算复杂度。

## 2 速率自适应机制

现有的速率自适应机制主要分为 2 类:

1) 通过统计过去一段时间内数据帧的接收情况来决定下一个数据帧的发送速率。此类速率自适应机制根据过去一段时间的信道信息估计当前的信道质量,不能对信道质量的快速变化及时作出反应。

2) 在发送数据帧前,先利用 RTS/CTS 握手信号对信道质量进行估计,然后决定数据帧的发送速率。此类速率自适应机制在发送数据帧之前利用 RTS/CTS 握手信号对信道质量进行估计,解决了前一类速率自适应机制不能准确估计当前信道质量的缺陷。此类机制通过在发送数据帧之前发送额外的控制帧对信道质量进行估计,对于有隐藏终端的网络拓扑,可以借助 RTS/CTS 控制帧进行信道估计,但对于无隐藏终端的网络拓扑,则会增加一定的网络开销。

针对上述两类速率自适应机制存在的缺陷,本文在 RACNC 协议中引入一种利用 ACK 帧进行速率反馈的速率自适应机制 (Rate-adaptive Mechanism with ACK Frame, RACK)。在该机制中,目的节点根据接收的上一个数据帧的信噪比 (Signal-to-Noise Ratio, SNR) 选取一个最佳传输速率,然后通过 ACK 帧反馈给源节点,告知源节点发送下一个数据帧应使用的传输速率。RACK 速率自适应机制既能及时对信道质量的快速变化做出反应,又相对上文第 2) 类速率自适应机制无需发送额外的 RTS/CTS 控制帧来确定发送速率,减小了网络开销。

RACK 机制通过在物理层使用不同的调制方式得到多速率,分别使用 BPSK, QPSK, 16-QAM, 64-QAM 和 256-QAM 调制得到 6 Mb/s, 12 Mb/s, 24 Mb/s, 36 Mb/s 和 48 Mb/s 的传输速率。

网络初始化时, RACK 机制为每个节点维护一张传输速率表,存储该节点与其邻居节点之间链路的距离信息和链路支持的传输速率。假设链路之间的距离信息可以通过 GPS 或相应的算法获得,然后根据获得的链路距离信息与文献 [15] 提供的链路距离信息与传输速率的对应关系初始化该传输速率表。节点维护的传输速率如表 1 所示,假设该节点为节点 A,且节点 A 的邻居节点有节点 B、节点 C、节点 D 和节点 E,则链路距离与传输速率的对应关系如表 2 所示。

表 1 节点 A 维护的传输速率

链路名称	链路距离	传输速率
AB	$d_{AB}$	$r_{AB}$
AC	$d_{AC}$	$r_{AC}$
AD	$d_{AD}$	$r_{AD}$
AE	$d_{AE}$	$r_{AE}$

表 2 链路距离与传输速率的对应关系

链路距离/m	传输速率/(Mb · s <sup>-1</sup> )
15	48
30	36
60	24
130	12
200	6

目的节点接收到源节点发送的数据帧后,根据接收到的数据帧的信噪比选取最佳的传输速率并通过 ACK 帧反馈给源节点。源节点接收到目的节点回复的 ACK 帧后,根据 ACK 帧反馈的传输速率值更新其维护的传输速率表。其中,信噪比 SNR 与传输速率的对应关系<sup>[7]</sup>如表 3 所示。

表 3 信噪比与传输速率的对应关系

信噪比/dB	传输速率/(Mb · s <sup>-1</sup> )
$SNR \leq 12.6$	6
$12.6 < SNR \leq 19.8$	12
$19.8 < SNR \leq 25.9$	24
$25.9 < SNR \leq 31.9$	36
$SNR > 31.9$	48

因为 RACK 机制使网络中的每个节点都维护一张传输速率表且传输速率表是根据信道状态实时更新的,所以节点在发送数据帧之前只需要查询传输速率表获得相应的传输速率,然后使用该传输速率发送数据帧。

## 3 RACNC 协议描述

为便于清楚地描述 RACNC 协议,首先给出简化的节点在网络中传输数据帧的示意图,如图 1 所示。

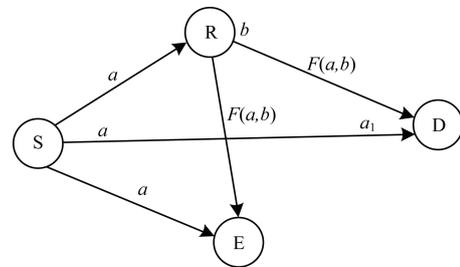


图 1 节点传输数据帧示意图

在图 1 中,节点 S 为源节点;节点 D 为目的节点;节点 R 同时在源节点 S 和目的节点 D 的通信范围内,即节点 R 均可接收到源节点 S 和目的节点 D 发送的数据,可以充当源节点 S 和目的节点 D 之间的中继节点(说明:节点 S 与 D 的中继节点可以有多个,为了简化,图中只画出了节点 R);节点 E, D 均在源节点 S 的通信范围内;a 为源节点 S 要发送给目的节点 D 的原始数据帧;a<sub>1</sub>为目的节点 D 接收

到的被破坏了原始数据帧; $b$ 为节点R自身待发送的数据帧,且其目的节点为节点E; $F(a,b)$ 为利用网络编码模型对数据帧 $a$ 与数据帧 $b$ 进行编码组合得到的编码数据帧。

为了将网络编码技术和速率自适应机制相结合,在减少数据帧传输次数的同时,又自适应选择传输速率发送数据帧,RACNC协议对802.11协议MAC帧的头部及ACK确认帧的帧格式进行了修改,增加了一些控制字段。

1)对MAC帧头部的修改。因为RACNC协议的网络编码模型采用的是随机线性网络编码,目的节点需要使用源节点从伽罗华域 $F(q)$ 内随机选取的编码系数对编码数据帧进行解码,所以RACNC协议在MAC帧的头部增加了一个字节的编码系数长度标志位,用于标识编码系数域在MAC帧中所占的长度,也用于目的节点判断接收的数据帧类型。其中,编码系数域是紧挨着MAC帧头部的,即编码系数存放在MAC帧数据区的最开始部分。

2)对ACK帧格式的修改。RACNC协议在802.11协议ACK帧格式的基础上增加了6个比特:第1个比特为编码帧标志位,用于标识该数据帧是原始数据帧还是编码帧;第2个和第3个比特为接收标志位,用于标识目的节点对源节点发送的数据帧的接收情况;最后3个比特为速率索引标志位,用于标识源节点发送下一个数据帧应使用的传输速率。其中,编码帧标志位为0,表示目的节点接收的上一个数据帧为原始数据帧;为1,表示目的节点接收的上一个数据帧为编码数据帧。接收标志位为00,表示目的节点没有成功接收到源节点发送的数据帧,只能由源节点重传;为01,表示目的节点接收到被破坏的数据帧,但其信噪比大于阈值 $A_{th}$ ,可以由中继节点进行编码重传;为11,表示目的节点成功接收到源节点发送的数据帧。速率索引标志位与传输速率的对应关系如表4所示。

表4 速率索引标志位与传输速率的对应关系

速率索引标志位	传输速率/(Mb·s <sup>-1</sup> )
001	6
010	12
011	24
100	36
101	48

RACNC协议在网络中传输数据帧主要分为2步:1)源节点直传数据帧;2)中继节点对数据帧进行网络编码,然后重传编码帧(若第1)步传输失败,才执行第2)步)。下面根据图1中使用RACNC协议对节点在网络中传输数据帧的各个操作步骤进行

具体描述。

**步骤1** 源节点正常发送原始数据帧。源节点S将MAC帧头部中的编码系数长度标志位置为0,然后查询传输速率表得到源节点S到目的节点D之间链路的传输速率,使用该传输速率发送原始数据帧,然后执行步骤2。

**步骤2** 目的节点根据对源节点发送的数据帧的接收情况采取不同的操作。若目的节点D成功接收到源节点S发送的数据帧,则执行步骤3-1;若目的节点D接收到被破坏的数据帧,但其信噪比大于阈值 $A_{th}$ ,则执行步骤3-2;否则,执行步骤3-3。

**步骤3-1** 目的节点D判断接收到的数据帧头部中的编码系数长度标志位为0,得知接收到的数据帧为原始数据帧,首先将ACK帧的编码帧标志位置为0,接收标志位置为11,然后根据接收的数据帧的信噪比和式(4)选取一个传输速率,再根据表3对速率索引标志位赋相应值,回复ACK帧,然后执行步骤4。

**步骤3-2** 目的节点D判断接收到的数据帧头部中的编码系数长度标志位为0,首先将ACK帧的编码帧标志位置为0,接收标志位置为01,然后根据接收的数据帧的信噪比、式(4)和表3对速率索引标志位赋值,回复ACK帧,然后执行步骤4。

**步骤3-3** 目的节点D判断接收到的数据帧头部中的编码系数长度标志位为0,首先将ACK帧的编码帧标志位置为0,接收标志位置为00,然后判断是否连续3次出现接收失败,若是,则根据接收的数据帧的信噪比、式(4)和表3对速率索引标志位赋值;否则,将ACK帧的速率索引标志位置000,回复ACK帧,然后执行步骤4。

**步骤4** 源节点和同时在源节点和目的节点通信范围内的其他节点(节点R)接收到ACK帧,根据ACK帧中的接收标志位和编码帧标志位的不同采取不同的操作。若编码帧标志位为0,接收标志位为11,则执行步骤5-1;若编码帧标志位为0,接收标志位为01,则执行步骤5-2;若编码帧标志位为0,接收标志位为00,则执行步骤5-3。

**步骤5-1** 本次数据帧传输成功,源节点根据ACK帧的速率索引标志位更新传输速率表,然后进行下一个数据帧的发送。接收到ACK帧的其他节点(非源节点)不做任何操作。

**步骤5-2** 源节点根据接收的ACK帧的速率索引标志位更新传输速率表。根据NCAC-WTC协议中的RSWC中继节点选择机制从所有既接收到源节点发送的原始数据帧又接收到目的节点回复的ACK帧的节点中选取一个中继节点(假设在图1中选取的中继节点为节点R),中继节点使用网络编码

模型对源节点发送的原始数据帧和该中继节点自身待发送的数据帧进行网络编码得到一个编码数据帧,并将选取的编码系数存储在 MAC 帧数据区的开始部分,根据编码系数所占的长度更新 MAC 帧头部的编码系数长度标志位,然后查询传输速率表,获得中继节点到目的节点之间链路(链路 RD)的传输速率和中继节点到中继节点自身待发送的数据帧的目的节点之间链路(链路 RE)的传输速率,比较这两个传输速率,选择一个较小的传输速率并使用此传输速率发送编码数据帧,然后执行步骤 6。

**步骤 5-3** 本次数据帧传输失败。若 ACK 帧的速率索引标志位不为 000,则源节点根据速率索引标志位更新传输速率表,否则,不更新传输速率表,然后执行重传数据帧。接收到 ACK 帧的其他节点(非源节点)不做任何操作。

**步骤 6** 目的节点接收到数据帧,判断数据帧头部中的编码系数长度标志位不为 0,得知接收到的数据帧为编码数据帧,使用 MIMO\_NC 对编码数据帧进行解码,然后根据解码的不同情况采取不同的操作。若成功解码,则执行步骤 7-1;否则,执行步骤 7-2。

**步骤 7-1** 目的节点将 ACK 帧的编码帧标志位置为 1,接收标志位置为 11,速率索引标志位置为 000,回复 ACK 帧,然后执行步骤 8。

**步骤 7-2** 目的节点将 ACK 帧的编码帧标志位置为 1,接收标志位置为 00,速率索引标志位置为 000,回复 ACK 帧,然后执行步骤 8。

**步骤 8** 源节点接收到目的节点回复的 ACK 帧,根据 ACK 帧中的接收标志位和编码帧标志位采取不同的操作。若编码帧标志位为 1,接收标志位为 11,则执行步骤 9;若编码帧标志位为 1,接收标志位为 00,则执行步骤 10。

**步骤 9** 本次数据帧传输成功,ACK 帧的速率索引标志位为 000,源节点不更新传输速率表,进行下一个数据帧的发送。接收到 ACK 帧的其他节点(非源节点)不做任何操作。

**步骤 10** 本次数据帧传输失败,ACK 帧的速率索引标志位为 000,源节点不更新传输速率表,重传数据帧。接收到 ACK 帧的其他节点(非源节点)不做任何操作。

以上步骤中对编码帧的解码过程只阐述了目的节点 D 对原始数据帧  $a$  解码后的一系列操作。其中,节点 E 按照文献[13]提出的自适应解码机制从编码帧中解码数据帧  $b$ 。节点 E 首先将 ACK 帧的编码帧标志位置为 1,若成功解码出数据帧  $b$ ,则将接收标志位置为 11;否则置为 00,然后根据接收的编码数据帧的信噪比以及表 3 和表 4 对速率索引标

志位赋值,回复 ACK 帧。中继节点 R 接收到 ACK 帧,首先根据 ACK 帧的速率索引标志位更新传输速率表,然后判断接收标志位,若为 11,则从缓存中删除数据帧  $b$ ;否则不做操作。

## 4 RACNC 协议性能的理论分析

为验证 RACNC 协议的正确性和有效性,下文对 RACNC 协议进行详细的理论分析,主要在相同的网络场景下对比 RACNC 协议中的 RACK 机制与传统的基于 RTS/CTS 握手信号的速率自适应机制的性能。

### 4.1 平均端到端时延

平均端到端时延定义为从源节点开始发送原始数据帧(包括估计信道的时延)到目的节点成功接收到原始数据帧所经过的时延。

在相同的网络场景下,假设:RACNC 协议与基于 RTS/CTS 握手信号的速率自适应协议均经过中继节点编码重传后,目的节点接收到编码帧并成功解码出原始数据帧;控制帧在网络上的传播时延为  $\Delta t$ ;从源节点开始发送原始数据帧到目的节点接收到被破坏的原始数据帧所经过的时延为  $t_1$ ;从中继节点开始发送编码数据帧到目的节点成功解码出原始数据帧所经过的时延为  $t_2$ 。

根据以上假设可得到 RACNC 协议的平均端到端时延为:

$$T = t_1 + \Delta t + t_2 + \Delta t = t_1 + t_2 + 2\Delta t \quad (4)$$

因为基于 RTS/CTS 握手信号的速率自适应协议在发送数据帧之前要先发送 RTS 帧对信道进行探测,然后根据目的节点回复的 CTS 帧确定数据帧的发送速率,所以可得到基于 RTS/CTS 握手信号的速率自适应协议的平均端到端时延为:

$$T' = \Delta t + \Delta t + t_1 + \Delta t + t_2 + \Delta t = t_1 + t_2 + 4\Delta t \quad (5)$$

比较式(4)、式(5)可知  $T < T'$ ,即 RACNC 协议的平均端到端时延小于基于 RTS/CTS 握手信号的速率自适应协议的平均端到端时延。

### 4.2 信道利用率

信道利用率定义为数据帧的传输时延与从节点开始发送数据帧到节点接收到针对该数据帧的接收标志位为 11 的 ACK 帧所经过的时延的比值。

在相同的网络场景下,假设:RACNC 协议与基于 RTS/CTS 握手信号的速率自适应协议发送的数据帧相同,长度为  $l$  bit;选取的传输速率也相同,均为  $c$  bit/s;网络的传播时延为  $\Delta t$ 。

在数据帧传输失败,经过重传才被成功接收的情况下,源节点发送数据帧与目的节点反馈 NACK 帧、中继节点发送编码重传数据帧与目的节点回复 ACK,此过程共经过的时间为  $4\Delta t$ ,所以,RACNC 协

议的信道利用率为:

$$\eta = \frac{\frac{l}{c}}{\frac{l}{c} + 4\Delta t} \quad (6)$$

由于 RTS/CTS 握手信号的速率自适应协议在源节点和中继节点发送数据帧前需要发送 RTS/CTS 进行信道质量探测,这段额外时长为  $4\Delta t$ ,因此数据帧经编码重传被正确接收所需的时间为  $8\Delta t$ ,其信道利用率为:

$$\eta' = \frac{\frac{l}{c}}{\frac{l}{c} + 8\Delta t} \quad (7)$$

比较式(6)、式(7)可知  $\eta > \eta'$ 。因为 RACNC 协议利用 RACK 机制进行信道质量探测,无需发送多余的 RTS/CTS 信号,所以信道利用率较高。

### 4.3 网络吞吐量

网络吞吐量定义为单位时间内通过网络的数据量,即目的节点成功接收的数据量与这些数据量在网络上传输所经过的时间的比值。

在相同的网络场景下,假设:RACNC 协议与基于 RTS/CTS 握手信号的速率自适应协议发送的数据帧大小均为  $M$ ;网络中有  $N$  个节点在同时发送数据,且彼此不产生干扰。

根据 4.1 节得出的平均端到端时延的表达式,在数据帧传输失败,经过重传才被成功接收的情况下,目的节点在时间  $T$  内的数据量为源节点直传的数据量和中继节点编码重传的数据量之和,因此,RACNC 协议的网络吞吐量为:

$$C = \sum_{k=1}^N \frac{2M}{T} \quad (8)$$

同理,基于 RTS/CTS 握手信号的速率自适应协议的网络吞吐量为:

$$C' = \sum_{k=1}^N \frac{2M}{T'} \quad (9)$$

因为  $T < T'$ ,显然得出  $C > C'$ ,即 RACNC 协议的网络吞吐量大于基于 RTS/CTS 握手信号的速率自适应协议的网络吞吐量。

## 5 仿真实验与结果分析

本文使用 OPNET14.5 软件进行网络建模和仿真,在相同场景条件下,分别以平均端到端时延、网络吞吐量和控制开销 3 个性能指标对 RACNC 协议进行仿真,并与 NCAC-WTC 协议进行对比。

### 5.1 仿真环境及参数设置

本文采用随机分布在  $300 \text{ m} \times 300 \text{ m}$  矩形区域的 35 个节点的无线单跳网络仿真模型,将节点每秒

可以产生的数据帧的个数作为自变量对网络性能进行仿真。如果节点产生数据帧的速率大于节点发送数据帧的速率,则节点将未及时发送的数据帧存储于缓存中,假设缓存无限大。SLOT, SIFS, DIFS 时隙长度与 IEEE802.11a 定义的相同,具体的仿真参数设置如表 5 所示。

表 5 仿真参数设置

参数名	参数值
$A_{th}/\text{dB}$	3
SLOT/ $\mu\text{s}$	9
SIFS/ $\mu\text{s}$	16
DIFS/ $\mu\text{s}$	34
ACK/bit	120
MAC 帧头/bit	232

## 5.2 结果分析

### 5.2.1 平均端到端时延

图 2 是节点产生数据帧的速率对不同协议平均端到端时延影响的仿真结果。从图中可以得知 RACNC 协议的平均端到端时延要比 NCAC-WTC 协议的平均端到端时延小,这主要是因为 NCAC-WTC 协议使用固定的基速率发送数据帧,而 RACNC 协议的速率自适应机制,在信道质量比较好的情况下,可使用更高的速率发送数据帧,可减小传输时延。

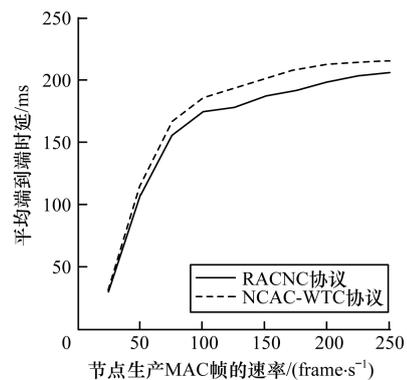


图 2 平均端到端时延对比

### 5.2.2 网络吞吐量

图 3 是节点产生数据帧的速率对不同协议网络吞吐量影响的仿真。从图中可以得知 RACNC 协议的网络吞吐量比 NCAC-WTC 协议的网络吞吐量大,主要原因是 RACNC 协议使用最佳传输速率发送数据帧。当信道质量较好时,RACNC 协议可提高数据帧的发送速率,使单位时间内网络中传输的数据帧更多,即网络吞吐量更大。

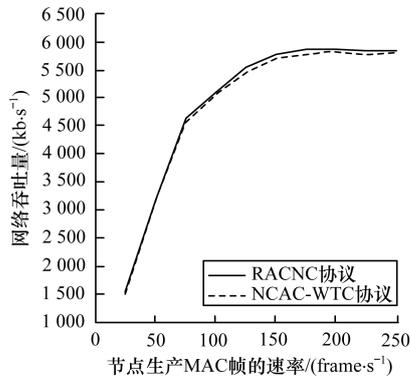


图 3 网络吞吐量对比

### 5.2.3 数据帧投递率

图 4 是节点产生数据帧的速率对不同协议数据帧投递率的仿真结果。从图中可以得知 RACNC 协议的数据帧投递率均高于 NCAC-WTC 协议的数据帧投递率,且随着节点产生数据帧速率的提升,数据帧投递率最后稳定在 96% 附近,主要是因为 RACNC 协议中发送节点依据信道质量情况实时调整发送速率,当信道质量较差时,发送节点可降低数据帧的发送速率,保障了数据帧的投递成功率。以最佳的发送速率发送数据帧能有效减小因发送速率与信道质量不匹配而造成的数据帧丢失,从而提高目的节点对数据帧接收的成功率。

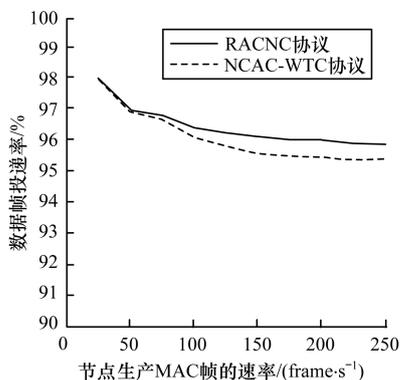


图 4 数据帧投递率对比

## 6 结束语

本文提出了一种基于网络编码的速率自适应协作 MAC 协议 RACNC。与基于网络编码的 NCAC - WTC 协议相比,RACNC 协议有效地减小了平均端到端时延,提高了网络吞吐量和数据帧投递率。但 RACNC 协议使用 RSWC 中继节点选择机制选取中继节点,而 RSWC 机制需要维护一张实时反映节点与其邻居及其邻居的邻居之间的链路信息的连通表,使 RACNC 协议只适用于静态网络或低速移动的动态网络,对于高速移动的动态网络,连通表中的信息将不准确,因此,下一步将针对高速移动的动态网络进行研究。

## 参考文献

- [1] AHLSSWEDE R, CAI Ning, LI S, et al. Network Information Flow[J]. IEEE Transactions on Information Theory, 2000, 46(4): 1204-1216.
- [2] 张建军,唐雅娟. 基于 IEEE 802.11 的速率自适应算法研究[J]. 重庆邮电大学学报(自然科学版), 2009, 21(5): 599-607.
- [3] 王高才,冯 鹏,王 淦,等. 一种速率自适应的能耗优化路由策略研究[J]. 计算机学报, 2015, 38(3): 555-566.
- [4] 李 楠,戚进勇,蔡跃明,等. 无线 Ad Hoc 网络中一种基于网络编码的协同 MAC 协议[J]. 电子与信息学报, 2011, 33(12): 2971-2977.
- [5] MUNARI A, ROSSETTO F, ZORZI M. Phoenix: Making Cooperation More Efficient Through Network Coding in Wireless Networks[J]. IEEE Transactions on Wireless Communications, 2009, 8(10): 5248-5258.
- [6] WANG Xiaoyan, LI Jie. Network Coding Aware Cooperative MAC Protocol for Wireless Ad Hoc Networks[J]. IEEE Transactions on Parallel and Distributed Systems, 2014, 25(1): 167-179.
- [7] KAMERMAN A, MONTEBAN L. WaveLAN®-II: A High-performance Wireless LAN for the Unlicensed Band[J]. Bell Labs Technical Journal, 1997, 2(3): 118-133.
- [8] HOLLAND G, VAIDYA N, BAHL P. A Rate-adaptive MAC Protocol for Multi-hop Wireless Networks[C]// Proceedings of MobiCom'01. New York, USA: ACM Press, 2001: 236-250.
- [9] 潘志鹏,吴 斌,叶甜春. 适用于 IEEE 802.11ac 协议的高效速率自适应算法[J]. 西安电子科技大学学报(自然科学版), 2016, 43(1): 136-142.
- [10] 曾登昊,鄢楚平,王朝翔,等. 基于 802.11 的混合链路调制速率自适应算法研究[J]. 计算机工程与设计, 2011, 32(6): 1898-1900.
- [11] 李科艳,雷 磊,罗 诚,等. WLAN Mesh 网络多信道速率自适应 MAC 协议[J]. 计算机工程, 2014, 40(11): 26-30.
- [12] 杨 林,郑 刚. 一种基于网络编码的速率自适应多播 MAC 协议[J]. 湖南大学学报(自然科学版), 2009, 36(11): 78-82.
- [13] 姚玉坤,朱丽青,杨及开,等. 基于编码感知的完全无碰撞协作 MAC 协议[J]. 华中科技大学学报(自然科学版), 2016, 44(5): 71-80.
- [14] FASOLO E, ROSSETTO F, ZORZI M. Network Coding Meets MIMO[C]// Proceedings of the 4th Workshop on Network Coding, Theory and Applications. Washington D. C., USA: IEEE Press, 2008: 1-6.
- [15] 习 勇,黄清艳,魏急波,等. 基于 IEEE 802.11 高速无线局域网的速率自适应 MAC 协议研究[J]. 电子与信息学报, 2007, 29(6): 1281-1285.