

# 无线 Mesh 网络中使用 双收发器的多信道 MAC 协议研究

何萍实<sup>a</sup>, 徐子平<sup>b</sup>

(解放军理工大学 a. 指挥自动化学院; b. 通信工程学院, 南京 210007)

**摘要:** 在无线 Mesh 网络中, 当底层有多个可用的物理信道时, 必须设计高效的媒体访问控制协议 (MAC) 才能充分利用所有可用信道, 克服多信道的隐终端和显终端问题。三类多信道 MAC 协议都只使用了两个无线收发器, 却能够在保持经济性的同时最大限度地利用多个信道, 并解决多信道环境下广播消息发送和信道切换等问题。分析比较表明, 该 MAC 协议能够在一定程度上利用多个信道增加网络的有效吞吐量, 但在保证网络负载均衡上还存在不足, 需要进一步研究。

**关键词:** 无线 Mesh 网络; 多信道; 媒体访问控制协议; 信道切换

**中图分类号:** TP393      **文献标志码:** A      **文章编号:** 1001-3695(2010)01-0327-03

doi:10.3969/j.issn.1001-3695.2010.01.097

## Research on multi-channel MAC protocols using dual transceivers in wireless Mesh networks

HE Ping-shi<sup>a</sup>, XU Zi-ping<sup>b</sup>

(a. Institute of Commanding Automation, b. Institute of Communication Engineering, PLA University of Science & Technology, Nanjing 210007, China)

**Abstract:** In wireless Mesh networks (WMNs), when multiple physical channels are available, high efficient media access control (MAC) protocols must be designed in order to fully exploit all the available channels and resolve the hidden and exposed terminal problems. This paper introduced three classes of multi-channel MAC protocol using dual transceivers. Multiple channels could be maximally utilized with little cost. The difficulties of broadcast messaged sending and channel switching could also be resolved. Conclusions were drawn through comparison that the protocols can utilize multiple channels in some extent to improve the total throughput, but cannot guarantee the load balancing of the networks and should be further studied.

**Key words:** WMN; multi-channel; MAC protocol; channel switching

无线 Mesh 网络 (WMN) 是一种自我组织、自动配置的网络, 节点自动以 Ad hoc 方式建立连接, 组成网状网络。通过多跳方式通信, 网络的覆盖范围可以有效扩大, 成为无线局域网或城域网连网方式之一。

媒体访问控制协议 MAC 用于解决访问媒体时可能存在的冲突和碰撞, 为网络中的无线设备分配资源。早期的研究中<sup>[1,2]</sup>, 所有网络节点共享一个公共的信道。由于在无线环境中信号的动态范围非常大, 无法使用冲突检测 CD (collision detection) 的方法检测冲突, 使用载波侦听 CS (carrier sense) 感知的信道状态并不一定准确。而且, 由于无线信号的能量有限, 其发送只能被部分节点正确接收, 而距离较远的节点并不知道信道的使用情况, 因而产生了隐终端和显终端问题, 严重影响了网络的有效吞吐量。被广泛应用的无线局域网 IEEE 802.11a/b/g 系列标准 (Wi-Fi) 使用了载波侦听冲突避免 CSMA/CA 策略, 并且使用 RTS/CTS 握手机制缓解隐终端问题, 但在无线多跳网络中无法获得满意的性能<sup>[3]</sup>。

在单信道 MAC 协议中, 控制分组 (如 RTS/CTS) 和业务分组在同一个信道上传输, 而业务分组相对于控制分组很大, 随着网络负载的增加, 频繁的碰撞降低了协议的效率。因此可以

通过增加信道数量来降低分组碰撞的概率; 同时, 增加网络中信道的数量可以增加网络的有效吞吐量, 减少分组的标准传输时延<sup>[4]</sup>。因此, 在无线多跳网络中, 使用多个信道是 MAC 协议未来的发展方向。

### 1 多信道 MAC 协议的类型及面临的主要问题

在 WMN 中, 网络节点按配备无线收发器的数量可以分为三类, 即单收发器、双收发器和多收发器协议。由于早期无线收发器的价格昂贵, 为每个网络节点配备多个收发器无疑是不经济的。而且, 对多信道 MAC 协议的研究大多是基于 Wi-Fi 的, 而 Wi-Fi 设备多只有一个收发器。然而, 单收发器协议或需要严格的时钟同步, 以便所有节点能够在同一时间集中于同一个信道广播控制报文, 或无法解决无线环境下的多信道隐终端和显终端问题, 因而无法获得满意的网络性能。

为网络节点配备多个无线收发器 (多于两个), 特别是每个信道都对应一个收发器, 毫无疑问能够增加网络的有效吞吐量, 但对于 Wi-Fi 路由器这样的家用小型设备, 在体积和成本控制上非常不利。因此, 使用多个信道同时配以两个收发器,

收稿日期: 2009-04-14; 修回日期: 2009-05-25

作者简介: 何萍实 (1979-), 男, 河北唐山人, 工程师, 博士研究生, 主要研究方向为无线 Mesh 网络、无线 Ad hoc 网络等 (jooney@126.com); 徐子平 (1956-), 男, 教授, 硕士, 主要研究方向为宽带交换技术、分组语音通信、ATM 交换机、集群通信系统等。

是一种成本和性能上的折中方案。

学术界针对双收发器的多信道 MAC 协议的研究并不多。这样的研究有两类: a) 网络中的信道由一个公共控制信道和  $N$  个业务信道组成, 节点两个收发器中的其中一个固定于公共控制信道上, 用于发送控制消息和必要的广播消息; 另一个信道在  $N$  个业务信道上切换, 用于发送具体的数据。 b) 并不区分控制信道和业务信道, 所有信道都用于发送控制消息和数据。节点的一个收发器固定于一个信道, 另一个收发器在其他信道上切换。不同的节点所固定的信道可以不同。每个节点都要向邻居通告其固定使用的信道, 这样发送节点可以清楚地知道接收节点固定使用的是哪个信道, 从而把其可切换的收发器切换到接收节点的固定信道上通信。

本文着重研究使用双收发器的多信道 MAC 协议。多信道 MAC 协议除了要面对传统的隐终端和显终端问题, 还要面临多信道的隐终端问题, 有时其对网络性能的影响甚至更大。例如, 使用 RTS/CTS 握手机制可以在很大程度上解决单信道的隐终端问题, 但在多信道环境中却不适用。例如在图 1 中, 节点  $B$  在信道 2 上向节点  $C$  发送数据, 此时, 节点  $A$  由于不知道节点  $B$  的信道使用情况, 在信道 1 上向节点  $A$  发送 RTS 消息。由于  $A$  收不到  $B$  的 CTS 消息, 超时后, 节点  $A$  继续在信道 1 上向  $B$  重发 RTS 消息, 导致网络带宽被浪费。

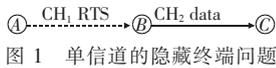


图 1 单信道的隐藏终端问题

在图 2 中, 节点  $A$  在信道 1 上向  $B$  发送 RTS 之前, 节点  $C$  在信道 2 上发送数据。  $B$  收到  $A$  发送的 RTS 消息后, 发送 CTS 作为响应; 之后,  $A$  向  $B$  在信道 1 上发送数据。此时节点  $C$  在信道 2 上的数据发送完毕, 又需要向  $B$  在信道 1 上发送数据,  $C$  的 RTS 消息会与  $A$  和  $B$  正在进行的发送相冲突。这两种情况分别称为 RTS 和 CTS 丢失问题, 也就是多信道上的隐终端问题。其原因都是由于节点使用了不同的信道, 而与邻居暂时处于不同的网络 (网络分割)。使用双收发器就能够加以解决。下文将详细介绍使用双收发器的协议。



图 2 多信道的隐藏终端问题

## 2 使用双收发器的多信道 MAC 协议

多信道 MAC 协议不仅需要解决单信道环境下隐终端和显终端问题, 还要面临多信道的隐终端问题。本文中介绍的协议即试图解决这样的问题。

### 2.1 动态信道分配协议 DCA 和使用功率控制的 DCA-PC

动态信道分配协议 DCA<sup>[5]</sup> 是第一个针对无线 Ad hoc 网络提出的双收发器多信道 MAC 协议。协议使用按需机制为网络节点分配信道, 并且网络中信道的数量和网络中节点的数量无关, 从而增加网络的扩展性。网络不需要严格的时钟同步, 可以降低实现的难度和设备的成本。DCA 为每个节点配有两个独立的半双工收发器。假设网络中可用信道的数量为  $N + 1$ , 其中  $N$  个信道是完全相同的正交数据信道, 用于发送具体的数据; 另一个信道作为公用控制信道使用, 用于发送控制消息。网络节点其中一个无线收发器固定于控制信道, 以保证网络的连通性, 另一个收发器在数据信道上切换。节点需要在本地维护两个数据表: 信道使用表 CUL 和空闲信道表 FCL。以节点  $X$

为例, 其 CUL[ $i$ ] 保存了  $X$  的邻居  $i$  使用信道的情况。CUL 有三个域:

- CUL[ $i$ ].host 表示  $i$  的身份标志 (如 MAC 地址)
- CUL[ $i$ ].ch 表示  $i$  具体使用的信道
- CUL[ $i$ ].rel\_time 表示  $i$  释放信道 ch 的时间

CUL 中保存的邻居信道的使用情况从控制信道中邻居的控制报文中得到。图 3 详细解释了节点间完成握手的过程。

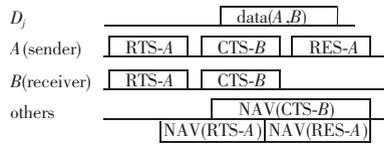


图 3 DCA 中节点通信的握手过程

节点  $A$  向  $B$  发 data 前, 先在控制信道上发送 RTS 消息, RTS 消息中含有其 FCL。  $B$  收到 RTS 后, 根据  $A$  的 FCL 和自己的 CUL, 选出可以使用的空闲信道, 然后发送 CTS 消息给  $A$ , 并切换到选定的数据信道  $D_j$  上;  $A$  收到 CTS 后, 发送 RES (预留) 消息, 通知其邻居信道的使用情况, 防止邻居在发送时使用该信道, 同时  $A$  切换到选定的信道  $D_j$  上发送 data,  $B$  收到 data 后返回 ACK 作为确认, 完成整个数据发送过程。  $B$  收到 RTS 消息后, 如果没有可以使用的信道,  $B$  会在 CTS 中发送其信道使用状态改变的时间  $T_{est}$ ,  $A$  收到 CTS 后, 在之后  $T_{est}$  重新发送 RTS 协商信道。

后来其作者又提出了 DCA 协议的功率控制版本 DCA-PC<sup>[6]</sup>, 通过功率控制减少节点之间不必要的干扰。设发送节点  $X$  和接收节点  $Y$  的发送功率和接收功率分别为  $P_t$  和  $P_r$ , 根据文献[7]的结果有

$$P_r = P_t (\lambda / 4\pi d)^n g_t g_r \quad (1)$$

其中:  $\lambda$  为载波的波长;  $d$  为  $X$  与  $Y$  之间的距离;  $g_t$  和  $g_r$  分别为  $X$  和  $Y$  的天线增益;  $n$  为路径损耗因子 (根据环境的不同, 在 2 与 6 之间取值, 典型值为 2)。因此, 可以通过控制发送功率来影响接收功率的大小。为此, 每个节点还需要增加一个功率表 power[], 保存其邻居节点发送分组使用功率的级别。DCA-PC 和 DCA 的区别在于, 控制信道上的 RTS/CTS/RES 报文使用最大功率发送, 而减小业务信道上 data 和 ACK 的发射功率, 从而减少对接收节点邻居的干扰, 以增加信道的空间重用范围。仿真结果表明, 与使用单信道的 IEEE802.11 协议相比, DCA 和 DCA-PC 都能显著提供系统的有效吞吐量, 并且在数据信道数量小于 7 时, DCA-PC 的性能优于 DCA。尤其是当仅使用三个数据信道时, DCA-PC 取得了最优的性能, 这对于 802.11b 这样只有三个互不重叠的数据信道的系统来说是好消息。由于只使用一个控制信道, 在数据信道数量增加时, 控制信道已经饱和, 无法有效地使用所有信道, 网络的总吞吐量反而下降。由于 DCA 和 DCA-PC 在控制信道上只是简单使用了 IEEE802.11 DCF 相同的策略, 节点需要使用控制报文协商信道, 致使控制报文长度增加, 进一步增加了控制报文碰撞的概率, 因而无法取得最优的信道利用率。

### 2.2 使用不同固定信道策略的双收发器多信道 MAC 协议

虽然与 DCA 一样使用了双收发器, 但文献[8]并没有让网络中所有节点的一个收发器固定在公共控制信道上。在文献[8]中, 网络中所有信道都相同, 并不区分数据信道和业务信道, 每个节点的两个收发器的其中一个固定在一个信道上, 另一个收发器在其他信道上切换。而网络中不同节点的固定信道可以不同。这样一个节点  $A$  需要与节点  $B$  通信, 而  $A$  和  $B$  的固定信道不同时,  $A$  的可切换收发器需要切换到  $B$  的固定信

道上,以完成通信。



图 4 可变信道切换示例

例如在图 4 中节点 A、B、C 的固定收发器分别在信道 CH<sub>1</sub>、CH<sub>2</sub> 和 CH<sub>3</sub> 上,而可切换收发器分别在信道 CH<sub>3</sub>、CH<sub>1</sub> 和 CH<sub>2</sub> 上。此时节点 A 向 B 发送数据,需要把其可切换收发器切换到 CH<sub>2</sub> 上,节点 B 向 C 发送数据,需要切换到 CH<sub>3</sub> 上,并且两个通信过程可以同时进行。该协议的核心是每个节点如何选取其固定的信道。为此,每个节点需要维护一张其邻居使用固定信道的表 neighborTable,并通过周期性地广播 hello 消息来互相交换信息。节点通过邻居的 hello 消息可以获取其两跳范围内节点所使用固定信道的情况。而且,节点可以改变自己的固定信道。开机或初始化时,节点随机地选择一个信道作为固定信道,然后与邻居节点交换 hello 消息。通过 hello 消息,节点可以获知使用每个信道作为固定信道的节点的数量,并从使用者较少的信道中挑选一个作为其固定使用的信道,然后以概率  $p$  把其固定收发器切换到这个信道上。如果节点完成了固定信道的切换,需要通过 hello 消息向邻居节点通告其信道的改变情况。

此协议的优点是,通过为每个节点合理地分配固定信道,可以有效地把网络负载分担到所有的可用信道上。而且,由于不使用专用控制信道,节点可以尽可能地使用网络中的所有信道,也不存在控制信道饱和的问题。由于网络中信道的数量与节点数量无关,可以轻松地增加信道数量以增加网络的吞吐量。但由于节点间没有同一个信道来保证网络的连通性,在需要发送广播消息时,每个节点都需要用其可切换信道在每个信道上发送一个副本。而信道切换存在不可忽略的时延(对于 IEEE802.11 的无线设备,时延可以达到几百毫秒),而协议却需要频繁的广播 hello 消息,影响了协议的性能。

### 2.3 BTMC:使用忙音的多信道协议

文献[9]提出了另一种解决多信道分配的 MAC 协议,为每个网络中的节点配备一个忙音接口和一个半双工收发器。忙音接口只完成忙音的发送和检测,成本比数据收发器低得多。为此,每个信道还需要配有一个忙音信道,所占带宽与数据信道相比可以忽略。协议使用一组  $k$  个哈希函数来解决信道分配问题:

$$c_i = h_j(M) \quad (i \in [1, m]; j \in [1, k]) \quad (2)$$

其中: $i$  为信道的编号,取值从  $1 \sim m$ ;  $j$  为哈希函数的编号,取值从  $1 \sim k$ ;  $M$  表示每个节点的 MAC 地址。每个哈希函数对节点的 MAC 地址作运算,结果都可对应于  $m$  个信道中的一个。等待接收的节点(如 B)通过计算  $h_j$  找到  $j$  取值最小的空闲信道,并在该信道上等待。发送节点(如 A)首先通过哈希函数对 B 的 MAC 地址作运算,找到 B 可能使用的空闲信道;然后, A 在该信道上发送 RTS 消息,然后等待 B 的 CTS 消息。在等待过程中,如果检测到忙音,那么该信道并不是空闲信道, A 继续使用哈希函数表中的函数计算 B 可能使用的下一个空闲信道,再次发送 RTS 消息。如果函数表中所有函数都计算完毕,但并没有找到空闲信道, A 执行退避过程,之后重新依次寻找空闲信道。B 收到 A 发送的 RTS 消息后,如果此时忙音信道空闲, B 向 A 发送 CTS 消息,并发送忙音,等待 A 发送的数据;如果 B 收到不是发往自己的 RTS 消息,或者 B 收到 RTS 消息时

检测到忙音,那么 B 重新寻找空闲信道。A 收到 B 的 CTS 消息,并检测到忙音,那么认为 B 成功收到了 RTS,可以发送数据,完成握手过程。

BTMC 不需要全网同步即可实现节点之间的信道分配,并且由于不使用控制信道,不会因为控制信道饱和而影响网络可用的信道数量。但由于需要通过信道切换的方式依次寻找空闲信道,信道切换时延的代价比较高。而且在每个信道上都需要 RTS/CTS 消息的交互才能完成通信,开销也比较大。

## 3 结束语

使用多个物理信道,在不增加网络带宽的情况下,通过设计高效的 MAC 协议,既能增加网络的有效吞吐量,也能减小发送数据的标准传播时延,必将是未来研究 MAC 协议的方向之一。受到成本和系统复杂性的限制,通常网络节点的无线收发器数量小于或远小于可用信道的数量。使用一个无线收发器,能够兼容现有的 IEEE802.11 设备,但在发送广播消息和信道切换时延上有着固有的缺点,无法有效利用所有可用信道带宽。使用多个无线收发器,又面临更多复杂的问题,只适用于无线骨干网中的一些大型设备。因此使用两个无线收发器,其中一个用于发送广播消息和控制消息,另一个用于在数据信道上切换无疑成为一种折中的解决方案。目前的研究结果还无法达到满意的程度,需要进一步深入的研究。多信道的使用,必然要求网络负载能够平均地分配到每个信道,以最大限度地提高信道的利用率。目前,根据信道的负载进行信道分配已经成为下一个研究的热点。但这需要对网络的负载进行正确的预测,或者对网络负载进行测量。而且,使用信道负载作为信道切换的标准还可能带来信道频繁切换的问题,切换时延是必须考虑的问题,需要进一步细致的分析和研究。

### 参考文献:

- [1] XU Shu-gong, TAREK S. Does the IEEE 802.11 MAC protocol work well in multihop wireless Ad hoc networks [J]. IEEE Communication Magazine, 2001, 39(6): 130-137.
- [2] PHIL K. MACA: a new channel access method for packet radio networks [C] // Proc of the 9th Computer Networking Conference on ARRL/CRRL Amateur Radio. London: [s. n.], 1999: 134-140.
- [3] BHARGHAVAN V, DEME R S, SHENKE R, et al. MACAW: a medium access protocol for wireless LAN's [C] // Proc of ACM SIGCOMM. New York: ACM Press, 1994: 212-225.
- [4] NASIPURI A, ZHUANG J, DAS S, et al. A multichannel CSMA MAC protocol for multihop wireless networks [C] // Proc of WCNC '99 IEEE. New Orleans: [s. n.], 1999: 1402-1406.
- [5] WU S L, LIN C Y, TSENG Y C, et al. A new multi-channel MAC protocol with on-demand channel assignment for multi-hop mobile Ad hoc networks [C] // Proc of International Symposium on Parallel Architectures, Algorithms and Networks (I-SPAN). Washington DC: IEEE Computer Society, 2000: 232-237.
- [6] WU S L, LIN C Y, TSENG Y C, et al. A multi-channel MAC protocol with power control for multi-hop mobile Ad hoc networks [J]. The Computer Journal, 2002, 45(1): 101-110.
- [7] ELLEN K W. Wireless multimedia communications: networking video, voice, and data [M]. Massachusetts: Addison-Wesley, 1998.
- [8] KYASANUR P, VAIDYA N H. Routing and interface assignment in multi-channel multi-interface wireless networks [C] // Proc of WCNC '05 IEEE. New Orleans: IEEE, 2005: 2051-2056.
- [9] EIHAWARY M, HAAS Z J. Busy tone multi-channel (BTMC): a new multi channel MAC protocol for Ad hoc networks [C] // Proc of the 6th Annual IEEE International Conference on Pervasive Computing and Communications. Washington DC: IEEE Computer Society, 2008: 234-238.