

# IEEE 802.11 无线网络中的冲突顺序解析算法

张棋飞<sup>1)</sup> 孙宝林<sup>1)</sup> 桂 超<sup>1)</sup> 刘 威<sup>2)</sup> 程文青<sup>2)</sup> 杨宗凯<sup>2)</sup>

<sup>1)</sup>(湖北经济学院计算机学院 武汉 430205)

<sup>2)</sup>(华中科技大学电子与信息工程系 武汉 430074)

**摘 要** 对分布式的基于竞争的 MAC 协议中的分组冲突进行合理的分类对于有效进行冲突解析起着关键作用。文中根据冲突节点所处退避阶段的不同将冲突划分为两类:交叉冲突和同级冲突,并且通过分析和实验证明了网络中的交叉冲突是普遍存在的。传统的退避算法并未考虑这两种冲突的不同特点,而是采用同样的方式进行处理,对系统性能造成了一定影响。作者认为,对于不同的冲突类型应该予以区别对待,因此提出了冲突顺序解析算法 CSR(Collision Sequential Resolution)。CSR 根据冲突发生的顺序,将冲突节点依次分布在一系列连续独立的基本窗口上,通过竞争窗口的离散化消除了交叉冲突;同时,通过选择合适的基本窗口大小在分组延迟和同级冲突概率之间取得折衷。仿真实验表明,同传统的退避算法相比,CSR 能够在冲突次数、吞吐量、延迟以及公平性方面提供全面的性能提升。

**关键词** IEEE 802.11;媒体接入控制;交叉冲突;同级冲突;冲突顺序解析

**中图法分类号** TP393

**DOI 号:** 10.3724/SP.J.1016.2009.01026

## Collision-Sequential Resolution Algorithm in IEEE 802.11 Wireless Networks

ZHANG Qi-Fei<sup>1)</sup> SUN Bao-Lin<sup>1)</sup> GUI Chao<sup>1)</sup> LIU Wei<sup>2)</sup> CHENG Wen-Qing<sup>2)</sup> YANG Zong-Kai<sup>2)</sup>

<sup>1)</sup>(School of Computing, Hubei University of Economics, Wuhan 430205)

<sup>2)</sup>(Department of Electronics and Information Engineering, Huazhong University of Science and Technology, Wuhan 430074)

**Abstract** Collision classification is crucial for collision resolution in distributed contention-based MAC protocols. Based on backoff stages the collided nodes stay, this paper classifies collisions into two categories: cross collision and intra collision, and then proves by analysis that cross collision is pervasive in networks, which is verified later by simulation results. Traditional backoff algorithms did not discriminate between these two collisions and treated them alike, which damages the system performance. However, the two collisions should be dealt with different policies and thus propose a novel backoff algorithm featuring Collision-Sequential Resolution (CSR) to address the problem. CSR redistributes the collided nodes in a series of consecutive separated elementary windows according to their occurrence sequence to eliminate cross collision completely. The intra collision is resolved with an appropriate elementary window size to achieve a tradeoff between delay and intra collision probability. Extensive simulations for collision times, throughput, delay and fairness demonstrate that the CSR provides comprehensive improvement compared with traditional backoff algorithms.

**Keywords** IEEE 802.11; medium access control; cross collision; intra collision; collision-sequential resolution

收稿日期:2006-10-07;最终修改稿收到日期:2008-09-20。本课题得到国家自然科学基金(60572049,60602029)、中国博士后科学基金(20070410955)以及湖北省教育厅科研项目(D20081904,T200902,Q20091903)资助。张棋飞,男,1977年生,博士,讲师,主要研究方向为无线自组织网络和无线传感器网络。E-mail: cheffly@gmail.com。孙宝林,男,1963年生,博士,教授,主要研究领域为高性能网络技术及无线通信网络。桂 超,男,1965年生,副教授,主要研究方向为无线自组织网络和无线传感器网络。刘 威,男,1977年生,博士,副教授,主要研究方向为无线自组织网络和认知无线电技术。程文青,女,1965年生,博士,教授,博士生导师,主要研究领域为无线传感器网络。杨宗凯,男,1963年生,博士,教授,博士生导师,主要研究领域为下一代互联网和无线自组织网络。

# 1 引言

媒体接入控制(Medium Access Control, MAC)协议是无线局域网(Wireless Local Area Networks, WLANs)的基础,主要负责解析节点间的竞争冲突,协调系统资源分配,对网络的性能起着决定性作用.由于WLAN的分布式特性,分组冲突不可避免. MAC协议通常采用退避算法(Backoff)进行冲突解析.退避的目的是协调节点的信道接入,在冲突发生时能够及时有效地进行恢复.退避机制决定了节点在发送报文之前需要推迟的时间,以便在多节点竞争接入同一信道时保证接入的有效性,达到合理利用信道资源的目的.退避算法既要尽量降低各节点间的冲突概率,又要避免因退避时间过长而降低信道利用率,同时还要保证各节点公平地访问信道.在分布式环境中,这些要求对算法的设计提出了挑战.

现有的退避算法主要是基于竞争窗口(Contention Window, CW)机制.以IEEE 802.11协议的BEB(Binary Exponential Backoff)<sup>[1]</sup>算法为例,每个节点维护一个本地竞争窗口值CW.需要进行退避时,节点在 $[0, CW-1]$ 范围内随机选取一个数,作为退避计数器BC(Backoff Counter)的值.每当信道空闲一个时隙,BC的值减1.倘若信道由闲变忙,退避进程暂时挂起,BC值被冻结,直到信道持续空闲DIFS(Distributed Coordination Function Inter-frame Space)后才开始继续递减BC值.一旦BC值为0,节点开始发送分组.如果发送失败,节点加倍CW直至 $CW_{\max}$ ,重新选择BC的值进行新一轮退避;若发送成功,则重设CW为 $CW_{\min}$ :

$$\begin{cases} CW_i = \min(2^i CW_{\min}, CW_{\max}), & \text{发生冲突} \\ CW_0 = CW_{\min} & \text{发送成功} \end{cases}$$

其中, $i$ 代表节点所处的退避阶段<sup>[2]</sup>,取决于节点自最近一次成功发送以来所遭遇的冲突次数.

BEB算法虽然简单,但却存在一个弊端:由于节点在发送成功以后立刻将竞争窗口值设为最小,使得该节点有可能再次接入信道,从而阻塞其它节点,降低了公平性.于是,提出了MILD(Multiplicative Increase and Linear Decrease)<sup>[3]</sup>算法.在MILD中,每当节点发送成功以后,竞争窗口并不直接重设为最小值 $CW_{\min}$ ,而是递减 $\alpha$ ,直至达到 $CW_{\min}$ .在每次分组发送失败后都将竞争窗口值增大 $\beta$ 倍( $\alpha$ 和 $\beta$ 分别设为1和1.5).MILD采用了竞争窗口拷贝机

制,在提高公平性的同时也带来了额外的开销,甚至会导致竞争窗口值迁移问题<sup>[4]</sup>.

基于竞争窗口机制的退避算法考虑的都是如何改变竞争窗口的变化规则,以反映信道的当前竞争状况. SD(Slow CW Decrease)<sup>[5]</sup>算法在成功发送一个报文后将节点的竞争窗口减小为原来的 $\delta$ 倍( $\delta=0.9$ ).而在MIMD(Multiplicative Increase Multiplicative Decrease)<sup>[6]</sup>中,窗口减小速度更快,每次发送成功后都将CW减半. EIED(Exponential Increase Exponential Decrease)<sup>[7]</sup>采用一种通用的算法,当发送成功后用 $CW/r_D$ 更新CW;若失败,则用 $CW/r_I$ 代替CW. MIMLD(Multiplicative Increase Multiplicative/Linear Decrease)<sup>[8]</sup>额外使用了一个参数 $CW_{\text{basic}}$ 作为区分网络竞争程度的标准.当 $CW > CW_{\text{basic}}$ 时,认为竞争较激烈,成功发送报文后将发送窗口减半;否则,线性递减CW. LMILD(Linear/Multiplicative Increase Linear Decrease)<sup>[9]</sup>将发送失败节点的CW值增大为原来的 $m_c$ 倍,同时,监听到冲突的其它节点将CW增大 $l_c$ .报文发送成功后,所有节点的CW值都线性减小.

以上算法都是在检测到分组成功发送后就认为信道竞争状况缓解,从而立刻减小竞争窗口值.由于节点一次发送成功的偶然性很大,因此若在此时就减小竞争窗口大小,有可能会引起进一步的冲突. GDCF(Gentle DCF)<sup>[10]</sup>算法规定:只有当节点连续成功发送若干个报文后才将竞争窗口值减小一半.这种保守的窗口变化方式更加温和,也更容易反映信道的真实竞争状况,尤其在竞争节点数较多的情况下, GDCF能够显著降低节点的冲突概率,获得比DCF更高的吞吐量和更多的公平性.

当前的冲突解析机制只是利用退避过程的随机性来规避冲突,通过竞争窗口范围的不同获取不同的BC值,以此实现冲突的解析.尽管竞争窗口的调整策略各不相同,但这些算法仍然无法完全避免分组冲突的发生.

下面以典型的IEEE 802.11协议为例,说明传统退避算法存在的问题.如图1所示,处于退避阶段 $i$ 的3个节点 $a, b$ 和 $c$ 在其竞争窗口 $[0, CW_i-1]$ 上随机分布,选取的BC值分别为 $j, j, k$ .由于 $a$ 和 $b$ 选取了在同一时隙进行分组发送,所以在时刻 $j$ 引发分组冲突.于是, $a$ 和 $b$ 进入退避阶段 $(i+1)$ ,并且新的竞争窗口 $[0, 2CW_i-1]$ 上重新选择BC值,确定各自的发送时隙分别为 $l$ 和 $k$ (用 $a'$ 和 $b'$ 分别表示).碰巧此时节点 $b'$ 选取的发送时隙 $k$ 已被处于

退避阶段  $i$  的  $c$  节点提前占用. 于是, 在  $k$  时刻, 处于退避阶段  $i$  的  $c$  节点和处于退避阶段  $(i+1)$  的  $b'$  节点同时发送分组, 造成新的分组冲突. 不难看出, 这次冲突是发生在两个处于不同退避阶段的节点之间的. 也就是说, 即使节点已经进行过退避 (如  $b'$  节点), 仍然不能保证分组的正常发送.

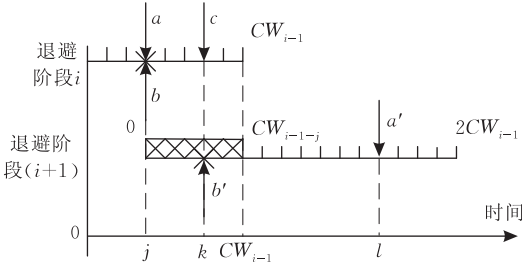


图 1 传统退避算法冲突解析示意图

这种由于退避效果不好产生的冲突不仅导致冲突解析过程失败, 更会对其它节点的正常发送造成干扰. 如上例中, 由于节点  $b'$  的介入, 使得本来应该成功发送的  $c$  节点也发生冲突.

通过进一步的分析我们发现: 在以上退避过程中, 节点  $b$  所经历的两次冲突性质是不一样的. 第一次冲突发生在两个处于同一退避阶段的节点之间, 而第二次冲突则是发生在两个处于不同退避阶段的节点之间. 于是, 我们可以根据冲突节点所处退避阶段的不同将冲突划分成两类: 交叉冲突 (Cross Collision) 和同级冲突 (Intra Collision). 所谓交叉冲突, 是指产生冲突的节点处于不同的退避阶段; 而在同级冲突中, 冲突节点都处于同一退避阶段. 如图 1 中,  $a$  和  $b$  之间的冲突是同级冲突, 而  $b'$  和  $c$  之间的冲突则为交叉冲突.

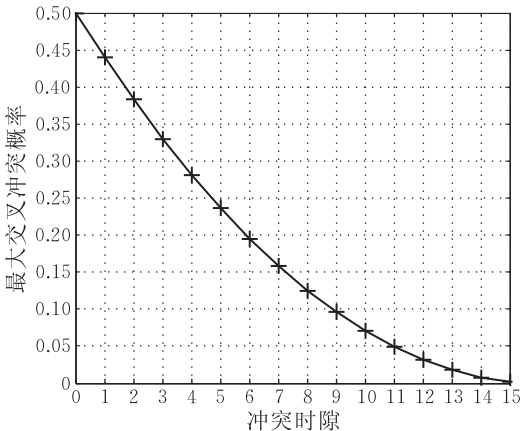


图 2 两个连续退避阶段之间的最大交叉冲突概率

我们认为, 对冲突性质进行有效的区分对于完成冲突解析具有重要意义. 下面, 我们将基于冲突划分标准对 IEEE 802.11 无线网络中的冲突解析算法

进行研究. 我们将在第 2 节中分析网络中交叉冲突发生概率的大小, 从而揭示其重要性; 第 3 节根据冲突划分标准, 提出冲突顺序解析算法 CSR (Collision Sequential Resolution); 仿真结果在第 4 节中给出; 最后对全文进行总结.

## 2 交叉冲突概率分析

IEEE 802.11 DCF 协议<sup>[11]</sup>规定: 在每次发送失败后, 节点加倍自己的竞争窗口, 重新选择 BC 值进行退避. 假定节点当前处于退避阶段  $i$ , 竞争窗口大小为  $CW_i$  ( $0 \leq i \leq i_{\max}$ ,  $CW_{\min} \leq CW_i \leq CW_{\max}$ ). 倘若在第  $j$  ( $0 \leq j \leq CW_i - 1$ ) 个时隙发生冲突, 节点进入退避阶段  $(i+1)$ , 同时加倍自己的竞争窗口, 如图 1 所示. 退避阶段  $(i+1)$  的竞争窗口为  $[0, 2CW_i - 1]$ , 图中的阴影部分表示两个退避窗口的交集.

为了便于分析, 我们首先只考虑两个连续的退避阶段:  $i$  和  $(i+1)$ . 在这两个连续的退避阶段之间发生交叉冲突的必要条件满足:

- (1) 处于退避阶段  $(i+1)$  的节点选取的 BC 值位于重叠区域内, 即  $BC_{i+1} \in [0, CW_i - 1 - j]$ ;
- (2) 在退避阶段  $i$ , 仍然有节点的 BC 值处于交集之中, 即  $BC_i \in [j, CW_i - 1]$ .

因此, 在两个连续的退避阶段  $i$  和  $(i+1)$  之间发生交叉冲突的概率  $Pr_{cr(i,i+1)}(j)$  满足

$$Pr_{cr(i,i+1)}(j) \leq Pr\{BC_i \in [j, CW_i - 1]\} \times Pr\{BC_{i+1} \in [0, CW_i - 1 - j]\} \quad (1)$$

由于退避计数器值在竞争窗口上均匀分布, 所以有

$$Pr\{BC_i \in [j, CW_i - 1]\} = \frac{CW_i - 1 - j + 1}{CW_i - 1 + 1} \quad (2)$$

$$Pr\{BC_{i+1} \in [0, CW_i - 1 - j]\} = \frac{CW_i - 1 - j + 1}{2CW_i - 1 + 1} \quad (3)$$

其中, 竞争窗口大小  $CW_i$  满足

$$CW_i = 2^i \times CW_{\min} \quad (4)$$

联立方程 (1)~(4), 得到

$$Pr_{cr(i,i+1)}(j) \leq \frac{(CW_i - j)^2}{2CW_i^2} = \frac{(2^i CW_{\min} - j)^2}{2^{2i+1} CW_{\min}^2} \quad (5)$$

显然, 最大交叉冲突概率函数  $\max Pr_{cr(i,i+1)}(j)$  满足

$$\max Pr_{cr(i,i+1)}(j) = \frac{(2^i CW_{\min} - j)^2}{2^{2i+1} CW_{\min}^2} \quad (6)$$

这是一个单调递减函数, 在其取值范围  $j \in [0, 2^i CW_{\min} - 1]$  内, 最大值为

$$\max Pr_{cr(i,i+1)}(0) = \frac{1}{2} \quad (7)$$

最小值为

$$\max Pr_{cr(i,i+1)}(2^i CW_{\min} - 1) = \frac{1}{2^{2i+1} CW_{\min}^2} \quad (8)$$

图 2 示出了在两个连续退避阶段  $i$  和  $(i+1)$  之间冲突时隙与最大交叉冲突概率之间的关系. 从图中容易看出: 冲突发生得越早, 引发交叉冲突的概率越大.

尤其当退避阶段  $i$  达到其上限值  $i_{\max}$  时, 有

$$CW_i = CW_{i+1} = CW_{\max} \quad (9)$$

此时, 最大交叉冲突概率函数满足

$$\max Pr_{cr(i_{\max}, i_{\max})}(j) = \frac{(2^i CW_{\min} - j)^2}{(2^i CW_{\min}^2)^2} \quad (10)$$

由此可见, 当满足条件:

$$2^i CW_{\min} \gg j \quad (11)$$

交叉冲突概率甚至可以无穷逼近 1.

虽然我们推导了在两个连续退避阶段之间发生交叉冲突概率的最大值, 但是应该强调的是: 交叉冲突不仅仅发生在两个连续的退避阶段之间. 只要节点的退避窗口存在着交集, 就有可能引发交叉冲突. 所以对当前处于某一退避阶段  $i$  的节点来讲, 它的交叉冲突概率应为

$$Pr_{cr(i)}(j) = \sum_{k=0}^{i-1} Pr_{cr(k,i)}(j) \quad (12)$$

显然, 有

$$Pr_{cr(i)}(j) \geq Pr_{cr(i-1,i)}(j) \quad (13)$$

可见, 交叉冲突是网络中的一类重要冲突.

### 3 冲突顺序解析算法 CSR

传统的退避算法并没有区分交叉冲突和同级冲突, 而是采用同一种方式进行处理. 特别地, IEEE 802.11 协议通过单纯增加竞争窗口的大小来避免冲突. 尽管该策略可以有效解析同级冲突, 但却引入了额外的空闲时隙, 增加了分组延迟. 而且, 由于退避算法的随机性, 只要节点的竞争窗口之间存在着交集, 就有可能引发交叉冲突. 尤其当竞争窗口到达最大值  $CW_{\max}$  后, 交叉冲突的概率甚至可以无穷逼近 1.

我们认为: 对于这两种不同类型的冲突应该采取不同的解析策略. 同交叉冲突相比, 同级冲突的解析策略比较简单, 仅仅通过扩大竞争窗口值就可以有效降低冲突概率. 然而, 至今却还没有提出一个针对交叉冲突的有效解决方案.

实际上, 交叉冲突和同级冲突间存在着某种程度的联系. 交叉冲突的存在, 不仅造成自身冲突解析过程的失败, 更有可能影响原本应该成功发送的节点, 带来新的冲突. 可以试想, 如果网络中没有交叉冲突, 而是仅仅存在同级冲突, 则从概率的角度分析, 也就意味着系统中发生过的冲突都得到了良好解析. 基于此, 我们提出一个冲突顺序解析算法 CSR. CSR 采用了一种冲突的顺序转化机制, 从解析交叉冲突入手, 实现从交叉冲突到同级冲突的有效转化, 最终使得系统收敛到一个无冲突的状态.

#### 3.1 IEEE 802.11 DCF

我们基于 IEEE 802.11 DCF 协议实现 CSR 算法. 图 3 是 DCF 基本接入模式示意图.

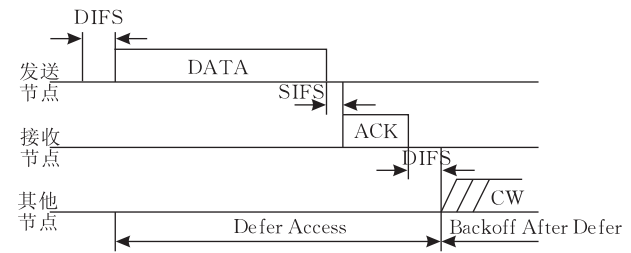


图 3 IEEE 802.11 DCF 基本接入模式示意图

根据信道上是否有分组发送, DCF 将信道分为空闲状态与忙状态, 而忙状态可以根据分组发送的结果进一步区分成传输状态和冲突状态. 节点在发送数据之前首先侦测信道状态. 如果信道忙, 节点从当前竞争窗口上随机选取退避计数器值, 以时隙为单位进行退避. 只要信道连续空闲 DIFS 后, 每空闲一个时隙, 节点递减计数器值. 一旦计数器递减到 0, 节点开始发送数据. 如果在退避过程中信道由闲变忙, 则退避进程挂起, 直到信道再次连续空闲 DIFS 后继续递减计数器值. 在收到发送方发来的数据分组后, 接收节点经过 SIFS (Short Interframe Space) 后回送一个确认分组 ACK (Acknowledgement). 如果发送节点没有及时收到 ACK, 表明传输失败, 于是增大竞争窗口, 重新选择退避计数器值进行新一轮退避.

#### 3.2 算法实现

我们从交叉冲突的定义中受到启发: 节点只有处于相互独立的竞争窗口中才能够完全消除重叠区域, 从而从根本上杜绝交叉冲突, 使得网络中只有同级冲突存在, 实现了冲突的有效转化. CSR 算法就是基于以上思想而提出. CSR 严格按照冲突发生的时间顺序为冲突节点动态分配独立的竞争窗口, 以此消除交叉冲突; 同时, 通过选择适当的窗口值大小

来减少同级冲突概率,保证分组延迟. CSR 将节点的竞争窗口划分为一个初始竞争窗口  $CW_0$  和若干个动态分配的基本窗口 EW (Elementary Window). 系统初始化时,节点均匀分布在初始竞争窗口上. 一旦发生冲突,根据冲突发生的顺序,将冲突节点依次分布在连续独立的基本窗口上,通过分布窗口的离散化消除交叉冲突;同时设置合适的基本窗口大小,在同级冲突概率和时延之间取得权衡.

为了消除交叉冲突,我们定义竞争级别 (Contention Level, CL) 为竞争区域上发生的信道接入次数,利用 CL 来指导基本窗口的动态分配. 事实上,CL 可以看成是信道竞争状态的一种表征.

CSR 算法实现如下:

1. 所有节点监测信道. 在信道连续空闲 DIFS 后,如果再空闲一个时隙,节点将其退避计数器值减 1. 一旦计数器值到达 0,节点马上发送分组. 倘若信道由闲变忙,退避进程挂起直到信道再次空闲 DIFS 后继续递减退避计数器值.

2. 一旦监测到忙状态 (不管是传输状态还是冲突状态),表明信道上有分组发送,于是所有节点都将 CL 值加 1:  
 $CL = CL + 1$  侦测到忙状态.

3. 节点完成数据分组发送后,如果在规定时间内没有收到接收方发来的确认信息,则认为发生了分组冲突<sup>①</sup>,于是检查其重传次数:

3.1. 如果重传次数超过了最大门限值  $Max\_Retrans\_Times$ ,节点将分组丢弃,并且重设竞争窗口为  $[0, CW_0 - 1]$ ,同时保留 CL 值以便继续指导后续的窗口分配;

3.2. 如果没有超过重传门限,节点重新设置其新的竞争窗口为

$$[CW_0 + (CL - 1) \times EW, CW_0 + CL \times EW - 1],$$

这样,冲突节点将会在当前竞争窗口末端的一个基本窗口上随机分布,从而实现了退避空间的独立性,避免了交叉冲突.

4. 在一次成功的分组传输后 (数据分组传输加上 ACK 传输),节点将 CL 值加 1,并据此设置其新的竞争窗口为

$$[CW_0 + (CL - 1) \times EW, CW_0 + CL \times EW - 1],$$

从而使得该节点完全独占第 CL 个基本窗口,保障了其后续传输都是在无干扰的情况下进行. 同时,由于这一系列基本窗口是连续排列的,节点需要经过一个周期的等待后才能够再次接入信道,从而为其它节点提供了接入机会,保障了算法的公平性.

图 4 描述了 CSR 算法. 图中的虚线箭头表示节点的重新分布. 假设节点 1 和 2 同时发送数据,产生一次冲突. 此时,网络中的  $CL = 1$ ,这两个节点在紧接初始竞争窗口后的基本窗口 1 上重新分布. 如果节点 3 发送成功,它设置  $CL = 2$ ,并将自己的分布范围设为紧接初始竞争窗口后的第 2 个基本窗口. 由于节点 3 独占了该窗口,保障了它的后续传输都能

够在无干扰的情况下进行. 通过这种方式,实现了退避窗口的不相交特性,完全消除了交叉冲突. 同时,算法使得发送成功的节点需要经过一轮的等待后才能够再次接入信道,以此获得更多的公平性.

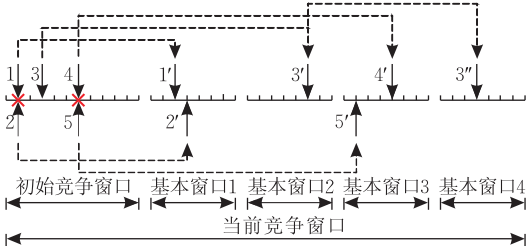


图 4 冲突顺序解析算法示意图

3.3 冲突概率比较

如果两个或多个节点选择在同一时隙发送分组,则产生冲突. 假定共有  $n$  个节点均匀分布在竞争窗口  $[0, m - 1]$  上,可以得到分组冲突概率  $Pr_m^n\{\text{冲突}\}$  为

$$Pr_m^n\{\text{冲突}\} = \begin{cases} 1, & n > m \\ 1 - \frac{m!}{m^n [m - n]!}, & n \leq m \end{cases} \quad (14)$$

我们定义节点的分布密度  $\rho$  为

$$\rho = \frac{n}{m} \quad (15)$$

于是得到分组发送成功的概率  $Pr_m^\rho\{\text{成功}\}$  满足

$$Pr_m^\rho\{\text{成功}\} = \begin{cases} 0, & \rho > 1 \\ \frac{m!}{m^{pm} (m - \rho m)!}, & 0 < \rho \leq 1 \end{cases} \quad (16)$$

考虑一个典型的 IEEE 802.11 退避场景,如图 1 所示. 假定在退避阶段  $i$  共有  $n (n < m)$  个节点均匀分布在窗口  $[0, m - 1]$  上. 在第  $j$  个时隙,共有  $k (k \in [2, \min(n, 2m - 2n)])$  个节点同时发送分组,产生冲突. 于是这  $k$  个节点全部进入退避阶段  $(i + 1)$  并且在新的竞争窗口  $[0, 2m - 1]$  上重新分布. 此时,对于处于退避阶段  $(i + 1)$  的节点来讲,其分布密度不再是均匀的. 新的竞争窗口也因此被分成两个区域:重叠区域 (Intersection Area) 以及独立区域 (Single Area). 这两个区域的分布密度分别为

$$\rho_{\text{Sin}} = \frac{k}{2m} \quad (17)$$

$$\rho_{\text{Int}} = \rho_i + \rho_{\text{Sin}} = \frac{n}{m} + \frac{k}{2m} \quad (18)$$

其中,  $\rho_i$  是退避阶段  $i$  的原始分布密度,  $\rho_{\text{Sin}}$  和  $\rho_{\text{Int}}$  分别代表退避阶段  $(i + 1)$  在独立区域以及重叠区域上的分布密度. 因此,对 DCF 协议而言,在新窗口上成

① 我们假设信道是无差错的,所有的分组丢失都由冲突造成.



功发送分组的概率满足

$$Pr_{DCF}\{\text{成功发送}\} = Pr\{\text{在重叠区域成功发送}\} \times Pr\{\text{在独立区域成功发送}\} \quad (19)$$

联立方程(16)~(19),我们得到 DCF 协议的冲突概率函数  $Pr_{DCF}(j)$  满足

$$Pr_{DCF}(j) = 1 - Pr_{DCF}\{\text{成功发送}\} = 1 - \frac{(m-j)!}{(m-j)^{\frac{2n+k}{2m}(m-j)} \left[ m-j - \frac{2n+k}{2m}(m-j) \right]!} \times \frac{(m+j)!}{(m+j)^{\frac{k}{2m}(m+j)} \left[ m+j - \frac{k}{2m}(m+j) \right]!} \quad (20)$$

对 CSR 算法而言,它将  $k$  个冲突节点均匀地分

布在一个完全独立的基本窗口上,避免了交叉冲突,此时只有同级冲突存在.假设

$$CW_0 = EW = m \quad (21)$$

可以推出 CSR 算法的冲突概率函数  $Pr_{CSR}(j)$  为

$$Pr_{CSR}(j) = 1 - \frac{(m-j)!}{(m-j)^{(m-j)\frac{n}{m}} \left[ m-j - \frac{n}{m}(m-j) \right]!} \times \frac{m!}{m^k (m-k)!} \quad (22)$$

我们固定窗口大小  $m$  以及冲突节点数  $k$ ,仅仅改变竞争节点总数  $n$ ,得到两种退避算法的冲突概率比较结果如图 5 所示.节点的增多势必加剧信道竞争,导致更高的冲突概率.然而在活跃节点数目相同的条件下,CSR 算法的冲突概率始终低于 DCF.

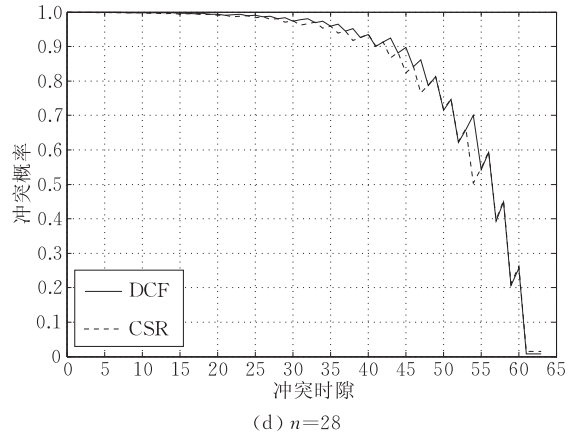
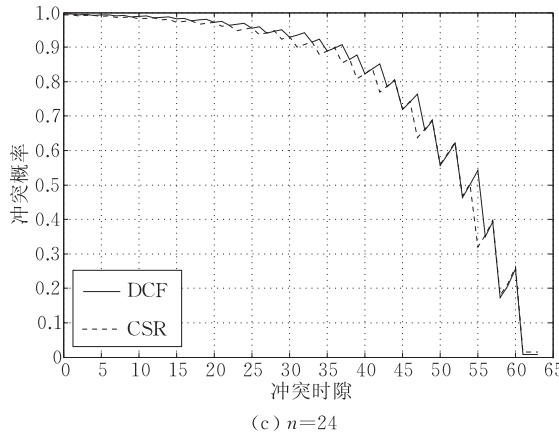
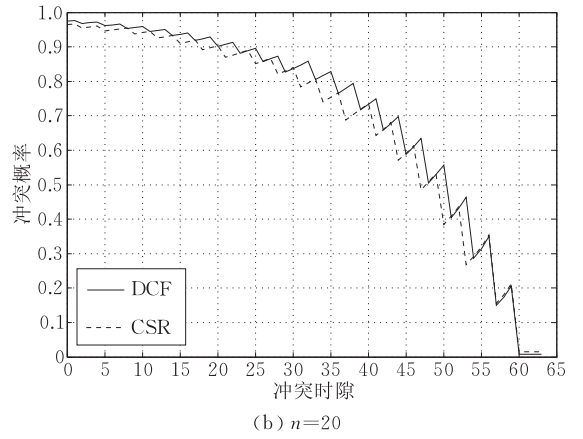
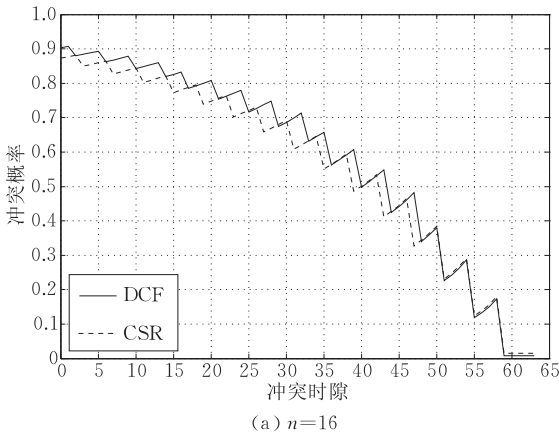


图 5 DCF 和 CSR 冲突概率比较( $m=64, k=2, n$  改变)

图 6 显示了冲突节点数目对冲突概率的影响规律. CSR 算法避免了交叉冲突,网络中只有同级冲突存在.在这种情况下,冲突节点数越多,发生同级冲突的概率也就越大.所以同 DCF 协议相比,CSR 算法的冲突概率下降相对缓慢.然而, $k$  个节点在同一时刻发生冲突的概率满足

$$Pr(k \text{ 个节点发生冲突}) = \frac{\binom{n}{k} (m-1)^{n-k}}{m^n} \quad (23)$$

图 7 给出了冲突发生概率与冲突节点数之间的关系.可以看出,随着冲突节点数的增多,冲突发生概率急剧下降并趋于 0.也就是说,多个节点同时发送分组导致冲突的概率是很小的.根据实际推断原理<sup>[12]</sup>: 概率很小的事件在一次试验中实际上几乎是不发生的.因此我们完全可以忽略在 CSR 算法中多节点同时冲突对系统性能造成的影响.

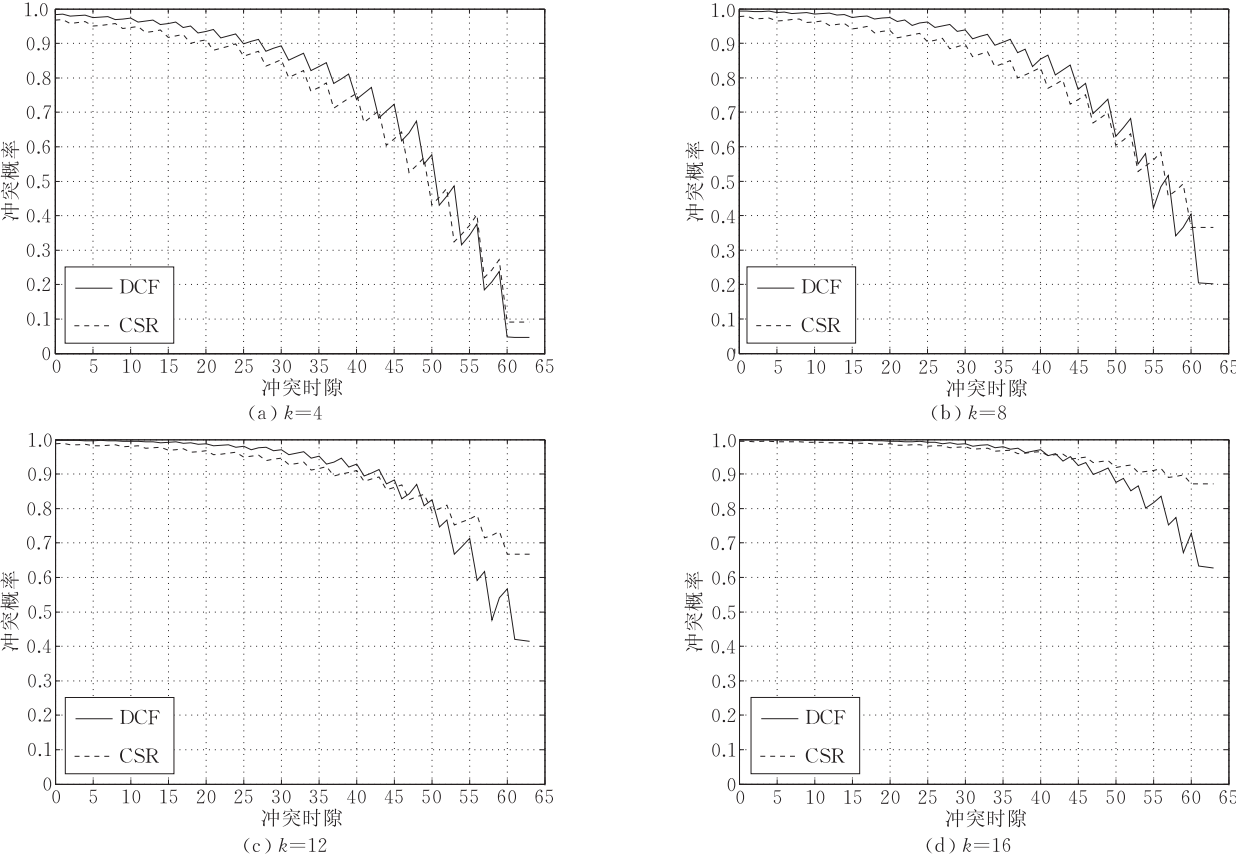


图 6 DCF 和 CSR 冲突概率比较( $m=64, n=20, k$  改变)

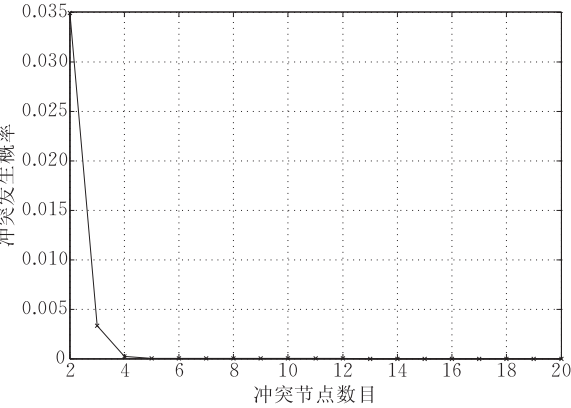


图 7 冲突节点数目对冲突发生概率的影响( $m=64, k=20$ )

在以上分析中,我们考虑的仅仅是一次状态转换过程中冲突概率的变化.事实上,经过多轮退避后,CSR 算法能够将节点依次顺序分布在时间轴上,通过竞争窗口的离散化消除交叉冲突,同时保障成功节点的后续传输.而 IEEE 802.11 DCF 却不得不面对接连不断的交叉冲突.可以想象,在这种情况下,CSR 算法的冲突解析性能更好.

3.4 基本窗口大小

一个合格的退避算法应该能够同时解析两种不同类型的冲突.我们知道,CSR 算法消除了交叉冲

突,网络中只有同级冲突存在,并且在 CSR 的调度之下,同级冲突主要发生在基本窗口上<sup>①</sup>.如果至少两个节点同时发送分组则会引发冲突.考虑到方程(23):多个节点同时冲突的概率可以忽略不计,因此此处我们仅仅考虑两个节点发生冲突时的情况,于是得到同级冲突概率满足

$$Pr\{\text{同级冲突}\} = \frac{1}{\text{sizeof(基本窗口)}} = \frac{1}{m} \quad (24)$$

此处  $m$  代表基本窗口大小.由此可见,基本窗口越大,发生同级冲突的概率越小.

同时,空闲时隙的数学期望  $E(T_{\text{Idle}})$  可以表示为

$$\begin{aligned} E(T_{\text{Idle}}) &= \sum_{i=0}^{m-1} i \times Pr\{BC=i\} \\ &= \sum_{i=0}^{m-1} i \times \frac{\binom{2}{1}^{(m-i)}}{m^2} = \frac{2}{m^2} \sum_{i=0}^{m-1} (m-i)i \end{aligned} \quad (25)$$

图 8 描述了空闲时隙的数学期望与基本窗口大小之间的关系.它是一个单调递增函数,基本窗口越大,平均空闲时间越长.

<sup>①</sup> 发生在初始竞争窗口上的同级冲突相对来说很少,可以忽略.

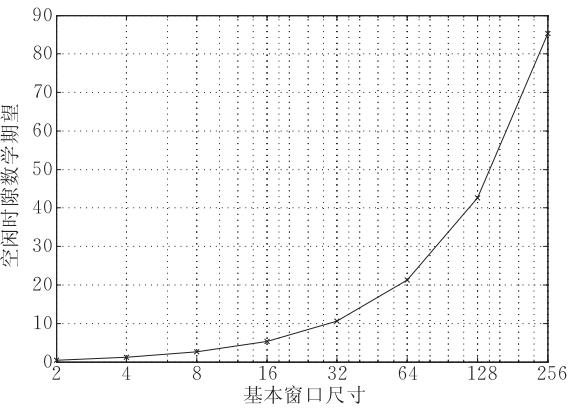


图 8 空闲时隙数学期望与基本窗口大小的关系

对比方程(24)和(25)我们知道,基本窗口的长度同时控制着空闲时隙的开销以及同级冲突的概率,而二者又是互相矛盾的,因此需要进行权衡,选择一个合适的窗口大小.

4 性能评估

4.1 仿真模型

我们利用 ns-2<sup>[13]</sup>进行仿真实验,比较 CSR, GDCF 和 IEEE 802.11 DCF 协议的性能. 仿真基于 DSSS 规范,实验参数如表 1 所示.

表 1 实验参数			
参数	值	参数	值
SIFS	10μs	CW <sub>min</sub>	32
DIFS	50μs	CW <sub>max</sub>	2048
Slot Interval	20μs	CW <sub>0</sub>	32
Preamble Length	144bit	EW	16
Channel Bit Rate	2Mbps	Packet Size	1024Byte

实验模拟 WLAN 环境,只设一个目的节点,即 AP. 源节点随机分布在一个 100m×100m 区域内,节点传输半径为 150m. 系统具有饱和的流量,采用基于 CBR 的 UDP 数据流进行通信.

4.2 结果和分析

由于分组传输失败对吞吐量和延迟有较大影响<sup>[14]</sup>,因此我们需要考察算法对分组冲突的解析能力,可以通过衡量网络中的分组冲突次数得到体现. 另外,吞吐量、延迟和公平性是考量退避算法性能的主要指标<sup>[4,15]</sup>,因此拟从以上几个方面对 3 个算法进行综合评估. 此外,还考察了基本窗口尺寸对 CSR 算法性能的影响.

(1) 平均冲突次数

我们定义平均冲突次数 (Average Collision Times, ACT) 为通信过程中单位时间内发生的分组冲突次数. 实际上,ACT 可以反映算法的冲突解析

效率. 我们改变网络中的活跃节点数,记录分组冲突次数,结果如图 9 所示. 随着竞争节点数目的增加,信道竞争状况加剧,导致冲突次数的增加. 从图中可以看出,CSR 的冲突次数是 3 个算法中最少的. 这是因为 CSR 通过严格的分时策略完全消除了交叉冲突,同时保障了成功节点的后续传输. 可以想象,在没有新节点加入的情况下,在 CSR 算法的调度下,系统最终会收敛到一个稳定状态. 此时,节点依次顺序接入信道,网络中没有任何冲突发生. 因此,图中 DCF 曲线和 CSR 曲线之间的差值就代表了网络中的交叉冲突次数. 显然,同同级冲突相比,网络中的交叉冲突更加普遍.

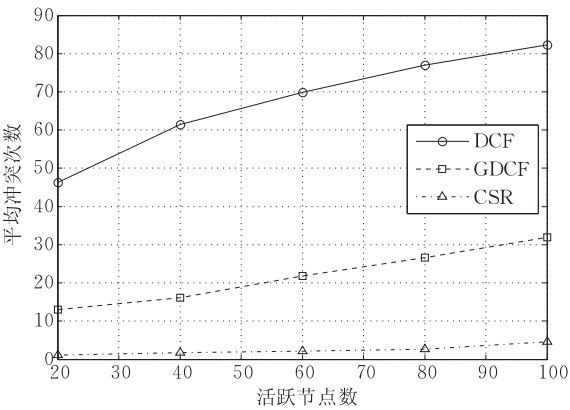


图 9 平均冲突次数比较结果

(2) 吞吐量

图 10 描述了算法的归一化吞吐量<sup>[16]</sup>比较结果. 对 DCF 协议而言,冲突概率随着节点数目的增多逐渐增大,最终造成吞吐量的下降. GDCF 则通过一种比较平缓的方式改变节点的竞争窗口,更加客观地反映网络的当前竞争状况,从而能够获得一定的性能提升. 而 CSR 算法则通过竞争窗口的离散化完全消除了网络中占主导地位的交叉冲突,从而提高了分组成功发送的概率,增大了系统吞吐量.

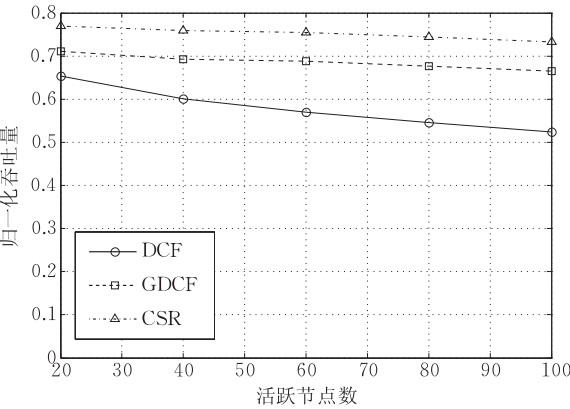


图 10 归一化吞吐量比较结果



(3) 分组延迟

从图 11 可以看出,随着节点数目的增加,CSR 算法延迟增加的幅度最小. DCF 频繁的冲突使得节点的空闲等待时间增多, GDCF 则由于窗口变化方式比较保守,对分组延迟也有一定影响. 在 CSR 中,虽然竞争窗口以基本窗口大小为步长逐步增加,但同时分组的冲突概率也大大降低,因此可以获得更低的分组延迟.

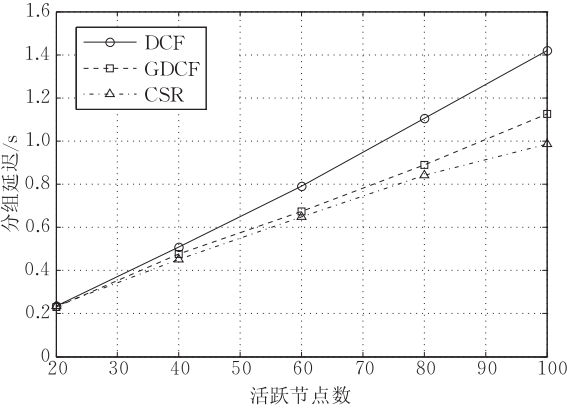


图 11 分组延迟比较结果

(4) 公平指数

公平指数 (Fairness Index, FI)<sup>[17]</sup> 定义如下:

$$FI = \frac{(\sum_{i=1}^n T_i / \phi_i)^2}{n \times \sum_{i=1}^n (T_i / \phi_i)^2},$$

其中,  $n$  表示数据流数目,  $T_i$  代表流  $i$  的吞吐量,  $\phi_i$  是流  $i$  的权重 (假设所有节点权重相同). 显然  $FI \leq 1$ , 且  $FI$  越趋近于 1, 公平性越好.

从图 12 中可以清楚看到: CSR 算法的公平性最好. 随着节点数目的增多, CSR 的  $FI$  曲线基本维持不变, GDCF 和 DCF 的曲线均有一定跌落. 这是由于在 CSR 算法的调度下, 各竞争节点依次顺序接

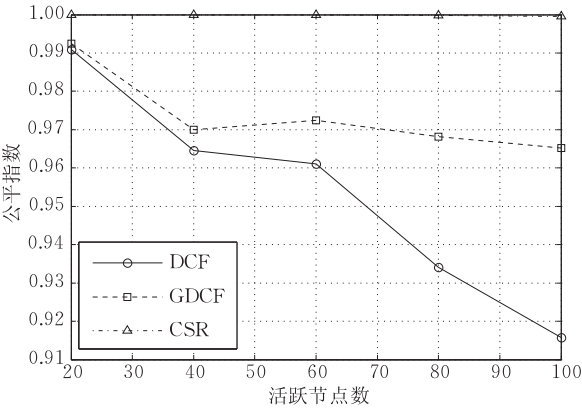


图 12 公平性比较结果

入信道, 节点每次发送后, 不管发送成功与否, 都需等待一轮周期过后才能再次接入信道, 从而为其它节点提供了接入机会, 保障了公平性.

(5) 基本窗口大小对算法性能的影响

正如前面提到, 基本窗口的大小对 CSR 算法的性能有着重要影响, 它同时决定着同级冲突的概率和分组延迟的大小. 我们选择 40 个节点进行实验, 通过改变基本窗口的大小观察协议性能的变化.

由于 CSR 中只存在同级冲突, 而大的窗口范围能够有效减少同级冲突, 因此随着基本窗口尺寸的增大, 冲突次数逐渐减小, 如图 13 所示. 由于 CSR 能够有效解析分组冲突, 因此就算采取最小的基本窗口大小 2, 平均冲突次数仍然在可以接受的范围内; 同时, 增大的窗口引入了更多的分组延迟, 且不同大小窗口引入的延迟相差较大 (如图 14 所示), 所以在这种综合作用下, 从图 15 中可以看出, 系统的吞吐量随着窗口的增大而减少.

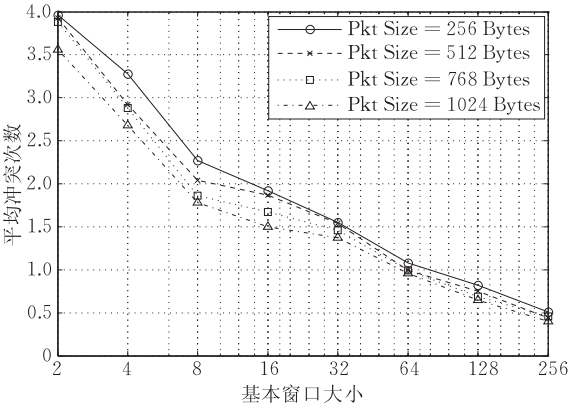


图 13 基本窗口大小对 CSR 算法冲突次数的影响

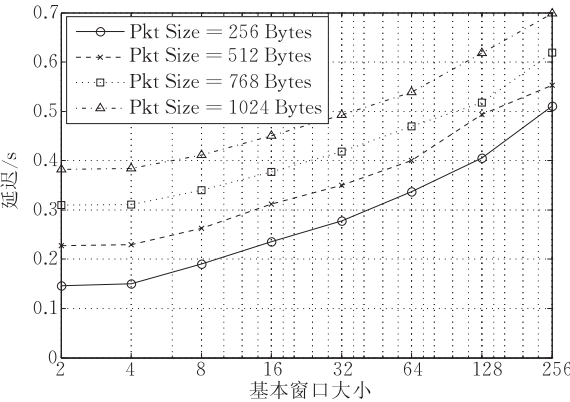


图 14 基本窗口大小对 CSR 算法分组延迟的影响

因此, 在应用 CSR 算法时, 可以忽略不同基本窗口大小带来的冲突次数差异, 选取较小的基本窗口尺寸以获得较小的分组延迟和较高的系统吞吐量.

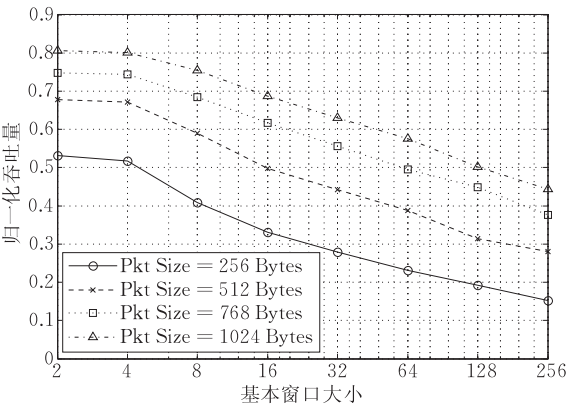


图 15 基本窗口大小对 CSR 算法吞吐量的影响

5 结 论

本文提出了分布式的基于竞争的 MAC 协议中的冲突划分标准,根据冲突节点所处退避阶段的不同,将冲突分为交叉冲突和同级冲突.随后,通过理论分析以及仿真实验证明:网络中的交叉冲突是普遍存在的.传统的退避算法并没有区分这两种不同类型的冲突,而是采用相同的方式进行处理,对系统性能造成了一定影响.我们认为,对于不同的冲突类型应该予以区别对待,因此提出了冲突顺序解析算法 CSR. CSR 采用冲突顺序转化机制,根据冲突发生的时间顺序为冲突节点动态分配独立的竞争窗口,从而实现了退避空间的离散化,消除了交叉冲突,使得系统中只有同级冲突存在,实现了从交叉冲突到同级冲突的有效转化;同时,通过设置适合的基本窗口大小,在同级冲突概率和分组延迟之间取得折衷.仿真实验表明,同传统退避算法相比,CSR 算法在冲突次数、吞吐量、延迟以及公平性方面均有较大提升.

参 考 文 献

[1] Kwak Byung-Jae, Song Nah-Oak, Miller Leonard E. Performance analysis of exponential backoff. *IEEE/ACM Transactions on Networking*, 2005, 13(2): 343-355

[2] Bianchi Giuseppe. Performance analysis of the IEEE 802.11 distributed coordination function. *IEEE Journal on Selected Areas in Communications*, 2000, 18(3): 535-547

[3] Bharghavan Vaduvur, Demers Alan, Shenker Scott, Zhang Lixia. MACAW: A media access protocol for wireless LAN's // *Proceedings of the ACM SIGCOMM*. London, 1994: 212-225

[4] Haas Zygmunt J, Deng Jing. On optimizing the backoff interval for random access schemes. *IEEE Transactions on Communications*, 2003, 51(12): 2081-2090

[5] Aad Imad, Ni Qiang, Barakat Chadi, Turletti Thierry. Enhancing IEEE 802.11 MAC in congested environments//*Proceedings of the 4th Workshop on Applications and Services in Wireless Networks*. Boston, USA, 2004: 82-91

[6] Wu Haitao, Cheng Shiduan, Peng Yong, Long Keping, Ma Jian. IEEE 802.11 distributed coordination function (DCF): Analysis and enhancement//*Proceedings of the IEEE ICC*. New York, USA, 2002: 605-609

[7] Song Nah-Oak, Kwak Byung-Jae, Song Jabin, Miller Leonard E. Enhancement of IEEE 802.11 distributed coordination function with exponential increase exponential decrease back-off algorithm//*Proceedings of the IEEE VTC*. Orlando, Florida, USA, 2003: 2775-2778

[8] Pang Qixiang, Liew Soung C, Lee Jack Y B, Leung Victor C M. Performance evaluation of an adaptive backoff scheme for wlan. *Wireless Communications and Mobile Computing*, 2004, (4): 867-879

[9] Deng Jing, Varshney Pramod K, Haas Zygmunt J. A new backoff algorithm for the IEEE 802.11 distributed coordination function//*Proceedings of the Communication Networks and Distributed Systems Modeling and Simulation*. San Diego, USA, 2004

[10] Wang Chonggang, Li Bo, Li Lemin. A new collision resolution mechanism to enhance the performance of IEEE 802.11 DCF. *IEEE Transactions on Vehicular Technology*, 2004, 53(4): 1235-1246

[11] ANSI/IEEE 802.116. The Institute of Electrical and Electronic Engineers, New York, 1999

[12] Sheng Zhou, Xie Shi-Qian, Pan Cheng-Yi. *Probability Theory and Statistics*. Beijing: High Education Press, 1989 (in Chinese)

(盛骤, 谢式千, 潘承毅. 概率论与数理统计. 北京: 高等教育出版社, 1989)

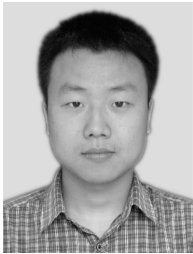
[13] Fall Kevin, Varadhan Kannan. The ns manual. UC Berkeley, LBL, USC/ISI, and Xerox PARC, 2002

[14] Xiao Yongkang, Shan Xiuming, Ren Yong. Game theory models for IEEE 802.11 DCF in wireless ad hoc networks. *IEEE Radio Communications*, 2005, 43(3): S22-S26

[15] Barowski Yawen, Biaz Saad, Agrawal Prathima. Towards the performance analysis of IEEE 802.11 in multi-hop ad-hoc networks//*Proceedings of the IEEE WCNC*. New Orleans, USA, 2005: 100-106

[16] Yang Xue, Vaidya Nitin H. Explicit and implicit pipelining for wireless medium access control//*Proceedings of the IEEE VTC*. Orlando, Florida, USA, 2003: 1427-1431

[17] Jain Raj, Duresi Arjan, Babic Gojko. Throughput fairness index: An explanation. *ATM Forum*, Document Number: ATM Forum/990045, Feb. , 1999



**ZHANG Qi-Fei**, born in 1977, Ph. D. , lecturer. His main research interests include wireless ad hoc networks and wireless sensor networks.

**SUN Bao-Lin**, born in 1963, Ph. D. , professor. His main research interests include high-performance computer networks and wireless communication networks.

**GUI Chao**, born in 1965, associate professor. His main

research interests include wireless ad hoc networks and wireless sensor networks.

**LIU Wei**, born in 1977, Ph. D. , associate professor. His main research interests include wireless ad hoc networks and cognitive radio networks.

**CHENG Wen-Qing**, born in 1965, Ph. D. , professor and Ph. D. supervisor. Her main research interests include wireless sensor networks.

**YANG Zong-Kai**, born in 1963, Ph. D. , professor and Ph. D. supervisor. His main research interests include NGI and wireless ad hoc networks.

**Background**

This work is supported by the National Natural Science Foundation of China under grants No.60572049 and No. 60602029, the Scientific Research Projects of Hubei Education Department under grants No.D20081904, No. T200902 and No. Q20091903, the Postdoctoral Science Foundation of China under grant No. 20070410955.

The IEEE 802.11 protocol has become the predominant technology for wireless local area networks, which can be a major reference to access mode research on wireless ad hoc networks. One of the key components of IEEE 802.11 is the Medium Access Control (MAC) protocol that primarily determines its performance.

The work in this paper investigates collision resolution

issues of IEEE 802.11 networks and then proposes a collision classification criterion. It classifies the collisions into two categories and then proposes Collision-Sequential Resolution (CSR) to resolve the two collisions with different policies. The extensive simulation results show that the CSR algorithm consistently excels traditional MAC protocols in terms of collision times, throughput, delay and fairness. Moreover, the CSR algorithm can be realized based on the IEEE 802.11 DCF with no significant modifications required. It is a novel exploring for collision resolution in IEEE 802.11 wireless networks and can be used to resolve medium access problems in wireless ad hoc networks and sensor networks.