Carrier-Sence Multiprle Access (CSMA) Protocols

Leonidas Georgiadis

February 13, 2002

1 Introduction

Communication of information between two or more parties takes place over a variety of physical media called channels. Such channels can be simple twisted pair cables, coaxial and optical cables, or the free space. Sometimes the channel is dedicated to a specific transmitter-receiver pair. This may be the case when two parties establish a telephone conversation over a dedicated cable. Channels of this type are called **point-to-point**. There are situations, however, where multiple users need to have access to the same channel. The most familiar one is human speech communication. When a number of humans are located in the same room they all use the same channel, the atmosphere, for their conversation exchange. Computer Local Area Networks (LANs) is another example: a common approach in this case is to attach a number of computers to the same cable as in Figure 1. Hence, each computer can listen to the transmission of every other computer attached to the same cable. For a third example, consider Satellite communication. As shown in Figure 2 a number of terminals need to communicate between each other but due to physical obstacles they cannot all listen to each other directly. Instead, each terminal first sends the information to the satellite through the upstream channel. The satellite listens to the upstream channel, receives the information and then retransmits to the downstream channel, to which all terminal can listen. Hence the upstream channel needs to be accessed by all terminals. Channels of this type are called **multiple-access**.

If the terminals in a multiple-access channel are left unchecked so that they can transmit information whenever the need to do so, then the possibility arises that more than one terminals attempt to use the channel at the same time. In such a situation, the concurrently transmitted messages interfere with each other and generally cannot be received correctly by the intended receivers. Hence, a fundamental issue in multiple-access channel communication is how to coordinate the transmitting terminals in order to avoid

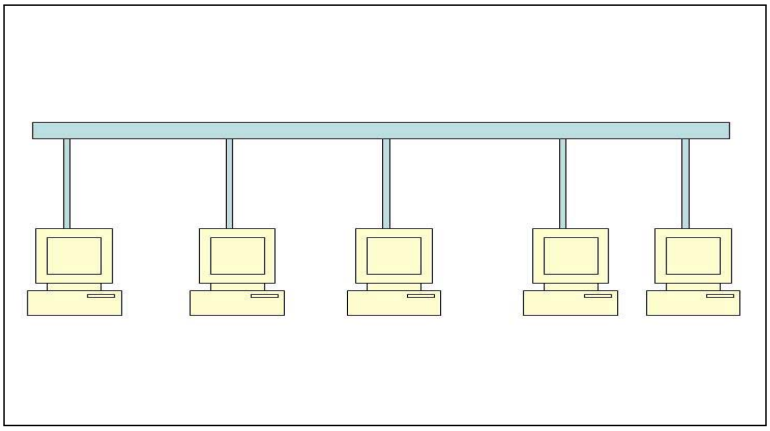

Figure 1: A Local Area Network

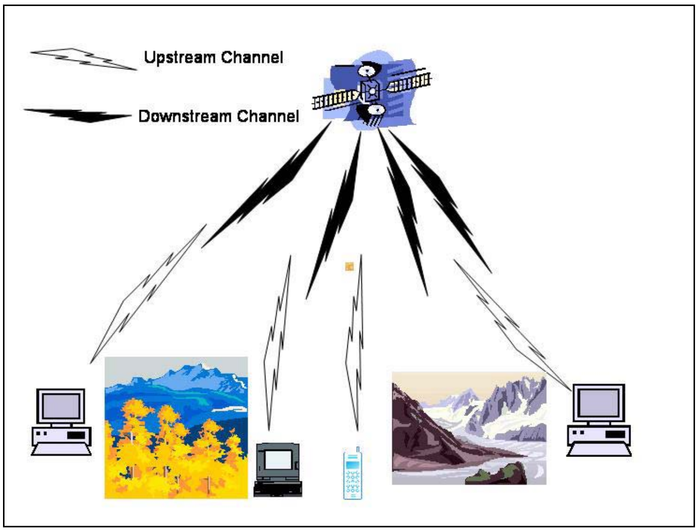

Figure 2: Satellite Communications

or recover from the interference that may result because of concurrent transmissions. The mechanisms by which this is achieved come by the name **multiple-access protocols**.

The simplest way to address the coordination problem in multiple-access communication is to avoid concurrent transmissions altogether. To be more specific we must make certain assumptions about the manner in which transmission of information takes place. First, as is very common today, we assume that all information whether sound, picture or text is transformed to a sequence of bits, 0 or 1, and that each terminal needs to transmit this sequence of bits to the receiving terminal - the receiver knows how to recover the original information from the received sequence of bits . Let us assume further that the sequence of bits is subdivided into groups called *packets*, and that the transmitter needs to transmit one packet at a time to the receiver. All packets contain the same number of bits B. If bits can be transmitted over the channel at a rate of R bits per second (bits/sec or bps), then each packet takes $T = B/R$ seconds to be transmitted. We refer to T as the "length" of the packet.

We are now ready to describe the protocol by which access to the channel is free of concurrent transmissions. We divide time into fixed intervals of length T called time *slots,* see Figure 3. Hence each slot fits exactly one packet. Let the number of terminals that can potentially use the channel be n. We group the time slots into *frames* where each frame contains n consecutive time slots. Terminal i is allowed to transmit in the ith time slot of each frame. The protocol just described is called Time Division Multiple Access (TDMA) protocol. Since slots are allocated exclusively to each user, no interference occurs and packets are transmitted successfully. Note that there is still a possibility that the packet may be received in error due to noise that naturally exists on the channel, but this is a lower level issue which is addressed by methods that are beyond the scope of the current discussion.

The TDMA protocol in effect divides the channel into n point-to-point channels. While simple, the protocol has some serious disadvantages. First, if a terminal does not have packets to transmit, the slots allocated exclusively to it cannot be used be anybody else, even if other terminals have a large number of packets to transmit and could use these slots. The second disadvantage is related to packet delays. Since the time interval between two successive slots during which terminal i can transmit is n time slots, a packet generated randomly at a terminal will take on the average $n/2$ time slots to be transmitted, a delay that can be very large if the number of terminals in the system is large. This will happen no matter whether the rest of the terminals have packets to transmit or not.

The disadvantages of TDMA are due to the fact that a terminal can transmit only during the slots allocated to it, even if other terminals are inactive. What if we dispense with the idea of allocating slots exclusively to transmitters? In fact, what if we take the exact opposite approach and allow a terminal

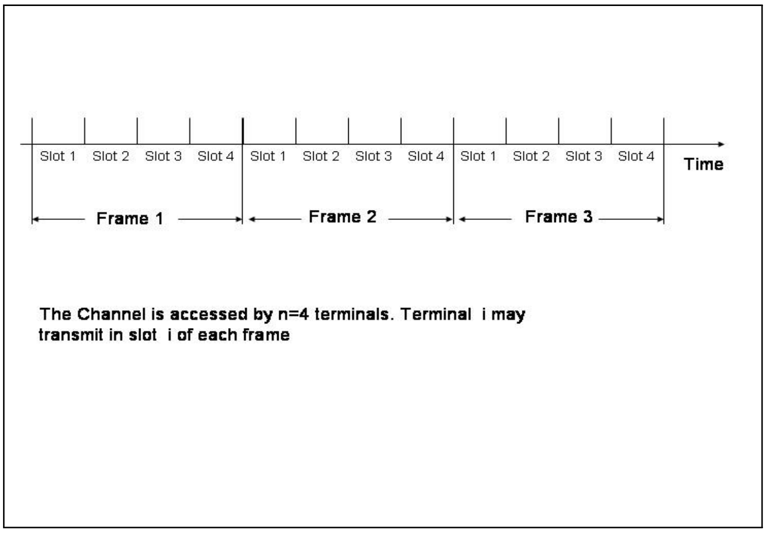

Figure 3: The TDMA protocol

to transmit in any slot when it has packets to transmit? In this case, if no other terminal has packets to transmit, then the given terminal can transmit a large number of packets with very low delay. However, if more than one terminals wish to send packets in the same slot then a "collision " will occur and no message will be received correctly. In this case one must specify how the terminals will react and cooperate in order to make sure that the packets are eventually delivered to their intended destinations. The simplest idea is to instruct the terminals to retransmit their collided packets. However, if more than one terminals pick again and again the same slot for retransmission the packets will keep colliding and will never be transmitted successfully. There are various methods to avoid this situation. We will concentrate on the most prevalent method encountered in practice, randomized retransmissions. If collisions occur then the terminals whose packets collided pick randomly some future time slot for retransmission. Hence, while collision may again occur, it is hoped that eventually the transmitting terminals will each pick different slots for transmission and thus their packets will be delivered successfully to their intended destination.

The algorithm just described comes by the name ALOHA protocol and will be described in more detail in Section 2. This algorithm constitutes the basis for the development of Carrier Sense Multiple Access (CSMA) protocol which takes advantage of certain channel features and transmitter capabilities in order to provide improved performance. The CSMA protocol will be described in Section 3.

2 The ALOHA protocol

The ALOHA protocol was designed by Abramson [1] to provide radio communication between several terminals scattered at various places over the islands of Hawaii. The terminals were sending their data packets to a central station over a common channel (the upstream channel). The central station was then retransmitting the packets to another channel (the downstream channel) that could be listened to by all the terminals. The situation is similar to the one described in Figure 2. Collisions could occur at the upstream channel if more than one terminals were attempting to transmit their packets. If this happened, the central computer was informing all the terminals that a collision had occurred.

There are two versions of the ALOHA protocol, slotted and unslotted. Slotted ALOHA requires time to be divided in time slots and terminals to transmit their packets at the beginning of each slot. Unslotted ALOHA permits the stations to transmit their packets at any time. The retransmission policy in case of collision is essentially the same for both protocols. In the next two sections we examine these two variants of the ALOHA protocol. Unslotted ALOHA was the precursor of Slotted, but it will be more instructive and simpler to concentrate on the Slotted ALOHA first.

2.1 Slotted ALOHA

Let us provide a model for this protocol. As in Section 1 the channel is divided in time slots. Terminals are synchronized to transmit their packets at the beginning of a time slot. At the end of each time slot terminals that transmitted their packets during that slot are informed whether there was a successful transmission or a collision in the slot. If the packet that a terminal transmitted collides with another packet, then the terminal attempts a retransmission in the next slot with probability p and defers for the end of the next slot with probability $1-p$. In case of deferral, at the end of the next slot the terminal attempts again retransmission with probability p and defers with probability $1-p$. Figure 4 shows an example of the operation of the ALOHA protocol. At slot 1 three terminals attempt to transmit their packets and there is a collision. Hence all three terminals will attempt to retransmit their packets. No terminal chooses to retransmit at slot 2 which is thus idle. Terminals a and b attempt to retransmit at slot 3 and hence there is again a collision. Terminal b is the only one attempting retransmission at slot 4 and its transmission is successful. The transmissions from terminals a and c collide again in slot 6, but they eventually pick different slots for retransmission and their packets are transmitted successfully in slots 8 and 9. Note that other terminals may become active (i.e., they may generate a new packet for transmission) while the retransmission process takes place. These terminals may cause additional

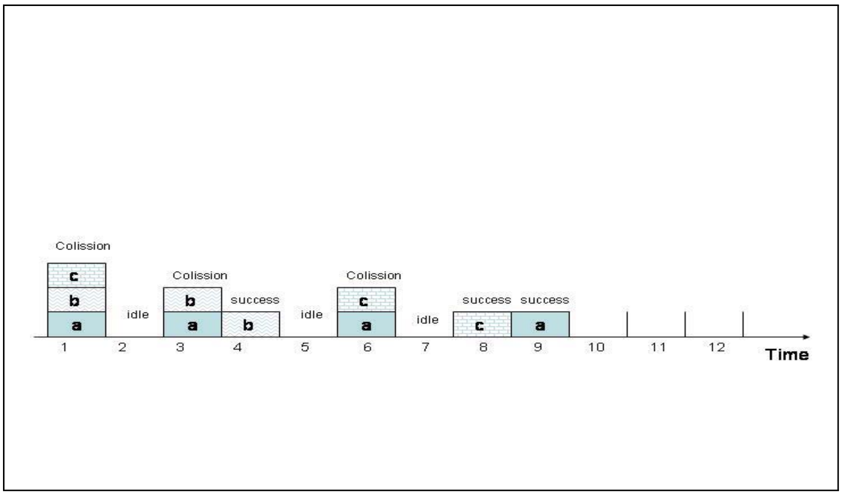

Figure 4: The operation of the ALOHA protocol

collisions. For example, if terminal d generates a new packet and attempts to transmit it in slot 8, an additional collision will occur. In the network designed by Abramson, all terminals (not only those that transmitted their packets) can listen to the downstream channel and hence can be informed about the status of the transmission at the end of the current slot, i.e., whether there was no transmission, a successful transmission or a collision during the slot. However, the ALOHA protocol does not make use of this extra information that a terminal can have.

The protocol just described has the desirable property that packets are not delayed at all if only one terminal needs to transmit at a given time slot. What happens, however, when more than one terminals attempt transmission? As the example in Figure 4 shows, in this case there will be collisions that will be followed by retransmission attempts. This results in two inefficiencies: a) slots may be wasted due to collisions and b) slots may remain idle even though there are terminals that need to transmit their packets; the latter will happen if all packets that attempt retransmission are deferring in the current slot and no new packets are generated. It is important therefore to know the useful information that can come out of the channel. An appropriate measure for this information is the average number of successfully transmitted packet, S, per slot. We refer to S as the *throughput* of the channel.

Next we provide a method for evaluating S. We need to make first an assumption regarding the statistics of new packets generation process: the number of new packets, K, generated for transmission during a time slot, is a Poisson random variable with rate λ packets/slot. That is, the probability that $K = k$ is given by

$$\Pr(K = k) = e^{-\lambda} \frac{\lambda^k}{k!}. \tag{1}$$

This model of packet generation is called “infinite population model” because it implies that the number of terminals in the system is potentially infinite (the probability that K is any large number is nonzero) and that each terminal generates packets infrequently, so that packet queues are not formed at the terminals. It is used because it is simple, a good approximation when the number of terminals is large and provides some important insights.

There are two sets of terminals that may attempt transmission at the beginning of a time slot. Those that generate new packets, and those whose generated packets have collided in some previous slot and attempt retransmission. In the latter case we say that the packets are “backlogged ”. Assume that the system can reach steady state and let M be the random number of packets (newly generated and backlogged) transmitted in a given slot in steady state. Denote G packets/slot the average value of M. Since M includes both newly generated and backlogged packets, it holds $G > \lambda$. Observe that a successful transmission occurs only when $M = 1$. Indeed, if $M \neq 1$ then either the slot will be idle (if $M = 0$) or there will be a collision in the slot (if $M \geq 2$). Therefore, by the definition of S we have

$$S = 0\,\Pr(M \neq 0) + 1\,\Pr(M = 1) = \Pr(M = 1)\,.$$

Hence, if we knew the statistics of M,then we would be able to evaluate S. The exact evaluation of the statistics of M is complicated. To simplify the situation we make the additional assumption that M is a Poisson random variable. Since the rate of M is G, we have from (1).

$$S = \Pr(M = 1) = e^{-G}G. \tag{2}$$

In Figure 5 we plot S as a function of G given by (2). It can be shown that the maximum value of S is $1/e \approx 0.368$ and is obtained at $G = 1$. Hence the maximum channel throughput of the slotted ALOHA protocol is 0.368 packets/slot. A conspicuous feature of the plot in Figure 5 is that a given channel throughput is achieved for *two* values of G a small, G_1, and a large G_2. The small value implies that the number of backlogged packets is small while for the large value this number is large. Clearly we would prefer to operate the system at the value G_1, but why two values appear and what is their meaning?

There are two flaws with the analysis presented above. First, the existence of steady state is assumed and second the probability distribution of M is assumed to be Poisson. For the Infinite Poisson model, both these assumptions turn out to be invalid! However, the derived bound on the achievable throughput is sill correct. A more detailed analysis of the system for finite number of users, which is beyond the scope

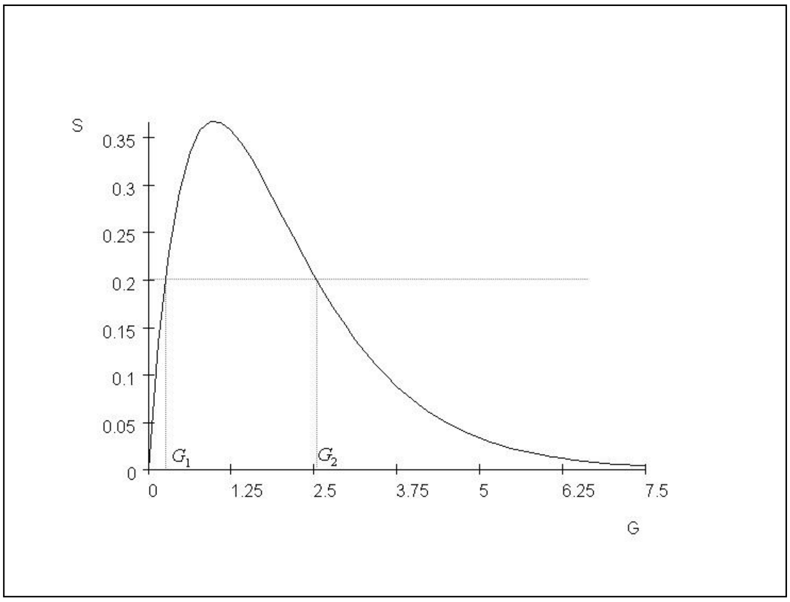

Figure 5: The Throughput of the ALOHA protocol

of this presentation, reveals that indeed the throughput of the system is at most $1/e$. Moreover it can be shown that the system behaves qualitatively as follows. There are long periods of time during which the number of backlogged packets in the system remains small and the system operates well inducing small packet delays. However, from time to time a large increase in the number of backlogged packets in the system will occur and system performance in terms of throughput and delay will degrade. Fortunately, it can also be shown that the time interval for the transition from the "good " state to the "bad " is generally very large. Hence this instability phenomenon of transiting from good to bad states is not usually a severe problem in real systems.

2.2 Unslotted ALOHA

In the previous section we assumed that the terminals are all synchronized to begin transmission of their packets at the beginning of each slot. If this feature is unavailable, the protocol can be easily modified to still operate. Indeed, the users can be allowed to transmit their new packets at packet generation time. If a collision occurs, then the terminal attempts a retransmission at a later randomly chosen time.

Let us evaluate the performance of the Unslotted ALOHA system. We adopt the infinite population model and the notation of Section 2.1. Taking into account the cautionary statements at the end of Section 2.1, let us assume the existence of steady state and that $M(\gamma)$, the number of terminal that attempt transmission in any time interval of length γT is a Poisson random variable with rate γG.

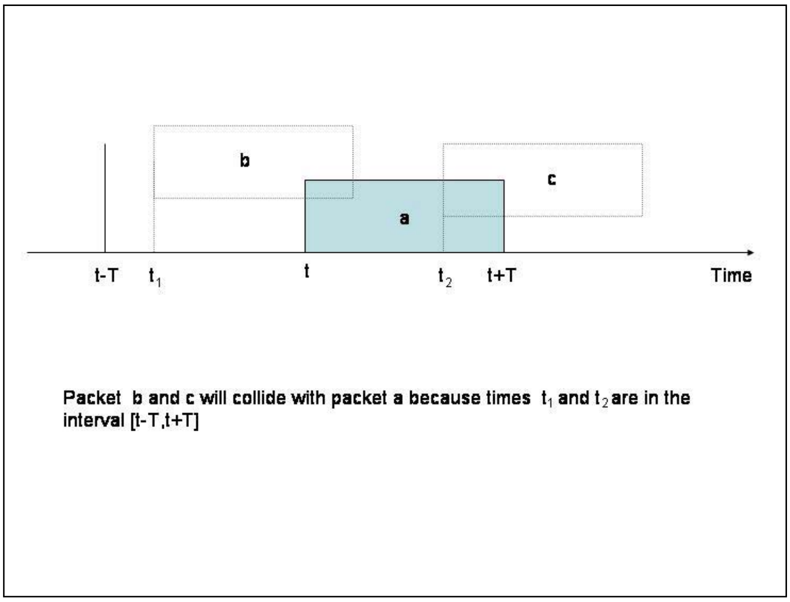

Figure 6: Possibility of colissions in the Unslotted ALOHA protocol

If terminal a begins transmission at time t , see Figure 6, its transmission will be successful if no other packet is begins transmission in the interval $[t-T, t+T]$. Since this interval has length $2T$, the probability that no packet (other than terminal a's packet) is transmitted in the interval $[t-T, t+T]$ is $P_s = P(M(2) = 0) = e^{-2G}$. We can interpret P_s as the proportion of attempted packet transmissions that are successful. Now, the rate (average number of packets per time T) by which packet transmissions are attempted is G and a proportion P_s of these transmissions are successful. Hence the rate of successful transmissions is

$$S = GP_s = Ge^{-2G} \tag{3}$$

From (3) it can be seen that the maximum throughput is $1/(2e)$ and is obtained for $G = 1/2$.

We see that the throughput of the Unslotted ALOHA is half the throughput of the Slotted one. However, Unslotted ALOHA does not require terminal synchronization. In any case, from the previous discussion we see that the throughput of both systems is much lower than one. Throughput one could be achieved if the terminals could be scheduled for transmission so that collisions are avoided. On the other hand, we have seen that the ALOHA protocol is very simple and distributed in the sense that the terminals operate independently of each other and require very small amount of feedback information to make their decisions. Moreover, the protocol induces very small packet delays when the system is lightly loaded. The question arises whether the throughput of the ALOHA protocol can be improved, while maintaining its desirable features. These considerations lead to the development of CSMA protocols which we discuss in the next section.

3 CSMA protocols

In this section we present the versions of CSMA protocols that have found wide application. One can think of the CSMA protocol as an evolution of ALOHA where certain terminal capabilities are exploited in order to attain improved performance. It turns out that in real systems the required terminal capabilities depend on the transmission media, i.e., whether communication takes place over wires - twisted pair, coaxial, optical - or through radio waves in the atmosphere - wireless communication. Accordingly we first discuss the CSMA and CSMA/CD protocols that are appropriate for wired communications and next examine the CSMA/CA protocol which is designed for wireless communications.

3.1 The CSMA and CSMA/CD protocols

As we saw in the previous sections, the throughput loss of the ALOHA protocol is due to the fact that slots are wasted due to collisions or remain idle while there are terminals having packets ready for transmission. Let us see whether we can improve this situation while maintaining the desirable features of the ALOHA system. The throughput of the system can be improved if

1. The likelihood of a collision is reduced

2. The time wasted transmitting garbled data when a collision occurs is reduced.

Consider the possibility of reducing collisions first. Let us assume that a terminal is able to listen to the channel and detect possible ongoing transmissions - busy channel. The ALOHA protocol can then be modified as follows. In case the terminal finds the channel busy, it defers transmission for a random time. Else it transmits its own packet. The protocol just described is called Carrier Sense Multiple Access Protocol. The term “Carrier Sense” signifies the capability of the terminal to listen to the channel and find out whether it is busy or not.

At first sight it seems that with CSMA we succeed in avoiding collisions altogether. Indeed, if all terminals transmit their packets only when the channel is not busy and pick a random retransmission time if they find the channel busy, then it seems that a collision will occur only when two or more terminals begin transmission simultaneously, an event that is quite unlikely. However, the situation is not as rosy as it seems, due to the finite time it takes for a signal to propagate from one terminal to

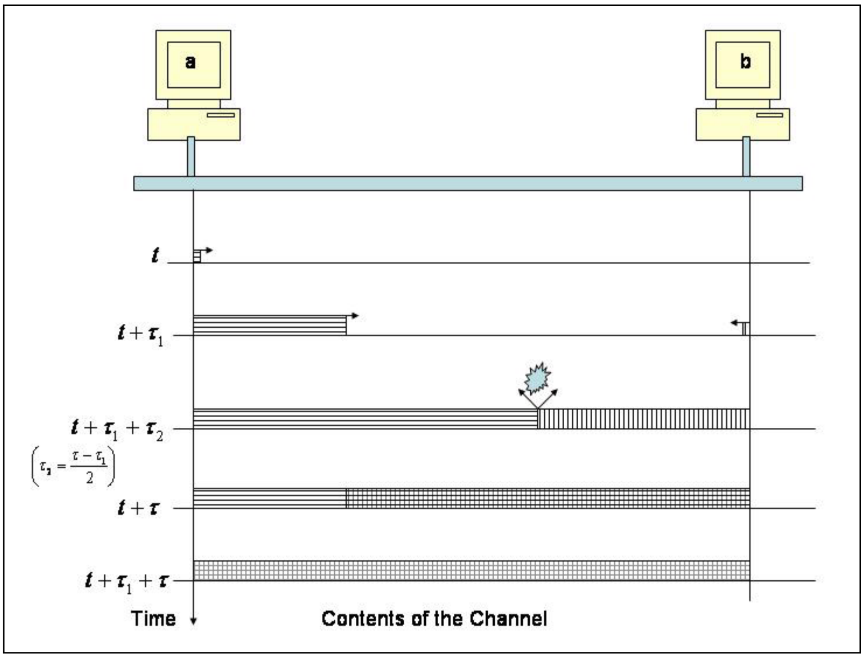

Figure 7: Collision occurence in CSMA protocol

another. Consider the example in figure 7. Assume that it takes τ seconds for a signal to be transferred from terminal a to b and vice versa. At time t terminal a sense the channel free and starts transmitting a packet. At time $\tau_1 < \tau$ terminal b sense the channel and finds it also free, although the packet from terminal a is well on its way on the channel. Terminal b starts transmitting its own packet, and $(\tau - \tau_1)/2$ seconds later the two packets begin to collide.

From the previous discussion we see that collisions will still occur with the CSMA protocol. However, we expect that the likelihood of a collision will indeed be reduced if the maximum signal propagation delay between two terminals in the system is small relative to the length of a packet. Indeed this is the case. It can be shown that the throughput of the CSMA protocol is approximately, for small τ/T,

$$S_{CSMA} \approx \frac{1}{1 + 2\left(\tau/T\right)^{1/2}}. \tag{4}$$

When $\tau \ll T$ the previous formula shows that S approaches one successful packet per packet duration time, i.e., the maximum possible.

Let us examine (4) more closely. If the length of the packet is B bits and the transmission rate at the channel is C bits/sec, then $T = B/C$. Therefore, we can rewrite (4) as

$$S_{CSMA} \approx \frac{1}{1 + 2\,(\tau C/B)^{1/2}}. \tag{5}$$

The channel propagation time, τ, is constant and independent of C and B. Therefore, if the network is extended to cover a wider area and as a result τ increases, then the throughput will be reduced. Assume next that we upgrade the channel to a higher transmission rate while maintaining the same arrangement of terminals (i.e. keep τ the same). What will happen to the channel throughput? We need to be careful here since throughput has been defined as the average number of successful packet transmissions per packet length T, and T changes as C varies and B remains constant. An appropriate measure in this case is the average number of successfully transmitted bits per second. This latter measure S^U_{CSMA} is simply related to S_{CSMA}, namely

$$S^U_{CSMA}(\text{bits/sec}) = \frac{SB}{T} = S_{CSMA}C \approx \frac{C^{1/2}}{1/C^{1/2} + 2\,(\tau/B)^{1/2}} \tag{6}$$

From (6) we see that the channel throughput in bits per second increases with C, however, the increase is proportional to $C^{1/2}$ and not C. In fact, *the throughput per channel transmission rate*, i.e., S^U_{CSMA}/C is equal to S_{CSMA}, which decreases as C increases. Also, as seen from (6), for constant C, S^U_{CSMA} increases as the packet length B increases. These considerations should be taken into account when deploying networks operating with the CSMA protocol.

We now turn our attention to the possibility of reducing the time wasted to collisions. Assume that a terminal is able to continue listening to the channel while it transmits its own packet. In case it detects that collision occurred, it interrupts its own transmission and attempts retransmission at a later time. Hence in general, if a collision occurs, a time interval smaller that the packet duration time will be wasted. In the example of Figure 7, terminals b and a will detect the collision at times $t + \tau$ and $t + \tau_1 + \tau$ respectively. The CSMA protocol where nodes are interrupting their transmissions when a collision is detected comes by the acronym CSMA/CD protocol - CD stands for Collision Detection. The throughput of the CSMA/CD protocol for τ/T small is given approximately by

$$S_{CSMA/CD} \approx \frac{1}{1 + 5\,(\tau/T)}. \tag{7}$$

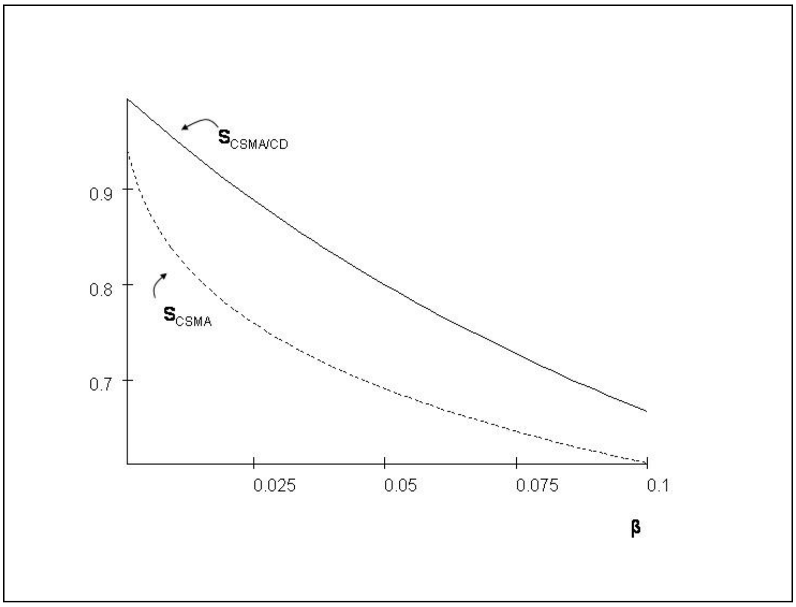

Figure 8: Comparison of CSMA and CSMA/CD protocols

Figure 8 shows the throughput of the CSMA and CSMA/CD protocols for various values of $\beta = \tau/T$. We see that both protocols can achieve much higher throughput than the original ALOHA system when β is small. In fact the throughput can be close to 1. We also see that for the same β, CSMA/CD can achieve significantly better throughput that CSMA. This improvement if due of course to the fact that less time is wasted to collisions in CSMA/CD systems than in CSMA.

Up to now we have specified that in case a terminal encounters a collision, it attempts a retransmission at some later random time. What is a good method of selecting such a random time? We discuss here one method that has found wide application. Intuitively, the random retransmission time, R, should depend on the number of the backlogged users: the larger the number of backlogged users, the more spread-out the distribution of R should be so that the likelihood of avoiding new collisions is reduced. Of course, R should not be too spread out because then terminals will attempt retransmissions rarely and a large portion of time will be left unused. In fact this intuition is correct and can be shown that if the number of backlogged terminals is known and the choice of R is based on this number, the instabilities of the CSMA protocol can be eliminated. However, in real systems the number of backlogged users is generally not known. As an alternative, a terminal may try to obtain an estimate of the number of the backlogged users based on its retransmission history. This estimate should increase as the number of collisions encountered during the attempt to transmit a packet increases. Hence the distribution of R should become more spread out as the number of such collisions increases.

The previous discussion justifies the following retransmission strategy: if a terminal encounters k collisions during the attempt to transmit a packet, then it attempts a retransmission at time R which is uniformly distributed in the time interval $(0, A2^k)$, where A is a constant. There are various variants of this strategy, however, the main characteristic of all of them is that the "spreading" of R increases exponentially with k. For this reason, this retransmission strategy is known as *exponential backoff.*

3.1.1 Applications of CSMA/CD Protocol

The foremost application of the CSMA protocol is in the technology that connects computer terminals located within a company, an institution, university campus etc., using wires. Such a technology is known as Local Area Network (LAN) technology. Over the past years there appeared several LAN technologies, but the first and by far the most prevalent one is the Ethernet technology also referred to as the IEEE 802.3 LAN technology.

The Ethernet technology was developed in the mid seventies by Bob Metcalfe and David Boggs. Since then, although it faced challenges by several alternative LAN technologies (token ring, FDDI, ATM), it still dominates the marketplace. One of the reasons for this success is that the hardware required for its deployment became very cheap, which in turn is due to the large production volume and to the simplicity of the multiple access protocol used for communication, which is the CSMA/CD protocol with exponential backoff. Moreover, the Ethernet technology proved capable of adapting itself to user demands for increased transmission rates. Currently, Ethernet LANs run at speeds of 10Mbps, 100Mbps, and even 1Gbps.

3.2 The CSMA/CA Protocol

The distributed nature of the CSMA protocol and the low delays it induces when the number of active terminals is small, make it a very attractive candidate for wireless communication. However, certain restrictions in such an environment do not permit the direct implementation of the protocol.

Let us recall that in order to be able to implement the CSMA/CD protocol, each terminal needs to be able to perform the following functions.

1. The terminal must be able to listen to the channel and hear whether one or more of the rest of the terminals in the channel is attempting a transmission - carrier sensing capability.

2. The terminal must be able to listen to the channel while transmitting and detect whether its transmission collided with the transmission of some other terminals - collision detection capability.

The collision detection capability implies that a terminal must be able to transmit and receive at the same time, which in a wireless environment can be expensive and is often avoided. Hence, the transmitting terminal may not able to even ensure the correct delivery of its packet. Moreover, as we will see below, even if the collision detection capability exist, it is still possible that a transmitting station does not detect a collision while it is transmitting a packet, but the transmission collides at the receiver. This lack of collision detection capability can be remedied by having the receiver inform the transmitter that the transmitted packet has been correctly received. To do this, the receiving terminal, upon correct reception of a packet, sends a short acknowledgement packet back to the transmitter. This packet is referred to as the ACK message.

Regarding the carrier sensing capability of the terminals, while possible, it is not always sufficient to ensure with high probability that the channel is free of transmissions. To understand this problem we must expand on the special restrictions imposed in a wireless environment. A characteristic of wireless transmission is that terminal a can deliver reliably information to b only if b is within a specified distance from a. Consider now the situation in Figure 9 where we assume that transmissions are symmetric in the sense that if terminal a can deliver information to b, then b can deliver information to a. The transmission from terminal a can reach b but not c. The transmission from c can reach b but not a. Using the standard CSMA protocol in this environment, certain collisions can still be avoided by sensing the channel. For example if b is transmitting to a, c can sense the ongoing transmission. However, assume that while a transmits to b, c receives a packet for transmission. If c listens to the channel, it will not hear a's transmission and therefore, if the standard CSMA protocol is employed, a collision will occur. This problem is known as the *hidden terminal* problem. Note that in this case, even if a is able to detect collisions, it will not be able to realize that a collision occurred since it cannot hear c's transmission - as we saw the latter problem is remedied by the use of the ACK message. Due to the restransmission policy of the basic CSMA protocol, the system can still operate in this environment in spite of the increased number of collisions, however, system throughput may decrease dramatically if packet sizes are large. In fact, plain carrier sensing is not always desirable in this environment. To clarify this point consider again the situation in Figure 9. Suppose that b is sending data to a and c wishes to send data to terminal d. If c senses the channel, it will find it busy and therefore will defer transmission. However, since c's transmission cannot reach a, c could in fact deliver its packet to d without colliding with b's transmission. As a result, plain carrier sensing in this case results in reduced utilization of the system. This problem is know as the *exposed terminal* problem.

We next provide a mechanism to address the above mentioned problems. Two control signals are

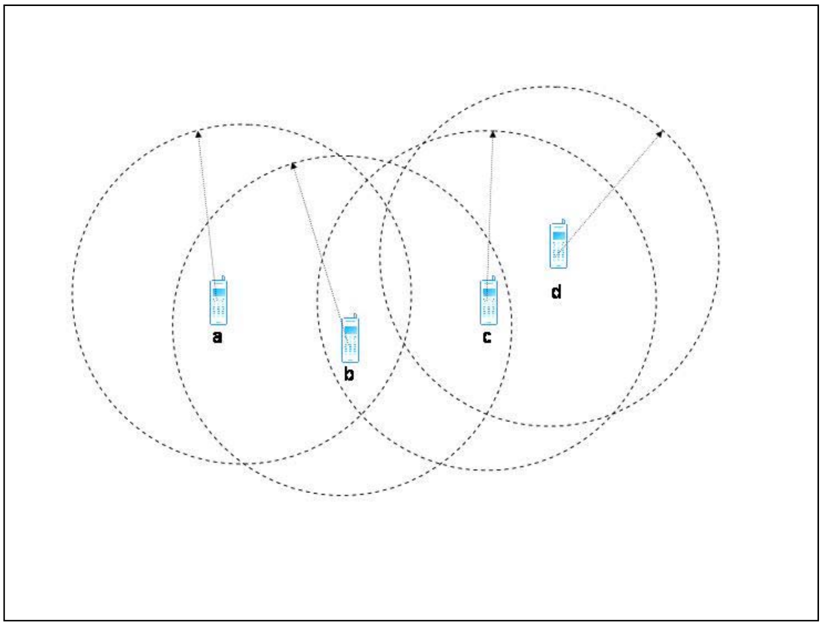

Figure 9: The Hidden and Exposed terminal problems.

introduced. These control signals are short messages (compared to packet sizes) that are exchanged between the transmitter and the receiver before the initiation of packet transmission. The first control signal is sent by the transmitter to the receiver and indicates that the transmitter is "Requesting To Send (RTS)" a packet. The receiver, upon correct reception of the RTS, replies that "it is Clear To Send (CTS)" the packet. Both RTS and CTS signals include a field indicating how long the packet transmission and the accompanied ACK message will last. The terminals now act as follows.

- If a terminal listens to a CTS signal, it waits until the end of the ongoing transmission; this is known since it is included in the CTS signal. It then waits for a random amount of time and attempts to initiate its own transmission process.

Let us see how this rule resolves the hidden terminal problem. Assume for the moment that the transmission of the CTS and RTS signals is instantaneous and let us return to the situation in Figure 9, where a needs to transmit a packet to b. Terminal a sends an RTS to b and b replies with a CTS signal. Terminal c receives the CTS signal and knows that a transmission has been initiated, so it defers its own transmission. Hence the hidden terminal problems is alleviated. In effect the exchange of CTS and RTS messages act as a virtual carrier sensing mechanism.

In fact, the RTS and CTS signals can also be used to also address the exposed terminal problem. Assume that we add the following rule.

- If a terminal listens to an RTS signal but not a CTS then it goes ahead with its own transmission, if any.

In Figure 9 assume that b sends an RTS to a and a replies with a CTS. Terminal c hears the RTS from b but not the CTS from a, and so it knows that its own transmission will not interfere with the b to a transmission. Hence it can start its own transmission at any time. Therefore the exposed terminal problem is avoided.

We assumed above that CTS and RTS signals are instantaneous. Of course, as described in Section 3.1, in a real system transmissions do not take place instantaneously and therefore one cannot assume that the RTS and CTS signals will be received correctly always and free of collisions. However, by now we know that by imposing appropriate retransmission rules the system can deal with occasion loss of RTS of CTS signals. The RTS and CTS are useful if packet sizes are large. For small packet sizes it is preferable to go ahead with the packet transmission rather than incurring the overhead of RTS-CTS message exchange.

The modified CSMA system whose principles of operation were described above, comes by the name CSMA/CA, where CA stands for Collision Avoidance. The acronym signifies that collisions are sought to be avoided and not that they are avoided altogether. Due to the retransmission policy of the CSMA system, collisions that may occur are not detrimental: in case of collision, the ACK message or RTS CTS messages will not be received and the transmitting terminal will defer its transmission for a later time. However, if the propagation delays are relatively large and the system is heavily loaded, collisions may degrade the performance of the system.

3.2.1 Applications of CSMA/CA Protocol

The principles of the CSMA/CA protocol have been applied to the specification of the MAC protocol for Wireless Local Area Networks (WLAN), known as the IEEE 802.11 standard[1]. Originally the transmission rates of IEEE 802.11 were 1 and 2 Mbps. The IEEE 802.11b extension to this standard specified 5.5 and 11 Mbps transmission rates, while there is ongoing work that will increase the rate to 20Mbps. There is currently a great interest in the development of WLAN technologies that not only support high data rates, but also multimedia communication such as video, audio, videoconfenence communication etc. The support for multimedia communication imposes additional requirements to the network, such as low packet delays, low packet loss etc. Networks that are able to provide such support are said to provide Quality Of Service (QOS). CSMA networks were not designed originally to provide QOS. There

[1] Currently the standard does not incorporate a mechanism for dealing with the exposed terminal problem.

is a large amount of onging works that either attempt to adapt the CSMA protocol to these additional requirements or investigate the feasibility of other approaches.

4 To Probe Further

The literature on the ALOHA and the various variants of the CSMA protocols is huge and is still expanding. We do not attempt to provide a detailed account of all the works that contributed to the development of these protocols. Instead we provide some key references to which the interested reader may turn either for a more in depth study, or for a more comprehensive account of related work.

The book by Rom and Sidi [2] provides an in depth analysis of the ALOHA, CSMA, CSMA/CD and various other multiple-access protocols. A nice and detailed exposition of the subject can also be found in the book of Bertsekas and Gallager [3]. Very readable accounts of the protocols can be found in the books of Tannenbaum [4] and Kurose and Ross [5]. Information on the IEEE 802.3 and IEEE 802.11 standards and related activities can be found in the web site [6].

References

[1] N. Abramson, The Aloha system - another alternative for computer communications, *Proc. Fall Joint Comput. Conf. AFIPS Conf., p 37. 1970.*

[2] R. Rom and M. Sidi, *Multiple Access Protocols Performance and Analysis,* Springer-Verlag, 1990.

[3] D. Bertsekas and R. Gallager, *Data Networks*, Prentice Hall, 1992, second edition.

[4] A. Tanenbaum, *Computer Networks*, Prentice Hall, 3rd edition, 1996.

[5] J. F. Kurose, K. W. Ross, *Computer Networking, A Top-Down Approach Featuring the Internet*, Addison Wesley, 2001.

[6] http://standards.ieee.org/getieee802/