Tactical Communications Using the IEEE 802.11 MAC Protocol

John A. Stine
Gustavo de Veciana

Department of Electrical and Computer Engineering,
The University of Texas at Austin, Austin, TX 78712

ABSTRACT

This paper proposes a traffic management protocol built upon the 802.11 MAC that is designed to provide the multiple qualities of service particular to tactical communications. Together, the protocols allow for the integration of voice and data traffic giving priority to voice without preempting data. Additional features allow for more efficient collection and dissemination of position information. The most attractive feature, however, is that the protocols allow mobile stations to conserve power, using one third to one fifth the power used otherwise. The paper presents a study that seeks the design parameters, both at the physical and protocol level, for a network to support a company sized unit. In so doing we present a design methodology and validate the performance of the network using a simulation that is absolutely faithful to the protocol operation. We conduct a sensitivity analysis to explore the effects of the design parameters on network performance.

1. Introduction

Over the past five years the Army has been attempting to develop a tactical wireless network that connects computers throughout a maneuvering combat force. From the time of its conception there has been an urgency to demonstrate a working implementation. As a result, the network was conceived and designed around existing equipment. But the Army's tactical radio, the Single Channel Ground Airborne Radio System (SINCGARS), was designed primarily for a single type use, i.e. single channel voice or data. The effort to combine digital and voice traffic using these radios forces the structure of the network to match that required for a pure voice network. On the surface this appears reasonable but voice and data traffic scale in different ways and this structure results in unnecessary overhead, excess complexity and an inefficient use of bandwidth. Multiplexing multiple voice nets onto a single digital channel offers solutions to these problems but up until just recently there have not been any commercial protocols for controlling medium access and for managing traffic. This paper proposes a traffic management protocol built upon the IEEE 802.11 Medium Access Control (MAC) protocol that is designed to support tactical communications over this type of digital channel. Besides solving many of the problems listed above, the combination also offers an astounding reduction in power consumption.

The paper first expounds upon the limitations of using single voice channel radios to form a network in order to motivate the use of multiple voice net digital channels for tactical communications. It then provides a brief description of the 802.11 MAC and the proposed traffic management protocol. Finally, it presents the design and simulation of a single channel digital network to support a maneuver company using these protocols. The description of the simulation provides details of the traffic model and explains how various parameters of the protocols affect the performance of the network. The simulation demonstrates that a digital channel with a bandwidth that supports 110 kbps of traffic will easily support the demands of the company network and that using a simple power saving feature in the protocol will save the average end user more than 70% of the power that is consumed without this feature.

2. Traffic on Tactical Networks

Figure 1 illustrates the traditional voice nets[1] at a company and below. Net A is the company command voice net and nets B through E are platoon nets. The bandwidth of each channel is just enough to support a simplex voice transmission. Users who want to transmit on the net wait until it is clear of voice transmissions and then transmit using push-to-talk access. In nets where the channel supports both voice and data transmission, voice users preempt data communications. Stations 6, 11, 16, and 21 are on two nets and have a radio for each. They serve as the gateway between the platoon and company command nets. In this network structure voice transmissions are confined to the channel on which they are broadcast but data may be routed to any other member of the network.

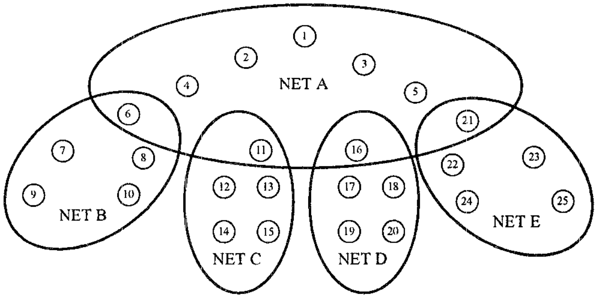

Figure 1. Station Association Among Voice Nets

The logical structure of the voice nets would remain the same regardless of technology since they mirror the lines of command and control. Data transmissions, however, require broader dissemination. There are three different types of data transmissions: 1) Position information that is disseminated to all users. Individual stations automatically report their positions. These reports are collated for each channel and are then rebroadcast both up and down the network hierarchy; 2) Small data transmissions indicative of specially designed short reports used by military units. Their dissemination is report dependent and may be either broadcast or routed up and down the network hierarchy; 3) Larger data transmissions that are indicative of orders, larger text messages or map overlays. Again, their dissemination is file dependent.

Voice and data traffic patterns in these networks are correlated. During periods of intense activity both the demand for voice access and the quantity of data transmissions increase. Voice, however, has a real time access requirement so it preempts data transmission. During voice transmissions, data will queue regardless of priority.

The use of single voice net channels for building a wireless network results in three distinct disadvantages. First, since the channels are optimized to carry voice traffic the channels may not be sufficient to carry the data traffic. Second, since voice nets are hierarchical, data messages must be repeated

[1] In this article, we use the words "net" and "network" to distinguish between the multicast groups for voice traffic and the grouping of all stations as a whole for both data and voice services.

0-7803-4506-1/98/$10.00 © 1998 IEEE.

across many layers. Third, on account of the hierarchical structűre of the network, the network itself becomes less flexible. There is greater overhead associated with maintaining routing tables and changing the topology is difficult. Topology changes are necessary in military networks in order to support task organizing.

3. The 802.11 MAC

The 802.11 standard defines a generic MAC protocol for wireless networking. This MAC protocol has two control functions, a distributed coordination function (DCF) and a point coordination function (PCF). The difference between the two is that in the DCF stations compete for access to the channel in a specified contention process while in the PCF a single station called the point coordinator (PC) manages station access. 802.11 compliant networks may employ just the DCF or a combination of the PCF and the DCF. Achieving multiple QOS requires that both control functions be used. The standard is very explicit on the operation of the DCF. By contrast, it defines the mechanism by which PC controls traffic but leaves traffic management up to the implementer. When operating both functions, the DCF and PCF alternate. The PCF always starts on a defined period but the actual duration of the PCF and DCF may vary. The period of time the DCF is operating is called the contention period (CP) implying the stations in the net can contend for access and the period of time the PCF is operating is called the contention free period (CFP) implying the stations cannot contend for access.

3.1 Distributed Coordination Function. The DCF defines the procedures by which all stations serviced by a single radio channel can gain access to that channel to send data. It is distributed in that the control is completely self contained at the station attempting to send traffic. The DCF uses a carrier sense multiple access - collision avoidance (CSMA-CA) scheme where stations contend and gain access through a handshake process after a random exponential backoff. A station contending for access listens for silence and then waits a random backoff time before attempting to gain control of the medium. If no other station requests access before the backoff expires, then the contending station may attempt to gain access through one of two modes. In the first, the station sends a request to send (RTS) frame. The receiving station upon hearing this request responds with a clear to send (CTS) frame. And finally, if there have been no collisions, the sending station sends its data. In the second, if the data to be transmitted is sufficiently small, the station will attempt to send the data immediately. If successfully received, the receiving station responds with an acknowledgement.

The key collision avoidance feature is a network allocation vector (NAV) that is included in each RTS, CTS, data, and acknowledgment (ACK) frame. This vector predicts the length of time until the stations complete the current transaction. Since both the sending and receiving stations transmit the NAV, all stations within listening range of either of these two stations will know how long the medium will remain busy. Each station in the net maintains a separate NAV timer which it updates upon hearing a transmission. This NAV timer serves as a mechanism to indicate to the station that the channel is busy even when it cannot monitor the transmissions of one of the parties in the current data exchange.

3.2 Point Coordination Function. The PCF is a centralized control mechanism where a single station, the PC, controls all the traffic on the network. The CSMA/CA mechanism of the DCF remains active during the PCF, but the PC is able to gain control of the medium by transmitting prior to any ordinary station in the network. This is accomplished using different interframe spaces which will be described in greater detail later. At the start of the CFP, the PC transmits a NAV that extends until the projected end of the CFP. So all stations who hear the NAV will not attempt to contend during the CFP. Once the PC has control, it then uses polling to manage transmissions of other stations or it transmits data itself. The polls either direct stations to send queued traffic or are used by the PC to gain information from the polled station. The CFP will last until the PC broadcasts an end of CFP message. At this time, monitoring stations will reset their internal NAV.

3.3 Power Saving Mechanism. In a network with both the DCF and PCF, beacons are transmitted by the PC to achieve proper synchronization. The beacon consists of a timestamp and a traffic indication map (TIM), a bitmap indicating which stations have traffic to be received. This beacon is transmitted at a regular interval but may be delayed if a data transmission is ongoing when it becomes due. The beacons are used in the implementation of the power save features for the mobile stations. A significant amount of power can be saved if the receiver is turned off when not needed. A mobile station with "limited traffic" that does not want to participate in the contention period may enter the power save mode by notifying the point coordination station. A mobile station operating in the power save mode turns off its receiver and on a periodic basis wakes up to see if it has any traffic. If traffic is queued, it continues to listen , otherwise it returns to a sleep mode. The period after which a source is required to wake-up is referred to as the Delivery Traffic Indication Map (DTIM) period and is defined as some integer number of beacons. The beacon that is broadcast at this time is referred to as a DTIM. Upon hearing the DTIM the power save station knows from the TIM whether it should continue to monitor the channel in order to receive data. The power save stations remain awake until the TIM of a subsequent beacon no longer lists it as a pending recipient of traffic.

Variable	Definition
BeaconPeriod	Time between beacons in units of time.
aDTIMPeriod	This is the time between DTIMs, the beacons when sleeping stations awaken to determine if they have traffic. It is defined as an integer number of BeaconPeriods
aCFPPeriod	Time between the starts of subsequent CFPs defined as an integer number of aDTIMPeriods.
aMaxCFPDuration	Maximum duration of the CFP defined in units of time. It must occur between the start of subsequent CFPs. The actual CFP may be shorter than this limit.

Table 1. Variables Affecting Transitions Between the CP and CFP

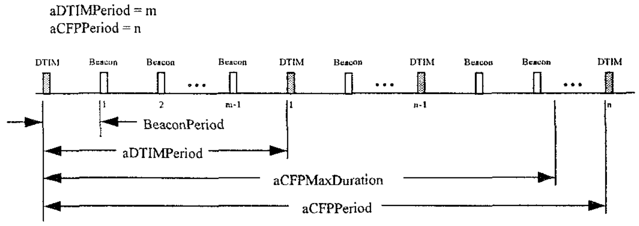

Figure 2. Network Timing

3.4 Defining the CFP Period. 802.11 has four variables that determine how the transitions between the CP and CFP occur. They are BeaconPeriod, aDTIMPeriod, aCFPPeriod, and aMaxCFPDuration. Table 1 defines each of these variables. These variables are the primary design variables that affect the operation of the network. More will be discussed on network design later. Figure 2 illustrates the relationship of these variables. The CFP always starts at the beginning of the aCFPPeriod and the CP always ends at the end. The transition from the CFP to the CP may occur at any time after a minimum CFP up to aCFPMaxDuration. The selection of aCFPMaxDuration must allow a minimum duration CP.

3.4 Interframe Spaces. The key to the operation of both the DCF and the PCF is the use of different interframe spaces. Interframe spaces are predefined periods of required silence prior to transmission. Each type of transmission must wait for a specific interframe space. The duration of the interframe space determines the priority that the transmission has within the protocol. For example, stations that want to contend for access must wait a longer interframe space than a station that needs to acknowledge the receipt of a message. Similarly, the PC seizes control by using a shorter interframe space than that used by a contending station but longer than the interframe space for an acknowledgement.

The 802.11 MAC defines four interframe spaces. The first and shortest is the Short Interframe Space (SIFS). The SIFS is the period of no transmission between the receipt and response to a frame. It is the space that exist between the CTS and RTS frames, data frames and the subsequent ACK, and the polling frames and responses during the operation of the point coordination function. The second, and next shortest in duration is the PCF Interframe Space (PIFS). The PIFS is used by the point coordinator to send beacons and to seize and maintain control of the medium during the contention free period. The third, and next longest in duration is the DCF Interframe Space (DIFS). This is the required duration of silence before contenders begin to count down their backoff timers. The fourth and longest is the Extended Interframe Space. It is used by a station waiting to contend instead of a DIFS when the traffic that it last detected was not a valid MAC frame.

The duration of the first three of these interframe spaces is dependent on the physical layer. They are determined by two values, aSIFSTime and aSlotTime. The aSIFSTime is the time it takes from the end of receipt of a message for a station to process the message and then respond with the first symbol of the preamble. The aSlotTime is the time it takes a station to recognize a channel is busy or idle plus the time it takes to process a frame, prepare a response, transmit it, and for it to propagate to the receiving station. The duration of the SIFS is aSIFSTime. The PIFS is one aSlotTime longer than the SIFS and the DIFS is two aSlotTimes longer the SIFS. The EIFS is equivalent to the time it takes to transmit 8 consecutive ACK frames plus the duration of one SIFS and one DIFS.

3.5 Transmitting Data. In a network operating both the DCF and the PCF, data traffic may be sent in either the CP or the CFP. The choice of how to send the data is made by the sending station. It either contends to send data immediately to the receiving station during the CP or it contends and requests that the PC mediate the transmission of the traffic during the CFP. The network is normally set up such that the stations will attempt to send data within the CP only if the file size is less than some threshold called the Maximum MAC Service Data Unit Size(MaxMSDUsize). (The MaxMSDUSize is not defined by the 802.11 MAC and is added as a part of this implementation.) This threshold is chosen such that sufficient time remains available during the CP for contending stations to contend. When data is sent in the CP it may either be fragmented or sent as a single transmission. Fragmentation occurs when the message is larger than a second threshold called aFragmentationThreshold. This threshold is used to minimize the overall transmission time of the file when retransmission on account of bit errors is considered. A third threshold exists for sending data during the CP called aRTSThreshold. This threshold determines when a file is sufficiently small such that it is more advantageous to send data without the RTS/CTS handshake. It's size is dependent on the probability of a collision on the wireless link. Table 2 lists the different thresholds and the actions taken for each.

Data Size	Station Action
FileSize ≤ aRTSThreshold	Send immediately upon successfully contending
aRTSThreshold < File Size ≤ aFragmentationThreshold	Contend using RTS/CTS handshake and send as a single transmission.
aFragmentationThreshold < File Size ≤ MaxMSDUsize	Contend using RTS/CTS handshake and send as a series of fragments.
MaxMSDUsize < File Size	Contend and coordinate with the PC to mediate the transmission.

Table 2. File Size Thresholds That Affect Method of Data Transmission

Each station maintains a counter variable which counts the number of times a station attempts to contend for a given message. If the station continues to contend without success at reaching the destination it may take one of two actions. Drop the transmission or request the PC manage it. The reason for not having success may be one or more of the following: the destination is no longer in the network; the destination cannot hear the transmitter; or the destination is in the sleep mode. Going to the PC assumes the latter of these three are the cause of the contention failure. If this is the case the PC will awaken the destination prior to orchestrating the transmission. If it is not the case, the PC knows which stations are on the channel and can assist the transmitting station in making the decision to drop the transmission. (This is not explicitly part of the 802.11 standard.)

3.6 Transmitting Voice. Voice transmissions require the network to provide a guaranteed continuous bit rate. This is only possible during the CFP where the PC can manage the bandwidth. So when a station wants to transmit a voice message it contends sending a message to the PC requesting that the PC control the traffic. If there is sufficient bandwidth, the PC will acknowledge the request otherwise the call is blocked. 802.11 does not define the specific process by which the PC manages the bandwidth during the CFP. An implementation is described in Section 5. (Note that [3] suggests a modification to the 802.11 MAC that allows voice communications during the DCF.)

3.7 Operation of the DCF. Figure 3 illustrates the contention process and the different types of traffic in a network with a PCF. The process starts when traffic arrives at a station and needs to be transmitted. That station then calculates a backoff using the equation.

$$Backoff = \left\lceil \min\left(\left(2^{i+3} - 1\right), 255\right) \times ran\# \right\rceil \times aSlotTime$$

The variable i in this equation is the number of times the station has previously contended to send this data. So the first time a station attempts to send the traffic, the backoff is between 1 and 7 aSlotTimes, the second time between 1 and 15, the third time between 1 and 31 and so on until the maximum of between 1 and 255 is reached. In Figure 3 there are 4 stations contending for backoff. The backoff timer of the first is 8 aSlotTimes and it has a data packet that is smaller than aRTSThreshold to send. The backoff timer of the second is 2 aSlotTimes and it has a packet larger than aRTSThreshold but smaller than aFragmentationThreshold. The backoff timer of the third is 12 aSlotTimes and it has a data packet larger than a aFragmentationThreshold. The fourth station has a backoff of 5 aSlotTimes and it is seeking access from the PC for either real-time traffic or a file that is too large to be efficiently transferred during the CP. Also shown in the illustration is a generic receiving station and the point coordinator. Note that after each contention period all backoff timers are decreased by the smallest backoff time. Observe the use of the interframe spaces. Since CTS and ACK frames are transmitted only after aSIFSTime they have priority over beacons and contention backoff. In turn, beacons are transmitted after aPIFSTime and therefore have priority over contention backoff.

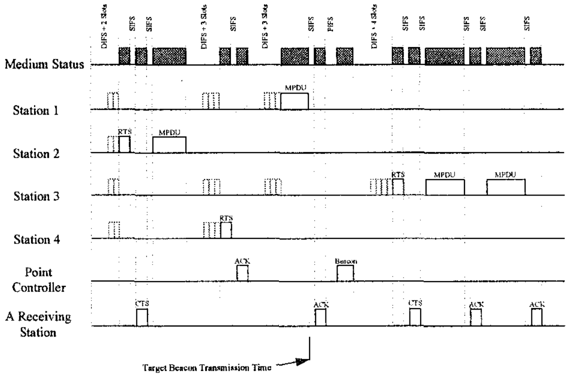

Figure 3. Example of the Distributed Coordination Function

3.8 Operation of the PCF. The operation of the PCF is not fully described by the 802.11 standard. Besides providing the mechanisms that allow the PC to seize control of the channel and defining the different types of polling frames, the standard leaves the rest up to the implementer. In this implementation of the point coordination function it is assumed the PC may control five types of traffic. Traffic from the one station to a second. Traffic from a station to the PC. Traffic from the PC to a station. Broadcast from a station. And finally, a broadcast from the PC. Figure 4 illustrates the operation of the point coordination function. The five types of traffic are illustrated. The following transmissions occur sequentially in the illustration. Station 1 is directed to transfer data to another station, not the point coordinator. Station 2 is directed to send data to the point coordinator. Station 3 receives data from the PC. Note that the PC's transmission includes the ACK to Station 2's transmission as well as both the Poll and Data for Station 3. Next, all stations receive a broadcast from the PC. This broadcast occurs in one fragment. The PC does not wait for an acknowledgement. Immediately after a SIFS the PC attempts to transmit data to a station but in this case the station does not hear the transmission and sends no ACK. Note that the PC uses the PIFS to regain control of the medium. In the final set of transmissions the PC directs Station 3 to send a broadcast transmission. Note again that the PC regains control of the network after waiting only a PIFS.

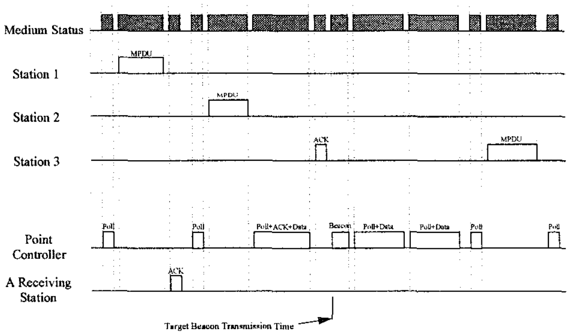

Figure 4. Example of the Point Coordination Function

4. Traffic Management Protocol

4.1 Voice Service. The purpose of a traffic management protocol is to provide adequate quality of service for all types of traffic in the network. The first challenge is to provide real time (RT) service for voice communications. The sequence for providing voice service starts with a station successfully contending and coordinating RT service from the PC. The PC then provides the service during the PCF by periodically polling the station and directing it to send transmissions. This continues until the station no longer responds or the station announces it has completed the transmission. In turn, the PC removes the station from the RT traffic list and stops polling it. The period of the polls matches the DTIM Period[2] in order to allow stations in the power save mode to receive the transmissions. The polls for all RT traffic follow immediately after each DTIM. Any number of DTIMs may occur during the CFP but a DTIM may not occur during the CP unless there is no RT traffic. The aCFPMaxDuration is therefore chosen to occur sometime after the last DTIM in the CFP-CP cycle (See Figure 2.). The time from the last DTIM in the CFP-CP cycle to the aMaxCFPDuration is the maximum time available each DTIM period to support RT traffic. The characteristics of the physical layer (i.e. real-time traffic bit rate, packet overhead, and transmission rate) together with the choice of aDTIMPeriod, aCFPPeriod, and aCFPMaxDuration determines the maximum number of the RT connections the point coordinator may support. The equations below define the relationships.

$$RTTrafficBitsPerDTIM = RTBitRate \times aDTIMPeriod \times BeaconPeriod$$

$$RTSlotTime = {(RTTrafficBitsPerDTIM + \#OHBitsPerPacket + \#\ PollBit)}\big/{TransmissionRate} + 2 \times aSIFSTime$$

$$RTFrameLength = aDTIMPeriod \times BeaconPeriod - (aCFPPeriod \times aDTIMPeriod \times BeaconPeriod - aCFPMaxDuration)$$

$$Max\#Calls = \lfloor (RTFrameLength - TimeToTransmitBeacons)/RTSlotTime \rfloor$$

Using this management protocol, the DTIM Period is the maximum delay contributed by the protocol and the time between the first and last RT slots is the maximum jitter. We assume that the network is designed such that a delay of a DTIM Period plus the maximum jitter time is not excessive. In this case, buffering can be used at the receiving station to eliminate any of the effects of the jitter.

In an effort to give some priority to voice, stations with both voice and data traffic pending access will contend for voice access first.

4.2 Traffic Management Phases. As noted above, not all of the time between DTIMs can be used for RT service. In fact, the length of time after the occurrence of the aCFPMaxDuration until the start of the next CFP-CP cycle is the minimum amount of time available between DTIMs that can be used for other services. The traffic management protocol takes advantage of this time by defining three distinct traffic management phases which may occur between each DTIM during the PCF. During the first phase, which begins immediately after the DTIM, the PC sequentially directs all stations with RT traffic to transmit. Once every RT station has transmitted, the second phase of the PCF begins. In this second phase, the PC cycles through the list of stations with pending PC controlled data traffic and directs each to send one MPDU each cycle. This continues until all stations have sent their data or the next DTIM occurs. Finally, if there is any time remaining until the next DTIM, the PC polls the active stations (stations that are not in the power save mode) to determine if they have any traffic pending. If a polled station is waiting to contend for real-time access, this feature allows the

[2] In this discussion "DTIM Period" refers to the period between DTIMs in units of time. aDTIMPeriod is the period between DTIMs in number of beacons.

station access without having to contend. The RT station will be included in the next real-time traffic phase after the next DTIM. If the station has data traffic pending access, the PC may direct it to transmit all or a portion of the data immediately or add it to the PC controlled data traffic list. The option chosen is based on the size of the data unit pending transmission. Similar to what occurs during the contention period, if the data is less than aFragmentationThreshold, then it may be transmitted in one transmission, otherwise it is fragmented. The maximum size of the data unit that may be sent in one poll is dynamically limited by the time until the next beacon. This limit serves two functions. It insures beacons, and more importantly DTIMs are not delayed and it contributes to a degree of fairness allowing multiple stations to be served. Note that in this implementation, if there is neither RT traffic nor PC controlled data traffic prior to the occurrence of aCFPMaxDuration, the point coordinator relinquishes control to the DC

4.3 Power Save Mode. The power save mode is the simplest possible. Individual stations choose to enter the power save mode whenever they have no traffic to send. At DTIMs they awaken to determine if they are to be a recipient of traffic. If they are they stay awake until a subsequent beacon reveals they are no longer a pending recipient. There is no effort to optimize this process by shaping traffic at the stations or by scheduling transmissions at the PC.

5. Simulation Model

The simulation used in this study explicitly models the 802.11 and the traffic management protocols described above. This includes modeling contention collisions amongst stations, transmission errors, and fading conditions. The simulation, however, does not model the hidden node phenomena. The simulation attempts to isolate the random events that occur to separate entities. Since traffic generation and channel characteristics are unique to each station, each station has its own series of random number generators. These random number generators are independent of each other affecting separate types of events such as files sizes, voice traffic duration, arrival times, and fading times. Each station has a total of 12 random numbers generators. The simulation models the 25 station organization illustrated in Figure 1 considering it to be representative of a maneuvering combat team. As will be described in Section 6.1, the network also has a set of random number generators to support the voice traffic model.

In a network there are various overheads for routing calls, providing training sequences for equalization, etc. The simulations in this study use the overheads that are defined in the 802.11 specification for the Direct Sequence Spread Spectrum (DSSS) physical layer.

5.1 Traffic Model. The traffic model for the simulation attempts to replicate both the voice and data traffic typical in military communications when traffic is at its peak. Voice nets are exactly as illustrated in Figure 1. The voice traffic on these nets tends to occur in bursts of transmissions. A user transmits a message that then gains a response from another user which results in a subsequent response and so on. A burst may involve several exchanges with any member of the voice network participating. Traffic generation in a voice network is dependent on the use of the network. That is users do not attempt to transmit until the channel is no longer busy. Each transmission in these bursts, however, requires the station to contend for access.

The voice traffic model attempts to replicate the voice traffic described above. At the network level each voice net is modeled as an on-off Markov chain alternating between idle and traffic burst periods. During the traffic bursts all members of the voice net have an equal probability of being the transmitter. The transmitting station is selected at the network level and then that station's own traffic generation parameters and random number generators are used to determine the time until it contends and the duration of the transmission. All stations and voice nets were modeled with the same traffic statistics. Table 3 lists and describes the parameters that make up the voice traffic model. Figure 5 illustrates the use of a voice net modeled in this manner.

NAME	DESCRIPTION	VALUE
Average Idle Time	The average time from the end of one burst till the beginning of the second.	20 Sec
Average Burst Time	The time period over which stations alternate transmitting in a voice net	30 Sec
Average Wait Time	The average time from the end of a voice transmission that a second user waits before attempting to transmit a reply.	0.5 Sec
Average Call Duration	The average duration of the voice transmission	4 Sec
Voice Coding Rate	The number of bits per second used to code a voice channel.	10 kbps

Table 3. Parameters of the Voice Traffic Model

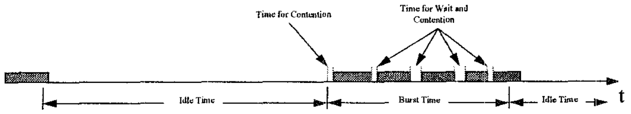

Figure 5. Model of Voice Traffic on a Single Voice Net

Data transmissions are as described in Section 2. The large and small files are generated as a Poisson process at each station. There are two sets of parameters, one for the small files and one for the large files. The parameters in each set is the average time between arrivals and the average file size. Files that arrive while other files are being transmitted are buffered at the stations. There is a single buffer in which both small and large files are stored until transmission. Files are serviced in order of arrival. The parameters used for the data arrivals is summarized in Table 4. All stations were modeled with the same data traffic parameters. These data transmissions may be either a point to point transmission between two stations in the net, a point-to-point transmission from a station to the PC, or a broadcast transmission of some kind. The simulation randomly selects the type of transmission and the destination with 20% being multicast and the remainder being split between the two point-to-point transmissions.

PARAMETER	VALUE
Large File Interarrival Time	240 Sec
Large File Average Size	50,000 bits
Small File Interarrival Time	25 Sec
Small File Average Size (bits)	800 bits

Table 4. The Station Data Traffic Generation Parameters.

The final data service is that which supports the automatically generated location reports. These messages have a constant and relatively small size. Although these location transmissions may be broadcast the primary issue is consolidation and dissemination of the data. In this network we consider the PC to have this function so the location transmissions are point-to-point from the stations to the PC. In existing military networks that report this information the stations initiate these transmissions. Due to their small size, the cost of contending, and the frequency at which polling occurs, we handle these reports in a different manner. Each station maintains a location buffer which is updated independently of the network operation. When a station is polled by the PC the station adds the location currently stored in the buffer to the overhead of its reply.

5.2 Error Model. The simulation uses an on-off Markov chain to model the effects of fading. The specific model is as suggested in [1].

6. Network Design

The objective in designing the network is to provide all the services needed using the least bandwidth. At the physical level, the designer must select a voice encoding bit rate and the transmission rate. At the MAC level the designer must select the thresholds for sending data and the timing parameters. The following is our design process.

6.1 Assume a voice bit rate, a maximum delay constraint, and an average BER. There are numerous voice encoding methods. As an example the U.S. digital cellular system uses a vector sum excited linear predictive coder (VSELP) that operates at a raw data rate of 7950 bits/s. As an arbitrary yet not unrealistic number we chose 10 kbps. The maximum delay the voice signals can tolerate is actually quite large. The transmissions are simplex without feedback to the sender. In a conversation, the delay manifests itself only in increasing the time till a response is heard. We assume the delay is constrained to 250 msec from transmission to receipt. In turn this contributes a maximum of 500 msec to the overall time from end of a transmission to the receipt of a response. The BER we chose comes directly from our error model which has an average error rate of about 5×10^{-5}.

6.2 Determine a feasible transmission rate. The voice bit rate, the number of simultaneous voice channels, the size of the DTIM, the minimum size of the CP, and the physical layer overhead all affect the choice of the transmission rate. The equations of section 4.1 characterize these requirements. The variable aCFPMaxDuration in these equations is selected based on the desired size for the CP. As a start we use the minimum size CP specified by the 802.11 MAC. It must be large enough to allow the transmission of one data file the size of MaxMSDUSize plus 8 ACKs. Figure 6 are the graphs of the minimum DTIM Period versus transmission rate that would allow a CP to support a MaxMSDUSize of 1000 bits and still have adequate time between DTIMs to support 5 concurrent voice transmissions. Note from these graphs the impact of the 802.11 physical layer overhead. At higher transmission rates the coding rate gain decreases as the overhead becomes the dominant factor in determining delay. Additionally, to get to lower transmission rates requires acceptance of a higher delay. Nevertheless it is still tolerable by our specification.

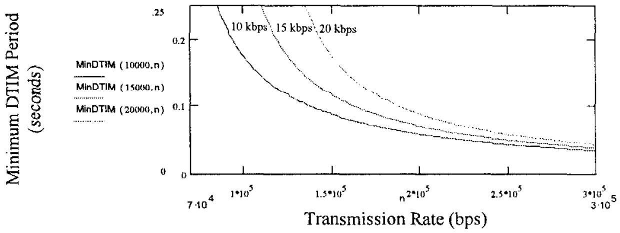

Figure 6. Minimum DTIM Period as a Function of Transmission Rate (bps) and Coding Rate.

From this graph we selected a transmission rate of 110 kbps. At this rate the minimum DTIM is 147 msec with a corresponding total delay for voice transmissions of 239 msec. This gives us some leeway to increase the CP if simulation shows the CP to be too short to support the contention process.

6.3 Calculate the aFragmentationThreshold.

The aFragmentationThreshold is the point where it is more advantageous to transmit data in fragments than in as a single transmission. It depends on the overhead and the BER of the channel. The solution to the following equation provides the threshold. The solution to this equation is the file size where it takes the same time on average to transmit the file in two packets as it does to send it in one.

$$2 \times \left(\frac{\left(ACK + PacketOH + \left(aFragmentationThreshold/2 \right) \right) \times TransmissionRate + SIFS\_Time}{(1 - BER)^{\left(ACK + PacketOH + \left(aFragmentationThreshold/2 \right) \right)}} \right)$$

$$= \left(\frac{\left(ACK + PacketOH + aFragmentationThreshold \right) \times TransmissionRate + SIFS\_Time}{(1 - BER)^{\left(ACK + PacketOH + aFragmentationThreshold \right)}} \right)$$

Again the overhead has an interesting effect. As the overhead gets larger so too does the aFragmentationThreshold. This is because the increased overhead has a greater impact on the duration of the smaller packets than the large packets thus allowing a larger threshold. Nevertheless, the probability of a successful transmission decreases as aFragmentationThreshold increases. Using the average bit error rate and the transmission rate selected in step 2 above we calculated the aFragmentationThreshold to be approximately 4700 bits. A packet this size has only a .77 probability of being received without error. Since this aFragmentationThreshold is larger than the initial guess for MaxMSDUSize we either needed to increase MaxMSDUSize and recalculate the minimum DTIM period or to choose aFragmentationThreshold to be the same as MaxMSDUDuration. We chose the latter to prevent the CP from being congested. A size close to aFragmentationThreshold, 4000, was selected for the MPDU size within the CFP.

6.4 Select the aRTSThreshold. If the data packet is small enough, i.e., less than aRTSThreshold, it is deemed better to attempt to send the traffic right away than suffer the delay of the RTS-CTS handshake. As an example, suppose the probability of a collision is 50% then aRTSThreshold is set by the following equality,

$$\left(\frac{(PacketOH + aRTSThreshold) \times TransmissionRate + DIFS + aSIFSTime}{.5} \right) + ACK\_Size \times TransmissionRate$$

$$= \left(\frac{RTS\_Size}{.5} + CTS\_Size + PacketOH + aRTSThreshold + ACK\_Size \right) \times TransmissionRate + \frac{DIFS}{.5} + 3 \times aSIFSTime$$

Our simulation does not simulate the hidden node collisions, only contention collisions, so we used a conservative parameter of 50% probability of a collision and determined the aRTSThreshold to be about 350.

6.5 Select the Beacon Period. The beacon period has an impact on the power save feature. Stations that have been awakened by a DTIM wait for a TIM in a beacon to tell them to return to the standby state. The more frequently the beacons occur the less time on average is spent receiving but the size and duration of the beacon is not insignificant and consumes precious bandwidth. Additionally, there is a point where there is no advantage to having more beacons. The beacon period should not be any shorter than the duration of a single voice slot. In our design the voice slots are longer than the CP. We chose aDTIMPeriod of 3 beacons which corresponds to a beacon occurring every two voice slots. The BeaconPeriod was then determined from the DTIM period.

6.6 Select the CFP Period. The choice of the CFP period balances the data transmission requirements during the CFP with the contention and data transmission requirements of the CP. The more DTIMs in the cycle the less time is available for contention access. This is not an issue when the network does not use the power save feature since stations can gain access during the CFP as part of phase 3 of the traffic management protocol but it is an issue in the power save network since stations transitioning from power save status must contend for access. In our design we chose aCFPPeriod of 2 to insure frequent CPs to support the contention process.

Parameter	Value
Voice Bit Rate	10 kbps
Transmission Rate	110 kbps
BER	See table 5
BeaconPeriod	49 msec
aDTIMPeriod	3
aCFPPeriod	2
aMaxCFPDuration	261 msec
aRTSThreshold	350 bits
aFragmentationThreshold (CP)	1000 bits
aFragmentationThreshold (CFP)	4000 bits
MaxMSDUSize	1000 bits

Table 5. Network Design Parameters

6.7 Select aMaxCFPDuration. The calculation of aMaxCFPDuration follows directly from what we have already determined. It is the difference of the time for the minimum CP from the time duration of the CFP Period.

As with most design processes, the one above is iterative. This was not emphasized in the description in order to keep it simple. After a number of process iterations, we decided on the parameters listed in Table 5.

7. Simulation Results

Multiple simulations were run both using and not using the power save mode. Each simulation starts in the zero state and runs 1 hour.. This is considered a good representation of the network going from a radio listening silence state to the intense traffic expected at contact with the enemy. Common random number seeds were used in the different modes to support comparison. Table 6 compares the networks using various criteria.

7.1 Improving Voice Access Performance by Increasing the CP . Of greatest concern in the performance of this network is the occasional voice access time that exceeds 1 sec. The intuitive solution is to increase the time available for contention. This gives us two options, increase the length of the CP and decrease the time in the CP used to send data. We attempted both of these solutions simultaneously. We increased the Beacon Period to 51.5 msec and aMaxCFPDuration to 271.6 msec to achieve a maximum voice delay of 250 msec. We also reduced the MaxMSDUSize from 1000 to 800 bits. These changes had little effect on the performance of the network.

Increasing the duration of the CP clearly reduced the average voice access time but did not eliminate the occasional delay greater than 1 second. Reducing the MaxMSDUDuration did not appear to have an effect. We traced through the simulation to get a better understanding of the causes of the long access times. These events occurred in clusters. When the power save feature was not being used, they occurred when the PC was mediating the transfer of a large data file and a relatively large number of stations were contending (more than 5). All access is then limited to that gained through the contention process. In the power save mode the factor was the large number of stations contending. Whether the PC mediates a large file transfer is not a factor since sleeping stations that have traffic to send will not be polled until they contend and leave the power save mode. Increasing the length of the CP remains the strategy of choice to reduce these sporadic long delays. Since the voice delay constraint limit has been reached the solution is either to relax the constraint, decrease the aCFPPeriod so CPs occur more frequently, or to attempt a faster transmission rate. Table 9 provides the results of decreasing the aCFPPeriod to one. All the parameters of the design are the same as those in Table 6 except the timing parameters were adjusted to the threshold where the maximum voice delay was 250 msec. As can be seen, the occurrence of long access delays decreased dramatically. In all the simulations of both modes there was only 1 occurrence of a delay greater than 1 second (1 in >9000 calls). Before, there was a minimum of 4 in each of the 6 simulations that were run.

7.2 Changing Transmission Rates. Table 8 provides the results of increasing the transmission rate. For each of the simulations the timing parameters were selected to support a maximum voice delay of 250 msec. Otherwise, the parameters were the same as those in Table 5. As expected, increasing the transmission rate improves all performance measures.

CRITERIA	W/O PSAVE	W PSAVE	OBSERVATION
Voice Access Time - The time it takes from a call arrival till it receives access. (seconds)	Average: 0.117 Std Dev: 0.106 High: 1.13	Average: 0.153 Std Dev: 0.130 High: 1.58	Performance decreased slightly but not enough to be an issue. The "highs" for access times were longer than desired.
Data Transmission Rate - The rate at which data bits are transmitted. The clock starts at file arrival and ends once all bits are transmitted. (bps)	Average: 9256 Std Dev: 10543 High: 71516 Low: 0.49	Average: 6764 Std Dev: 9298 High: 69158 Low: 1.64	Performance decreases by 27% when power save mode was used but no severe backlogs of data were observed in either case.
Poll Arrival Time - The time between polls to the same station. (seconds)	Average: 0.362 Std Dev: 3.41 High: 28.07	Average: 0.408 Std Dev: 21.81 High: 182.5	Effectiveness of polling procedure to get position data is adequate for the non-psave network but results in long interludes of no reports in psave networks. A possible fix is to implement a timer where stations initiate reports when they have not been polled.
Power Consumption - Average power consumed by each station. Reported as time in seconds in each of the three states; transmitting, monitoring, standing-by, and in a dimensionless usage factor[3]	Transmit: 79 Monitor: 3521 Standby: 0 Usage: 3679	Transmit: 77 Monitor: 863 Standby: 2660 Usage: 1017	Power consumption decreases dramatically and equates to using less than a third the number of batteries in the transmitters of the mobile stations of the psave network.

Table 6. Simulation Results on Initial Network Design

[3] Power consumption depends on the hardware used. [4] identifies two different transmitters with power consumption ratios of 1.8:0.6:0.05 and 1.7:1.5:0.08 between the transmit, monitor, and standby modes respectively. We use a 2:1:0 ratio in our analysis.

CRITERIA	Measure	WO PSave	W PSave
Voice Access Time	Average:	0.066	0.071
	Std Dev:	0.051	0.054
	High:	0.676	1.19
Data Transmission Rate	Average:	11174	8338
	Std Dev:	10742	9767
	High:	67732	70480
	Low:	2.16	2.10
Poll Time	Average:	0.462	0.524
	Std Dev:	3.50	37.47
	High:	41.37	205.63
Power Consumption	Transmit:	74	71
	Monitor:	3526	920
	Standby:	0	2609
	Usage:	3674	1062

Table 7. Performance After Decreasing aCFPPeriod

Criteria	Measure	120000 bps W PSave	130000 bps W PSave	150000 bps W PSave
Voice Access Time	Average:	0.134	0.125	0.111
	Std Dev:	0.098	0.105	0.096
	High:	0.633	0.409	0.334
Data Transmission Rate	Average:	8005	8852	10449
	Std Dev:	11198	12464	14986
	High:	79676	80453	105637
	Low:	2.70	2.44	2.38
Poll Time	Average:	0.372	0.361	0.325
	Std Dev:	18.33	20.22	15.90
	High:	186.34	147.20	159.94
Power Consumption	Transmit:	73	70	65
	Monitor:	851	823	824
	Standby:	2676	2707	2711
	Usage:	997	963	954

Table 8. Network Performance After Increasing Transmission Rate

7.3 Reducing Power Consumed During Power Save Mode by Increasing aDTIMPeriod. As a final test of the protocol we evaluated the effect of increasing aDTIMPeriod on the power consumed in the power save mode. (This amounts to increasing the number of beacons in a fixed length DTIM period.) We did this analysis using the transmission rate of 150 kbps. The expected effect would be a reduction in power consumption with little effect on the other performance measures. Figure 7 confirms this expectation. The simulation using 9 beacons per DTIM consumed 23% less power than that consumed using 3. This is also a mere 20% of that which is consumed not using the power save mode. There is a practical limit that can be achieved by increasing the aDTIMPeriod. Note the leveling off of improvement over 5 to 7 beacons per DTIM. Most of the power consumed in this network is listening to the voice transmissions which are broadcast. These transmission are relatively long. If the target beacon transmission time occurs during one of these transmissions then it is delayed for the duration and if more than one target beacon transmission times occur during a single voice packet transmission then only one of the beacons will be transmitted. In the case of aDTIMPeriods of 5, 6, and 7 the effective frequency of the beacons during the voice transmissions remained the same for this reason. In these transmissions 5 beacons per DTIM corresponds to one beacon every other voice packet and 9 beacons per DTIM corresponds to a beacon being transmitted after each voice packet transmission. This is considered the practical limit to improvement in this network.

7.3 Other Improvement Options. Although a reasonable design was achieved using a total transmission rate of 110 kbps it was very obvious that the physical layer has a significant effect on the performance of this type of wireless network. The 802.11 DSSS physical layer used in these simulations contributes 192 bits of overhead to every transmission. Reducing this overhead would make smaller transmission rates feasible.

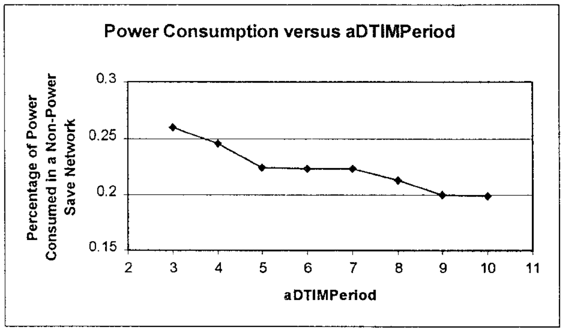

Figure 7. Power Consumption as a Function of the Number of Beacons per DTIM

8. Conclusion

This paper presented a new approach for a tactical communications network that offers many improvements over existing networks. These improvements not only include a more flexible structure allowing easier reorganization but also a design flexibility to offer different qualities of service and different power consumption rates. This paper contributed a description of the problems in current tactical networks, a digital network alternative, a traffic management protocol, a network design methodology, a high fidelity simulation, and for the first time to the authors' knowledge a simulation of an 802.11 based network that examines its power save mode. Although migrating to this type of network will require the development of new radio systems the improvements would be well worth the investment. The analysis presented in this paper demonstrated that the difference in power consumption can be as much as a factor of 5. Light forces using battery powered transceivers would consume one fifth the number of batteries Additionally, fewer transceivers would be needed since multiple nets are serviced on a single channel.

REFERENCES

[1] B. Crow, et al., "IEEE 802.11 Wireless Local Area Networks," IEEE Communications Magazine, September 1997, pp 116-126.

[2] J. Sobrinho and A.S. Krishnakumar, "Real-Time Traffic over the IEEE 802.11 Medium Access Control Layer," Bell Labs Technical Journal, Autumn 1996, pp 172-187.

[3] P802.11, Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, IEEE Press, May 1997.

[4] J. Chen, et al, "A Comparison of MAC Protocols for Wireless Local Networks Based on Battery Power Consumption," INFOCOM 98, March 1998, pp. 150 - 157.

John A. Stine is a Major in the U.S. Army Corps of Engineers. He served as an assistant professor in the Department of Electrical Engineering and Computer Science at the United States Military Academy at West Point and as an analyst in the Army's recent Task Force XXI experiment. He is now a student at The University of Texas at Austin.

Gustavo de Veciana is an Associate Professor in the Department of Electrical and Computer Engineering at The University of Texas at Austin. His research focusses on issues in the design and control of telecommunication networks. He was the recipient of a 1996 National Science Foundation CAREER Award.