A Software Framework for Mobile Ad hoc Data Communications Using Legacy Voice-Centric Tactical Radios

Steven R. Brand Geoffrey G. Xie John H. Gibson

Abstract

Currently, small ground units such as those operating in Iraq have very limited data communication capabilities between soldiers. Tactical chat or file transfer is available only at the battalion level or higher. To address this problem, we have developed a software application, which can leverage existing voice-centric radios to provide data services including tactical chat and file transfer capabilities to frontline ground units. The software embodies two key innovations: (1) a data link protocol to allow the radios to form wireless LANs, and (2) an operation-aware ad hoc routing protocol to support reliable and resource-efficient multi-hop message relays. In this paper, we describe the design, implementation, and functional testing of these two protocols.

1. Introduction

Among the capabilities of the Knowledge Area underlying the Net-Centric Environment, as defined by the Net-Centric Environment Joint Functional Concept [NCE05], are the ability to collaborate, synchronize actions, share situational awareness and understanding, and achieve constructive interdependence. These capabilities must be enabled by the underlying network architecture. Efforts are underway to acquire the tactical networking infrastructure necessary to ensure the interoperability crucial to achieving these knowledge capabilities. However, current tactical maneuver element networking capacities fall far short of meeting the agility, robustness, and flexibility necessary to satisfy the information needs of constructive interdependence. As noted in the Net-Centric Operational Environment Joint Integrating Concept, "The Joint Force and mission partners require, but currently lack, a seamless sharing of required information and knowledge through an assured, protected network." [NCE05] Particularly, while data networking resources are often abundant at command headquarters, they *remain scarce* for frontline force elements, precisely where constructive interdependence is required. Most of the communication within the frontline units is carried out using voice-centric ground radios such as the Single Channel Ground Air Radio System (SINCGARS) [FAS99][EngDoc00].

Whereas high-speed networks are crucial to high resolution video- and imagery-based communications, several rudimentary functionalities have long proven their worth to command and control collaboration while, for the most part, *not demanding high bandwidths*. These include basic text-based e-mail, instant messaging, and file transfer. The value of e-mail to collaboration has long been recognized. Interactive on-line chat functions, often referred to as instant messaging, grew out of the bulletin-board services and were standardized by RFC 1459. The value of this function in support to situational awareness was clearly demonstrated in both the 1991 Soviet Coup and Kuwaiti force operations [IRC]. Finally, file transfer is one of the core capabilities of the networking and underlies many applications, such as the File Transfer Protocol (FTP) and Hyper Text Transfer Protocol (HTTP), and remains fundamental to collaboration activities. Most of the

voice-centric ground radios including the SINCGARS have a built-in data interface (port). The data interface has *sufficient* bandwidth to support tactical chat and transfer of small files. However, it is designed for *manual* establishment of *point-to-point* data connections and does not have advanced data networking capabilities such as *automatic local area networking* and *multi-hop relays*. Since these advanced capabilities are crucial to the deployment of tactical chat and file transfer applications, tactical chat and file transfer are not widely used by frontline troops.

Several solutions to this problem have been envisioned but they have severe limitations in terms of functionality, cost, or timeliness. There exist commercial off-the-shelf products capable of supporting mobile ad hoc data communications, such as would benefit small maneuver forces. For example, several venders provide wireless local area networking (WLAN) equipment. These products offer many benefits, such as rapid network establishment and some degree of support for mobile users. Unfortunately, they are often incompatible with operational security requirements, particularly in the area of limitations in levels of classification supported or likelihood of detection.

Concurrently, there are several commercial ventures specifically targeted for the military's tactical data network requirements. In particular, three vendors, Raytheon, ViaSat, and Harris, have developed solutions to address the lack of a tactical data network capability for maneuver forces. [VEND] Both Raytheon and ViaSat developed interface hardware, either in the form of an external "black box" or a PCMCIA card which provides a hardware-based link layer interface to the tactical radio set. Harris, however, developed a complete radio system, the Falcon II, to provide networking functionality through a native IP stack. All provide tactical LAN functionality but are proprietary, expensive, and limited to single-hop topologies. No support for "virtual" beyond-line-of-sight communications, via frame forwarding, is offered by these systems.

Harris also provides two specific products targeted for secure wireless local area networking arena, the SECNET11 and the SECNET54. The former is built upon the IEEE 802.11 standard and the latter, while currently available as an IEEE 802.11 compliant device, is projected to be marketed with other physical and link layer capabilities, such as IEEE 802.3 or Ethernet. The SECNET11 is certified by NSA for classified use, up to Secret. This certification simply means the data cannot be extracted without use of an appropriately keyed SECNET11 device. Further, since it provides encryption of the entire MAC frame, with the exception of the preamble, traffic and nodal analysis is prevented or mitigated. However, the physical layer is constrained by its compliance to the IEEE standard as is the preamble; therefore, the spread spectrum signal is detectable and the products are susceptible to the same detection techniques, such as war driving, as are non-secure IEEE 802.11 products [PEIKARI03; HAWK05; HARRIS05, OROS06]. Finally, the SECNET11 architecture allows either ad hoc or infrastructure mode deployment. All nodes must be within range of each other in the ad hoc mode, only the infrastructure mode allows for multi-hop connectivity. This capability is provided by way of wireless bridges that function in a wireless point-to-point mode to interconnect wireless enclaves.

Thus, even the very capable off-the-shelf capability requires investment in additional hardware and their cost-benefit must take into consideration the U.S. Services' on-going

tactical radio replacement program, embodied in the Joint Tactical Radio System (JTRS). This system will have advanced data networking capabilities built-in. It is an evolutionary development effort that is helping to establish the software-based radio (SBR) standards [SCA05]. JTRS will enable operators to reprogram the radio according to the tactical scenario. Such reprogramming includes both the signal processing and media access mechanisms. The program loads would essentially be configuration files maintained within the system and allow the radio to be upgraded without hardware replacement. As the configuration files define how the radio functions it can be configured to interoperate with other fielded systems, provided the appropriate configuration file is available. However, the JTRS devices should be viewed as a future solution and will not likely reach operational forces until 2008 or later.

This paper describes an *open-source stop-gap* system intended to demonstrate both the utility of data networking between small tactical maneuver elements and the viability of software defined media access control, combining the ruggedness of the SINCGARS radio as the physical layer, to include signal generation and reception, with the flexibility of a software-based media access control. The former, tactical data networking, is achieved using a low-overhead routing protocol that uses the notion of tactical maneuver formations to make bandwidth efficient routing decisions. The latter, software defined MAC, illustrates the flexibility and utility of implementing access protocols separate from the underlying physical layer, thus allowing the radio to be used to support various access protocols without requiring changes to the radio hardware. It takes advantage of the data port available on the SINCGARS radio and implements a derivative of the basic Aloha protocol while retaining the "voice-traffic" priority orientation of the radio. As noted, the SINCGARS radio was not designed to support multi-hop topologies. However, implementing a multi-hop ad hoc network via a ruggedized PDA or notebook computer attached to the radio's serial data port is feasible provided the link layer protocol provides for a form of bridging or packet forwarding.

This paper will focus on the two key innovations embodied in the software system: (1) a data link protocol to allow the radios to form wireless LANs, and (2) an operation-aware ad hoc routing protocol to support reliable and resource-efficient multi-hop message relays. The remainder of the paper is organized as follows. The system concept is addressed in Section II, while a design of the software-based link layer protocol is provided in Section III. Section IV then provides the key design aspects of the frame forwarding methodology that implements the multi-hop topology support of the proposed solution. This methodology takes advantage of the well-known maneuver tactics of a tank unit. The implementation and testing of the protocols are described in Section V, which introduces a simple yet effective chat application used to demonstrate the networking functionality. Finally, Section VI provides concluding observations and highlights areas which bear further exploration.

2. System Concept

This section provides a system description for a tactical data network using a software-based link protocol rather than conventional link components that are integrated in the hardware

that also provides the physical layer signal support. The section then provides a tank platoon scenario that demonstrates the utility of a multi-hop network implanted by the software-based architecture. Finally, the section presents basic design concepts that the proposed system must satisfy. Note that while the demonstration platform uses the military's SINCGARS as the physical layer component any radio that provides an RS-232 interface port may be used to provide the physical layer support, thus offering wide flexibility in the deployment of the software host. A radio providing a higher effective bandwidth would reduce the data transmission rate constraint imposed by the SINCGARS.

2.1 Components of System

Commercial wireless network systems, such as the IEEE 802.11 based wireless local area network (WLAN), integrate the physical and link layers in a single hardware device. This integration limits the employment of the WLAN to the radio spectrum and signal waveform hardwired into the network interface device. This paper proposes *decoupling* the link functionality from the physical layer functionality and allowing the link layer to be implemented independently from the physical layer, as illustrated in Figure 1. The SINCGARS provides a useful platform for demonstrating this concept without requiring any modifications to the hardware.

The SINCGARS provides both voice and data communications support, however, the radio is used predominantly as a voice network by ground maneuver forces. The data communications capability is provided by a single RS-232 compliant serial interface on the radio. This connection to the interface is via a circular six-pen connector. In order to connect a notebook computer to the radio the user must have a cable with a serial connector, for connection to the computer, on one end and a properly configured circular connector, for connection to the radio, on the other. The radio accepts the serial data from the computer and generates a waveform for transmission according to the data rate and hopping pattern manually selected by the user. The SINCGARS effectively serves as the physical layer for the connected data device. It is the responsibility of the data device to send the data to the radio set *as a bit stream* and collect incoming bits from the radio set into frames. Additionally, all link layer functionality must be performed by the data device.

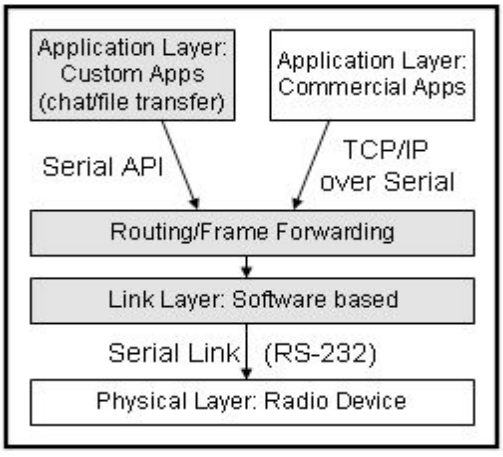

Figure 1: System Design

As the SINCGARS is extremely constrained by the available bandwidth, the link layer must be very efficient in order to ensure the effective use of the supportable transmission rates. Functions included in the demonstration system include media access control, error detection and recovery, framing, and frame forwarding. The first is essential as the media is shared and access must be coordinated. Traditional wireless access measures might be incorporated to provide this coordination, and indeed, the simple CSMA methodology of 802.11x family networks for small frames was modeled for one of the access

control mechanisms. As wireless communications are prone to interference due to multipath-reception and hidden terminals, the bit error rates of the links are much less robust than wired or fiber optic links. Thus, a means of detecting and recovering from such induced errors is essential. Framing organizes the raw bits into meaningful fields, while frame forwarding is essential for extending the range of the network beyond line-of-sight (LOS). It should be noted that the single hop range of the SINCGARS is superior to the range of the commercial wireless LAN devices, however, it is still insufficient to support maneuver forces on the move, as terrain may often mask the reception of forces that would otherwise be in range. Multi-hop networking offers a means of overcoming this masking. This is provided by way of frame forwarding which bases the next hop destination on route information derived from the routing protocol described in Section 4.

Each of these link layer functions, though normally implemented in the network interface device firmware, may be implemented in software. Figure 1 is a system diagram showing the basic components of a network host where the physical layer, providing signal transmission and reception, is provided by an off-the-shelf radio, while the remaining network functionality is provided in software. The shaded blocks of the diagram represent those functions implemented by the effort reported in this paper. In particular, since the SINCGARS provides an external serial interface port, it is possible to implement the link layer functions in the data device, making them independent from the radio set. The link layer functions are separated here from the routing and frame forwarding functions. This is to ensure the routing or forwarding algorithms are sufficiently independent of the other functions so that various routing algorithms may be introduced without modification to the link layer. The current Java library includes an application programming interface (API) for communicating directly with the serial port analogous to interfacing with files. It should be noted, however, that the Java serial API does not provide for TCP/UDP socket functionality. Thus, the transport layer must be independently implemented if the desired applications require socket support. Such socket support may be available from venders and incorporated in the host to provide easier adoption of commercial-off-the-shelf applications, such as e-mail, or government-off-the-shelf applications such as C2PC, a PC-based command and control system.

As noted, conventionally, link layer protocols are implemented in hardware, perhaps using application-specific integrated circuits (ASICs). This limits the system administrator's or user's ability to adapt the link protocol to the target environment. By implementing the link protocol in software the system can be appropriately configured for the deployment situation. Further, it allows the physical layer component to be exchanged for an alternate device without requiring the link layer to be modified.

2.2 System Utility

Military units maneuver in formations to increase command and control effectiveness. The formations are established according to established tactics, techniques, and procedures and establish positions sectors or areas of responsibility. The formation may change in response to changing tactical situations. In general, the formations for wheeled and tracked vehicles may be classified as a column, a line, or a wedge, the first normally used when transiting an

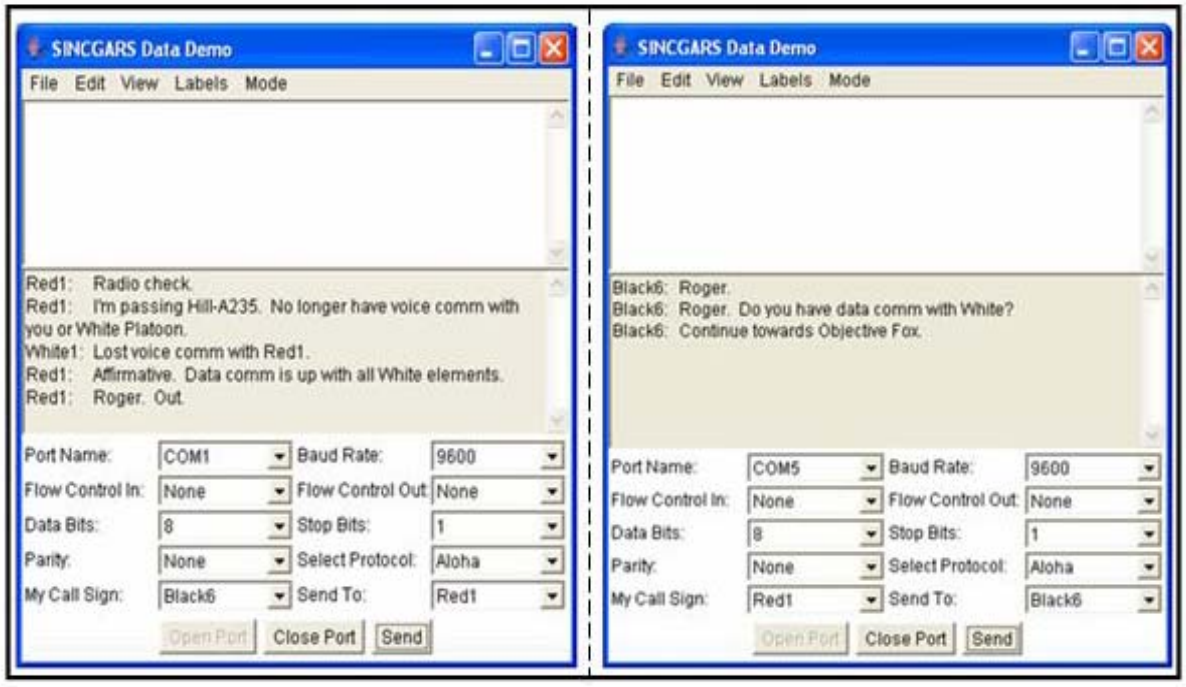

Figure 2: Tactical Chat Application Screenshot

area and the latter two for engaging enemy forces. Each entity, or vehicle, has a paired companion, or wingmen, with which it must maintain radio contact regardless of the tactical situation. This paper specifically uses a tank company as the basis for designing the frame forwarding protocol. A tank company is nominally comprised of three platoons, each of which contains four tanks. A company commander is responsible for the operation of the company as a whole. Each platoon is assigned a commander who is responsible to the company commander for the direction and control of its platoon. Commands, directives, and requests for tactical information typically flow from the company commander to the platoon commanders for execution, while reports and responses to requests flow from the platoon elements back up through the platoon commander to the company commander.

As link layer functions are transparent to the user, this paper presents a custom chat application that uses the link layer functions and the routing and frame forwarding capability in order to demonstrate their execution. Figure 2 shows a typical exchange between platoon commanders using the custom chat application. The left pane is the display seen by Black6, the company commander, while the right pane is the display on Red1's terminal. In the example a tank company is moving towards its objective. As the lead platoon moves past a large hill, its commander, Red1, notices that he has lost voice communication with other elements of the company that are on the far side of the hill. Red1 uses the data network to notify his commander, Black6. Black6 also receives a report from White1, indicating a loss of voice communication with Red1. Black6 then ensures that Red1 is able to communicate with other elements of the company via the chat application that uses the multi-hop, frame forwarding network described in this paper. Satisfied that all units can still synchronize actions and maintain a shared situational awareness, Balck6 directs Red1 to continue to move towards the objective.

Without the frame forwarding functionality provided by this effort the lead element would have been cut-off from the other units until all had passed the obstruction, thus inhibiting the ability of the company to react to evolving tactical conditions.

2.3 Design Requirements

As the proposed system is intended to support tactical maneuver forces, the system should consider the tactics, techniques, and procedures employed by such forces. Small tactical units extensively employ voice communications. The design of the data network utility should not adversely impact the use of the underlying physical device for its primary function, voice communications. The key system requirements, driven by the target applications, include:

a) Retain the priority of voice traffic. Voice traffic remains a crucial C2 tool. Data transmissions must not impair voice traffic;

b) Provide reliable transfer of data. Data integrity is essential to C2. The link must not lose data or degrade its quality.

c) Assure sustainable throughput. The network must provide assured data transfer. It must be able to support varying traffic loads without significant degradation.

d) Easy to configure. The typical user will not be a network administrator; therefore, the user interface must be intuitive.

e) Overcome LOS limitations. The network must overcome the line-of-sight limitations of the underlying infrastructure. To do so it must support frame forwarding and provide a route determination protocol.

f) Enhance the interoperability of maneuver elements. Unless the proposed system enables the maneuver elements to better coordinate their activities it provides little or no operational benefit.

The final two requirements listed above are particularly focused on the nature of the tactical maneuver forces for which this network capability is intended. The modern battlefield is characterized by highly mobile forces. This mobility serves several functions, among which are movement to engagement of enemy forces, massing of combat fires on adversary centers of gravity, and force survivability. The impact of the failure to maintain force maneuverability can be readily seen by the devastation experienced by the Iraqi forces during the Persian Gulf War of 1991 [MC2000]. That conflict also serves to highlight the speed with which maneuver forces may operate, such that what was at one time a single hop requirement, with respect to network connectivity, may evolve into a multi-hop requirement. At the speed at which the requirement evolves, neither the systems administrator nor the user can afford to reconfigure the network – it must adapt autonomously. Two key tenets of maneuver forces, massing of fires and force dispersal, require that maneuver forces be able to exchange information rapidly. The modern battlefield may just as likely be an urban location, as experienced in Mogadishu, Somalia or Baghdad, Iraq. In such environments the

radio line of sight may be severely hampered by manmade structures. Even in open areas, hills or rock escarpments may reduce radio coverage. In such cases, the ability to relay information autonomously can significantly enhance force interoperability. As radio voice communications do not provide for autonomous relay, tactical chat may provide a significant advantage over current capabilities. Further, depending on bandwidth constraints and the various radio platforms available, implementing Voice-over-IP applications may provide switched voice capabilities to maneuver forces equipped with minimal radio systems.

3. Design of Data Link Protocol

Voice-centric tactical radios are characterized by low data rates when operated in data modes. For example, the serial data interface of SINCGARS has a maximum baud rate of 9600. Therefore, the main design objective for the data link protocol is to maximize the throughput of a data connection while maintaining some level of fairness when multiple data connections are active at the same time. The data link protocol described in this paper is called the SINCGARS Layer 2 Interface (SL2I). In the rest of this section, we describe the three key components of the protocol: media access control, framing, and error control.

3.1 Media Access Control

Nodes in an ad hoc network share a common transmission channel. If two or more nodes send data into that channel simultaneously, their signals may collide, resulting in garbled bits at the receivers. Therefore, some form of media access control (MAC) is required to arbitrate each node's access to the channel. A whole spectrum of MAC design choices is possible. At one extreme, contention-free protocols, such as Frequency Division Multiple Access (FDMA) and Time Division Multiple Access (TDMA), seek to eliminate collisions via the use of specialized hardware or through the exchange of control messages among the transmitting nodes. At the other end, contention-based protocols do not try to prevent collisions, but attempt to minimize them or efficiently overcome collisions when they occur.

We observed that without hardware assistance, contention-based protocols incur significantly less communication overhead than contention-free protocols. Since we are targeting small, bandwidth-constrained networks made of legacy equipment, we have chosen to base SL2I on contention-based protocols, which are variations of the classic Aloha and Carrier Sense Multiple Access (CSMA).

Aloha Functionality. The simplest of contention-based MAC protocols, Aloha allows for random access to the shared channel [Abramson 70][KR05]. Nodes with data to send simply transmit at will. The source node of a given data frame considers the transmission successful only if an acknowledgement (ACK) frame is received in response. The absence (or late arrival) of the ACK frame will automatically trigger a retransmission of the data frame. The retransmitted frames are sent only after a randomly chosen back off period has expired. Nodes retransmitting frames are not likely to choose the same back off period, and are thus not likely to experience subsequent collisions for a given frame. SL2I includes the

Aloha protocol as its default MAC algorithm. In that mode, a node will make up to 5 attempts to deliver each frame.

CSMA Functionality. An improvement over Aloha, the CSMA protocol avoids many collisions by sensing the shared medium prior to sending each frame. When the channel is busy, the frame is held until the channel is free [KR05]. SINCGARS clears the voltage on one of the pins of the serial interface whenever it is busy transmitting or receiving. SL2I may be set to operate in the CSMA mode to take advantage of this feature. In that mode, SL2I monitors the voltage level of that particular pin. The presence of a voltage indicates an idle channel. In the absence of this voltage, SL2I nodes in CSMA mode will buffer pending data frames until the channel is idle. The CSMA protocol cannot, however, prevent all collisions. It is possible for two or more nodes to simultaneously sense an idle channel and then send frames into the medium resulting in a collision. In such an event, SL2I nodes in CSMA mode will retransmit their respective frames after a random back off period, with up to 5 attempts.

Figure 3 illustrates the basic steps of the SL2I MAC algorithm. In Aloha mode, data is encapsulated and sent immediately. In CSMA mode, encapsulated data is buffered until an idle channel is detected. As a stop-and-wait algorithm, SL2I then awaits the receipt of an acknowledgement frame. The random back off algorithm is a simple variation of the classic exponential back-off algorithm. If no ACK is received in a timely fashion, a random back off period of 0 – 1000 milliseconds is chosen. Subsequent ACK timeouts do not lead to any expansion of the set of values used to determine the back off period. This deviation from the classic exponential back-off algorithm reflects the expectation that the user population of the tactical data network established with the tactical radios will be small, resulting in less risk of multiple collisions. Only upon expiration of the back off period may the frame be retransmitted.

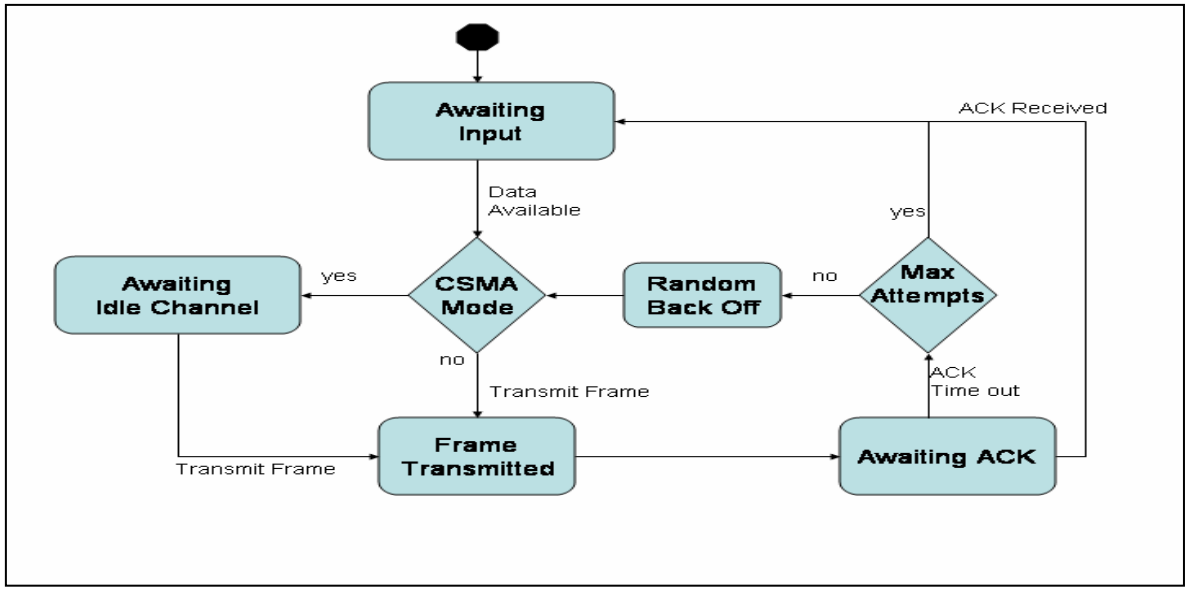

Figure 3. SL2I Media Access Control Flow Diagram

3.2 SL2I Framing

The design of SL2I is driven by throughput efficiency. To this end, an SL21 frame header contains only six fields. Maximizing per-frame payload, frame headers are kept small, with each field being one byte in length. Likewise, we have attempted to minimize the number of link-layer control messages.

SL2I's maximum payload size is 600 bytes, resulting in a maximum frame size of 608 bytes or about 500 ms of transmission time when the link capacity is 9600 bps. This size is intentionally chosen to balance throughput efficiency with fairness. A larger size would be desired from the perspective of maximizing throughput efficiency. However, too large a payload size could lead to one radio holding the transmission medium for too long and starving new data connections. Messages that exceed this maximum are fragmented into multiple frames. Each frame is pre-pended with a layer two header. This header is only six bytes long and is depicted in Figure 4. It should be noted that the intermediate address fields are for routing, a layer three concern which will be discussed in the Section 4.

1 byte	1 byte	1 byte	1 byte	1 byte	1 byte	0-600 bytes	1 byte	1 byte
Frame Type	Source Address	Destination Address	Source Address	Destination Address	Sequence Number	**Payload**	Checksum	End Of Frame
			Intermediate Addresses					

Figure 4. SL2I Frame Header Format

The frame header is comprised of a frame type field, four address fields, and a sequence number field. These six fields are the first six fields in each frame type. An 8-bit checksum value is calculated, based upon frame payload and header content, and appended to the frame before transmission. Finally, an ASCII End of Transmission Block (ETB) character is inserted into the End of Frame field. The only value allowed in this field is the ASCII ETB (decimal 23).

Four types of data or control frames are currently defined. They are listed in Table 1. Only the Data and File Transfer types can be designated by upper layer processes. Acknowledgement (ACK) frames are generated by SL2I. All others are control message frames related to routing and are discussed in Section 4.

Frame Type	Field Value (ASCII character)
Data	"T"
Acknowledgement	"A"
File Transfer	"I"
Routing Control	
Hello or Hello Response	"H"
Route Request	"Q"
Route Response	"R"

Table 1. SL2I Frame Types

3.3 Error Control and Flow Control

SL2I uses a one-byte checksum value for error detection. Each node receiving a properly formed frame will calculate a checksum for the frame based on its header and payload. This calculated checksum is compared to the checksum value appended to the received frame. The incoming frame is passed up to the upper layer and an ACK for the frame is sent only if the checksum values match. As mentioned previously, frames for which no ACK is received are resent. A frame may be retransmitted up to a maximum number of five times and if an ACK is still not received after all the retransmissions, the frame will be dropped. Then it is up to the upper layer application process to take corrective actions.

SL2I is a stop and wait protocol, i.e., it will not transmit a new frame until the previous frame has been either acknowledged by the receiver or dropped after a maximum number of retransmission failures. This simple form of flow control is motivated by two observations. First, most tactical messages are short. Second, it is more important to minimize the channel access time for all nodes than to maximize the throughput of one node. Allowing one node to transmit multiple frames at a time would reduce channel availability for other nodes.

4. Design of Ad Hoc Routing Protocol

We have developed a hybrid ad hoc routing algorithm to achieve multi-hop message relay. Termed "Expected Relative Positioning routing with Congestion Avoidance (ERP/CA)", the protocol combines several features of proactive, table-driven routing algorithms with those of reactive, on-demand algorithms. Its novel approach to route selection draws upon knowledge of military units on the move and is designed specifically for tactical mobile ad hoc networks (MANETs). ERP's low overhead and persistent-path preference make it a good fit for low-bandwidth networks like those dependent upon legacy SINCGARS radios.

The most salient features of the ERP/CA algorithm are presented in this section. There are two conceptual goals that underlie the entire design effort: (i) operation-awareness, and (ii) bandwidth-efficiency. These goals and the main approaches to achieve them are described first to shed light into the detailed and sometimes subtle trade-offs that have been made for the ERP/CA algorithm.

4.1 Operation-Aware: Exploiting Operational Knowledge about Node Movement

Most MANET routing algorithms are designed specifically for civilian networks. A civilian MANET can be expected to have a random topology. Military units however almost never move in random directions at random speeds. Instead, military units travel in tactical formations. These formations are designed primarily to facilitate Tactics, Techniques, and Procedures (TTPs) that provide a combat advantage over the enemy. The formations are also designed to maintain communication between unit leaders and their subordinate elements. A platoon commander, for example will tend to maintain a relative position in the formation that facilitates LOS radio communication with all members of his platoon. At the

lowest level, individual platoon members tend to maintain radio contact with their respective wingmen. Individual elements of a tactical formation do not normally move alone.

Figure 5 depicts a tank company (comprised of three platoons) in a wedge formation. In each platoon, the 1-element is a Platoon Commander and wingman for the 2-element. In each platoon, nodes three and four are wingmen for each other. Charlie-6 and Charlie-5 are Company Commander and Executive Officer, respectively, and are wingmen for one another.

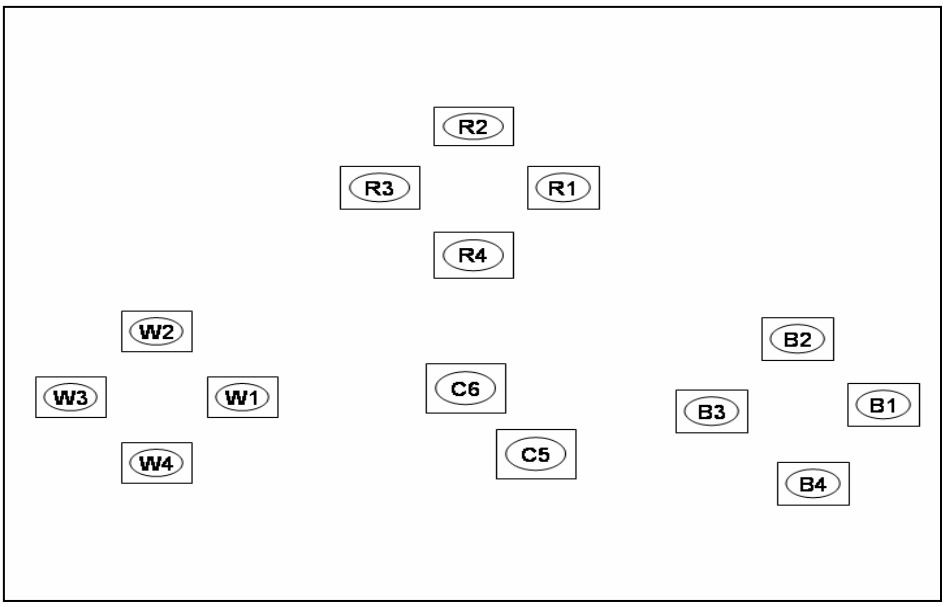

Figure 5. Tank Company in Wedge Formation

The call sign used by a given element of the company inherently encompasses information about the element's status and relationship to other elements of the company. The call sign "Red2" (R2 in Figure 5), for example, encompasses the following:

a) Member of Red Platoon
b) Not a commander
c) Wingman for Red1
d) Expected to move with Red1 in particular and Red platoon in general

Network addresses used by ERP/CA are simple integers. These addresses are unique, numerical representations of the call signs and so encompass the same TTP-based information. Military TTPs govern the movement of elements in a formation. ERP/CA uses this information to route data along paths that tend to remain persistent. Because it is aware of policies governing node movement, ERP/CA would, for example, prefer a path through a wingman where no direct path exists. No other multi-hop path can be expected to be more persistent than that which passes through the destination node's wingman.

Shortest path is the metric of last resort for ERP/CA. Primarily, route choices are based upon minimizing the frequency of route changes. Persistent, reliable routes are preferred to those that are likely to change frequently due to node mobility and radio range limitations.
It tends to be true that data destined for any node within a given platoon can be reliably routed through its respective platoon commander node. Data routed through the node's wingman will have an even greater expectancy of persistently, reliable delivery. ERP/CA favors such routes, when a direct link is not available.

Related Work. Most current MANET routing protocols, including Optimized Link State Routing Protocol (OLSR) [Johnson99][Perkins99][CJ03], make routing decisions based upon more temporary characteristics like the existence of link or multi-link connectivity. Because these protocols ignore the policy-based relationships, the choice of intermediate nodes along a multi-hop path tends to be somewhat arbitrary and subject to frequent change. Other protocols like Associativity Based Routing (ABR) also attempt to route based on node relationships [Toh96]. But these protocols estimate node relationships at run time, and as such produce significantly more overhead than ERP/CA, which receives node relationships as explicit user input to the routing protocol. Because node relationships in a military context are policy-based, ERP/CA routing decisions can be made with a valid expectation of persistence with minimum communication overhead.

4.2 Bandwidth-Efficient: Minimizing Overhead of Control Traffic

Legacy tactical radios, like the SINCGARS, were designed primarily for voice traffic. The bandwidth for data communications is limited. Specifically, the SINCGARS has a maximum data rate of 9,600 bps. Routing algorithms with excessive overhead would take away too much of this scarce resource from data packets. ERP/CA's control and administrative messages only minimally interfere with the transfer of data.

As a reactive protocol, ERP/CA uses no announcement frames to proactively discover routes. Likewise, no routing tables are exchanged—completely eliminating the resource draining overhead associated with such control frames. Route Request Frames are sent only in response to user demand. In the absence of such demand, new routes will not be discovered and stale routes will not be rediscovered. Valid routes are, however, cached to avoid unnecessary rediscovery of recently used routes.

The protocol also attempts to minimize the effects of flooding during route request sequences. Intermediate nodes with valid routes to the destination prepare Route Response Messages instead of forwarding the request. Also, as described in Section 4.4, each node will enter a unique wait-period prior to responding to a route request. Because the first response to a request preempts all pending responses, overhead traffic is further reduced.

Finally, even for proactive protocols, route maintenance can contribute significantly to administrative overhead. ERP/CA does not use any route error messages. Instead, any node along the path from source to destination that discovers a bad link will, on-demand, dynamically discover a new route using the process discussed in Section 4.4. A link is

considered bad when it becomes stale due to inactivity or upon repeated failures to acknowledge frames sent across the link. The ERP algorithm's preference for persistent routes minimizes the need for route changes, even with multiple hops between source and destination.

4.3 Initial Neighbor Discovery

Nodes attempting to enter the network broadcast a single Hello Message. At no other time does a node send a hello message. The hello is not flooded. When a node sends a Hello Message, the recipients, all one-hop neighbors, send a Hello Response. Upon receipt of each hello response the initiating node will add the responding node to its routing table. Likewise, nodes receiving a Hello Message from the initiator will add the new node to their routing table. This allows ERP nodes to take advantage of the proactive, low-latency characteristics of table-driven protocols for their initial one-hop neighborhood. As only a single Hello Message is sent per node, the overhead associated with proactive protocols is minimized for the one-hop neighborhood. This savings in overhead traffic is further compounded by the fact that Hello Messages are not flooded, and no routing tables are ever exchanged.

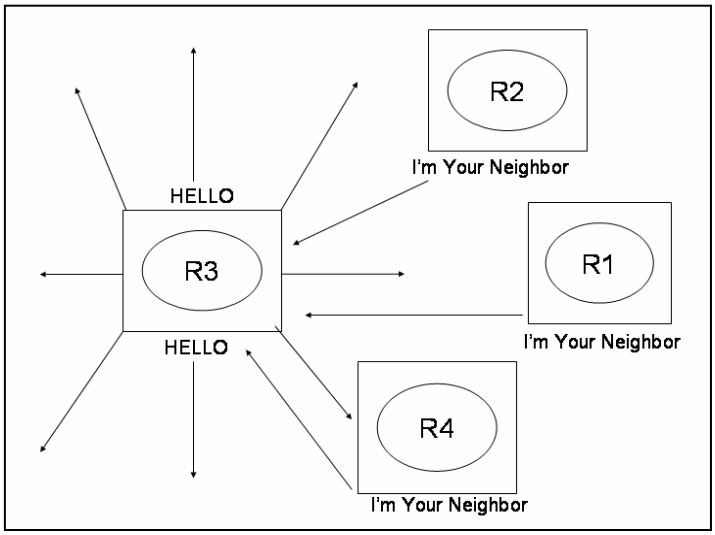

Figure 6. Initial Neighbor Discovery

In Figure 6, Red-3 sends a Hello Message to discover its initial neighborhood. The Hello Message is broadcasted and processed by all nodes in the one-hop neighborhood (i.e., fellow Red Platoon members).

4.4 Dynamic Route Discovery

ERP/CA allows for the dynamic discovery of multi-hop routes. This process begins with a broadcasted Route Request Message. In Figure 7, White-3 is requesting a route for Red-3. In this scenario, assume that no other white element has a valid route to Red-3 either, and so, the request is flooded. Further, assume that Red-1 and Red-4 both have valid routes to

Red-3. The flooded Route Request stops at Red-1 and Red-4. Each node with a valid route to a requested node will prepare a response after receiving a Route Request message. Before sending its response, however, each node will enter a unique Route Response Wait (RRW) period *that is a function of its expected relationship to the requested node*. In general, the wait is shorter for a node that has a "closer" relationship. The exact algorithm for determining this wait period is given in Section 4.6.

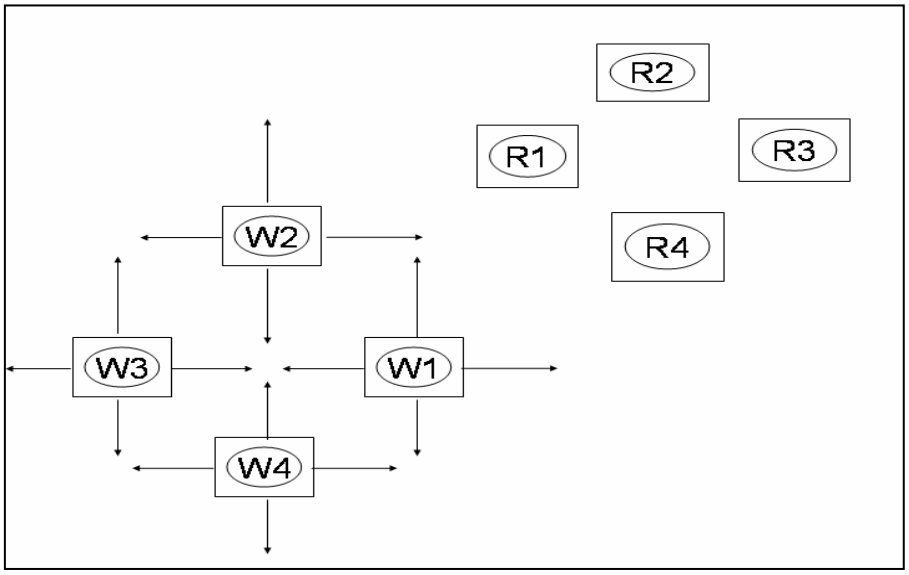

Figure 7. Controlled Flooding of Route Request

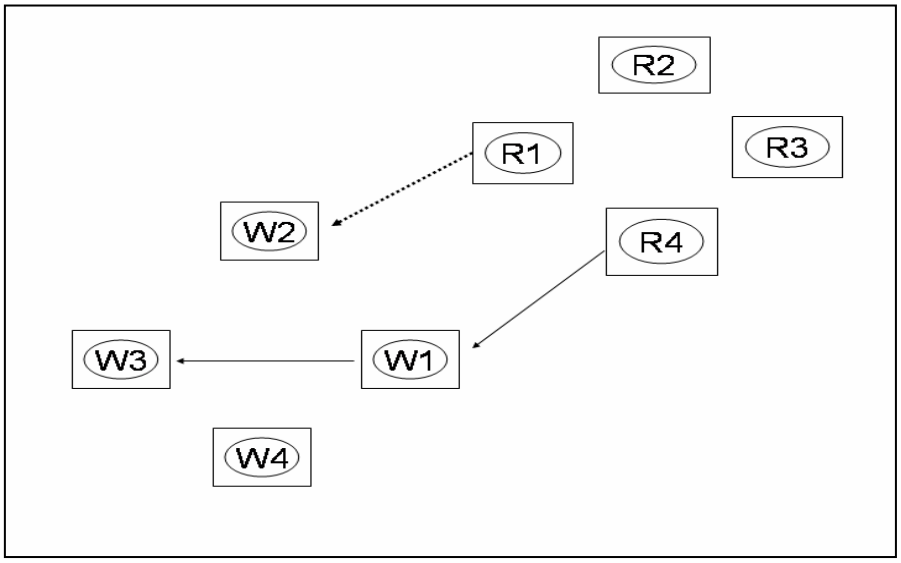

Figure 8. Unicast Multihop Route Response

Since there is no direct link to Red-3 from any node in white-platoon, the response from Red-3's wingman (Red-4) will be the first. This Route Response Message from Red-4 preempts Red-1's response. Figure 8 shows the unicast response from Red-4 to White-1 and then from White-1 to White-3. The dotted line from Red-1 to White-2 indicates a

pending response that was never sent. White-3 will cache the first response to its request and discard any others.

4.5 Route Caching

Each node maintains a cache of valid routes it has recently used to deliver both locally generated and forwarded frames. The route cache includes only the destination node's address, the next-hop node's address, and a route freshness timestamp. By default, routes are purged after ten minutes of in activity. Route freshness, however, is updated upon receipt of each ACK from the respective next-hop node. Repeated failure to receive acknowledgement of frames sent across a link are indicators of excessive congestion or possible link failure. Such failures lead to removal of routes containing the bad link.

4.6 Congestion Avoidance

Network congestion causes delays in data delivery. To facilitate the timely delivery of data ERP/CA considers node congestion in selecting routes. Nodes that are heavily congested are avoided in favor of those that are less congested. The RRW parameter mentioned in Section 4.4 is increased in proportion to the size of a node's route cache. (The exact algorithm is given in Section 4.7.) Since stale routes are removed from the cache, nodes with large caches have recent activity with a large number of other nodes. These large-cache nodes are more likely to experience traffic congestion.

Some routing algorithms use periodic probes or announcements to determine or distribute per node congestion information. Probes and announcements compete with data for access to the shared medium and consume limited bandwidth. They contribute to the congestion that they intend to measure. With ERP/CA, congestion avoidance is the responsibility of nodes responding to a route request, not the requestor. This is opposite from algorithms that use probes or announcements, which provide information to source nodes to make routing decisions that avoid congestion.

Avoiding congestion in routing decisions serves two purposes. It helps to minimize latency and it supports ERP's persistent path preference. When congestion causes retransmissions, the delivery of data is delayed. When congestion is so great that frames are dropped, reliability and persistence of the link is reduced. A path from source to destination is only as persistent and reliable as its weakest link. As such, ERP/CA seeks to avoid heavily congested routes.

Figure 9 depicts a scenario in which White-3 has data to send to Blue-3. Blue-3 is beyond the LOS range of White-3. Blue-3 will respond to the flooded route request before its wingman or Platoon commander does. As a result, none of the subsequently pending route responses related to W3's request will include a path through Blue-3's wingman or Platoon Commander. In this scenario, the route selected by the ERP/CA algorithm will be based only on avoiding likely congestion. Among those preparing a response to W3, including D5 and D6 as a potential choice for the next hop to the destination, the first node to unicast a response to White-3 will be the node with the least congestion.

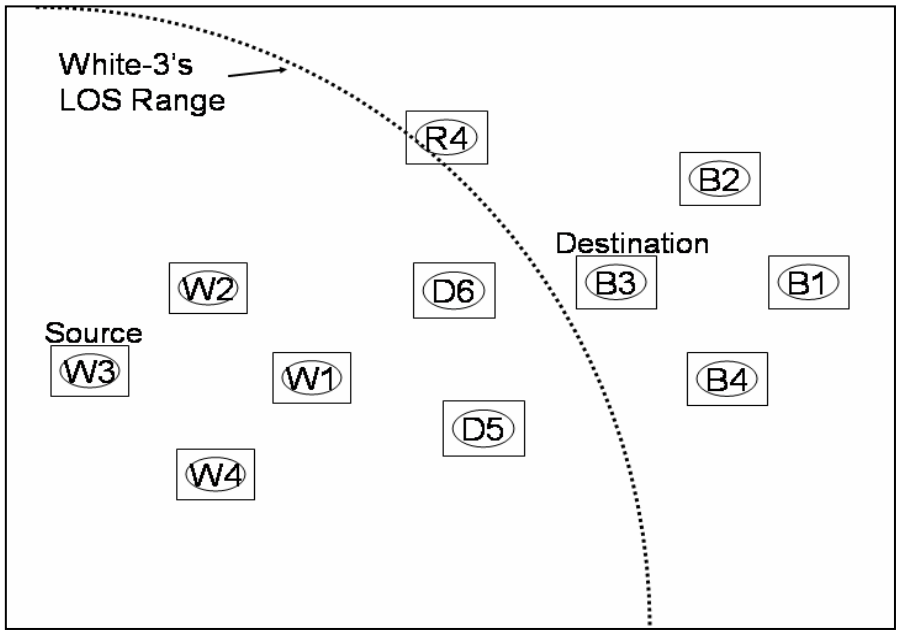

Figure 9. Congestion Avoidance Example

4.7. Route Selection Algorithm

For ERP/CA the preference of a route is embedded in how long nodes will wait before sending their response to a Route Request. The Route Response Waiting period (RRW) has three components, Categorical Wait (CW), Congestion Avoidance Value (CAV), and Individual Response Wait (IRW), as follows:

$$\mathbf{RRW = CW + CAV + IRW} \quad (1)$$

Upon receipt of a Route Request, nodes with a valid route to the requested destination will self-assign values to each of the RRW components. Assigned values are based upon a node's relation to the requested destination node. A destination node and its wingman will always set the CAV and IRW to zero.

Categorical Wait (CW). CW's are weighted heaviest and have four potential values as shown in Table 2. The Good category is self-assigned to any node with a route to the destination and not in one of the other categories. Of the categorical wait values, Good incurs the largest wait time. The next category is Better. The Better wait value is smaller than Good, and is used by the platoon commander of the destination node. The wingmen

relationship is assigned the Best category with an even smaller CW value. For wingmen, the RRW is equal to the CW. Obviously, the ultimate and most reliable path form source to destination is a direct link. Destination nodes are in the Direct Link category and self-assigned a CW value of zero and an RRW of zero. Destination nodes respond without delay to Route Requests.

Category	Wait Time Assigned (ms)
GOOD	1500
BETTER	1000
BEST	500
DIRECT LINK	0

Table 2. CW Values

Congestion Avoidance Value (CAV). CAV values are determined by multiplying the number of routing table entries by the CAV-weight, with a default value of 21.5 ms. This value is then rounded to the nearest integer value. Each node's CAV is measured in milliseconds and is proportionate to the size of its routing table. The larger the table, the larger the CAV.

Individual Response Wait (IRW). IRW values are determined by multiplying the node's unique integer address by the IRW-weight, with a default value of 12.25 ms. This value is then rounded to the nearest integer value. Each node self-assigns an IRW value, which is used to correlate its network address with a temporal value in milliseconds. This unique IRW value is used to break ties when RRW values would otherwise be equal, and it is used to ensure that multiple nodes, in general, will not transmit Route Responses simultaneously.

5. Implementation and Initial Testing of Protocols

The protocols described in this paper were implemented in Java using all standard packages. The implementation contains a total of 14 Java classes and about 5,500 lines of code. To facilitate the functional testing of the data link and routing protocols, a sample chat application (see Figure 2) was developed. This application ran directly above SL2I, and allowed the user to make a per-message designation of message destination. Finally, the application also allowed users to designate files to be transferred via SL2I.

We recently successfully demonstrated the system functionality to the US Marine Corp Systems Command (MARSYSCOM). Demonstration of the data link protocols involved a simple three node network. These nodes were labeled A, B, and C. Each node was comprised of a laptop PC connected directly to SINCGARS radio via custom serial cable. Initially, each node was placed within range of all others, and all nodes operated in Aloha mode. Each node was able to send and receive text messages with all other nodes. File transfers were equally successful. This portion of the functional testing concluded with a simultaneous transmission of frames. At each node, SL2I was able to de-conflict the transmissions via random back-offs. All frames were successfully delivered. Each of these

tests was repeated with each node using the CSMA MAC protocol. Again, each test was successful.

Voice-centric radios, like the SINCGARS, are designed such that "voice-traffic" is given priority over "data-traffic". The functional testing of SL2I included a scenario in which attempts were made to send voice and data simultaneously. In accordance with tactical radio procedures, voice transmissions were brief during this testing. The text messages sent were constructed to ensure that multiple frames would be required for full transmission. The testing showed that SL2I "data-traffic" did not inhibit "voice-traffic" in any way. Further, each data frame was successfully received before all send attempts were exhausted.

Demonstration of the routing protocol included the use of an RF attenuator. Connecting this device between the radio's antenna connector and its antenna at one node, C, allowed us to simulate extreme distance. Nodes A and B were configured as discussed above. With the attenuator in use, and Node-B placed in close proximity to Node-C, each node began its Initial Neighbor Discovery. Nodes A and C discovered only Node-B, and Node-B discovered both A and C, producing a classic hidden node scenario. On demand, Node-A was able to successfully discover and utilize the route (via Node-B) to Node-C. Node-C was equally successful at discovering the route to Node-A. Text messages and file transfers were successfully exchanged between all the nodes.

6. Concluding Remarks

We have successfully demonstrated it is feasible to create mobile ad hoc *data* networking software that directly runs over legacy voice-centric tactical radios, requiring no third-party hardware or proprietary protocols. However, this is just an initial step. It is our desire to make the software widely available so that the system can be further improved and ready for field deployment. In the near term, the following four issues require attention:

a) Security certification. We do not anticipate major problems since the system is external to the radios. The data transmitted using the system will receive the same level of protection as the current voice traffic.

b) Performance Optimization. More tests involving large topologies are required to determine strategies for fine-tuning the parameters (e.g., the CW values) of the protocols based on operational scenarios. Computer simulation can be used to overcome the limited availability of the radios.

c) Embedded Reporting of Position Location Information (PLI). One solution is to expand the SL2I header to include a field for PLI content. SL2I will then report updates of PLI to the upper layer applications.

d) External Connectivity. It would greatly enhance the utility of the system to the users if the system can be integrated with existing tactical data networks, thereby extending the reach of tactical chat and file transfer applications across multiple units. Another question worthy of research is whether it is feasible to combine SL2I with IP-over-serial modules and create a system that is IP compatible.

7. References

[Abramson 70] N. Abramson, "The Aloha System - Another Alternative for Computer Communications," in *Proc. Fall Joint Computer Conference, AFIPS Conference*, Vol. 37, pp. 281-285, 1970.

[Bates04] Jason Bates. "Congress Worried JTRS, Combat System Timelines Don't Mesh". July 15, 2004.
(http://www.isrjournal.com/story.php?F=328018. Last visited 23 December 2005)

[CJ03] Clausen, T., and Jackquet, P., "Optimized Link State Routing Protocol," RFC 3626, 2003.

[EngDoc00] SINCGARS Program Management (PM) Office, U.S. Army PM-TRCS, Fort Monmouth, New Jersey, "System Engineering Document For The Ground Radio SINCGARS System Improvement Program (SIP And ASIP)", Volume 1, JANUARY 24, 2000.

[FAS99] Federation of American Scientists, "Single Channel Ground and Airborne Radio System," FAS Military Analysis Network, March 1999
(http://www.fas.org/man/dod-101/sys/land/sincgars.htm. Last visited 11 January 2006)

[Haas98] Z. J. Haas and M.R. Pearlman, "The Performance of Query Control Schemes for the Zone Routing Protocol," in *Proc. ACM SIGCOMM'98*.

[HAWK05] Hawk, Jeff, "Wireles Warriors Secure in Their Knowledge," SIGNAL, AFCEA, August 2005 (http://www.afcea.org/signal/articles/anmviewer.asp?a=1000)

[HARRIS05] "Secure Communications Solutions: SecNet11; SecNet54," Harris Corporation Government Communications Systems Division (GCSD) Product Information, (http://www.govcomm.harris.com/secure-comm/ last visited 18 January 2006)

[IRC] "Internet Relay Chat," Wikimedia Foundation Inc.
(http://en.wikipedia.org/wiki/Internet_Relay_Chat. Last visited 11 January 2006)

[Johnson99] David B. Johnson, Davis A. Maltz, "The Dynamic Source Routing Protocol for Mobile Ad Hoc Networks," IETF Draft, October 1999.

[KR05] Kurose, J.F., and Ross, K.W., *Computer Networking: A Top-down Approach Featuring the Internet,* 3rd ed., Addison-Wesley, 2005.

[MC2000] McCaffrey, "Lessons of Desert Storm," Joint Forces Quarterly, Winter 2000
http://www.dtic.mil/doctrine/jel/jfq_pubs/1834.pdf

[NCE05] Department of Defense Joint Staff J7, Net-Centric Environment Joint Functional Concept, v1.0. April 2005 (http://www.dtic.mil/futurejointwarfare/concepts/netcentric_jfc.pdf. Last visited 11 January 2006)

[NCE05] Department of Defense Joint Staff J7, Net-Centric Operational Environment Joint Integrating Concept, v1.0; October 05 (http://www.dtic.mil/futurejointwarfare/concepts/netcentric_jfc.pdf. Last visited 11 January 2006)

[OROS06] extracts from an e-mail exchanges between NPS research faculty members Major Carl Oros, USMC; Brian Steckler; Rex Buddenberg; and the authors, February 9, 2006

[Peikari03] Cyrus Peikari and Seth Fogie, *An Insider's Guide to Protecting Your Wireless Network*

[Perkins99] Charles E. Perkins, Elizabeth M. Roye
r, Samir R. Das, "Ad Hoc On-demand Distance Vector Routing," IETF Draft, October 1999.

[SCA05] JTRS Joint Program Executive Office (SPAWAR Systems Command), "Technical Overview: Software Communications Architecture," August 2005 (http://jtrs.army.mil/sections/technicalinformation/fset_technical.html. Last visited 11 January 2006. Also see http://jtrs.army.mil/documents/jtrs%2Bbrochure.pdf.)

[Toh96] Chai-Keong Toh, "A Novel Distributed Routing Protocol to Support Ad hoc Mobile Computing," in *Proc. 1996 IEEE 15th Annual Int'l. Phoenix Conf. Comp. and Commun.,* pp. 480-86, March 1996.

[Toh99] C.-K. Toh, "Long-lived Ad-Hoc Routing based on the concept of Associativity," IETF Draft, March 1999.

[VEND] vendor web sites:

http://www.viasat.com/datacontrollers/

http://www.raytheon.com/products/stellent/groups/public/documents/content/cms01_058051.pdf

http://www.rfcomm.harris.com/products/tactical-radio-communications/an-prc117f.pdf

http://download.harris.com/app/public_download.asp?fid=843