NAVAL POSTGRADUATE SCHOOL

MONTEREY, CALIFORNIA

THESIS

A SOFTWARE-BASED NETWORK INFRASTRUCTURE FOR MOBILE AD HOC DATA NETWORKING IN SUPPORT OF SMALL TACTICAL UNITS USING THE SINCGARS RADIO

by

Steven R. Brand

March 2006

Thesis Advisor:	Geoffrey Xie
Thesis Co-advisor:	John Gibson

Approved for public release; distribution is unlimited

THIS PAGE INTENTIONALLY LEFT BLANK

REPORT DOCUMENTATION PAGE	*Form Approved OMB No. 0704-0188*

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instruction, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188) Washington DC 20503.

1. AGENCY USE ONLY *(Leave blank)*	**2. REPORT DATE** March 2006	**3. REPORT TYPE AND DATES COVERED** Master's Thesis

4. TITLE AND SUBTITLE: A Software-Based Network Infrastructure for Mobile Ad Hoc Data Networking in Support of Small Tactical Units Using the SINCGARS Radio	**5. FUNDING NUMBERS**
6. AUTHOR(S) Steven R. Brand	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Postgraduate School Monterey, CA 93943-5000	**8. PERFORMING ORGANIZATION REPORT NUMBER**
9. SPONSORING /MONITORING AGENCY NAME(S) AND ADDRESS(ES) Marine Corps System Command (MCSC) Quantico, VA	**10. SPONSORING/MONITORING AGENCY REPORT NUMBER** N/A

11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

12a. DISTRIBUTION / AVAILABILITY STATEMENT Approved for public release; distribution is unlimited	**12b. DISTRIBUTION CODE**

13. ABSTRACT (maximum 200 words)

Currently, there is no infrastructure in place to provide data networking capabilities to ground-based tactical units below the battalion level. Legacy, voice-centric radios, organic to these units, possesses no inherent packet switched networking capability. The infrastructure for such a network is presented in this thesis.

Specifically, with the SINCGARS providing the Physical Layer, a software-implemented Data Link Layer is presented. Both an Aloha-like and a CSMA protocol are implemented for media access control. Additionally, a novel routing algorithm, Expected Relative Positioning with Congestion Avoidance (ERP/CA), is presented as the Network Layer protocol. This protocol is optimized for military application, using policies regarding movement and positioning within formations to inform its routing selections.

Finally, a prototype application is presented to demonstrate the use of the proposed small tactical unit, mobile ad hoc network infrastructure. The application is used in the functional testing of the layer 2 and layer 3 protocols. Results of the functional testing are presented.

14. SUBJECT TERMS Mobile Ad Hoc Networks, Tactical Networks, MANET, Routing Protocol, Expected Relative Positioning, SINCGARS, Data Link, Wireless, Media Access Control, ERP/CA	**15. NUMBER OF PAGES** 104
	16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT Unclassified	**18. SECURITY CLASSIFICATION OF THIS PAGE** Unclassified	**19. SECURITY CLASSIFICATION OF ABSTRACT** Unclassified	**20. LIMITATION OF ABSTRACT** UL

THIS PAGE INTENTIONALLY LEFT BLANK

A SOFTWARE-BASED NETWORK INFRASTRUCTURE FOR MOBILE AD HOC DATA NETWORKING IN SUPPORT OF SMALL TACTICAL UNITS USING THE SINCGARS RADIO

Steven R. Brand
Captain, United States Marine Corps
B.A. (Political Science), The Ohio State University, 1998
B.S. (Business Administration), The Ohio State University, 1998

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN COMPUTER SCIENCE

from the

NAVAL POSTGRADUATE SCHOOL
March 2006

Author: Steven R. Brand

Approved by: Geoffrey Xie
Thesis Advisor

John Gibson
Co-Advisor

Peter J. Denning
Chairman, Department of Computer Science

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

Currently, there is no infrastructure in place to provide data networking capabilities to ground-based tactical units below the battalion level. Legacy, voice-centric radios, organic to these units, possesses no inherent packet switched networking capability. The infrastructure for such a network is presented in this thesis.

Specifically, with the SINCGARS providing the Physical Layer, a software-implemented Data Link Layer is presented. Both an Aloha-like and a CSMA protocol are implemented for media access control. Additionally, a novel routing algorithm, Expected Relative Positioning with Congestion Avoidance (ERP/CA), is presented as the Network Layer protocol. This protocol is optimized for military application, using policies regarding movement and positioning within formations to inform its routing selections.

Finally, a prototype application is presented to demonstrate the use of the proposed small tactical unit, mobile ad hoc network infrastructure. The application is used in the functional testing of the layer 2 and layer 3 protocols. Results of the functional testing are presented.

THIS PAGE INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

I. INTRODUCTION 1
  A. OBJECTIVE 2
  B. WHY THE SINCGARS 3
  C. RESEARCH QUESTIONS 3
  D. ORGANIZATION 4

II. BACKGROUND AND RELATED WORK 7
  A. INTRODUCTION 7
  B. LAYER TWO SERVICES 8
    1. Encapsulation 8
    2. Error Detection and Reliability 8
    3. Media Access Control 8
  C. CONTENTION-FREE MAC PROTOCOLS 9
    1. Polling 9
    2. Time Division Multiple Access 9
    3. Frequency Division Multiple Access 10
    4. Code Division Multiple Access 10
    5. RTS-CTS 11
  D. CONTENTION-BASED MAC PROTOCOLS 11
    1. ALOHA 11
    2. Slotted ALOHA 12
    3. Carrier Sense Multiple Access 12
      a. The Hidden Node Problem 13
    4. Carrier Sense Multiple Access with Collision Avoidance 14
  E. 802.11B 14
    1. Virtual Carrier Sense and RTS-CTS 14
    2. Service Set Identifiers 15
    3. Beacons and Probes 15
    4. Authentication 16
    5. Encryption 17
    6. Clock Synchronization 17
  F. INDUSTRY EFFORTS, TACTICAL IP NETWORK SYSTEMS 17
    1. ViaSat Data Controllers 17
    2. TacLink 3000 Tactical Modems 18
    3. Harris Falcon II Tactical Network System 19
    4. CONDOR 19

III. LAYER TWO DESIGN, IMPLEMENTATION, AND FUNCTIONAL TESTING 21
  A. INTRODUTION 21
  B. CURRENT DATA NETWORKING CAPABILITY 21
    1. SINCGARS RS-232 Interface 22

2. The SINCGARS Audio/Data Connector 22
C. DESIGN 24
1. SL2I Addressing 25
2. SL2I Framing 25
3. SL2I Error Control 26
4. SL2I Reliability 27
5. SL2I MAC Functionality 28
D. IMPLEMENTATION 28
1. Programming Language and Serial Port Access ..28
2. SL2I API 29
3. Addressing Implementation 29
4. Framing Implementation 30
5. Error Control Implementation 30
6. Reliability Implementation 31
7. MAC Implementations 31
a. CSMA Protocol Implementation 31
b. Aloha Protocol Implementation 32
8. File Transfer 32
E. FUNCTIONAL TESTING 33
1. SINCGARS Data Demo Application 33
2. The Functional Test 35
a. Test Setup 35
b. Free Text 37
c. Intentional Collision 37
d. Free Text with Voice 38
e. File Transfer 38

IV. LAYER THREE DESIGN, IMPLEMENTATION, AND FUNCTIONAL TESTING 39
A. INTRODUCTION 39
B. EXISTING AD HOC ROUTING PROTOCOLS 41
1. Table-driven Routing Protocols 41
2. On-demand Routing Protocols 41
C. DESIGN OF THE ERP/CA ALGORITHM 42
1. Key Design Factors 42
a. Designed to Exploit Domain Knowledge 42
b. Flexible Enough to Accommodate Mobility Model Exceptions 45
c. Designed to Find Persistent and Reliable Routes 45
d. Designed to Avoid Congestion 46
e. Designed to Have Low Overhead 46
f. Designed to Prevent Routing Loops 47
2. Control Message Types 47
3. Initial Node Discovery 49
4. ERP Route Discovery 50
5. ERP/CA Distributed Route Selection Algorithm .52

a. Categorical Wait Value 53
b. Congestion Avoidance Value 54
c. Individual Response Wait 54
6. Examples of Applying ERP/CA Algorithm 54
a. Multi-hop Routing Examples 54
b. Congestion Avoidance Example 59
7. ERP Routing Tables 60
8. Route Maintenance 60
D. ERP/CA IMPLEMENTATION 61
1. Encoding of TTP-Based Knowledge 62
2. Implementation of the Distributed Route Selection Algorithm 62
a. CW Value Assignment 63
b. CAV Value Assignment 63
c. IRW Value Assignment 63
3. Congestion Avoidance Implementation 63
4. Control Message Implementation 64
5. Route Maintenance Implementation 64
6. ERP Routing Tables 65
7. Routing Loop Avoidance 65
E. FUNCTIONAL TESTING OF THE ERP/CA ALGORITHM 65
1. Test Setup 65
2. New-Join Discovery 67
3. Dynamic Discovery of Multi-Hop Paths 67
4. Dynamic Discovery of Single-Hop Paths 68
5. Path Discovery via Route Request Flooding 68
6. Congestion Avoidance 69

V. CONCLUSION, RECOMMENDATIONS, AND FUTURE WORK 71
A. OVERVIEW 71
B. CONCLUSION AND RECOMMENDATIONS FOR SL2I 71
C. CONCLUSION AND RECOMMENDATIONS FOR ERP/CA 72
D. FUTURE WORK 73
1. SL2I 73
a. Issue: Data Delivery Reports 73
b. Future Work: Data Delivery Reports 73
c. Issue: Position Location Information (PLI) 74
d. Future Work: Position Location Information (PLI) 74
e. Issue: Porting to Personal Data Assistants (PDAs) 74
f. Future Work: Porting to Personal Data Assistants (PDAs) 75
2. ERP/CA 75
a. Issue: Route Maintenance 75

b. Future Work: Anticipatory Link (Route) Verification 76
c. Issue: Multicasting 76
d. Future Work: Multicasting 77

APPENDIX 79

LIST OF REFERENCES 83

INITIAL DISTRIBUTION LIST 85

LIST OF FIGURES

Figure 1. Hidden Node Problem..............................13
Figure 2. Pin-out For SINCGARS (W4) Audio/Data Connector...23
Figure 3. Pin-out For DB9 Connector........................23
Figure 4. RS-232 Signals, By Pin...........................24
Figure 5. SL2I Header Format...............................25
Figure 6. Data Demo’s GUI..................................34
Figure 7. LOS Range and RF Shadow..........................40
Figure 8. Tank Company In Wedge Formation..................43
Figure 9. Hello and Hello Response Message Format..........48
Figure 10. Route Request Format.............................49
Figure 11. Route Response Format............................49
Figure 12. Node I Must Route Data From S To D...............51
Figure 13. Two Platoons In A MANET..........................55
Figure 14. Unicast Route Response...........................56
Figure 15. Most Persistent Route From Red-2 To Blue-3.......56
Figure 16. Flooded Route Request Stops At Red-2 and Red-1...57
Figure 17. Multi-hop Route Response.........................58
Figure 18. Most Persistent Route From Blue-3 To Red-3.......58
Figure 19. Potential Routes To Red-1........................60
Figure 20. Effective Topology After Attenuation Of Blue-1...66
Figure 21. Initial State and Transitions....................79
Figure 22. Sending Data to Another Node.....................80
Figure 23. Receiving Unicast Data from Another Node.........80
Figure 24. Processing a Route Request.......................81
Figure 25. Receiving ACK, Updating Route Freshness..........81

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF TABLES

Table 1. Cost of Procuring Tactical Networking Equipment..20
Table 2. Cable Splicing Specification......................23
Table 3. Data Link Layer Services Provided By SL2I........24
Table 4. Frame Types and Symbols...........................26
Table 5. Categorical Response Wait Values..................53
Table 6. ERP/CA Control Message Frame Types...............64

THIS PAGE INTENTIONALLY LEFT BLANK

ACKNOWLEDGMENTS

This thesis is dedicated to my wife and children. Throughout the entire process, my wife, Valerie, provided me with support and encouragement. My children, Marquis, Travis, and Ezra, provided my first ‘user’ evaluations and served as testing assistants on more than one occasion. Without my family’s love, patience, and inspiration this work would not have been possible. I thank you most of all.

I would like to thank my thesis advisor, Professor Geoffrey Xie, for holding me to such a high standard of academic performance. You ‘set the bar’ very high, and then raised it again. Thank you!

I would like to thank my co-advisor, Professor John Gibson, for his willingness to talk with me for hours at a time. I learn best by discussing the various aspects of a given topic and you listened as I strayed far and wide and helped me to find my way back to the true course. Often, it was these conversations that reinvigorated my efforts. Thank you.

I would like to thank Professor Craig Martell for helping me keep my right-brain engaged while I was immersed in left-brain studies. You provided me with assistance, advice, insight, and inspiration throughout my time at NPS. You were a teacher, a mentor, and a friend. Thank you.

I would like to thank Professor Richard Riehle for his eclectic style and for his encouragement. I hope that in this thesis you will find a beautiful snake—without feet.

I would like to thank Inspector-Instructor Capt Pollard Ham, and his Marines, of Company A(-) Reinforced, 4th LSB, 4th MLG. Your generous support in the form of time and equipment were invaluable. Semper Fi.

I. INTRODUCTION

Data networks offer an unparalleled means for distributing information. Military units enjoy this capability when in non-tactical or administrative status via wired, and sometimes wireless, Local Area Networks (LANs). Most, however, have to forego all such network functionality upon embarking on tactical training or actual combat. This is especially the case for small military units (those below the battalion level). These units are not normally equipped with the required networking hardware organic to units at and above the battalion level.

Although platoon and company level units lack the modern organic equipment that provides dedicated data networking capabilities, these units would greatly benefit from access to data networks. Furthermore, an ability to improvise data networks with legacy voice-centric communications equipment, specifically the Single Channel Ground and Airborne Radio System (SINCGARS), would be even more valuable.

The make-up and physical location of small units is far more dynamic than that of larger units. To offer the greatest utility, any data networks used by these small units must be ad hoc. In mobile ad hoc networks, a change in a node’s physical location (a change in the network topology) does not lead to a (long-term) loss of connectivity. Similarly, the presence or absence of any specific node, or nodes, will not eliminate overall connectivity.

The most salient characteristic of an ad hoc network is its dynamic membership. Nodes in such a network may join and leave the network at will. The topology of mobile ad hoc networks can potentially change from moment to moment, and there are no “central nodes” that direct or coordinate the activities of all other nodes. Mobile ad hoc networks are temporary and may be formed, joined, and destroyed, spontaneously. Members in such a network typically do not have a fixed physical location.

Rapid, agile, and flexible communications are essential to accurate situation awareness and command and control. Unfortunately, such communications are difficult for forces on the move. Mobile ad hoc networks provide the members of a small unit with the means to share information digitally. This shared information leads to a shared awareness of the situation, allowing for more effective operations and more efficient mission accomplishment.

A. OBJECTIVE

This research effort investigated the feasibility of providing a mobile ad hoc data networking capability over the SINCGARS SIP radio. This radio currently serves as the principal communications device for forward forces at the company level and below. However, it does not currently provide multipoint local area networking capabilities analogous to the ad hoc wireless Ethernet, standardized by IEEE 802.11. Further, the programmed replacement radio, the Joint Tactical Radio System (JTRS) will not be available to troops for several years [Bates 04]. This research proposes an open source, stopgap capability that will provide a means of extending the utility of the SINCGARS until the deployment of the more capable JRTS, thereby providing a migration path to the wireless network waveform to be provided by JTRS. More specifically, this research investigated the feasibility of providing a software-based logical link and Media Access Control (MAC) mechanism by which the SINCGARS, as the physical link, may be used to support a tactical ad hoc data network.

B. WHY THE SINCGARS

Like most legacy tactical radios, the SINCGARS (RT-1523C/D/E) is designed to interface with the serial port of Data Terminal Equipment (DTE), such as a laptop computer, via an RS232 protocol. These radios "act as half-duplex modems and are therefore considered [Data Communications Equipment] DCE equipment" [EngDoc00 p482]. To have the widest impact, the radio used in this research had to provide this well known interface.

In RS232 Data Mode, the RT-1523C/D/E is capable of a maximum baud rate of 9600 bps. This is very low compared with the baud rates of radios designed specifically for data transmissions. The SINCGARS, however, is the most-widely deployed tactical radio in the US military. Providing a data networking capability to a legacy radio will have the greatest impact when implemented with the SINCGARS radio. Small units are more likely to have these radios than any other. Furthermore, the units are not likely to have radios specifically designed for data communications.

Finally, because the SINCGARS has inherent frequency hop and encryption capabilities, it can provide a secure physical layer for mobile ad hoc tactical networks.

C. RESEARCH QUESTIONS

1. What characteristics of the SINCGARS make it difficult to use in implementing an ad hoc network?

2. What is the typical topology for an ad hoc network supporting a Marine deployment?

3. What degree of interconnection will be required with other units? Or can the ad hoc network be non-routable?

4. What will be the impact of the implementation of an ad hoc network over the SINCGARS with respect to the normal operation of the radio? Will the use of the radio to support an ad hoc network require fielding separate radios to support the current radio usage or can the current requirement also be satisfied with the ad hoc network, perhaps by using Voice over IP (VoIP) technology?

5. Can the serial port of the SINCGARS support an ad hoc network implementation?

7. It is known that the SINCGARS serial link can support a HyperTerminal session between two PCs. It is also known that the HyperACCESS application, a more capable version of HyperTerminal, provides an application programming interface for various scripting languages. Can this, or any other API, be used to support the implementation of a MAC protocol over the SINCGARS?

D. ORGANIZATION

The organization of this thesis is as follows: Chapter II provides an overview of the Data Link Layer of the OSI Model and a discussion of related research to include current industry efforts. Chapter III begins with a discussion of the current data networking capability of the SINCGRS radio. This is followed by a discussion of the design and implementation of the layer two protocols. The chapter concludes with a description of the prototype application developed as a result of this research and how it was used in the functional testing of the data link layer.

Chapter IV begins with a discussion of existing ad hoc routing protocols. The design and implementation of the layer three (networking) protocol, Expected Relative

Positioning (ERP), is discussed next. The chapter concludes with a description of ERP’s functional testing. Chapters V and VI provide evaluations of the layer two and three protocols, respectively. The evaluations are based on simulations using OPNET Modeler 11.0. Chapter VII examines the applicability of this networking capability for small tactical military units and provides conclusions and recommendations for follow-on research.

THIS PAGE INTENTIONALLY LEFT BLANK

II. BACKGROUND AND RELATED WORK

A. INTRODUCTION

The Open System Interconnection (OSI) model is a widely used model of network communication. It depicts a layered set of protocols, each providing services to the layer above and making use of services provided by underlying layers. All together, the protocols in this model allow for the efficient, reliable transfer of data in most modern networks.

Layer two of this model is the Data Link layer. It makes us of the physical layer, which is the physical link between nodes in a network. The Data Link layer is subdivided into Media Access Control (MAC) and Logical Link Control (LLC) layers. The various services provided by the LLC include per-link reliability, bit error detection or correction, and encapsulation. The MAC provides access to the shared medium.

This chapter begins with a more detailed look at some of the layer two services. It then describes MAC protocols proposed for wireless networks. These protocols can be classified into two groups: contention-free and contention-based. As the most widely used wireless Data Link layer protocol suite, the more salient features of the 802.11b specification are given special attention. Finally, because proprietary WLAN implementations for tactical networks are available, these industry efforts are also discussed.

B. LAYER TWO SERVICES

1. Encapsulation

Encapsulation involves the pre-pending of a layer two header to the data received by the data link layer from the upper layers. Encapsulation usually includes the addition of a trailer that includes a checksum value or Cyclic Redundancy Check (CRC) value as well. This value uniquely represents the data contained in the frame and is used by the receiving node to detect bit errors occurring in the physical link between nodes. Together, the header, CRC, and data payload make up the layer two data unit called the frame.

2. Error Detection and Reliability

At the data link layer, reliability is measured across a single link. A node reliably sends a frame across the link by resending lost or corrupted frames. Checksums or CRC values are used to detect corrupted frames and request retransmission from the source node. The LLC provides this error detection and reliability without requiring any resubmission of data from the upper layers of the protocol stack.

3. Media Access Control

Nodes in an ad hoc network share a common channel. If two or more nodes send data into that channel simultaneously, there will likely be collisions. This is undesirable. The successful transmission of a layer two frame requires that the frame be received, intact, by the receiving node. Collisions cause corruption of these frames. In order to minimize the loss or corruption of data frames, medium access control protocols are utilized at layer two.

There are two general categorizations of MAC protocols. These categories are contention-based and contention-free. While contention-based protocols allow nodes to access the shared medium at random, contention-free protocols ensure that no two nodes access the medium simultaneously. When only one node at a time has access to the medium, there will be no frame collisions. Contention-based protocols tend to be less complicated, but must provide some means of resolving the contention. More precisely, when a collision occurs, the transmitting nodes must take measures to retransmit their frames without future collision. This is most frequently accomplished by having each of the transmitting nodes wait for a random “back-off” period of time before attempting to retransmit their respective frames.

C. CONTENTION-FREE MAC PROTOCOLS

1. Polling

Polling is perhaps the simplest form of a contention-free MAC protocol. It utilizes a centralized control node called an Access Point (AP). All nodes in the network must be within radio range of the AP. All data is sent and received via the AP. No node may send data until it is polled by the AP. Each node in the network is polled by the AP in a round-robin fashion or according to another scheduling method, such as priority-based polling. Polling is wasteful of the shared channel when polled nodes have no data to send, and it lacks the flexibility required for mobile ad hoc networking.

2. Time Division Multiple Access

The Time Division Multiple Access (TDMA) protocol may be used with or without an AP. It does, however, require some means of synchronizing the clocks at each node in the network. If no AP is used to synchronize clocks, Global Positioning Satellite (GPS) receivers are frequently used at each node.

The basic principal behind this protocol is to evenly partition the shared channel, assigning a unique time-slot to each node. Various algorithms have been developed to dynamically assign these slots, so that nodes can join and leave the network without causing a network collapse.

TDMA provides contention-free access to the shared medium by allowing an individual node to send data across a link only during its assigned time-slot. If a given node has no data to send during its time-slot, that slot is wasted. No node may use the slot assigned to another node.

3. Frequency Division Multiple Access

Frequency Division Multiple Access (FDMA) is very similar to TDMA. It also evenly partitions the shared medium. FDMA, however, assigns each node a unique frequency, slightly different from that of all other nodes in the network.

Instead of dividing time, FDMA divides the channel’s allotted bandwidth into multiple channels and assigns a single channel to each node. Use of FDMA requires special radio hardware. Also, protocol complexity is increased if channels are assigned on demand. Whenever an assigned channel is unused, that slice of bandwidth is wasted.

4. Code Division Multiple Access

With Code Division Multiple Access (CDMA), all nodes use the entire channel bandwidth to send each frame of data. No time slots are used; each node sends frames when data is made available from upper layers. This data is encoded by the LLC using a chipping code that is unique to each individual node.

Special hardware is used to encode and decode the signals sent and received at each node. A node, J, would use the chipping code for node K to decode data sent from K. Although CDMA schemes have been widely used in cellular phone networks, it has been facilitated by expensive hardware.

5. RTS-CTS

Contention-free access to the medium can also be achieved by dynamically reserving the medium. Before sending a data frame, a node may send a special control frame called a Request-To-Send (RTS) frame. Nodes hearing this frame will refrain from transmitting for a specified period of time. The intended recipient of the data frame will send a Clear-To-Send (CTS) frame to the requesting node. Nodes hearing this CTS frame will also refrain from transmitting. The RTS and CTS frames, effectively, reserve the channel for the RTS source node.

D. CONTENTION-BASED MAC PROTOCOLS

1. ALOHA

The earliest contention-based MAC protocol was detailed in *The Aloha System - Another Alternative for Computer Communications*, Proceedings of Fall Joint Computer Conference, AFIPS Conference. This simplest of MAC protocols allows for random access to the shared channel. 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 ALOHA protocol makes no attempts to prevent frame collisions. Its maximum channel utilization is only 17 percent. For some networks, (i.e. those for which maximum size and traffic load are low) this may be sufficient.

2. Slotted ALOHA

A variation of the ALOHA protocol, called Slotted ALOHA, doubles the channel utilization to 34 percent. Slotted ALOHA requires clock synchronization among the nodes in the network. This synchronization allows time to be divided into slots equal to the transmission time of one frame. Nodes are only allowed to begin transmission of a frame at the start of a slot, thus reducing the likelihood of collisions but not eliminating them. Like ALOHA, Slotted ALOHA frames are retransmitted when no ACK is received.

3. Carrier Sense Multiple Access

Carrier Sense Multiple Access (CSMA) protocols are an improvement over ALOHA and Slotted Aloha. Nodes executing CSMA protocols physically sense the carrier medium before transmitting frames. When the medium is idle, nodes with data to send may transmit. When the medium is in use by a node, no other node is allowed to transmit.

Upon sensing an idle channel, nodes transmit their frames immediately. If two simultaneously nodes sense an idle channel, they are likely to transmit simultaneously. This will lead to collisions when these two nodes are within radio range of one another. As a result, the two will retransmit their frames upon sensing an idle carrier, possibly resulting in another collision.

Nodes on a wired network can sense collisions by measuring voltage on the wire. This allows the use of Carrier Sense Multiple Access with Collision Detection (CSMA/CD). Collision detection, however, is not available to nodes on a wireless network. For wireless networks, the hidden node problem is a consequence of this inability to detect collisions.

a. The Hidden Node Problem

In Figure 1, nodes A and B are within range of one another, as are nodes B and C. Nodes A and C however are not within radio range of one another. Node B is able to sense the carrier and detect any transmissions sent from node A. Using CSMA, node B will not transmit while A is transmitting. Node C, however, will not be able to sense transmissions by node A. Nodes A and C are hidden from one another. As a result, nodes A and C could send frames to node B simultaneously. These frames are likely to be corrupted, upon arrival at node C, or lost entirely.

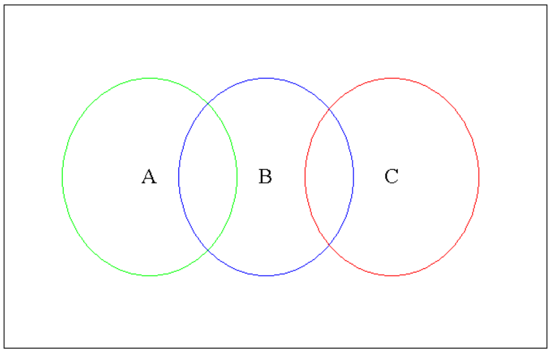


Figure 1. Hidden Node Problem.

4. Carrier Sense Multiple Access with Collision Avoidance

Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) is a MAC protocol that combines the functionality and advantages of CSMA and RTS-CTS. This combination reduces the likelihood of data frame collisions when multiple nodes simultaneously sense an idle carrier. Assuming a given node senses an idle carrier, it will still not transmit its data frame until after it receives a valid RTS frame in response to its CTS frame.

CSMA/CA protocols include additional functionality to prevent repeated data frame collisions. If nodes A and B simultaneously sense an idle carrier and simultaneously transmit their data frames, the retransmissions are not likely to occur simultaneously. This is because both A and B will choose random back-off values. The back-off value represents a period of time during which the respective nodes will refrain from transmitting. As nodes A and B are not likely to choose the same back-off value, they are not likely to retransmit their frames at the same time, thus a subsequent collision will be avoided.

E. 802.11B

The most commonly used MAC protocol is specified by IEEE 802.11b [Intel05]. The 802.11b standard actually specifies the Data Link and Physical layers. It is a complex set of protocols with a large set of functionality. For example, 802.llb uses CSMA/CA and addresses the hidden node problem via the use of RTS-CTS messages.

1. Virtual Carrier Sense and RTS-CTS

In addition to physical carrier sensing, 802.11b uses virtual carrier sensing as well. A Network Allocation Vector (NAV) value is maintained at each node. When a node detects an RTS from another node it sets its NAV to a value found in the duration field of the RTS frame. The RTS duration field is used to reserve the channel for enough time to send the data frame and receive an ACK. The CTS frame also contains a duration field. Stations not in range of the requesting node when it sends its RTS, but in range of the recipient of the RTS, will update their respective NAVs upon hearing the CTS. Stations wishing to transmit must wait until both the physical channel is idle and the virtual carrier sense indicates the channel is idle (meaning the NAV equals zero).

Other functionality in the 802.11b standard includes the following:

2. Service Set Identifiers

Service Set Identifiers (SSIDs) are strings, up to 32 bytes in length, used to identify networks (wireless LANs). SSIDs are often called the network name because human readable ASCII characters are normally used, though not required by the standard.

Basic Service Set Identifiers (BSSIDs) are also used to identify infrastructure and ad hoc wireless LANs. In an infrastructure network, the BSSID is the MAC address of the access point. Ad hoc networks use a randomly generated, 48-bit BSSID. BSSIDs allow multiple 802.11 networks, existing in the same area, to filter link layer broadcasts, thereby creating independent, logical LANs.

3. Beacons and Probes

Beacons and Probes are management frames used to announce and discover 802.11 networks, respectively. Both use SSIDs as network identifiers. In an ad hoc network there is no Access Point (AP) to broadcast its (and thus the network's) SSID, but at least one node in the ad hoc network will take on this responsibility. A node wishing to join the network will listen for the broadcast of the SSID. Alternatively, a node may solicit a Probe Response Frame, which contains the SSID, via a Probe Frame. Beacon and Probe Response Frames contain all the information a node needs to join the network, to include physical layer specifications, coordination function specification, and supported data rates.

4. Authentication

Nodes wishing to join an ad hoc 802.11b network must first be authenticated by the network. Authentication is the process by which a node identifies itself to the network. The soliciting node's MAC Address (the source in its Authentication Frame) is used as its station identifier. When using Open-System Authentication, no additional information or calculations are required. With few exceptions, stations requesting authentication are authenticated.

Although, the 802.11b standard only requires Open-System Authentication, the use of MAC Filtering and Wired Equivalency Protocol (WEP) are also common. With MAC filtering, authentication is granted only to those stations whose MAC addresses are found on an authorization list. When WEP is used, the soliciting station will be presented with an unencrypted challenge text. Proper encryption of the challenge text proves possession of the shared WEP key and earns authentication.

Regardless of the authentication technique used, nodes may be authenticated with multiple networks simultaneously but may be associated with only one at a time.

5. Encryption

Data secrecy and integrity are called for in the 802.11b standard. As of this writing, Wired Equivalent Privacy (WEP) is the most commonly used encryption technique in 802.11b networks, though the use of Wi-Fi Protected Access (WPA) is growing. Both WEP and WPA provide some degree of protection for the payload of Data Link layer frames.

6. Clock Synchronization

Clock Synchronization is used to schedule beacon frames. Additionally, 802.11b has power save functionality for battery powered devices. Power Save Mode allows a device to “sleep” during the interval between Beacons. Upon awakening, the device will listen for special frames indicating that it is the intended recipient of a buffered data frame. Clock synchronization ensures that nodes are awake simultaneously to send and receive data while in Power Save Mode.

The above is a portion of the rich set of functions available in the 802.11b standard. In Chapter V, we compare its performance to that of SINCGARS Layer-2 Interface (SL2I), the layer-two protocol designed in this thesis. We believe SL2I provides the minimal subset of 802.11b’s functionality.

F. INDUSTRY EFFORTS, TACTICAL IP NETWORK SYSTEMS

1. ViaSat Data Controllers

ViaSat develops digital communications products for commercial and government markets. Their ViaSat Data Controllers (VDCs) and application software have been purchased and used by some USMC units to pass data across a tactical LAN. However, ViaSat Data Controllers (and associated software) have not been designated as a program of record in the USMC, and have not been widely deployed.

The VDCs provide “black box,” proprietary data networking capabilities to tactical radio networks. Each node on these networks consists of a laptop PC or PDA connected to a VDC. The VDC-500 provides the TCP/IP stack and is connected to the data connector on the tactical radio via an RS-232 cable. Alternatively, the VDC-600 Personal Data Controller II (a PCMCIA card) may be used.

ViaSat Data Controllers provide the following functionality:

- Built-in message compression
- Adaptive forward error correction
- CSMA channel access and contention resolution
- Channel quality estimation
- Automatic request for retransmission
- Supports channel rates from 75bps to 128 kbps

2. TacLink 3000 Tactical Modems

Produced by Raytheon, the TacLink 3000 Tactical Modem is also a PCMCIA Card. It is also the interface between a tactical radio and a PC or PDA. Together, these three elements would comprise a node on a tactical data network.

“A front-end communications processor, the ruggedized TacLink supports wide-area network communications for a variety of commercial and military host computers through open standard interfaces,” according to Raytheon’s website (http://www.raytheon.com/products/taclink/). Like the VDCs, the TacLink 3000s are layer two devices providing controlled access to the shared medium.

3. Harris Falcon II Tactical Network System

Harris provides similar tactical networking (LAN) functionality exclusively through its FALCON II radios. The Falcon II (AN/PRC-117F(C)) has a native IP stack implementation eliminating the need for external hardware between the PC and radio. It is a programmable VHF and UHF radio that can be operated in SINCGARS mode. The PRC-117F is intended to be a replacement for the SINCGARS and other legacy radios.

4. CONDOR

CONDOR is the Marine Corps' Command and Control On-the-move Network Digital Over-the-horizon Relay. The complete CONDOR system uses three types of vehicles: the Gateway Vehicles, the Jump Command and Control Vehicles, and the Point of Presence Vehicle (PoP-V). The Jump Command and Control Vehicles provide connectivity for mobile command and control elements. The Gateway Vehicles use Enhanced Position Location Reporting System (EPLRS) or satellite radios to overcome Line of Sight (LOS) barriers separating EPLRS networks, and the PoP-V provides the means for other tactical radios to connect to the Tactical Data Network. The PoP-V will, for example, connect a SINCGARS network to a Gateway Vehicle, and thus, the wider network.

Designed by CenGen, CONDOR is a Marine Corps program of record. CONDOR will provide mobile, Beyond Line of Sight (BLOS) data networking capabilities to units using tactical data radios. CenGen is currently evaluating the PRC-117F, the ViaSat Data Controllers, and the TacLink 3000s for use in tactical LANs. The Data Controllers or

TackLink 3000s will provide layer two functionality to tactical radios like the SINCGARS, in the CONDOR system. C2PC and IRC Chat are the projected applications for these networks.

Product	**Price**	**As Of**	**Source**
VDC-600	$3,500	22 Dec 05	www.viasat.com/datacontrollers/ordering/
TacLink 3000	$2,832	22 Dec 05	NSN: 5895-01-518-9747
Falcon II (AN/PRC-117F(C))	$26,000	21 Dec 05	www.rfcomm.harris.com/contact/ informal quote via phone

Table 1. Cost of Procuring Tactical Networking Equipment

VDCs, the TacLink-3000s, and the Falcon IIs all provide expensive, proprietary, single-hop LAN functionality. No wireless network routing protocols are employed. The cost to procure this commercial, networking hardware is listed in Table 1. As an open source, software-based Data Link layer, SL2I provides analogous wireless network connectivity to small tactical units at a little to no cost.

III. LAYER TWO DESIGN, IMPLEMENTATION, AND FUNCTIONAL TESTING

A. INTRODUTION

Today, many PDAs and most laptop computers are capable of wireless networking. These devices either use removable cards or have some integrated wireless (802.11x) capability. In either case, the physical layer (the radio) and the data link layer are tightly coupled in one hardware/firmware package. Removing the card, or chip, in the case of integrated capability, removes the wireless networking functionality.

The operating systems on these devices are designed to use the layer two services provided by the wireless cards/chips. Attempts to reroute IP data packets to another I/O port require new hardware solutions or modifications to the operating system.

This chapter will describe a non-IP-based solution. This solution does not involve the addition of hardware, and does not require OS modification. First, the current point-to-point data networking capabilities of the SINCGARS radio are discussed. This provides some context for the layer two design and implementation decisions made in the protocol development. After the discussion of the SINCGARS’ limited current capabilities, a detailed description of the design, implementation, and testing of a new, more capable, data networking functionality is presented.

B. CURRENT DATA NETWORKING CAPABILITY

Legacy tactical radios in the US military, represented in the research by the SINCGARS (SIP and ASIP), do not have built-in 802.11 or Ethernet interfaces. They have serial, RS-232, data interfaces. Any data link layer designed to operate in conjunction with the SINCGARS as the physical layer would have to do so via the radio's serial interface. This interface is designed to facilitate only a rudimentary point-to-point networking capability, via the use of third-party software like HyperTerminal, for example. The pin-out of this interface is provided in Figure 2.

1. SINCGARS RS-232 Interface

> The SINCGARS RS-232 data interface uses a transmit data line (input to the radio), a receive data line (output from the radio), and a Clear-to-Send (CTS) line for hardware flow control of the transmitter; flow control is implemented one-way only. When the radio de-asserts the CTS line, the DTE can not send data.
>
> When the radio is operated in the RS-232 data mode, it expects a 10-bit character from the attached DTE. Each character is constructed as one start bit, 8 bits of data followed by one stop bit… The transmitting radio will remove the start and stop bits before RF transmission. Reconstruction of the 10-bit pattern is performed at the receiving radio [EngDoc00].

2. The SINCGARS Audio/Data Connector

With a PDA or PC acting as Data Terminal Equipment (DTE), the SINCGARS acts as Data Circuit-Terminating Equipment (DCE). Data to be transmitted is sent from the DTE to the DCE, while received data is sent to the DTE from the DCE. The cable between the DTE and the SINCGARS must be custom made, as there are none in the DoD supply system that can be used without modification. Cables used in this research were constructed by splicing together a serial crossover cable (with female DB-9 connectors) with an interface cable from a Lightweight Digital Facsimile (AN/UXC-7), which has a SINCGARS W4 connector on one end.

Specifications for the cable splicing are listed in Table 2. Figures 2 and 3 show the pin-outs for the W4 and DB9 connectors, respectively. The signals carried on each pin are displayed on Figure 4.

W4	**Connects To**	**DB9**
A	---------------------------------	5
B	---------------------------------	2
E	---------------------------------	8
F	----------------------------------	3

Table 2. Cable Splicing Specification

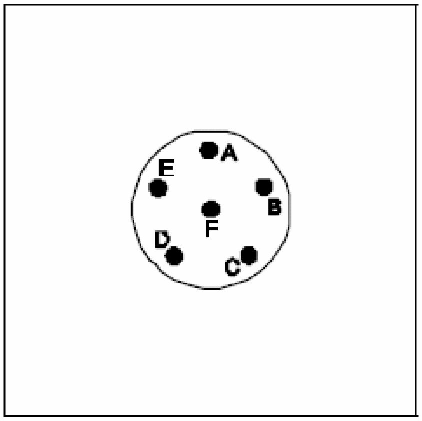


Figure 2. Pin-out For SINCGARS (W4) Audio/Data Connector

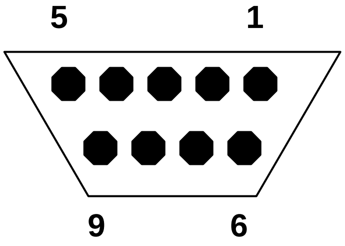


Figure 3. Pin-out For DB9 Connector

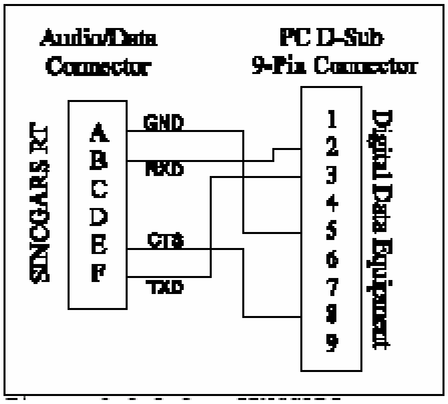


Figure 4. RS-232 Signals, By Pin

C. DESIGN

The set of data link protocols used in this thesis, called the SINCGARS Layer-2 Interface (SL2I), were implemented exclusively in user level software and without any modification to OS kernel level code. SL2I, a basic data link layer tied to the device's serial port, was developed to take advantage of the physical layer services provided by the SINCGARS and to work within its limitations. SL2I is designed to provide traditional data link layer services to upper layers via its Application Program Interface (API). It sends encapsulated data to the tactical radio and de-encapsulates data frames received from the radio across the serial interface. The services provided are listed in Table 3. SL2I enables the SINCGARS radio to be used for ad hoc, multipoint data networking.

Media Access Control	**Logical Link Services**			
Aloha	Reliability	Variable Frame Size	Encapsulation	Check Sum
CSMA	Reliability	Variable Frame Size	Encapsulation	Check Sum

Table 3. Data Link Layer Services Provided By SL2I

1. SL2I Addressing

Instead of using the physical addresses (MAC address) normally provided by a Network Interface Card (NIC), SL2I uses a custom addressing scheme, as no NICs are incorporated in the design. To reduce frame overhead, each SL2I node is assigned a unique 8-bit address, as opposed to the 48-bit MAC address assigned to each NIC.

2. SL2I Framing

SL2I’s maximum payload size is 600 bytes. Datagrams 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 5. It should be noted that the intermediate address fields are for routing, a layer three concern which will be discussed in the next chapter.

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).

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 5. SL2I Header Format

Frame types are listed in Table 4. Only the Data and File Transfer types can be designated by upper layer processes. Acknowledgement (ACK) frames are generated by SL2I. All others are related to routing and are discussed in Chapter IV.

Frame Type	Symbol	Use
Data	T	Indicates data frame
Acknowledgement	A	Indicates an ACK frame
File Transfer	I	Indicates frame used in file transfer
Control	R,Q,H	Indicates a control frame used for routing

Table 4. Frame Types and Symbols

The frame format and field sizes assist SL2I in identifying invalid frames. Since ACK frames have no payload, the smallest valid frame is comprised solely of a 6-byte header. Any frames smaller than this are discarded as invalid. Similarly, frames larger than 607 bytes (header, payload, and checksum) are also discarded as corrupted frames.

3. SL2I Error Control

SL2I uses a one byte checksum value for error control. Each node receiving a valid 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. An ACK for the frame is sent only if the checksum values match.

Calculation of the checksum value is accomplished by converting each byte of the frame (header and payload) into an integer. The first and second bytes are summed. Then the integer value of the third byte is subtracted from this sum. Following this pattern, each subsequent integer is added to or subtracted from the running total. Ensuring only positive integers are used, the absolute value of the final running total is used in the last step of the calculation. Modular arithmetic is used to ensure that the final checksum value is not too large to be represented by a single byte. The result of the running total modulo 128 is used as the checksum value. This simple checksum calculation is sufficient for a SINCGARS-based network as the SINCGARS, in data mode, provides Reed Solomon Forward Error Correction at the physical layer.

4. SL2I Reliability

Successful delivery of a data frame requires two components. The first is that the destination node receives a properly formatted, properly addressed, and uncorrupted data frame. The second component is that the source node receives a timely, properly formatted, properly addressed, and uncorrupted ACK frame.

Corruption, congestion, or frame collision can prevent the achievement of either component of successful frame delivery. Regardless of the cause of the failure, the outcome will be the same-- no ACK will be received. Until an ACK is received, SL2I will repeatedly attempt to deliver the frame. After a maximum of five attempts, the source node will drop the frame and notify any layer-three protocol.

5. SL2I MAC Functionality

SL2I may be set to operate in either Aloha mode or CSMA mode. When Aloha is chosen as the MAC protocol, data delivered to SL2I from an upper layer process is formatted, encapsulated, and immediately transmitted. No attempt is made to avoid collisions.

Alternatively, CSMA may be chosen as the MAC protocol. In this mode, SL2I make use of an additional physical layer service provided by the SINCGARS radio, and reduces the likelihood of frame collisions. Acting as a DCE, the SINCGARS will notify the associated DTE of carrier status. As seen in Figure 4, the SINCARS uses Pin E of its Audio/Data Connector to indicate that it is clear for the DTE to send data. SL2I monitors the signals on this line, when set to CSMA mode, and will not transmit frames unless the carrier is idle.

D. IMPLEMENTATION

1. Programming Language and Serial Port Access

To take advantage of cross-platform portability, SL2I was written completely in Java. High-level access to the PC’s serial port was provided by the Sun Javax.Comm extension to the virtual machine. It contains support for parallel ports as well as RS-232 serial ports, allowing synchronous and asynchronous input/output operations. This extension, as well as detailed instructions for installation, may be downloaded from Sun, at http://java.sun.com/products/javacomm/.

The Javax.Comm download includes sample applications that use the serial and parallel ports. One application, SerialDemo, was the basis of the prototype application used in this thesis. Its supporting classes served as a starting point for the development of SL2I.

2. SL2I API

Processes wishing to make use of the services provided by SL2I have several options. Access to SL2I services is provided by the SincgarsSerialConnection Class. Data to be transmitted may be submitted to SL2I by calling the sendText method. The method accepts a byte array and has no return type. Processes may poll SL2I to determine if data is available to pass to upper layers via the isTextAvailable method, which returns a boolean, and receive the data via the receiveText method. Alternatively, data may be delivered automatically (without polling) to the upper layer. Calling the setAutoReceive method and passing a TextArea or PrintStream reference can configure SL2I to push de-encapsulated data to the upper layer.

Configuration options provided by SL2I include serial port selection (for devices with multiple serial ports) and baud rate selection. It is also possible to separately set flow control in and out for hardware, software, or none at all. The number of data bits used per character may be set from 5 to 8 and stop bits may be set to 1, 1.5, or 2. Finally, Sl2I supports both Aloha and CSMA MAC protocols.

3. Addressing Implementation

SL2I's addressing scheme is designed to be simple and flexible. The application layer process provides a list of valid call sign strings, via the setCallSignList method. Since military procedures require each unit element to have a unique call sign, SL2I uses the index of each entry as the node's unique address value.

Local host applications designate the local call sign by calling the setThisCallSign method. By searching the list of call signs, SL2I identifies the index-based address value for the local host. This process is duplicated when SL2I is given a call sign for the designated destination node. The current call sign list, local host call sign, and destination call sign are stored by SL2I.

4. Framing Implementation

Data passed to SL2I from upper layers is first checked for size in the sendText method. When these data packets, plus the SL2I header exceed 607 bytes, the packets are fragmented into multiple data frames.

After any necessary fragmentation, the SL2I header construction begins. The frame type is first determined. The previously set address values are then inserted into the address fields. Finally, the frame sequence number is inserted and incremented. The content of each header field is determined and inserted in the applyLayer2Header method.

5. Error Control Implementation

The use of CRC values provides a level of error control that is superior to that of checksums. The calculation and comparison of CRC values is, however, far more computationally intensive. For data link protocols, these computations are usually hardware-based, and thus very fast. The use of such a computationally intensive algorithm would have produced a throughput penalty for the software-based SL2I. As a tradeoff, checksums were chosen instead.

A simple, but novel approach was chosen to calculate the checksum values. The mySimpleChecksum method converts each byte of the frame (header and payload) into an integer. Then it alternately adds or subtracts this integer value to that of the previous, and it returns the result modulo 128.

6. Reliability Implementation

SL2I spawns a new thread for each data packet delivered to it from an upper layer process. Packets that must be fragmented into multiple frames spawn a single new thread. Although the receipt and initial processing of ACK frames is implemented in the main thread, all other aspects of layer two reliability are implemented in the newly spawned thread. Once the maximum number of resend attempts has been reached or successful frame delivery is achieved, the new thread is terminated.

Upon the initial transmission of a data frame, an ACK timer is started. Three and a half seconds are allotted for delivery of the data frame, processing by the receiving node, and receipt of the ACK frame. If the timer expires before the receipt of the ACK, the frame is transmitted again, and the transmission attempt counter is incremented. The new thread remains in this loop until one of the following events occurs, receipt of an ACK for the transmitted frame or achieving the maximum send attempt threshold.

7. MAC Implementations

a. CSMA Protocol Implementation

Before sending a data frame to the SINCGARS for transmission, SL2I will first evaluate the status of a Boolean variable, carrierIdleStatus. The value of carrierIdleStatus is set in accordance with the signal on pin E, the CTS line. When voltage on the CTS line is asserted, a Java event is triggered. The event causes the value of carrierIdleStatus to be set to true, indicating that the carrier is idle. When the SINCGARS is transmitting or receiving data, it de-asserts the voltage on the CTS line. This is treated as a new event, which causes carrierIdleStatus to be set to false.

While in CSMA mode, a frame that is ready for transmission will be held in an SL2I buffer until the carrier is idle. Once the CTS line is asserted, carrierIdleStatus will be set to true, and the buffered frame will be sent to the SINCGARS for transmission.

b. Aloha Protocol Implementation

The Aloha protocol does not include any carrier sensing. When SL2I is set to operate in Aloha mode, carrierIdleStatus is always set to true. Events triggered by the voltage changes on the CTS line are not monitored. Because the carrier is always assumed idle, SL2I immediately sends all frames to the SINCGARS for transmission. Frames that collide in transport between the DTE and DCE are dropped by both.

8. File Transfer

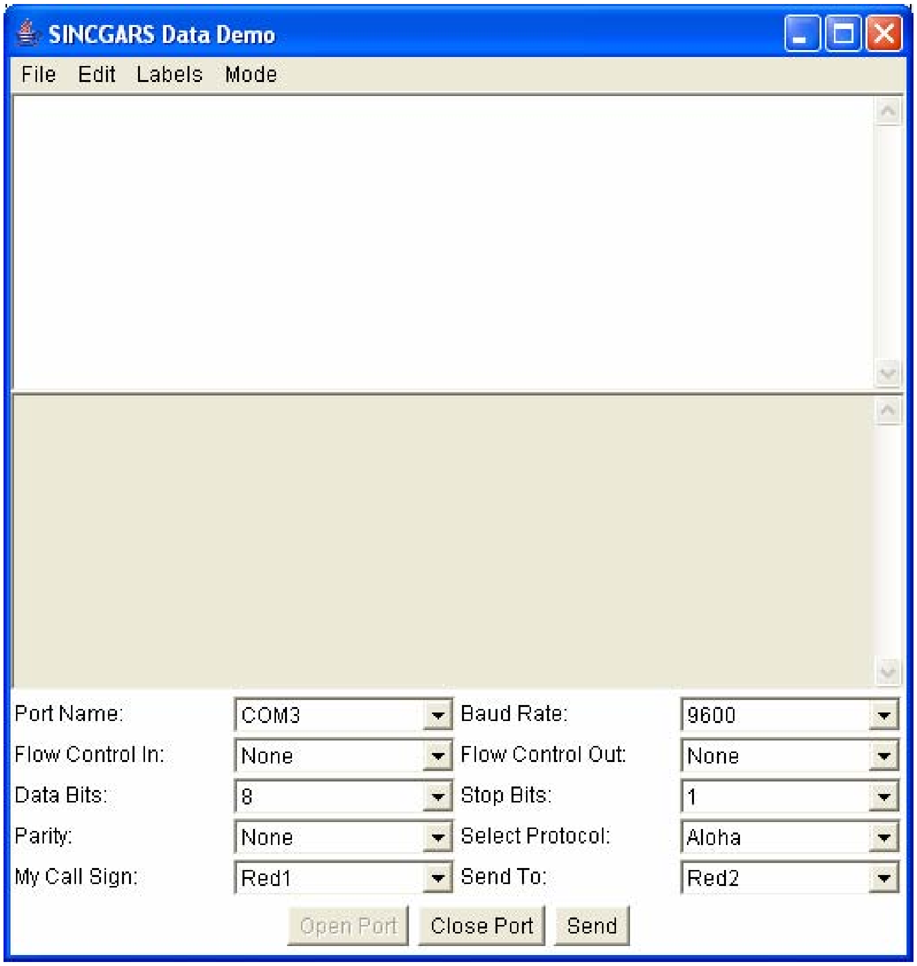
The flow of data to and from SL2I may be altered by configuring it for file transfer. The send-file configuration will cause raw bytes to be retrieved from a designated file and sent to the destination node. When configured to receive a file, the payload of received frames will be stored in a file. SL2I’s send-file and receive-file configurations are mutually exclusive. A single node may not simultaneously send and receive a file.

E. FUNCTIONAL TESTING

1. SINCGARS Data Demo Application

Data Link protocols do not generate data. Instead, they encapsulate and transfer, to another node, any data received from upper layers. This thesis research began with an application, the SINCGARS Data Demo, passing data directly to the data link layer. No transport or network layer protocols were used. Most messages sent by Data Demo are unicast and are sent to the node corresponding to the call sign selected in the “Send To” window. Messages may also be broadcast to all nodes within radio range. The local node will display messages with addresses that match the call sign listed in the “My Call Sign” window, as well as any broadcasted messages. Data Demo’s graphical interface is shown in Figure 6.

SINCGARS Data Demo is a tactical chat application, written in Java. It allows users to send and receive point-to-multipoint text messages. The application’s GUI contains two text windows. Data Demo’s top window, the send window, displays text messages as they are being composed. Completed messages remain in the send window until the user clicks the SEND button. These messages may be edited until the SEND button is clicked. Sent messages are automatically removed from the send window, but may be displayed in the receive window with the ECHO option toggled on. ECHO is found in the Edit drop-down menu.



Port Name:	COM3	Baud Rate:	9600
Flow Control In:	None	Flow Control Out:	None
Data Bits:	8	Stop Bits:	1
Parity:	None	Select Protocol:	Aloha
My Call Sign:	Red1	Send To:	Red2

Figure 6. Data Demo’s GUI

SINCGARS Data Demo’s bottom window is the receive window. It displays all messages addressed to the local node. All received messages (and echoed messages) remain in the receive window but may be cleared by the user via the CLEAR RECEIVE WINDOW option in the Edit drop-down menu. By default, each received message is labeled with the sender’s call sign. This option may be toggled on and off.

SINCGARS Data Demo may also be used to transfer files. The Mode drop-down menu allows users to select “Send Image” or “Receive Image”, which activates the corresponding SL2I file transfer modes. Users designate the file to send by typing its name in the send widow and clicking the SEND button.

2. The Functional Test

Functional testing was conducted to test the integration of various software modules or classes. The classes that comprise SL2I were tested to evaluate their ability to jointly provide data link layer services to upper layer processes. The functional testing was conducted using the SINCGARS Data Demo application running directly above the SL2I. For small unit tactical data networks, traffic loading is expected to be light. The expected traffic and small network size make the use of the TCP/IP stack superfluous. The hierarchical addressing scheme of IP is not required in small networks, and when traffic-load is light, multi-layer reliability is not necessary. During functional testing, the reliability provided by the data link layer proved sufficient.

a. Test Setup

Initial testing was conducted using the Aloha MAC protocol. Three nodes were used in the conduct of this functional test. Each node was within radio range of all other nodes. Two of the nodes were comprised of Dell Latitude C640 laptop computers. These were running Windows XP Professional, with 384MB of RAM, a 200Mhz P4 processor, and a 30GB hard drive. The Latitude C640 has a built in serial port. Each of these PC’s was connected directly to the Audio/Data connector on the front panel of a RT-1523(c)D SINCGARS.

The third node used an RT-1523(c)F, commonly referred to as the “Fox” variant. With respect to its serial interface, the Fox has the same functionality as earlier variants. The physical dimensions of the Fox, however, are roughly half that of the D variant. With respect to this functional test, the Fox provided no greater or lesser capability as compared to the RT-1523(c)D.

This third node used a Dell Latitude D410 laptop computer. The D410 was running Windows XP Professional, with a 1.6GHz processor, 512MB of RAM, and a 40GB hard drive. The D410 has no serial port. One of its three USB ports was converted into a serial port via a USB-to-serial adapter cable from Prolific. The Prolific adapter cable comes with a driver, which was also used. With the exception of this adapter cable, the connection to the local SINCGARS was identical to that at the other two nodes.

At each node, the Data Demo was the only application running during the functional test. Each radio was set to its lowest power setting, and whip antennas were used. The test site was an indoor classroom. No frequency hopping or encryption was used. Each radio was set to RS-232 Data Mode.

Users at each station selected a unique call sign from the options presented in Data Demo drop-down menu and conveyed this selection, by voice, to the other operators. One of the RT-1523(c)D nodes chose “Red-1”, while the other chose “White-1”. “Blue-1” was chosen as the call sign at the Fox node. Four test modes were used, Free Text Only, Intentional Collision, Free Text With Voice, and File Transfer.

b. Free Text

Users were allowed to send any desired text at any desired time. As expected of actual tactical messages, these test messages tended to be one or two sentences long, well below maximum frame size. Collisions were rare, as users tended to await a response to one message before sending another. SL2I properly filtered valid frames based upon destination, allowing only data addressed to the local host to be passed up to the application. No frames were lost during the test.

c. Intentional Collision

This mode was designed to force frame collisions and evaluate SL2I's reliability functionality. Messages of at least 1200 bytes were cut from a text document and pasted into the Data Demo send window at two stations, Red-1 and White-1. These long messages required fragmentation into at least two frames. With voice coordination, users at these stations sent the messages simultaneously, ensuring collisions on at least the initial frames. The collisions caused an ACK-time out, and at each node, SL2I chose a random back-off period from 0 to 1000ms. Upon expiration of the back-off period, each node retransmitted its frame. All messages were received intact and entirely. No frames were lost.

A similar test was conducted during which intentional collisions were induced. In this test, Red-1 sent messages addressed to White-1, while White-1 sent messages to Blue-1. Again, users coordinated the sending of the messages to ensure collisions. Again, all messages were delivered intact and entirely.

d. Free Text with Voice

The SINCGARS radio is designed to allow analog voice transmissions while in data mode. In this test mode, Red-1 sent short text messages to White-1 while White-1 transmitted brief voice messages. Frames sent during voice transmissions were lost and had to be retransmitted, but the retransmissions were at the data link layer. When the voice transmissions were brief, SL2I was able to successfully deliver frames without the user having to resend them. Otherwise, data frames were lost due to collisions and repeated ACK time outs.

The SINCGARS radio will give priority to voice transmissions over data transmissions, so a station transiting a multi-frame text message can always override the text with a voice transmission. Red-1, however was able to successfully send a multi-frame message to White-1 while also sending brief voice messages.

e. File Transfer

At Red-1, the “Send Image” mode was selected in the Data Demo “Mode” drop-down menu. “Receive Image” was selected at Blue-1. A text message was sent from Blue-1 to Red-1, indicating that the station was ready to receive the image. During the file transfer, no other frames were transmitted. A 21.2KB JEPEG was successfully transferred.

IV. LAYER THREE DESIGN, IMPLEMENTATION, AND FUNCTIONAL TESTING

A. INTRODUCTION

Wireless communications systems have a finite physical range. In the absence of obstacles, this maximum physical range approximates a circle around an omni-directional transmitter like the SINCGARS. The range is dependent on the transmitter power, the receiver sensitivity, and the loss in signal power due to channel characteristics (range, multi-path interference, etc.). Line Of Sight (LOS) range is a distance from source to destination along a radius for this circle. The power of the transmission and the effects of diffraction may be sufficient to overcome many obstacles along any LOS, but large obstacles (i.e. hills) cast RF shadows.

A destination node is said to be in a hill's RF shadow when the hill is in the LOS between the source and destination, the source and destination nodes are on opposite sides of the hill, and there is not enough diffraction to provide a path around the hill. See Figure 7. The RF shadow, effectively, reduces the LOS range of the source, S, and prevents the destination, D, from receiving transmissions from S.

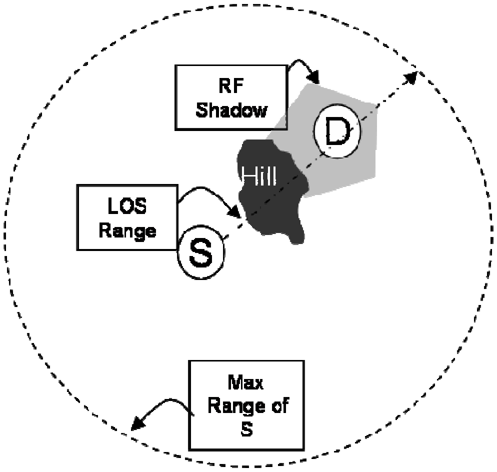


Figure 7. LOS Range and RF Shadow

The layer two protocols described in the previous chapter allow any two nodes within LOS radio range of one another to send and received data frames between them. However, the layer two protocols alone do not allow for data transfers between nodes that are not with in LOS range of one another. The LOS restriction can be overcome, however by an intermediate node (or nodes) that links the source and destination nodes. The intermediate nodes would route the data along a path to the destination node. This is the function of a layer three, Internetworking Layer, protocol.

In wireless networks, routing protocols can be divided into two broad categories: table-driven and on-demand. While nodes using table driven protocols maintain tables listing current routes to each available destination node, nodes using on-demand protocols dynamically discover routes to destination nodes only as required. There are positives and negatives associated with each of these two approaches. This chapter describes the design, implementation, and testing of a new ad hoc routing protocol called Expected

Relative Positioning Routing with Congestion Avoidance (ERP/CA). The protocol draws upon the advantages of existing approaches, while attempting to minimize their disadvantages.

This chapter begins with a brief discussion of the attributes of wireless table-driven and on-demand routing protocols. This overview provides a context within which ERP/CA may be positioned. The overview is followed by a detailed description of ERP/CA design and implementation. Finally, a discussion of the functional testing of ERP/CA is presented.

B. EXISTING AD HOC ROUTING PROTOCOLS

1. Table-driven Routing Protocols

Table-driven routing protocols are proactive and thus, react quickly to topology changes. Because tables are maintained for routes to all available destinations, there is no per-packet overhead associated with finding a route to a given destination. The proactive nature of these protocols, however, requires frequent “keep alive/ hello” messages that allow nodes to maintain an accurate picture of the network topology, and contributes significantly to general routing overhead. Also, to use table-driven protocols, it is necessary for nodes to periodically exchange complete routing tables. Table updates are also sent upon network topology changes. Table-driven protocols include Destination-Sequenced Distance-Vector (DSDV) and Zone Routing Protocol (ZRP) [Haas98].

2. On-demand Routing Protocols

Conversely, on-demand routing protocols do not require any exchange of routing tables. Instead, nodes reactively send route requests when data is available to be sent to a destination. This eliminates the overhead of the “keep-alive/ hello” messages but adds some latency to the delivery of data packets, due to route discovery. In general, on-demand protocols react more slowly to topology changes, but never maintain “stale” routes. Additionally, there is no need for nodes to send error messages in response to topology changes when on-demand routing protocols are used. On-demand protocols include Ad hoc On-demand Distance Vector Routing (AODV) [Perkins99], Dynamic Source Routing [Johnson99], and Associatively Based Routing (ABR) [Toh96, Toh99].

C. DESIGN OF THE ERP/CA ALGORITHM

This thesis develops and evaluates a hybrid ad hoc routing algorithm. ERP/CA 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 MANETs. ERP’s low overhead and persistent-path preference make it ideal for low-bandwidth networks like those dependent upon legacy SINCGARS radios. The most salient design features of the ERP/CA algorithm are presented in this section.

1. Key Design Factors

a. Designed to Exploit Domain Knowledge

Most MANET routing algorithms are designed with civilian networks in mind. 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.

(1) Military Unit Formations. Tactical formations are usually built around traditional military hierarchies. In accordance with TTPs, unit leaders, like Squad Leaders, Platoon Commanders, and Company Commanders, will position themselves in such a way that they can maintain visual and LOS radio contact with subordinate unit leaders or unit members. Figure 8 shows a tank company in a Company Wedge Formation.

In Figure 8, Black 5 and Black 6 represent the company Executive Officer and Commanding Officer, respectively. Red 1, White 1, and Blue 1 are each platoon commanders of Red, White, and Blue platoons, respectively. A platoon commander’s tank and three others comprise a tank platoon. To facilitate command and control, each platoon commander will maintain radio contact with the members of his platoon. Likewise, the company commander, Black 6, will maintain radio contact with each platoon commander.

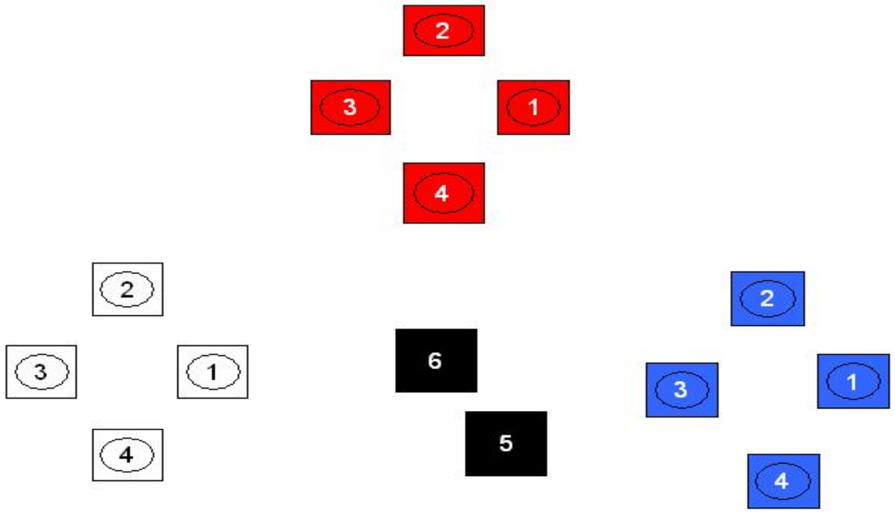


Figure 8. Tank Company In Wedge Formation

(2) Tactical Movement Techniques. Most people are familiar with the “wingman” concept employed by military pilots. Each pilot has a wingman that is not to be left. This concept is also employed by ground units, both wheeled and tracked. A tank commander, for example, will not leave his wingman. Their relative positions tend to remain constant. One can be expected to remain next to, and in radio contact with, one’s wingman, under normal circumstances, even while on the move.

Similarly, a platoon commander can be expected to remain in contact with the members of his platoon. Because military units travel in military formations, the relative positions of a platoon commander to that of his subordinate elements can be expected to remain the same. Company commanders tend to position themselves centrally, maintaining command and control over platoon commanders.

(3) Incorporation of TTP-Based Knowledge. The Wedge is just one possible tactical formation. Formations change in response to situation changes. Formation changes can lead to some relative positioning changes. Still, a tank will not normally move with out its wingman. Platoons will not be ordered to move without their platoon commanders.

Though mobile, the members of a company can be expected to maintain their relative positioning. ERP uses this TTP-based information to facilitate efficient routing that favors paths that are least likely to fail due to node mobility. ERP seeks to utilize the route that is most likely to have the greatest persistence and reliability. Route choices are based upon the expected relative position of the next hop to the destination.

Optimized by the information inherent to the mobility model, the expected relative position of a given node to the destination node is translated into expected likelihood of radio contact and is the primary factor in ERP/CA route selection.

b. Flexible Enough to Accommodate Mobility Model Exceptions

Military operations are event driven. Events sometimes cause elements of a military unit to deviate from normal TTPs. During such events, nodes in the network may not move in accordance with the mobility model and, therefore, may tend not to maintain their expected relative positioning. Such events include enemy contact, orders from a senior commander, and mechanical failure. Mechanical failure may cause a node to drastically change its position relative to others. During any of these occasions, ERP/CA is designed to revert to a simple shortest path first algorithm.

c. Designed to Find Persistent and Reliable Routes

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.

d. Designed to Avoid Congestion

The value of data is greatest when it is delivered in a timely fashion. 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.

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.

e. Designed to Have Low Overhead

Legacy tactical radios, like the SINCGARS, were designed primarily for voice traffic. The bandwidth for data communications is limited. Specifically, the SINCGARS has a 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.

f. Designed to Prevent Routing Loops

Data traveling along a routing loop will either be lost or trapped indefinitely, which serves only to consume scarce bandwidth. Avoiding routing loops allows ERP/CA to more efficiently utilize the shared medium and more reliably deliver data. Routes chosen by the EPR/CA algorithm are loop-free.

ERP/CA uses a form of Poison Reverse to avoid routing loops. A Route Paradox exists when two nodes consider each other as the next hop in the route to a given destination. Any node, I, receiving a Route Request to a destination node, D, will first perform a Route Paradox evaluation. Node I will determine if the next hop in its route to D is the node S, the source of the request. If so, a route paradox exists. Because ignoring this paradox could lead to a routing loop, node I will instead ignore the Route Request, allowing other nodes with valid routes to respond.

2. Control Message Types

ERP/CA uses four message types to control its routing functions. These message types are Hello, Hello Response, Route Requests, and Route Request Response. Duplicate control messages are considered stale and are discarded. Sequence numbers are incremented for each message, and duplicate messages are determined via examination of complete header content.

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

Figure 9. Hello and Hello Response Message Format

Hello and Hello Response Messages are used for the initial discovery of a node’s one-hop neighborhood. After this initial discovery, a node will not issue another Hello Message, but it will send Hello Response Messages. Both Hello and Hello Response messages are indicated by message type ‘H’. The destination address fields are used to distinguish between the two. A Hello Message is broadcasted by using address ‘00’ in both destination address fields. All Hello Response messages are unicatst and addressed to the requestor node.

Route Request Messages are used to identify previously unknown single and multi-hop routes. Routes to nodes in the single-hop neighborhood may become stale due to inactivity. On demand, these routes are rediscovered via Route Request Messages.

Figure 10 shows the format of Route Request Messages. These messages are indicated with by a ‘Q’ in the type field and have no payload. A Route Request Message is broadcasted by using address ‘00’ in both destination address fields. Route Request Messages include two additional header fields. The Requested Destination Node ID field is used to specify the node to which a route is requested. The Keep Alive field is used to limit the range of flooding for request messages. This value is set to 5 by the request originator and reduced by 1, at each node that forwards the request, until it reaches zero.

1 byte	**4 bytes**	**1 byte**	**1 byte**	**1 byte**	**1 byte**	**1 byte**
Frame Type	All Address Fields	Sequence Number	Requested Destination Node ID	Keep Alive	Checksum	End Of Frame
Q						

Figure 10. Route Request Format

Nodes with a valid route to a requested destination send a unicast Route Response Message in response to a Route Request Message. Route Response Messages are only sent in response to a Route Request. They are never spontaneously generated or used for route maintenance. Figure 11 shows the format of Route Response Messages. These messages are indicated with by an ‘R’ in the type field and have no payload. The Requested Destination Node ID field is used to specify the destination node to which this route refers.

1 byte	**4 bytes**	**1 byte**	**1 byte**	**1 byte**	**1 byte**	**1 byte**
Frame Type	All Address Fields	Sequence Number	Requested Destination Node ID	Keep Alive	Checksum	End Of Frame
R						

Figure 11. Route Response Format

3. Initial Node Discovery

Nodes attempting to enter the network broadcast a single Hello Message. At no other time will 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.

4. ERP Route Discovery

Discovery of new entrants to the one-hop neighborhood is insufficient. Nodes in a MANET must be able to dynamically discover routes to other nodes as well. Assume that there are three nodes in such a network, S, I, and D. Further, assume that the source and destination nodes, S and D, respectively, are not within radio range of one another, and assume that neither has a path to the other in its routing table. The intermediate node, I, has both S and D in its routing table. (Note that it is not essential that node S be in the routing table of node I at the start of the route discovery process.) This topology is depicted in Figure 12.

Further minimizing overhead, node S will not attempt to discover a route to node D unless, and until, S has data to deliver to D. When S has data to deliver to D, it will broadcast a “Route Request Message”. All nodes within range of S will prepare a “Route Response Message,” if they have a route to D. Those that do not have a route will prepare to flood the route request. Route Requests are sent with a Time-To-Live (TTL) of five hops, and no node will re-broadcast a previously received request. A Route Response from node I will cause all nodes hearing it to cancel any pending Route Response to the request from S. As a function of congestion avoidance, S will choose the first Route Response received, in the event of multiple responses.

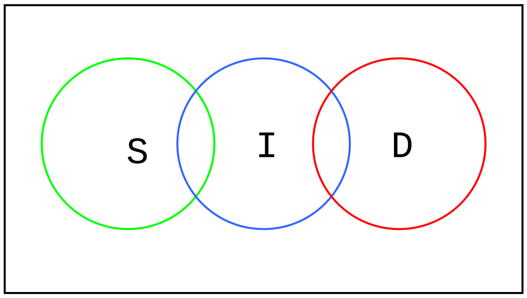


Figure 12. Node I Must Route Data From S To D

Upon receipt of the Route Response from I, S will add the direct path to node I to its routing table, if it were not previously there, and it will add the path to node D to its routing table as well. Data from S to D will then be sent via I.

In seeking routes to a destination node, the ERP algorithm assigns weights to each potential next hop node based upon its expected relative position to the destination. The destination node itself is given the highest weight. The next two highest weights are assigned to the wingman and platoon commander, in that order.

Finally, the lowest weight is assigned to a node with no expected relative position, for example a node from another platoon. There may be times when the best node through which to route data to a destination is not the destination node’s wingman or platoon commander. This could happen, for example, when both the wingman and platoon commander are out of contact with the destination node as a result of radio or mechanical failure. Vehicle mechanical failure could prevent one node from maintaining its expected relative positioning.

In military units, each element knows its tactical role in advance. Company commander, platoon commander, and wingman roles are assigned and well known well in advance of any tactical operation. Call signs, like “Red-1” and “Blue-4” can inherently connote such roles. “Red-1” is assigned to the platoon commander of Red Platoon. Blue-4 is the wingman of Blue-3, etc. ERP incorporates this knowledge to facilitate reliable routing. Nodes responding to Route Requests do so based upon their respective relationship to the destination node. Each node self-assigns a relative weight for each potential destination node. The weights are then multiplied by a constant temporal value. Nodes receiving a route request will respond only after waiting for a period of time equal to the relative weight multiplied by the temporal constant. This assures that the destination node, wingman, and platoon commander will respond to route requests in that order. Because nodes hearing a Route Response will cancel their pending responses, and because source nodes respond only to the first response, the most persistent and reliable routes are chosen.

5. ERP/CA Distributed Route Selection Algorithm

The distributed route selection algorithm is used to make multi-hop route selections. It determines how long a given node will wait before sending its 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).

RRW = CW + CAV + IRW

a. Categorical Wait Value

CW's are weighted heaviest and have four potential values. 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* is the largest. The next category is *Better*. The Better delay value is smaller than *Good*, and is used by the platoon commander of the destination node. Wingmen are self-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-assign a CW value of zero and an RRW of zero. Destination nodes respond without delay to Route Requests.

Category	**Time Value Assigned**
GOOD	1500ms
BETTER	1000ms
BEST	500ms
DIRECT LINK	0ms

Table 5. Categorical Response Wait Values

b. Congestion Avoidance Value

CAV values are determined by multiplying the number of routing table entries by the CAV-weight, 21.5. 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.

c. Individual Response Wait

IRW values are determined by multiplying the node’s unique integer address by the IRW-weight, 12.25. 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.

6. Examples of Applying ERP/CA Algorithm

For simplification, the below examples discuss expected relative position calculations and congestion avoidance calculations as though they were mutually exclusive. In practice, however, the two metrics are used jointly, in the distributed algorithm, to make route selection decisions.

a. Multi-hop Routing Examples

Two tank platoons, Red and Blue, are shown in Figure 13. These two platoons represent a portion of a larger unit and larger ad hoc network. In Figure 13, the “1” elements are platoon commanders, and they are wingmen for their respective “2” elements. In each platoon, elements “3” and “4” are wingmen for each other. Each node in a platoon is in range of each other node in that platoon, but only Red-2, Red-1, Blue-2, and Blue-4 have cross-platoon connectivity.

Suppose that Red-2 has data to send to Blue-3 but has no route. Red-2 will broadcast a Route Request. All nodes within range will either prepare a Route Response (if they have a valid route to the destination), or forward the request (if they do not have a valid route). Duplicate requests are ignored. Blue-2 and Blue-4 will both prepare a response. As the wingman for Red-3, Red-4 will be the first to respond to the request, preempting any response from Red-2. Figure 14 shows the unicast Route Response from Blue-4, and Figure 15 shows the data routed from source to destination.

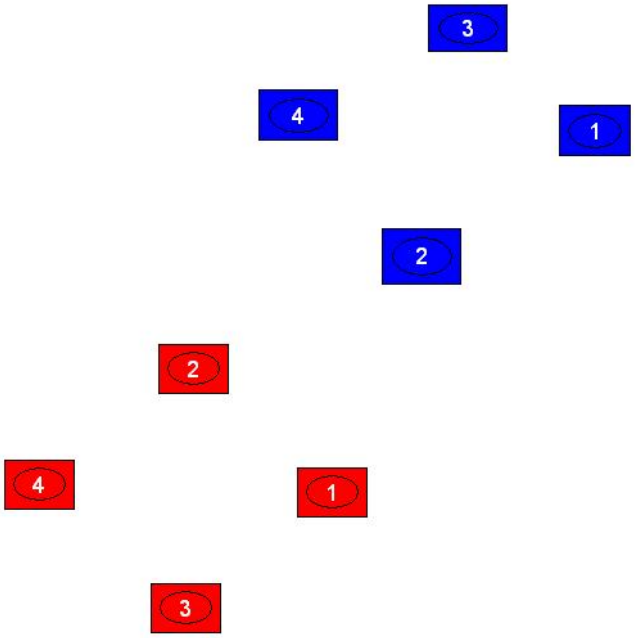


Figure 13. Two Platoons In A MANET

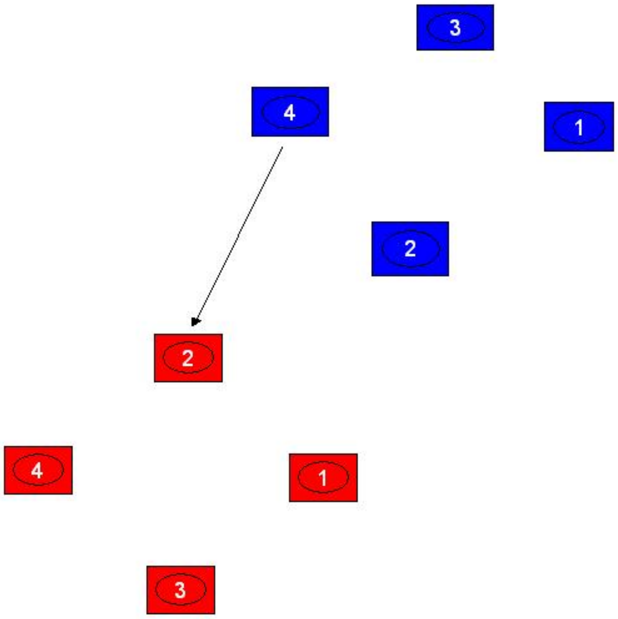


Figure 14. Unicast Route Response

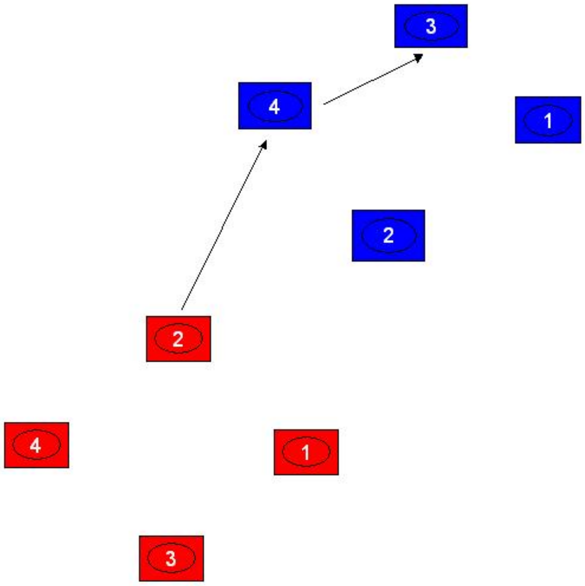


Figure 15. Most Persistent Route From Red-2 To Blue-3

With knowledge of the TTPs and military mobility patterns, it is clear that no two-hop route from Red-2 to

Blue-3 will have greater persistence and reliability than the route: *Red-2* → *Blue-4* → *Blue-3*.

Assume now that Blue-3 has data to send to Red-3 and that no Blue node has a route to Red-3. Blue-3's Route Request broadcast will be forwarded by all Blue nodes. See Figure 16. Red-2 and Red-1, both with routes to Red-3, will receive the forwarded Route Request. Neither of these nodes is a wingman for Red-3, but Red-1 is the platoon commander node for Red-3. As such, Red-1's response will preempt the response from Red-2. Assuming that Red-1 first received the request from Blue-2, Red-1 will unicast a Route Response to Blue-2. As shown in Figure 17, Blue-2 will then unicast its Route Response to Blue-3. As shown in Figure 18, *Blue-3* → *Blue-2* → *Red-1* → *Red-3* is the most persistent path from Blue-3 to Red-3.

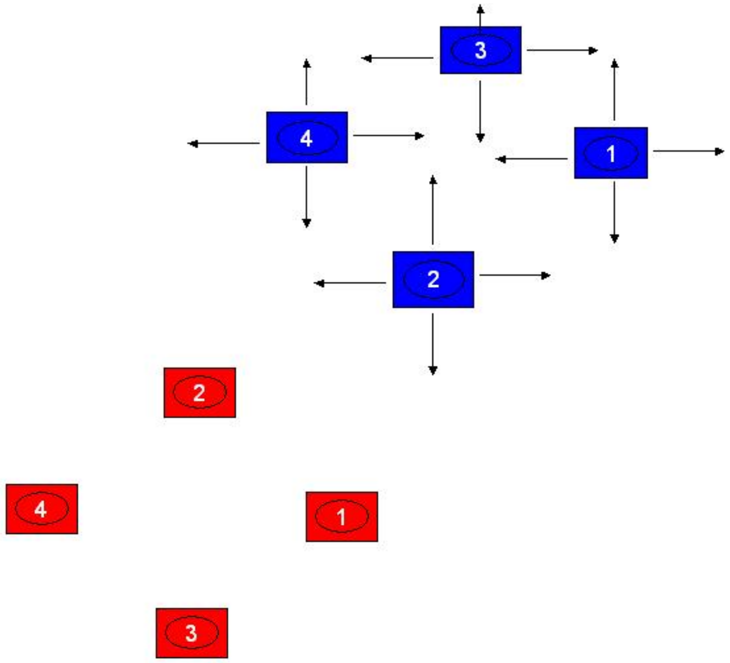


Figure 16. Flooded Route Request Stops At Red-2 and Red-1

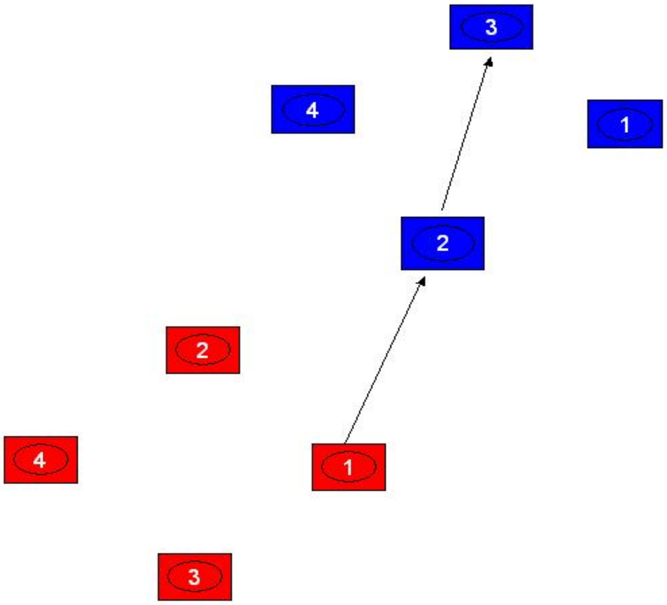


Figure 17. Multi-hop Route Response

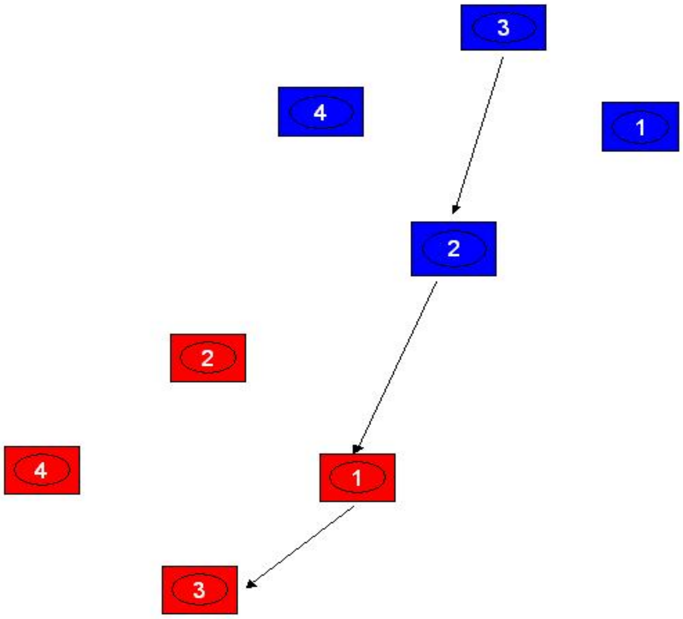


Figure 18. Most Persistent Route From Blue-3 To Red-3

b. Congestion Avoidance Example

Assume that Blue-3 now has data for Red-1 but no route to Red-1. Further, assume that both Blue-4 and Blue-2 have direct paths to Red-1 and that Blue-1 has a two-hop path, via Blue-2. Figure 19 shows multiple paths, one from each of these nodes, to Red-1. Because Blue-1, 2, and 4 all have a path to Red-1, a Route Request for Red-1 from Blue-3 will not be flooded. The first response from any of these nodes will preempt all others.

As none of these nodes is a wingman or platoon commander for the destination node, node-specific response waiting periods will be based solely upon routing table size. Since route discovery is largely demand based, nodes with larger routing tables have a greater likelihood of high traffic volume. More traffic leads to more congestion. The node least likely to experience congestion will be the first to respond to the route request from Blue-3. Note that the least congested, and thus most reliable, route may be the longer three-hop route via Bue-1. Ties caused by nodes having equally sized routing tables are broken in favor of the node with the smallest node address value.

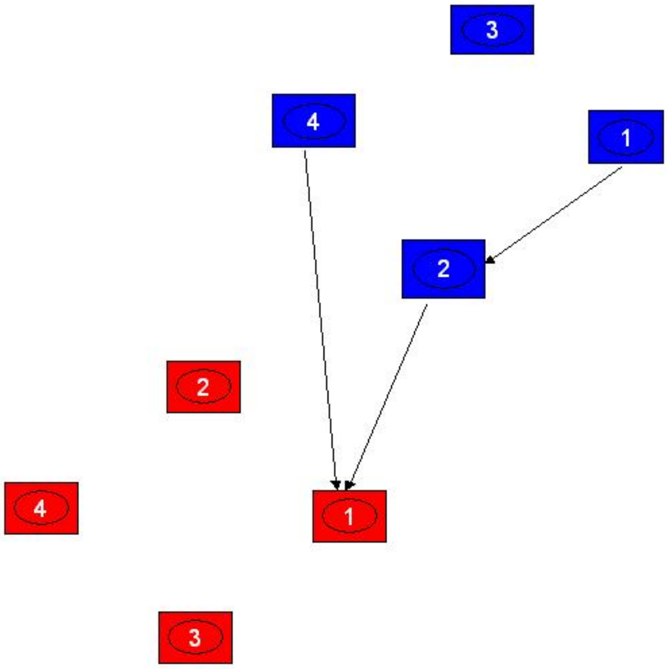


Figure 19. Potential Routes To Red-1

7. ERP Routing Tables

ERP/CA routing tables do not contain full routes to any destinations beyond a node’s one-hop neighborhood. Instead, for a given node, S, the entry in its routing table for destination node, D, will consist only of the next-hop node and a timestamp indicating freshness. In the routing table of S, an entry of “B” as the route to D means that node B is the next hop from S to D.

8. Route Maintenance

Paths are considered stale and removed from the routing table after ten minutes of inactivity. Each node also removes routes that include a bad link to the next hop. ERP/CA receives reports of suspected bad links from SL2I. After two such reports, the link is labeled bad. Subsequent execution of the route discovery process may lead to reestablishment of links previously labeled as bad.

Nodes in the one-hop neighborhood are most likely members of the same unit (i.e. same platoon). These single-hop paths are least likely to change and are kept fresh in the routing table when frequently used. When node S sends data to its single-hop neighbor, D, and receives an ACK from D, the timestamp on the path to D is updated. Using ACKs as indication of valid routes ensures that only valid two-way paths are refreshed. It also eliminates the need for additional overhead to maintain the route freshness.

The ERP algorithm's preference for persistent routes minimizes the need for route changes, even with multiple hops between source and destination. Situations will occur, however, such that destination nodes are no longer available via originally selected routes. The link failures, mentioned above, will cause at least one node along the path to drop the route to the destination node. This allows for on-demand route repair without the overhead of route error messages.

The next time that data is available for that destination node, a new route will be dynamically discovered. The reactive nature of ERP towards route failure means that data packets may be delayed or dropped during the repair. This risk is considered an acceptable trade-off to the additional overhead that comes with the flooding of route failure messages.

D. ERP/CA IMPLEMENTATION

The primary SL2I logic and the implementation of the ERP/CA algorithm are executed in separate Java classes. The Router class handles all ERP/CA functionality.

1. Encoding of TTP-Based Knowledge

ERP/CA depends upon upper layer processes to deliver the encoded TTP-based knowledge in the form of a list of node call signs. The Router assumes that the list is sorted in a way that makes wingman and platoon commander relationships evident. The first and second elements on the list are treated as wingmen for one another. The second and third, fourth and fifth, etc are also treated as wingmen for one another. The first element and every fourth element after it on the list are platoon commanders (i.e., elements 1, 5, 9, etc).

This allows the Router to determine the role of the local node based upon its call sign’s position on the list. If the node’s call sign is first on the list, it is a wingman for the second node, and it is a platoon commander for nodes 2 through 4. This knowledge allows the router to assign appropriate values to the Categorical Wait variable in the distributed route selection algorithm.

2. Implementation of the Distributed Route Selection Algorithm

The distributed route selection algorithm is used to determine a given node’s RRW. Variables in the algorithm include the nodes CW, CAV, and IRW. The encoding of the TTP-based knowledge, described above, allows each node to “know” its expected relative position to that of any other given node.

The distributed route selection algorithm is implemented in the Router class. The result of the RRW calculation is returned by the getRouteResponseDelay method.

a. CW Value Assignment

The time-values associated with each wait category are hard-coded into integer variables. Given, the destination node to which a route is requested, each node will determine its categorical wait based upon its expected relative position to the destination.

b. CAV Value Assignment

The integer variable, nodeCount, is used to track the number of destination nodes to which a given node has valid routes. This value is incremented when new routes are discovered and decremented when routes are removed from the routing table. CAV values are determined by multiplying the nodeCount value by the CAV-weight, 21.5. This value is then rounded to the nearest integer value.

c. IRW Value Assignment

Integer values are used to uniquely identify each node. IRW values are determined by multiplying the local node’s unique integer address by the IRW-weight, 12.25. This value is then rounded to the nearest integer value.

3. Congestion Avoidance Implementation

As mentioned above, CAVs are based upon the size of a node’s routing table. Higher weights are assigned to larger tables, and these weights lead to longer wait periods before a node’s response to a Route Request. As a result, Route Responses from nodes that are likely to be heavily congested have a longer wait period than those that are not likely to be heavily congested. Because nodes hearing a Route Response will cancel their pending responses, and because source nodes respond only to the first response, paths that are least likely to be congested are chosen.

4. Control Message Implementation

ERP/CA control Messages are not treated as payload for SL2I. Instead, SL2I’s implementation of the ERP/CA algorithm simply incorporates additional, ERP-specific, frame types. Each control message is assigned a frame-type identifier. See Table 6.

A frame of type-H is treated as a Hello Message when it is a broadcast frame. Unicast H-frames are processed as Hello Response Messages. Hello Messages received by SL2I are processed by the Router class, and Hello Responses are generated by the Router class as well. Similarly, Route Response Messages are generated by the Router class. They are generated only in response to Route Requests received by SL2I and processed by the Router. Hello and Hello Response Messages are distinguished by their addressing types—unicast or broadcast.

Message Type	Indicator	Addressing Type
Hello	H	Broadcast
Hello Response	H	Unicast
Route Request	Q	Broadcast
Request Response	R	Unicast

Table 6. ERP/CA Control Message Frame Types

5. Route Maintenance Implementation

Failure to receive an ACK to a message after the maximum resend attempts constitutes an ACK failure. These failures are reported to the Router class via its nonResponseNotification method. Multiple ACK failures are indicative of a possible link failure. For example, if attempts to send data from S to D, via I are not ACK'ed by I, node S will record an ACK failure. Multiple ACK failures, will cause S to remove the paths to both I and D from its routing table.

6. ERP Routing Tables

Routing tables are stored as arrays in the Router class. Each entry in the routing table consists only of Destination Node ID, ID of the Next Hop to the destination, and a time stamp. Time stamps are generated when the route is discovered and refreshed (via the resetRouteTimer method) upon receipt of an ACK from the route's next-hop node.

7. Routing Loop Avoidance

Route Paradox evaluations are performed in the Router class by the checkForRoutingParadox method, which returns a boolean value. Route Responses are generated only for requests that do not generate a Route Paradox.

E. FUNCTIONAL TESTING OF THE ERP/CA ALGORITHM

The Java class that implements the ERP/CA algorithm was tested to evaluate its ability to dynamically discover routes to other nodes. Additionally, the functional testing was conducted to test the integration of the ERP/CA module with the SL2I modules. Once again, the testing was conducted using the SINCGARS Data Demo application running directly above ERP/CA and SL2I.

1. Test Setup

Test setup was identical to that described in Chapter II, with one exception. At each node, the PC's were connected directly to the Audio/Data connector on the front panel of the SINCGARS, but the antenna connector for the

Fox radio was connected to the radio via an RF attenuator. This device was used to attenuate RF signals sent and received by the Fox radio.

By attenuating the RF signals, extreme distance/RF shadow was simulated. Although each node was physically located within the same room, the 20dB attenuation allowed for the effective topology shown in Figure 20. With White-1 and Blue-1 (the Fox node) effectively out of LOS range of one another, Red-1 would have to be used to route traffic between the two. Throughout the experiment, a two-way link was maintained between Red-1 and White-1 and between Red-1 and Blue-1. The Five test modes used were New-Join Discovery, Dynamic Discovery of Multi-Hop Paths, Dynamic Discovery of One-Hop Paths, Path Discovery via Route Request Flooding, and Congestion Avoidance.

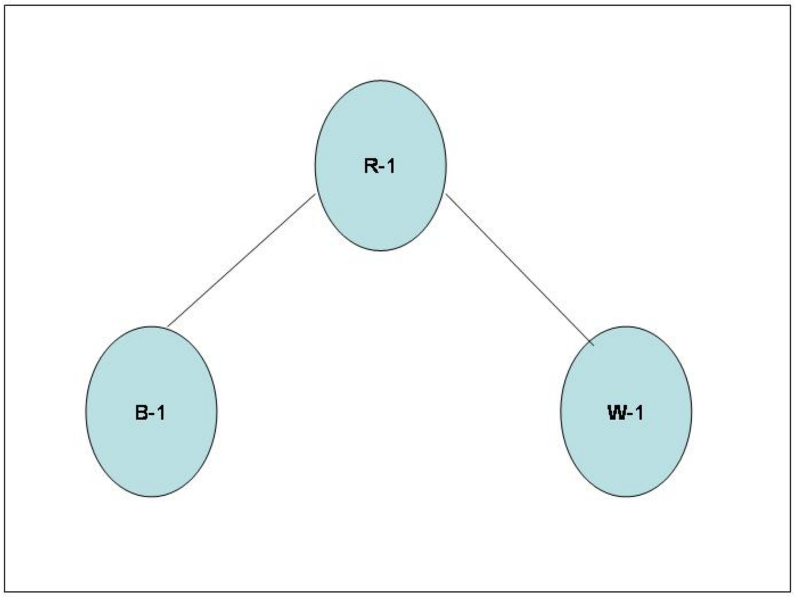


Figure 20. Effective Topology After Attenuation Of Blue-1

2. New-Join Discovery

Discovery of nodes initially joining the network is the most basic of ERP functionality and the easiest to test. When the user clicks the “Open Port” button on the Data Demo GUI, an ERP Hello Message is triggered. Red-1 was first to transmit its Hello Message. Since there were no other nodes online to respond to the message, Red-1’s routing table remained empty.

Blue-1 was brought online next. Red-1 responded to its Hello message, and the two nodes added the direct paths to each other to their respective routing tables. When White-1 was brought online, its Hello Message was heard only by Red-1, as White-1 was effectively out of range of Blue-1. White-1 added the direct path to Red-1, upon receiving the Red-1’s response to the Hello Message. At this point, Red-1 had valid paths to each of the other two nodes, while White-1 and Blue-1 each had a path to Red-1 only.

With equal success, this test was repeated with each node serving as the first and last node to join the network. After new-join discovery, Red-1 was able to directly send and receive messages to each of the other two nodes.

3. Dynamic Discovery of Multi-Hop Paths

With Blue-1’s routing table containing only a path to Red-1, the user at Blue-1 composed a message for White-1. The user selected the “Send” button on the Data Demo. Before transmission of the data, Blue-1 first transmitted a Route Request. The request was heard only by Red-1, which delivered a response. Blue-1 then updated its routing table and sent its data via Red-1. Red-1 sent an ACK to

Blue-1 and successfully delivered the data to White-1. This test was repeated, with equal success, with White-1 discovering the two-hop path to Blue-1.

4. Dynamic Discovery of Single-Hop Paths

In a MANET, a node's one-hop neighborhood will be altered not only by nodes joining and leaving the network but also by node mobility. A node that has previously sent its Hello Message and joined the network may move into a new one-hop neighborhood. Nodes using ERP/CA must be able to discover direct routes to these nodes.

After the New Join Discovery tests, Red-1, White-1, and Blue-1, were assigned the call signs Red-2, White-2, and Blue-2, respectively. Because a call sign change does not generate a new Hello Message, this effectively represented three new nodes moving into the one-hop neighborhood. An attempt by the user to send data from Red-2 to Blue-2 required a Route Request broadcast. Upon receiving the request, Blue-2 added Red-2 to its routing table. Blue-2's Route Response allowed Red-2 to update its routing table, and send data to Blue-2 directly. On demand, each node was able to discover direct paths to nodes within radio range and successfully deliver data.

5. Path Discovery via Route Request Flooding

The network topology in Figure 16 could be the result of all nodes moving to their current positions from another where they had previously joined the network. The nodes would not generate new Hello Messages as a result of the move. This is the scenario for the Multi-Node Path Discovery test mode.

The test begins with none of the nodes having a route to another node. As in previous tests, this state was achieved with call sign changes after the generation of the Hello Messages. With data to send to Blue-1, White-1 generated a Route Request. Red-1 was the only recipient of the request and had no route to Blue-1. Red-1 forwarded the request for a route to Blue-1. Red-1’s request was received and responded to by Blue-1.

Red-1 first updated its routing table, adding Blue-1, and then sent a Route Response Message to White-1. After updating its routing table, White-1 successfully delivered its data to Blue-1, via Red-1. Subsequently, Blue-1 was able to successfully deliver data to Blue-1 after dynamically discovering a route as well.

6. Congestion Avoidance

Only three radios were available to support the conduct of this research and testing. The limited resources prevented the full testing of the congestion avoidance functionality. Only one aspect was tested.

Failure of a node to send an ACK after maximum attempts to send a data frame may be indicative of a link failure. At a minimum, it is indicative of congestion. Multiple failures should cause removal of the path from a sending node’s routing table.

This test began with each node having a valid route to each other node. Data was successfully transferred between each pair of nodes. Then Red-1’s call sign was changed to Red-2. A message for White-1 was composed and sent by the user at Blue-1. At Blue-1, SL2I attempted the max number of resends, but received no ACK from Red-1. The potential link failure was reported to the ERP/CA module. A second message met with the same fate and caused Blue-1 to drop the routes to Red-1 and White-1. The call sign at Red-2 was changed back to Red-1.

Assuming that the failure to successfully deliver the data was due to congestion and not node failure, Blue-1 broadcasted a new Route Request when data for Blue-1 was again available. The “new” route was discovered, routing tables were updated, and the data was successfully delivered.

V. CONCLUSION, RECOMMENDATIONS, AND FUTURE WORK

A. OVERVIEW

This thesis proposes a custom layer 2 protocol, SL2I and a novel MANET routing protocol, ERP/CA. SL2I provides an interface between legacy, voice-centric, tactical radios (like the SINCGARS), and the serial port of a PC. SL2I provides link-by-link, reliable delivery of frames, error detection, and media access control. ERP/CA provides dynamic, on-demand, paths for delivering data across a multi-hop wireless network.

Informed by TTPs governing mobility in tactical units, the ERP/CA algorithm uses path preference as the primary metric for route selection. Routes chosen by ERP/CA are least likely change in the face of node mobility in tactical environments. Further, the congestion avoidance nature of the ERP/CA algorithm can be expected to aid in the mitigation of network latency frequently found in reactive MANET protocols.

B. CONCLUSION AND RECOMMENDATIONS FOR SL2I

SL2I was designed as a proof of concept solution. It provides software-based data link services to disadvantaged tactical units—those having only legacy, voice-centric radios. SL2I provides a reliable means of point-to-multipoint, packet-switched communication over a shared wireless medium.

The functional testing of SL2I provided promising results. It is clear that the hardware-based COTS solutions currently being marketed to various military commands are not the only solution. This testing, however, was limited and provides no insights into how the protocol will scale. Additional characterization of SL2I’s performance should be conducted. Further, it is recommended that additional experimentation be conducted to further optimize the protocol.

C. CONCLUSION AND RECOMMENDATIONS FOR ERP/CA

MANET routing protocols have traditionally been designed to with civilian applications in mind. Efforts have largely focused on ways to optimize the protocols, given the expectation of random movement patterns among the nodes in the network. This optimization has frequently been achieved with the use of heavy control and administrative overhead traffic.

Tactical units do not move in random patterns. Their mobility is governed by TTPs. ERP/CA makes use of this knowledge to discover optimal routes without overburdening the network with excessive control traffic. Because the tactical relationships between nodes in the network are policy based, this information is available in advance. Instead of discovering the unique tactical relationships between the nodes in the network, it is encoded into the ERP/CA protocol. Given the limited bandwidth of voice-centric radios, ERP’s path persistence minimizes route requests, minimizes the need for route repairs, and minimizes the overall negative impact of routing traffic on data throughput.

The functional testing of ERP/CA provided promising results. The implementation allowed for the dynamic discovery of multihop routes that were consistent with the persistent tactical relationships. It can be expected that this algorithm will yield routes that require far less maintenance (due to node mobility) than those selected by random-mobility-based protocols. Further, it can be expected that ERP/CA will discover these routes with far less overhead.

This testing of ERP/CA in this thesis, however, was limited and provides no insights into how the protocol will scale. Additional characterization of ERP/CA’s performance should be conducted. Further, it is recommended that additional experimentation be conducted to further optimize the protocol.

D. FUTURE WORK

1. SL2I

a. Issue: Data Delivery Reports

As currently designed and implemented, SL2I does not make delivery status reports available to applications. In the event of undeliverable data (meaning maximum attempts to transfer data across a link have been exhausted), the data is dropped without warning. The infrastructure presented in this thesis does not include a transport layer, which could resend the data without additional user intervention. (Note:ERP/CA is dependent upon link failure reports for proper operation, and it currently does receive them from SL2I.)

b. Future Work: Data Delivery Reports

The addition of transport layer has been considered. Even a fairly simple transport layer, which would automatically attempt to resend data without user intervention, would improve the reliability of data transmissions. A simple protocol that made, for example a maximum three attempts should be investigated. This protocol, should avoid the virtual connection setup and teardown associated with TCP, in order to preserve the low overhead objectives of the protocols described in this thesis.

c. Issue: Position Location Information (PLI)

Tactical data networks are used extensively for establishing and maintaining a common situational awareness. This awareness is significantly dependent upon knowledge of friendly position locations. Although applications could be written to exchange PLI as payload in SL2I frames, SL2I currently has no inherent support for exchanging PLI automatically.

d. Future Work: Position Location Information (PLI)

Today it is becoming increasingly common to find portable Global Positioning Satellite (GPS) receivers that connect to laptop computers via USB cables. Many GPS receivers are even available as PCMCIA cards. These devices allow for each node to know its own location. SL2I could be modified to include PLI data in the header of each frame, facilitating automatic updates to the collective situational awareness of all nodes in the network. To enhance the tactical applicability of SL2I, future work should include an exploration of this modification.

e. Issue: Porting to Personal Data Assistants (PDAs)

In the Marine Corps, laptop computers in the form of the Mounted-Data Automated Communications Terminal (M-DACT) have become commonplace in tactical vehicles (like tanks and Light Armored Vehicles (LAVs)). The dismounted version of this device, the D-DACT is a ruggedized HP iPAQ5500 series Pocket PC is being fielded to infantry units throughout the Marine Corps.

SL2I was written in Java to provide a cross-platform solution. During this thesis research, there was limited success with porting the SL2I implementation to a PDA. Attempts were made to port the software to the Windows CE-based iPAQ5555, with limited success. Time constraints prevented the complete cross-platform implementation.

f. Future Work: Porting to Personal Data Assistants (PDAs)

Providing MANET capability to dismounted units will be equally, if not more, beneficial than providing the capability to mounted units. The tactical radios used by infantry units are battery powered and have significantly less range, as compared to vehicle mounted radios. As a result, D-DACT equipped infantry units could benefit greatly from the reliability and potential range improvements afforded by a multihop networking infrastructure. To expand the tactical applicability of SL2I, future work should include porting to the Windows CE platform.

2. ERP/CA

a. Issue: Route Maintenance

The default value used for route freshness is ten minutes. It remains to be seen if this value is optimal for the ERP/CA algorithm. It is clear, however, that a route failure during this interval will generate no route error messages. As a result, data sent incident to the route failure is lost. In the absence of route error reports, source nodes must depend on users to recover from such dropped data. For the expected environment, this potential burdening of the user was seen as acceptable, in exchange for the savings in overhead.

b. Future Work: Anticipatory Link (Route) Verification

In addition to optimizing the ERP/CA route freshness parameter, future experimentation should include anticipatory link/route verification. In the absence of frequent data exchanges, which produce ACKs used to update ERP/CA route freshness, routes may become invalid prior to the expiration of route freshness intervals.

Without all the overhead associated with broadcasted route requests, a node wishing to verify the status of the link to the next-hop node could send a verification frame to that next-hop node. This verification frame would be unicast and have no payload. An ACK to this verification frame would justify continued use of routes using this link. When no ACK is received, the route discovery process could be reinitiated.

The verification frame should be sent only on demand, and only for routes whose freshness has not expired. Further, anticipatory verification should be used only when the route's freshness has reached or exceeded half of the route freshness timer; in the default case, this threshold is five minutes. The use of anticipatory route verification seems a promising solution to reduce the likelihood of data loss due to undetected stale routes in ERP/CA, without having to introduce route error messages. Future work should include an exploration of this concept.

c. Issue: Multicasting

Multicasting is commonly used in tactical data networks, but ERP/CA currently has no multicasting functionality.

d. Future Work: Multicasting

Future additions to ERP/CA’s functionality should include multicasting. The inherently hierarchical nature of military units seems to facilitate multicasting in much the same way that it facilitates ERP’s routing algorithm. Future work should include an investigation of commander-based multicasting. A company commander, for example can usually transmit multicast frames to each platoon commander with a single transmission. Each platoon commander can similarly transmit such frames to the respective platoon members with a single transmission. Those nodes registered to receive the particular multicast will process the frames. All others will discard them. Registration would be internal to each node.

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX

In the figures below, a Finite State Machine (FSM) specifying the operation of the ERP/CA algorithm is given. The entire FSM is too large for a single page, and is thus presented in segments.

Figure 21 presents the machines initial states and transitions. States marked with an asterisk are those which have transition listed in later figures. These are presented in Figures 22 through 25.

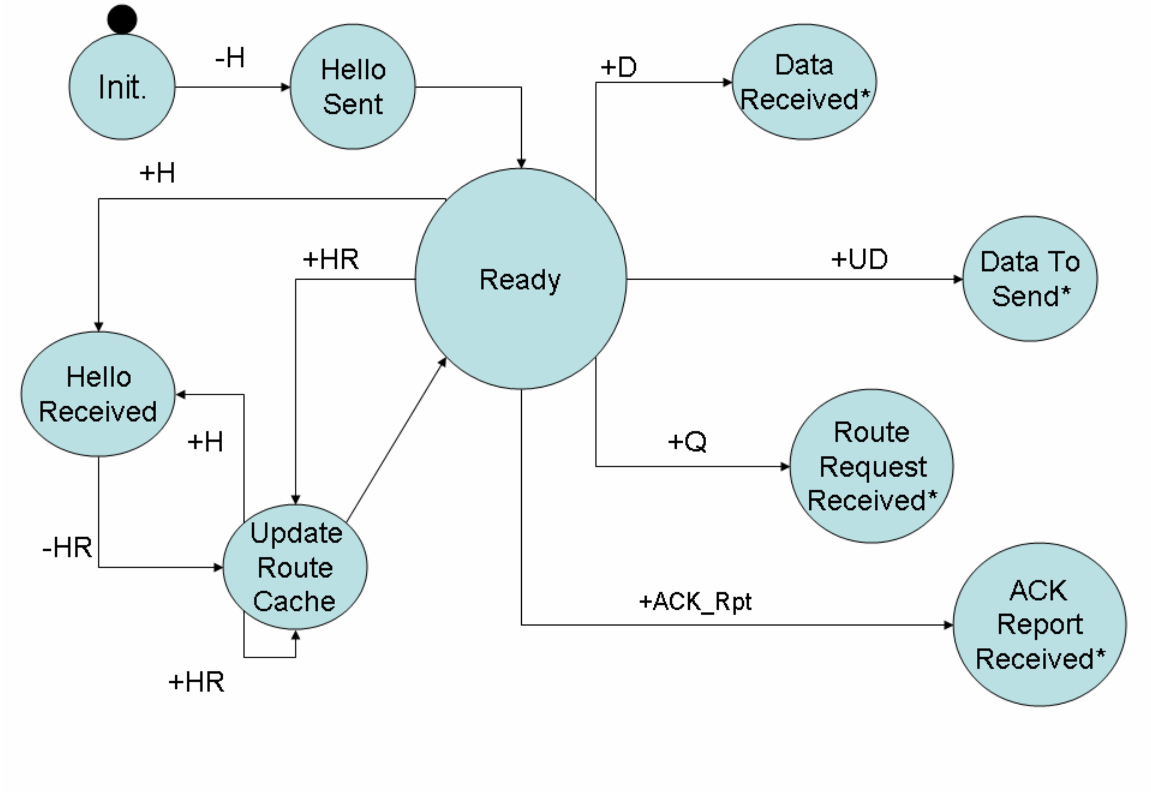


Figure 21. Initial State and Transitions

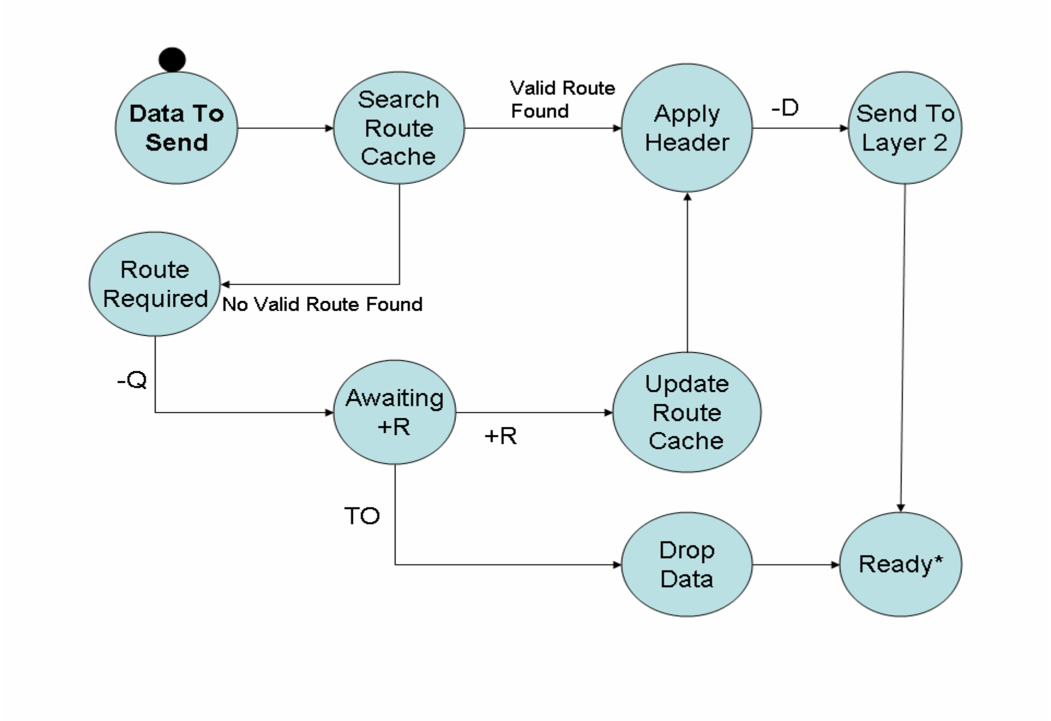


Figure 22. Sending Data to Another Node

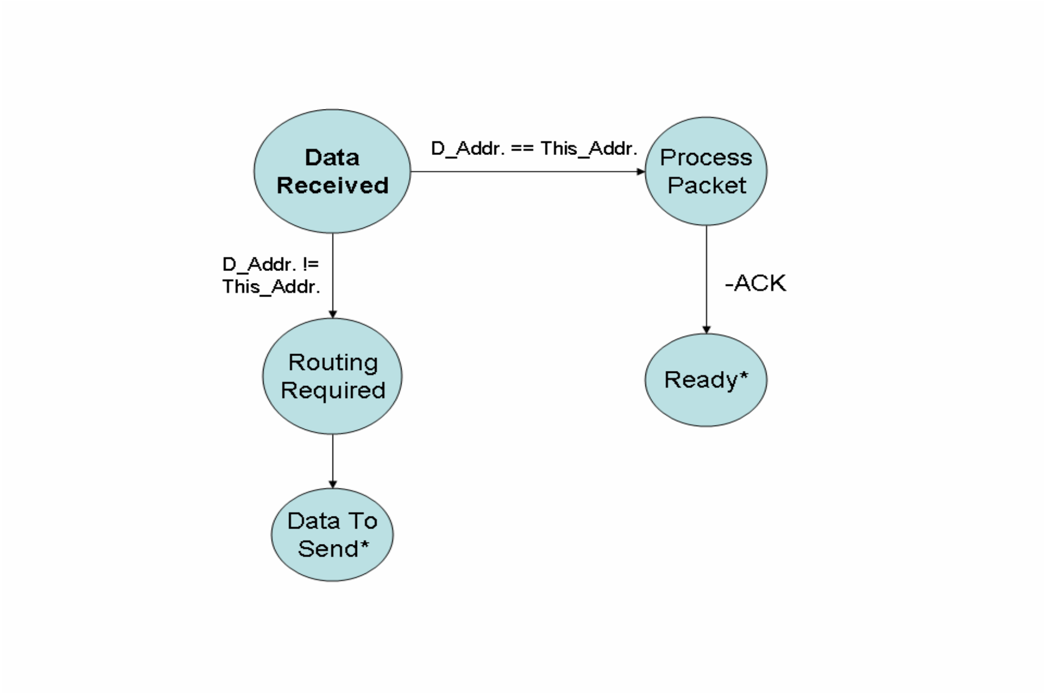


Figure 23. Receiving Unicast Data from Another Node

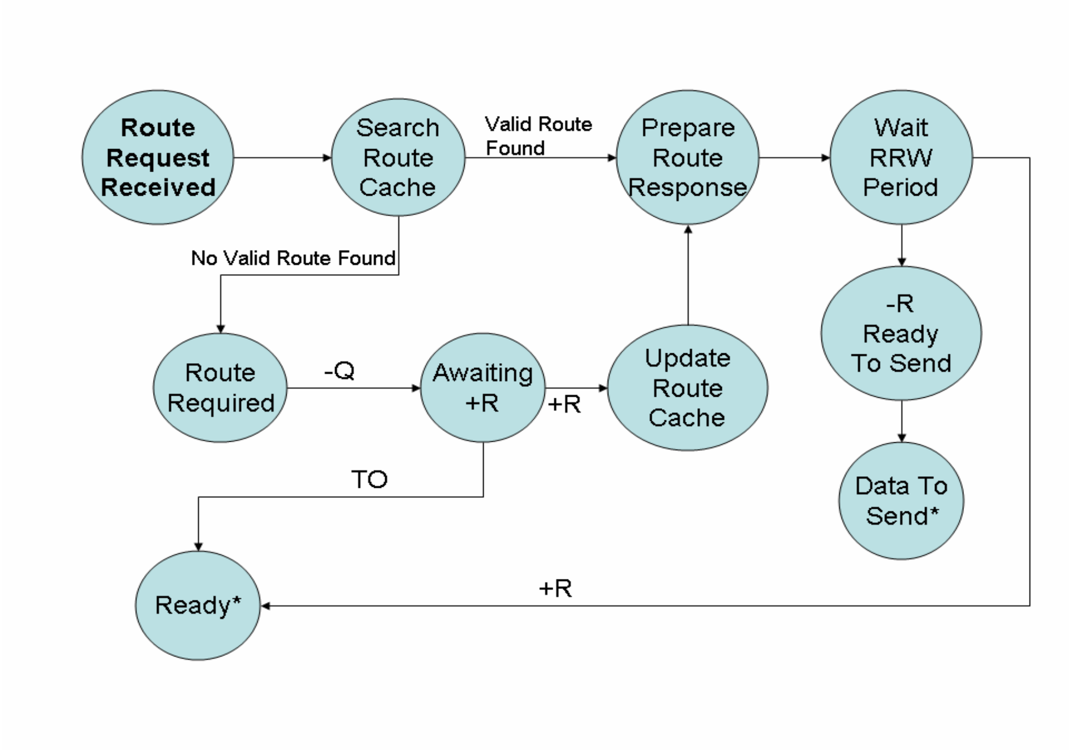


Figure 24. Processing a Route Request

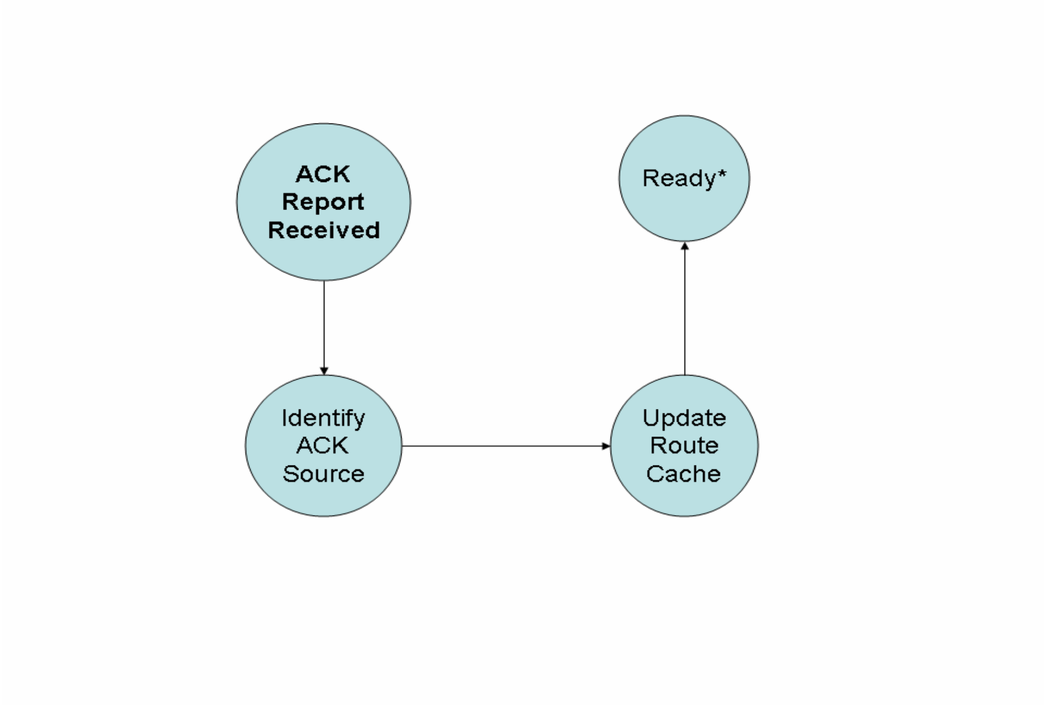


Figure 25. Receiving ACK, Updating Route Freshness

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF REFERENCES

[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 05.

[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, 782 pages.

[Forouzan01] Behrouz A. Forouzan. *Data Communications and Networking*, 2nd ed., McGraw-Hill, 2001.

[Groombridge04] CW2 Kenton Groombridge. “Setup of the ASIP SINCGARS for Data Communications,” 2004. http://team-signal.net/html/Groombridge.htm. Last visited 26 January 2006.

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

[Harold99] Elliotte Rusty Harold. *Java I/O*, O’Reilly, 1999.

[Howell04] Ryan A. Howell and Craig B. Abler. “The Digital Battery ‘TOC’,” FA Journal Sep-Oct, 2004.

[Intel05] Intel and Cisco. “Deploy Wireless LANs with Confidence: A Guide to Secure, End-to-End WirelessLAN Solutions,” 2005. www.intel.com/business/smallbusiness/ wireless/smb_wireless_guide.pdf. Last visited 26 January 2006.

[Johnson99] David B. Johnson, Davis A. Maltz, "The Dynamic Source Routing Protocol for Mobile Ad Hoc Networks," October 1999 IETF Draft, 49 pages.

[Kurose05] Kurose, J.F., and Ross, K.W., *Computer Networking: A Top-down Approach Featuring the Internet*, 3rd ed., Addison-Wesley, 2005.

[Perkins99] Charles E. Perkins, Elizabeth M. Royer, Samir R. Das, "Ad Hoc On-demand Distance Vector Routing," October 99 IETF Draft, 33 pages.

[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).

[Tanenbaum81] Tanenbaum, Andrew S., *Computer Networks*, Prentice-Hall, 1981.

[Toh96] Chai-Keong Toh, "A novel distributed routing protocol to support Ad hoc mobile computing," Proc. 1996 IEEE 15th Annual Int'l. Phoenix Conf. Comp. and Commun., March 1996, pp. 480-86.

[Toh99] C.-K. Toh, "Long-lived Ad-Hoc Routing based on the concept of Associativity," March 1999 IETF Draft, 8 pages.

INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center
Ft. Belvoir, VA

2. Dudley Knox Library
Naval Postgraduate School
Monterey, CA

3. Marine Corps Representative
Naval Postgraduate School
Monterey, CA

4. Director, Training and Education
MCCDC, Code C46
Quantico, VA

5. Director, Marine Corps Research Center
MCCDC, Code C40RC
Quantico, VA

6. Marine Corps Tactical Systems Support Activity
(Attn: Operations Officer)
Camp Pendleton, CA

7. Maj Barry A. Dowdy
Marine Corps Systems Command (PMM122)
Quantico, VA

8. Professor Geoffrey Xie
Naval Postgraduate School
Monterey, CA

9. Professor John Gibson
Naval Postgraduate School
Monterey, CA

10. Professor Craig Martell
Naval Postgraduate School
Monterey, CA

11. Professor Richard Riehle
Naval Postgraduate School
Monterey, CA

12. Professor Frank Kragh
Naval Postgraduate School
Monterey, CA