CORE Metadata, citation and similar papers at core.ac.uk

İSTANBUL TECHNICAL UNIVERSITY ★ INSTITUTE OF SCIENCE AND TECHNOLOGY

1.1.

VIRTUAL CELL LAYOUT BASED DYNAMIC SOURCE ROUTING (VB-DSR) FOR THE MOBILE SUBSYSTEM OF THE NEXT GENERATION TACTICAL COMMUNICATIONS SYSTEMS

MS Thesis by

Müjdat SOYTÜRK

504990131011

Date of submission : 13 May 2002

Date of Defence Examination : 04 June 2002

Supervisor (Chairman) : Prof. Dr. A. Emre HARMANCI

Members of the Examining Committee: Prof. Dr. Bülent ÖRENCİK

Assoc. Prof. Dr. Cem ERSOY (B.U.)

MAY 2002

ACKNOWLEDGEMENT

First of all, I would like to thank to my supervisor Prof.Dr. A. Emre Harmancı and Dr. Erdal Çayırcı for their continuous support and guidance. I also appreciate them for their great patience and contributions to this thesis after long hours of studies.

This thesis is completed within the time, my life is changed completely. While continuing my MS education, I engaged and married with my wife, Gülden Soytürk, and now we are waiting for a baby. We spent our first marriage years during the completion of this thesis. I want to give my very special thanks to my wife for her great support and patience.

Finally, I am thankful to Turkish Navy for giving opportunity to educate in Istanbul Technical University.

May 2002 Müjdat Soytürk

TABLE OF CONTENTS

LIST OF ACRONYMS

ABR : Associativity Based Routing
ACK : Acknowledgement
AODV : Ad hoc On-Demand Distance Vector Routing Protocol
ATM : Asynchronous Transfer Mode
BS : Base Station
CAX : Computer Aided Exercise
CBR : Cluster Based Routing
CDMA : Code Division Multiple Access
CGSR : Cluster Gateway Switching Routing
CLR : Clear
CSMA : Carrier Sense Multiple Access
CSR : Clustered Spine Routing
DBF : Distributed Bellmann-Ford
DRP : Dynamic Routing Protocol
DSDV : Destination Sequenced Distance Vector Routing
DSR : Dynamic Source Routing
DV : Distance Vector
ECM : Electronic Counter Measures
FDMA : Frequency Division Multiple Access
FIFO : First In First Out
FP : Forwarding Protocol
FSR : Fisheye State Routing
GB : Gafni-Bertsekas Protocol
GPS : Global Positioning System
GSM : Global System for Mobile Communications
GSR : Global State Routing
HLR : Home Location Register
HSR : Hierarchical state Routing
ID : Identification
IERP : Inter-Zone Routing Protocol
JTLS : Joint Theater Level Simulation
LAN : Local Area Network
LCC : Least Cluster Change
LMS : Location Management Server
LRU : Least Recently Used
LS : Link State Protocol
MA : Multiple Access
MAC : Multiple Access Control
MMWN : Multimedia Support For Wireless Networks System
MPR : Man Packed Radio
MPRT : MPR Tier
MRL : Message Retransmission List
MS : Mobile Subsystem

MT : Mobile Terminal
PCS : Personal Communications Services
PR : Packet Radio
PRMA : Packet Reservation Multiple Access
PSN : Physical Subnet
PSR : Partial Knowledge Spine Routing
QoS : Quality of Services
QRY : Query
RAMA : Resource Auction Multiple Access
RAP : Radio Access Point
RAPT : RAP Tier
RD : Route Deletion
RERR : Route Error
RRC : Route Reconstruction
RREQ : Route Request
RREP : Route Reply
RT : Routing Table
SATT : Satellite Tier
SC : Strong Channel
SSA : Signal Stability Based Adaptive Routing
SST : Signal Stability Table
SYNC : Synchronization
TDMA : Time Division Multiple Access
TTL : Time-To-Live
TORA : Temporarily Ordered Routing Algorithm
UAV : Unmanned Aerial Vehicle
UAVT : UAV Tier
UMTS : Universal Mobile Telecommunications Systems
UPD : Update
UTRA : UMTS Terrestrial Radio Access
VC : Virtual Circuit
VCL : Virtual Cell Layout
VG : Virtual Gateway
VLR : Visitor Location Register
VSN : Virtual Subnet Routing
WAS : Wide Area Subsystem
WC : Weak Channel
WRP : Wireless Routing Protocol
ZRP : Zone Routing Protocol

LIST OF TABLES

Page Number

LIST OF FIGURES

Page Number

LIST OF SYMBOLS

E_b/N_o	: Bit energy to noise density ratio
r	: VCL cell radius
T_d	: Propagation delay
T_s	: Bit transmission time

ÖZET

GELECEK NESİL TAKTİK İLETİŞİM SİSTEMLERİNİN GEZGİN ALTSİSTEMİ İÇİN SANAL HÜCRE KALIBI TABANLI DİNAMİK KAYNAK YOLATAMA

Telsiz ağlar, 1970'lerde ortaya çıkışlarından itibaren bilgisayar endüstrisinde çok yaygınlaşmıştır. Telsiz ağlar gezgin kullanıcılara nerede olduklarına bağlı olmadan heryerde iletişim kurma ve bilgiye erişim imkanı sağlar. Hiçbir sabit altyapıya gerek duymadan bu imkanı sağlayan ad hoc ağların zaman içinde gelişmesiyle ve askeri, ticari ve özel maksatlar için tercih edilir hale gelmesiyle, uygulamada bazı zorluklar başgöstermiştir. Çalışmamızda, bu zorlukları ve ad hoc ağlardan beklenenleri dikkate alarak Sanal Hücre Kalıbı (SHK) tabanlı Dinamik Kaynak Yol Atama (DKYA) yaklaşımını bir çözüm olarak sunuyoruz. SHK, etkili bir özkaynak yönetimi ve ağ ölçeklenebilirliği sağlayan çok-katmanlı öbek-yapılı bir yaklaşımdır. Biz SHK yapısının üzerine Dinamik Kaynak Yolatama protokolunu uyguladık. Sunduğumuz yaklaşım, SHK yapılı DKYA, iki düzeyli hiyerarşik öbek yapılı bir sistemdir.

SHK yaklaşımında önerilen benzetim yaklaşımını geliştiriyoruz ve taktik iletişimlerin başarımlarının değerlendirmesinde bu benzetim yaklaşımını kullanıyoruz. Bu yöntemde, önceden gerçekleştirilen bilgisayar destekli askeri tatbikatlarda girilen emirleri, çok sayıdaki birliğe ait hareket, görev ve durum gibi verileri toplayan bir yapıcı (muharebe) model kullanarak tekrar işlemekteyiz. Daha sonra toplanan bu veriler daha da detaylandırılarak, başarı ölçütlerimize ait değerleri üreten benzetim kullanılmaktadır. Sistemi değerlendiren başarım ölçütleri, SHK-tabanlı DKYA'nın gezgin ad hoc ağlardan beklenenleri sağladığını göstermektedir.

SUMMARY

VIRTUAL CELL LAYOUT BASED DYNAMIC SOURCE ROUTING (VB-DSR) FOR THE MOBILE SUBSYSTEM OF THE NEXT GENERATION TACTICAL COMMUNICATIONS SYSTEMS SUMMARY

Wireless networks have become very popular in the computing industry after their emergence in the 1970's. Wireless networks provide mobile user with ubiquitous communication capability and information access regardless of location. Mobile ad hoc networks, that manage it without a need to infrastructure networks, as evolved in time, exhibit some challenges to implement as they become more preferable for military, commercial and special purposes. By considering the challenges and expectations of mobile ad hoc networks, we propose an approach called VCL based DSR (VB-DSR), which uses Dynamic Source Routing (DSR) protocol over Virtual Cell Layout (VCL) structure. VCL is a multi-tier cluster-based approach that provides an efficient resource management and network scalability. We implemented Dynamic Source Routing protocol over VCL structure. Proposed approach, VB-DSR, is a cluster-based two level hierarchical scheme.

We also enhance the simulation approach proposed by VCL structure, and use it for the evaluation of tactical communication systems. In this approach, the commands entered during the military computer aided exercises are replayed by running a constructive (combat) model which generates mobility, posture and status data for a number of units, then these data are enhanced and drive a simulation which produces the data related to the performance metrics. The evaluated performance of the system shows that VB-DSR approach satisfies the requirements of mobile ad hoc networks.

2. INTRODUCTION

If we define the principal reasons for implementing wireless communications systems, they include support for terminal mobility, and more rapid, widespread access to communications services, without any expensive requirements like in wired systems. Briefly, wireless systems provide mobility, flexibility and cost savings. There are two variants of mobile wireless networks; infrastructured networks, known as cellular networks, and infrastructureless networks, known as ad hoc networks.

In cellular networks, mobile terminals are access to a fixed infrastructure through a single hop wireless link to an access point. The advantage of the infrastructured wireless networks is that existing wired networks can be leveraged to support access for mobile users without modifications to the network's control structure. The disadvantage is that it requires a fixed infrastructure—constraining node mobility, limiting network deployability, and increasing installation and management costs. Infrastructured wireless networks are not well suited for rapid network deployment, temporary networking for mobile devices, or for environments where it is difficult to achieve adequate base station (BS) coverage, or the installation of fixed infrastructure is not feasible. To address these shortcomings, ad hoc networks emerged. In an area where there is no or little communication infrastructure or accessing to the existing infrastructure is ineffective or impossible, wireless mobile users may still be able to maintain the communication through the ad hoc networks [1].

2.1. Contribution of the Thesis

In our thesis, we lay out the challenges of wireless mobile ad hoc networks and made a survey of proposed routing protocols and schemes for it. We described the structure and behavior of the proposed algorithms and schemes, and emphasized their properties, advantages and drawbacks. As each proposed scheme presents a new valuable approach, they suffer from specific drawbacks. Some of these drawbacks are common to all these proposed schemes [1]. We take into the consideration of the advantages and drawbacks of the proposed schemes and try to present an acceptable solution for the mobile subsystem of the next generation tactical communications systems.

We propose an approach named VCL based Dynamic Source Routing (VB-DSR) that uses Dynamic Source Routing protocol [2] with VCL [3, 4] that enables the management of scarce resources efficiently in a mobile environment with a mobile infrastructure.

In VB-DSR, we benefit the advantages of VCL structure. We implement DSR as the routing protocol, which does not rely on periodic message dissemination, hence reduces the bandwidth overload and power consumption. Mobile terminals manage calls via the radio access points if they have access to them, else they manage calls in an ad hoc manner.

We also enhance the simulator proposed in [5] by implementing the dynamic source routing protocol and CSMA/CA access scheme for ad hoc nodes. We use the approach presented in [5] to evaluate the performance of the tactical communication systems. In this approach, the commands entered during a Computer Aided Exercise (CAX) are recorded. These recorded commands are replayed, and the results of these commands are translated into a database, which stores some mobility and posture information about a number of units. The translated database is used to drive a simulation software which enhances the resolution of the information produced by the constructive models, generates the calls and events, and collects the data related to the predetermined performance metrics for the proposed communication system. A final module analyzes the data collected for the performance metrics.

We evaluate the performance of VB-DSR by this enhanced simulation system. The evaluated performance of the system shows that the VB-DSR has low routing overhead, provides very low energy consumption, and has low route construction delay than other proposed schemes. Constructed routes are short and stable. VB-DSR is scalable for large sized networks by the aid of VCL structure.

2.2. Structure of the Thesis

In Chapter 2, the properties and the structure of mobile ad hoc networks, and the challenges it posed are described. We present a review of proposed routing schemes for mobile ad hoc networks in the rest of this section. The review is organized according to the classification of the proposed routing schemes. We identify major contributions and the drawbacks of these routing schemes.

We propose an approach in Chapter 3 to overcome seen shortcomings of proposed schemes, by using the identified properties and drawbacks of the proposed schemes in Chapter 2. The parameters of the proposed approach, the structure and design criteria used to implement, is presented in this chapter.

We define the proposed system architecture in Chapter 4. This chapter includes VB-DSR approach and the algorithms and parameters used.

The performance of the proposed approach is presented in the Chapter 5.

We conclude the proposed approach in the last chapter.

3. MOBILE AD HOC NETWORKS

3.1. Properties of Ad Hoc Networks

Ad-hoc networks are self-organizing, rapidly deployable, and require no fixed infrastructure. They are comprised of wireless nodes, which can be deployed anywhere, and must cooperate in order to dynamically establish communications using limited network management. Nodes in an ad-hoc network may be highly mobile, or stationary, and may vary widely in terms of their capabilities and uses[1-3, 6-12].

The primary objectives of ad hoc network architecture are to achieve increased flexibility, mobility and ease of management relative to infrastructured wireless networks. This is achieved by eliminating the need for fixed BSs and routers; thereby, enabling instant infrastructure wherever ad-hoc nodes are activated, and eliminating many of the constraints to node mobility. An ad-hoc network is itself mobile because the network moves anywhere the nodes locate themselves. In effect, the end nodes themselves must act as mobile routers and BSs. Hence, an ad-hoc network is a dynamic entity, which requires adaptive control algorithms in order to be responsive to node mobility, and to operate with minimal administrative intervention. Because of these properties, ad hoc networks can be used in many different areas and for many different purposes [1, 12 - 13]:

- Military (tactical) communication – for fast establishment of communication infrastructure during deployment of forces in a foreign (hostile) terrain.
- Rescue missions – for communication in areas without adequate wireless coverage.
- National security – for communication in times of national crisis when the existing communication infrastructure is non-operational due to a natural disaster or global war.
- Law enforcement – similar to tactical communications.

- Commercial use – for setting up communication in exhibitions, conferences, or sale presentations.
- Education – for operation of virtual classrooms.
- Sensor networks – for communication between intelligent sensors mounted on mobile platforms.

The fundamental property, which distinguishes ad-hoc networks from other wireless architectures, is that node mobility causes the network topology to be continuously reconfigurable. In a wireless ad-hoc network environment, transmission range is limited and variable due to numerous system and environmental factors, including transmission power, receiver sensitivity, noise and other channel effects, namely, path-loss, shadow fading, Raleigh fading, Doppler shift, and interference. Node mobility may exacerbate several of these capacity-limiting effects. Furthermore, signal range may be limited by design in order to increase system throughput by minimizing channel access contention, and to increase battery lifetime by minimizing transmission power. In general, a node's transmission range is neither fixed, nor symmetric; it demonstrates temporal and spatial variability. Consequently, the links of an ad-hoc network are not fixed entities—their status changes over time and is dependent on the relative spatial location of the nodes, transmitter and receiver characteristics, and the signal propagation properties of the environment. Wireless channel effects and their impact on link status is not unique to ad-hoc networks, although the effects may be more pronounced when both ends of a wireless link are mobile. However, the crucial difference is that all the links in an ad-hoc network are wireless, potentially with rapidly moving end-points. Hence, there is no fixed infrastructure in an ad-hoc network. Consequently, the links not only represent wireless end-points, as in infrastructured wireless networks, they represent the network topology itself. Thus, as nodes move freely and independently, the topology of an ad-hoc network changes dynamically and arbitrarily.

The lack of a fixed network and the mobility of the nodes lead to two important features of ad hoc networks, namely, multi-hop packet routing and mobile (end-system) routers. Unlike infrastructured networks, ad-hoc networks cannot rely on dedicated BSs and routers to forward traffic across fixed network segments between mobile users. Furthermore, direct communications between all nodes is infeasible due to limited transmission range and node mobility. Consequently, store-and-forward packet routing is required over multiple-hop wireless paths. Therefore, the mobile nodes themselves must cooperate in order to maintain routes dynamically and forward traffic on behalf of other nodes—the mobile nodes themselves must be routers. In order to maintain communications subject to router mobility and the subsequent dynamic status of the wireless network links, the routers must implement adaptive algorithms that are responsive to dynamics in the network topology, without over-utilization of network resources.

If we summarize all of these properties, they are [1, 12, 14]:

1. There is no centralized authority for network control, routing or administration.
2. Network devices, including user terminals, routers, and other potential service platforms are free to move rapidly and arbitrarily in time and space.
3. All communication, user data and control information, is carried over the wireless medium. There are no wired communication links.
4. Resources, including energy, bandwidth, processing capacity and memory, that are relatively abundant in wired environments, are strictly limited and must be preserved.
5. Mobile nodes that are end-points for user communications and process user applications must act cooperatively to handle network functions, mostly notably routing, without specialized routers.

3.2. Structure of Ad Hoc Networks

Properties of ad hoc networks described above. These properties also cause the challenges for implementing a successful wireless ad hoc networking technology. Most of these challenges are the same of the infrastructured networks, but some of them are unique to ad hoc networks. The challenges stemming from the properties described above affect every aspect of system design and performance: from issues related to physical and MAC-layer design, to network-layer issues including routing, addressing and mobility management, and where real-time communications are required, connection admissions control and real-time resource management, and finally application-layer issues. Therefore, firstly, we must survey the structure of ad hoc networks.

3.2.1. Physical-layer

Because of the highly re-configurable nature of ad hoc networks, the temporal and spatial variability of link quality, the mobility of the terminals, and the limited power, it is necessary to understand the basic challenges relevant to physical-layer components of an ad hoc network.

In particular, it is crucial to understand that there is a fundamental tradeoff, which couples the physical-layer, MAC-layer, and routing algorithm design and performance. Specifically, the range of the radio transceivers is chosen as a tradeoff between network connectivity, the reuse of available spectrum, and the power consumption [1, 6, 12]. This tradeoff, which has a direct impact on channel contention, routing and battery lifetime, can be stated as follows:

Power versus Bandwidth Tradeoff: In a wireless ad-hoc network, signals can be transmitted at lower power in order to reduce channel contention and conserve energy. Reducing channel contention can increase system utilization and mobile terminal battery lifetime. However, signal range will be reduced and channel effects such as path-loss may be increased. Therefore, nodes, which could have communicated directly over a single-hop, are forced to communicate over multi-hop wireless paths. Hence, average path-length will increase, and, for a given mobility pattern and environment, the link failure rate will increase. Consequently, more power and bandwidth will be consumed forwarding data packets and control information. This in turn may reduce system throughput and possibly hasten battery failure. Therefore, a critical transmission range must be defined, which is the minimum range required to maintain connectivity [1].

In a different manner from the infrastructured systems, mobility and environmental factors such as physical obstructions make the challenges more difficult. Wireless channel effects represent the first and most fundamental challenge to ad-hoc network system design. The effects of signal propagation impose a floor on the quality, the information carrying capacity, the stability, and the signal range of the wireless links. The precise impact depends upon specific system factors, including, antenna design, signal transmission power and receiver sensitivity, modulation and detection schemes, use of signal processing, transmission bandwidth and carrier frequencies, the presence and dielectric characteristics of physical obstructions, and node mobility. Each of these factors will affect, in some way, the ability of an ad-hoc node to accurately transmit and receive information. The challenge is to design system components that can operate effectively in a range of environments and subject to expected mobility patterns.

The second major physical-layer challenge in ad-hoc network system design relates to energy consumption and battery lifetime. These issues represent a critical design factor in any wireless system, but particularly so in an ad-hoc network where there are no fixed infrastructure components. Two approaches have been proposed to address these issues, namely, either to build a more powerful battery, or to use existing batteries more efficiently. As the technology evolve, more powerful batteries are developed. More crucial is utilizing the power more efficiently. To achieve this goal, communications protocols should attempt to minimize energy consumption by optimizing channel access and eliminating unnecessary transmissions. Furthermore, it becomes effective to devise energy-based routing metrics that can minimize the effects of routing on power consumption [6].

The final challenge is to acquire and utilize mobility or location information in order to adapt physical-layer parameters to the dynamically changing environment, or to provide information to upper layer entities for enhancing system performance. For example, a strategy to adapt the physical-layer attributes to mobility information would increase the transmission power as the speed of a mobile node increases. Hence, multi-level signal strength architecture could be envisioned that is similar to the multi-level schemes proposed for cellular networks. The effect is to reduce the rate of link failure, which might otherwise become excessive for rapidly moving nodes if signal transmission range is very short. Furthermore, mobility and location information could potentially be used by a routing algorithm to select more stable routes, or to improve the efficiency of the routing protocol.

3.2.2. MAC-layer

The basic challenge to overcome when dealing with a shared transmission medium is how to control access to the communications channel in a fair and efficient manner. Medium-Access-Control (MAC) provides this functionality. Because of scarce bandwidth in wireless networks, bandwidth must not be consumed wastefully. Therefore, to access to the transmission medium, an efficient multiple access (MA) technique must be used.

Multiple Access (MA) techniques can be classified into two; conflict-free techniques and contention-based techniques [7]. Conflict-free techniques are those ensuring that a transmission, whenever made a successful one, will not be interfered by another transmission. Conflict-free transmission can be achieved by allocating the channel to the users either statically or dynamically. In static conflict-free techniques, the channel is allocated to the users either on a frequency basis using Frequency Division MA (FDMA), on a time basis using Time Division MA (TDMA), or on a code basis using Code Division MA (CDMA). In static conflict-free techniques, users continue to use allocated resource without sharing with any other user [15].

Contrary to the static conflict-free techniques, the dynamic conflict-free technique allocates the channel temporarily to the users on demand basis. Hence, resources are used only by users those required to.

In contention-based techniques, the transmission is not guaranteed to be successful. The technique must prescribe a way to resolve conflicts once they occur; so all messages are eventually transmitted successfully. The resolution process does consume resources and is one of the major differences among the various contention methods. As in the conflict free case, here too, both static and dynamic resolution methods exist. Static resolution means that the actual behavior is not influenced by the dynamics of the system. To avoid contention to occur, priority values or probabilistic values are used. In dynamic resolution methods, contention is avoided to occur by monitoring and tracking the changes in the system.

Fixed allocation schemes are not appropriate for ad-hoc networks. However, dynamic channel allocation schemes can provide a user with probabilistic bounds on access delay and minimum bandwidth.

In transmission media, propagation effects vary randomly and transmission range is limited, variable and asymmetric. Hence, the transmission medium can be accessed by multiple users at the same time that causes an event called collision or interference. Interference is classified into two as primary interference and secondary interference.

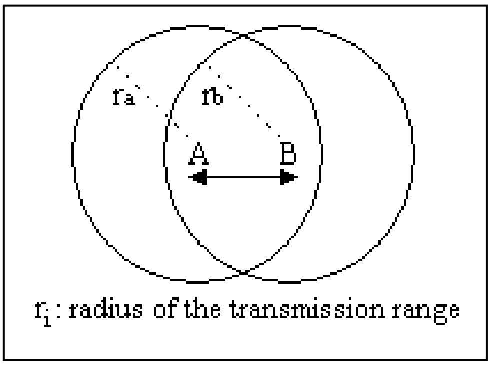

Figure 3.1 Primary interference - I.

Primary interference is said to occur when a transceiver is expected to perform more than one operation at the same time, such as receiving from two different transmitters at the same time or transmitting and receiving at the same time. There are two types of primary interference. To explain first type of primary interference, let two transceivers *a* and *b* be within the transmission range of each other (Figure 2.1). In this case, if *a* and *b* start transmission at the same time, then both transceivers will be expected to transmit and receive simultaneously.

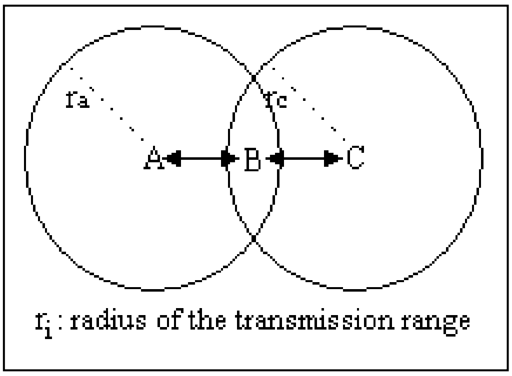

Figure 3.2 Primary interference - II (Hidden Terminal Problem).

We can explain second type of primary interference, also known as hidden terminal problem, with three transceivers as *a*, *b*, and *c* in Figure 2.2. The transceivers *a* and *c* are not within the range of one another, but third transceiver *b* is within the transmission range of both *a* and *c*. If *a* and *c* start simultaneous transmissions, then *b* will be expected to receive from both *a* and *c* at the same time.

Secondary interference occurs when a transmission from a neighboring transmitter unwillingly interferes at the receiving end of a communication between a transmitter and receiver.

We can explain secondary interference with an additional transceiver *d* that is in transmission range of *c* but not *a* (Figure 2.3). Let *a* is in communication with *b*, and *c* needed to communicate with *d*. Since *b* is within transmission range of both *a* and *c*, the transmissions of *c* is interfered with the transmission of *a*, although *c* does not intend to transmit any packet to *b*.

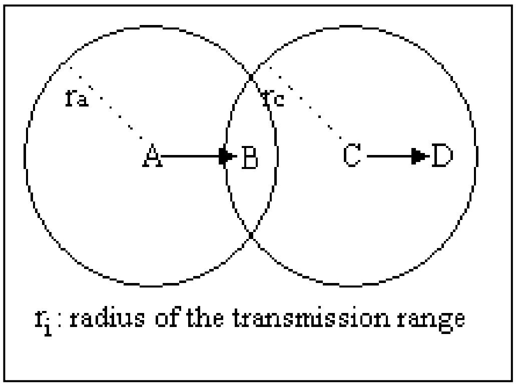

Figure 3.3 Secondary interference.

3.2.3. Network layer: The routing algorithm

An ad-hoc network is a cooperative engagement of mobile hosts. These hosts use wireless communications with constraints on signal transmission range, bandwidth and power. Peer-to-peer communications must be supported between arbitrary hosts without the need to involve specialized routers or requiring direct single-hop communications. Consequently, the mobile hosts must cooperate in order to establish and maintain routes between arbitrary end-points. All the links are expected to be wireless, and any intermediate node or end-point is free to move arbitrarily in time and space. Consequently, routes must be adapted rapidly to node movement and variability in link quality without over utilizing network resources. Therefore, the wireless ad-hoc network routing problem presents a very difficult challenge that can be posed as a classic tradeoff between responsiveness and efficiency. This tradeoff must balance the need to rapidly adapt the network to node mobility and changes in link quality, against the overhead associated with responding to frequent topology changes. In a wireless ad-hoc network overhead is primarily measured in terms of bandwidth utilization, power consumption and the processing requirements on the mobile nodes. Finding a strategy for efficiently balancing these competing needs forms the basis of the routing challenge [1, 12].

3.3. Structures of Routing Algorithms

The preliminary proposed routing protocols for mobile networks were adaptations of routing protocols those used in fixed networks. These protocols used in fixed networks outperform worse results in dynamic networks since they were designed for fixed networks. As the network becomes more dynamic, the overload of routing processes increases. Network resources can be exhausted rapidly or may become unusable, if this overload uncontrolled.

As these techniques become insufficient, new techniques proposed for mobile networks. It will be useful to survey the structure of general routing protocols before examining them. Routing techniques can be classified as flat routing and hierarchical routing according to their structure .

3.3.1. Flat routing techniques

In flat-routed algorithms, each node maintains routing information to some or all of nodes in the network in one more tables. Table size can be acceptable in a small sized network, but as the network size becomes greater, the routing tables maintained becomes greater. Communication load and process time for routing increases significantly. Table updating processes and table processing time cause overhead in the network. Therefore, flat routing algorithms are not scalable for large networks, and have poor performance. There are many proposed flat routed protocols in the literature [1, 10, 12, 16].

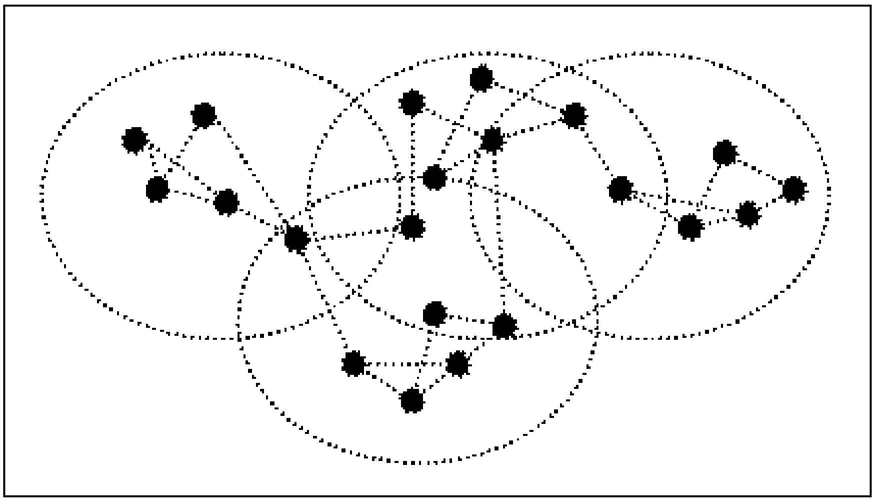

Figure 3.4 A flat ad hoc network.

3.3.2. Hierarchical routing techniques

In large networks, hierarchical routing techniques are used for scalability. The main advantage of hierarchical routing is that it minimizes the routing table size, hence decreases the routing process time significantly.

A network is consisting of end-point nodes and switches. Switches manage routing function. Communicating entities are the end-point nodes, and each end-point node in case, acts as a switch and manages routing process for its neighbors.

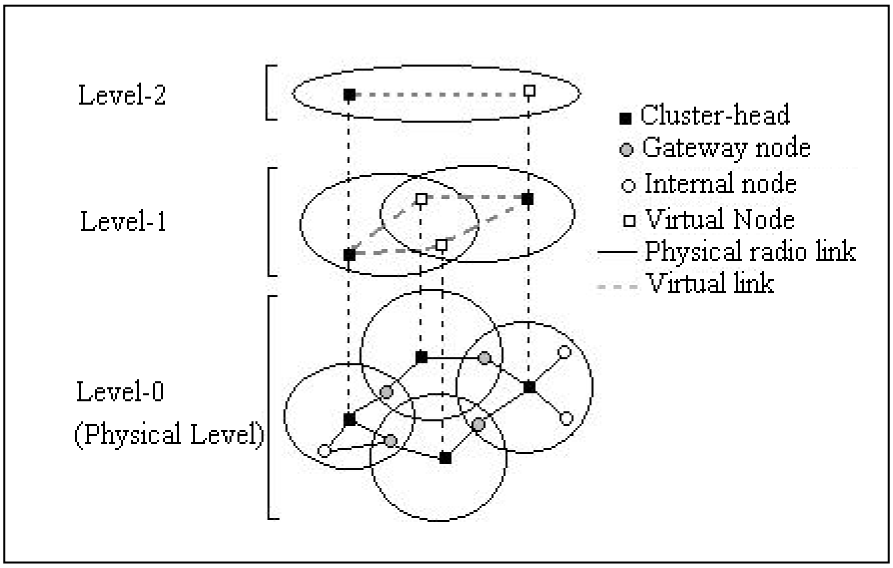

Figure 3.5 A hierarchical ad hoc network.

In a hierarchical network, the lowest level consists of end-point nodes (Figure 2.5). Neighboring end-points organize into clusters and at each cluster a node is selected as cluster-head. Cluster-heads act as switches. Cluster-heads also organize into clusters to make the upper level and at each cluster; a node is selected as cluster-head for that level. And so on for a complete hierarchical structure [1, 10, 12, 16-18].

At each level, a node in a cluster only maintains the routing information of the members of that cluster. For the nodes in different clusters, routes are established via cluster-heads, being also a member of upper level cluster. Therefore routing table size and routing process load decrease. But the found route may not be optimal route.

3.4. Routing Algorithms for Mobile Ad Hoc Networks

Traditional routing algorithms tend to exhibit their least desirable behavior under highly dynamic conditions [19]. Routing protocol overhead typically increases dramatically with increased network dynamics. If the protocol overhead is uncontrolled, it can easily overwhelm network resources. Furthermore, traditional routing protocols require substantial inter-nodal coordination or global flooding in order to maintain consistent routing information and avoid routing table loops. These techniques increase routing protocol overhead and convergence times. Consequently, although they are well adapted to operate in environments where bandwidth is plentiful and the network links are relatively stable, the efficiency of these techniques conflict with routing requirements in ad-hoc networks. It, therefore, appears that new routing strategies are required for ad-hoc networks that are capable of effectively managing the tradeoff between responsiveness and efficiency. The following definitions present the most commonly used means of classifying routing protocols that have been designed for ad-hoc networks [1, 12]:

Proactive Routing is defined as a strategy in which routes are continuously maintained for all reachable network destinations. This approach requires periodic dissemination of routing updates to reflect the up-to-date state of the network.

Reactive Routing is defined as a strategy in which routes are established and maintained on a demand basis—only if they are needed for communications. This approach requires procedures to acquire new routes and to maintain routes following topology changes.

Hybrid Routing is defined as a strategy, which selectively applies either proactive or reactive routing techniques, based upon either predefined or adaptive criteria.

3.4.1. Proactive routing algorithms

Proactive routing protocols periodically distribute routing information throughout the network in order to pre-compute paths to all possible destinations. Hence, each node maintains a priory calculated routing information to all destinations, regardless as to whether or not a given node actually needs to reach each such destination, or lies along a path of another node that does. All nodes update these tables to maintain a consistent and up-to-date view of the network. When the network topology changes, the nodes propagate update messages throughout the network in order to maintain consistent and up-to-date routing information about the whole network. These routing protocols differ in the method by which the topology change information is distributed across the network and the number of necessary routing-related tables. Although this approach can ensure higher quality routes in a static topology, it does not scale well to large, highly dynamic networks. This routing strategy is also referred to as table-driven routing, because protocols that adopt this strategy attempt to maintain consistent information in the routing tables distributed throughout the network [1,7-9, 12, 19-22].

3.4.1.1. Destination-sequenced distance-vector routing protocol

The Destination-Sequenced Distance-Vector (DSDV) Routing Algorithm [19] is the earliest published work that directly addresses the routing problem of ad hoc networks. It is a benchmark for comparison as improved strategies evolved. The DSDV protocol is an adaptation of traditional Distance-Vector (DV) algorithm, with modifications intended to improve its efficiency and convergence characteristics for the mobile environment. DSDV is based on Bellman-Ford shortest path routing algorithm with certain improvements.

Every mobile station maintains a routing table that lists all available destinations, the number of hops to reach the destination and the sequence number assigned by the destination node. The sequence number is used to distinguish old routes from new ones and thus avoid the formation of loops. The stations periodically transmit their routing tables to their immediate neighbors. A station also transmits its routing table if a significant change has occurred in its table from the last update sent. Therefore, the update is both time-driven and event-driven. The routing table updates can be sent in two ways: a "full dump" or an incremental update. A full dump sends the full routing table to the neighbors and could span many packets, whereas in an incremental update only those entries from the routing table are sent that produce a metric change since the last update and it must fit in a packet. If there is space in the incremental update packets then those entries may be included whose sequence number has changed. When the network is relatively stable, incremental updates are sent to avoid extra traffic and full dumps are relatively infrequent.

In a fast-changing network, incremental packets can grow in size so that full dumps will be more frequent. Each route update packet, in addition to the routing table information, also contains a unique sequence number assigned by the transmitter. The route labeled with the highest (i.e., most recent) sequence number is used. If two routes have the same sequence number, then the route with the best metric (i.e. shortest route) is used. Based on the history, the stations estimate the creation time of routes. The stations delay the transmission of a routing update by creation time so as to eliminate those updates that would occur if a better route were found very soon..

DSDV prevents loops that is in DBF, but it is still relies on periodic routing updates that involve every node in the network. To prevent looping and improve stability, mechanisms were added that impose substantial control and coordination among the nodes. This increases protocol overhead and slows convergence times. Consequently, DSDV cannot support very high rates of node mobility and is not sufficiently scalable. Another shortcoming of DSDV is that it requires bi-directional links. Unidirectional links, which are expected to be common in wireless ad-hoc networks, cannot be used for routing. This may significantly reduce the network's effective connectivity—increasing average path-length and partitioning the network when a destination is reachable only over a directed path.

3.4.1.2. The wireless routing protocol (WRP)

The Wireless Routing Protocol (WRP) described in [20] is a table-based protocol with the goal of maintaining routing information among all nodes in the network. is a table-based distance-vector routing protocol. Each node in the network maintains a Distance table, a Routing table, a Link-Cost table and a Message Retransmission list.

The Distance table of a node x contains the distance of each destination node y via each neighbor z of x. It also contains the downstream neighbor of z through which this path is realized. The Routing table of node x contains the distance of each destination node y from node x, the predecessor and the successor of node x on this path. It also contains a tag to identify if the entry is a simple path, a loop, or invalid. Storing predecessor and successor in the table is beneficial in detecting loops and avoiding counting-to-infinity problems. The Link-Cost table contains cost of link to each neighbor of the node and the number of timeouts since an error-free message was received from that neighbor. The Message Retransmission List (MRL) contains information to let a node know which of its neighbor has not acknowledged its update message.

Nodes exchange routing tables with their neighbors using update messages periodically as well as on link changes. The nodes present on the response list of update message (formed using MRL) are required to acknowledge the receipt of update message. If there is no change in the routing table since the last update, then the node is required to send an idle "Hello" message to ensure connectivity. On receiving an update message, the node modifies its distance table and looks for better paths using new information. Any new path so found is relayed back to the original nodes so that they can update their tables. The node also updates its routing table if the new path is better than the existing path. On receiving an ACK, the mode updates its MRL. A unique feature of this algorithm is that it checks the consistency of all its neighbors every time it detects a change in link of any of its neighbors. Consistency checking in this manner helps eliminate looping situations in a better way and also has fast convergence.

WRP falls short in that it still requires significant periodic information exchange. The volume of routing overhead required to maintain shortest-path trees to all destinations will be substantial when nodes become highly mobile or the network becomes large. Consequently, protocol scalability and rapid adaptation in highly dynamic environments is unlikely. Furthermore, because the algorithm maintains shortest-path trees, every node that uses a failed in its tree will be involved in reaction to the link failure. Consequently, the effects of node mobility are typically far reaching, and cannot be bounded.

3.4.1.3. Global state routing

Global State Routing (GSR) [21] is similar to DSDV described in Section 2.4.1.1. GSR is modeled to utilize the routing accuracy and fast convergence of Link State scheme but at the same time avoiding flooding of routing messages. Because mobile ad hoc environment has limited bandwidth, the dissemination method used in DBF is adopted, which has the advantage of no flooding.

In GSR (like in LS) link states are not propagated, a full topology map is kept at each node, and shortest paths are computed using this map. In this algorithm, each node maintains a Neighbor list, a Topology table, a Next Hop table and a Distance table. The Neighbor list of a node contains the list of its neighbors; here all nodes that can be heard by a node are assumed to be its neighbors. For each destination node, the Topology table contains the link-state information as reported by the destination and the timestamp of the information. For each destination, the Next Hop table contains the next hop to which the packets for this destination must be forwarded. The Distance table contains the shortest distance to each destination node.

The routing messages are generated on a link change as in link-state protocols. On receiving a routing message, the node updates its Topology table if the sequence number of the message is newer than the sequence number stored in the table. After this the node reconstructs its routing table and broadcasts the information to its neighbors.

The drawbacks of GSR are the large size update message, which consumes considerable amount of bandwidth and the latency of the link state propagation, which depends on the update period. It takes the idea of link-state routing but improves it by avoiding flooding of routing messages.

3.4.1.4. Fisheye state routing

Fisheye State Routing (FSR) Algorithm [1] is based on GSR. In mobile Ad hoc network the bandwidth available is limited and large size update messages like in GSR consumes a considerable amount of bandwidth. The fisheye technique reduces the size of update messages without seriously affecting routing accuracy. The eye of the fish captures with high detail the pixels near the focal point. The detail decreases as the distance from the local point increases. In routing, it translates to maintaining accurate distance and path quality information about the intermediate neighborhood of a node, with progressively less detail as the distance increases.

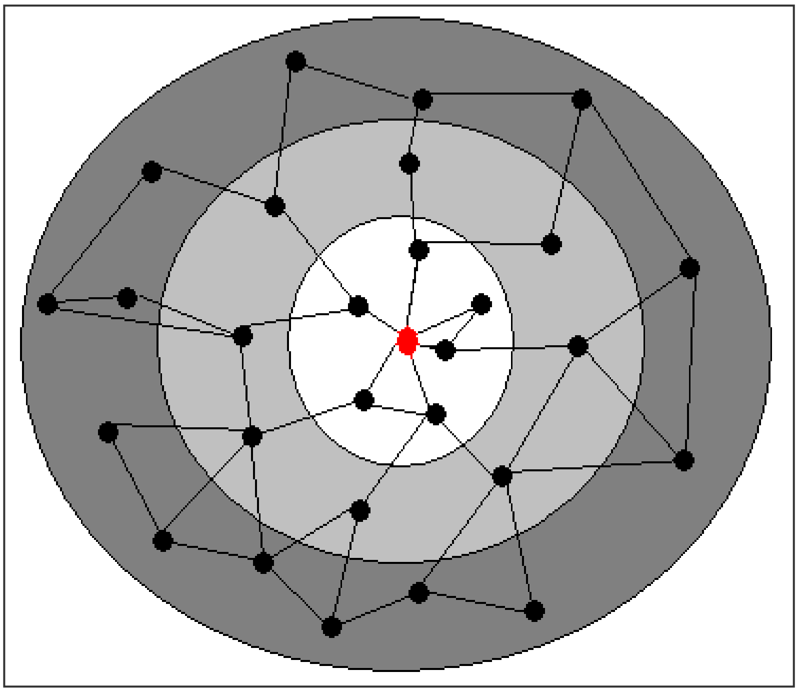

Figure 3.6 Accuracy of information in FSR.

In FSR, each update message does not contain information about all nodes. Instead, it exchanges information about closer nodes more frequently than it does about farther nodes, thus reducing the update message size. Therefore, each node gets accurate information about its neighbors. The detail and accuracy of information decreases as the distance from the node increases. Figure 2.6 defines the scope of fisheye for the center node. The scope is defined in terms of the nodes that can be reached in a certain number of hops. The center node has the most accurate information about all nodes in the white circle and so on. Even though a node does not have accurate information about distant nodes, the packets are routed correctly because the route information becomes increasingly accurate as the packet moves closer to the destination.

This strategy produces timely updates from near stations, but create large latencies from stations that are afar. Even though the nodes do not have accurate information about the distant nodes, the packet routed correctly because when a packet approaches its destination, it finds increasingly accurate routing instructions as it enters sectors with a higher refresh rate.

It provides lower latency for access to frequently used destinations. It has a lower control traffic overhead in dense traffic situations. Though compared to other flat table driven schemes (such as DSDV, GSR), it scales better but it still has scalability limitations due to flat addressing scheme.

3.4.1.5. Hierarchical state routing

In a mobile network, hierarchical routing has some drawback in mobility and location management. Hierarchical State Routing (HSR) [7, 8] was developed to overcome these problems that combines dynamic, distributed multi-level hierarchical clustering with an efficient location management. HSR maintains a hierarchical topology. The network is partitioned into clusters and a cluster-head elected as in a cluster-based algorithm. The cluster-heads again organize themselves into clusters and so on. The goals of clustering are the efficient utilization of radio channel resources and the reduction of network-layer routing overhead (i.e., routing table storage, processing and transmission). The nodes of a physical cluster broadcast their link information to each other. The cluster-head summarizes its cluster's information and sends it to neighboring cluster-heads via a gateway.

As shown in Figure 2.7, these cluster-heads are members of the cluster at the next higher level and they exchange their link information as well as the summarized lower-level information among each other and so on (e.g., ATM PNNI). A node at each level floods to its lower level the information that it obtains after the algorithm has run at that level. Therefore, the lower level has hierarchical topology information. Each node has a hierarchical address. One way to assign a hierarchical address is the cluster numbers on the way from root to the node as shown in Figure 4. A gateway can be reached from the root via more than one path, so the gateway can have more than one hierarchical address. A hierarchical address is enough to ensure delivery from anywhere in the network to the host.

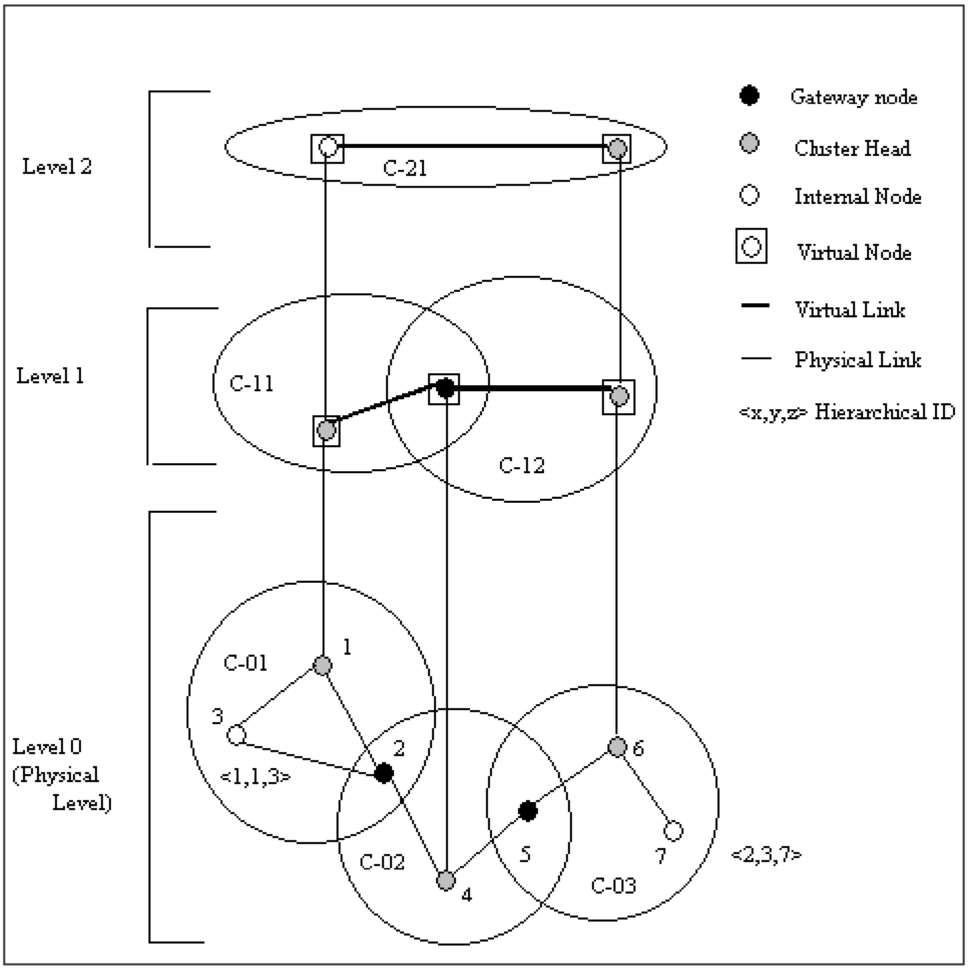

Figure 3.7 An example of clustering in HSR.

In addition to multilevel clustering, HSR also provides multilevel logical partitioning. Nodes are partitioned into logical subnetworks. While clustering is based on geographical (i.e. physical) relationship between nodes, (hence it will be referred to as physical clustering), logical partitioning is based logical, functional affinity between nodes (e.g. employees of the same company, members of the same family, etc). Each node is assigned a logical address <subnet, host>. Each subnetwork has a location management server (LMS). All the nodes of that subnet register their logical address with the LMS. The LMS advertise their hierarchical address to the top levels and the information is sent down to all LMS too. The transport layer sends a packet to the network layer with the logical address of the destination. The network layer finds the hierarchical address of the hierarchical address of the destination LMS from its LMS and then sends the packet to it. The destination LMS forwards the packet to the destination. Once the source and destination know each other's hierarchical addresses, they can bypass the LMS and communicate directly. Since logical address/hierarchical address is used for routing, it is adaptable to network changes.

Logical partitions play a key role in location management. HSR location management scheme tracks the mobile nodes, while keeping the control message overhead low.

The drawbacks of HSR with respect to flat link state routing are the need to maintain longer hierarchical addresses and the cost of continuously updating the cluster hierarchy and the hierarchical addresses as nodes move.

3.4.1.6. Clusterhead gateway switch routing protocol

Clusterhead Gateway Switch Routing (CGSR) protocol [22] differs from the other protocols in the type of addressing and network organization scheme employed. Instead of a flat network, CGSR is a clustered multihop wireless network with several heuristic routing schemes. According to the protocol, by having a cluster head controlling a group of ad hoc nodes, a framework for code separation (among clusters), channel access, routing and bandwidth allocation can be achieved. The basic idea is that a packet will be routed alternatively between cluster heads and the gateway until finally it reaches the destination cluster head which then forwards it to the actual destination which is within its cluster.

The mobile nodes are aggregated into clusters and a clusterhead is elected. All nodes that are in the communication range of the clusterhead belong to its cluster. A gateway node is a node that is in the communication range of two or more clusterheads. A dynamic network clusterhead scheme can cause performance degradation due to frequent clusterhead elections, so CGSR uses a Least Cluster

Change (LCC) algorithm. In LCC, a clusterhead change occurs only if a change in network causes two clusterheads to come into one cluster or one of the nodes moves out of the range of all the clusterheads.

The general algorithm works in the following manner. The source of the packet transmits the packet to its clusterhead. From this clusterhead, the packet is sent to the gateway node that connects this cluster-head and the next clusterhead along the route to the destination. The gateway sends the packet to that clusterhead and so on until the destination clusterhead is reached in this way. The destination clusterhead then transmits the packet to the destination. Figure 2.8 shows an example of the CGSR routing scheme.

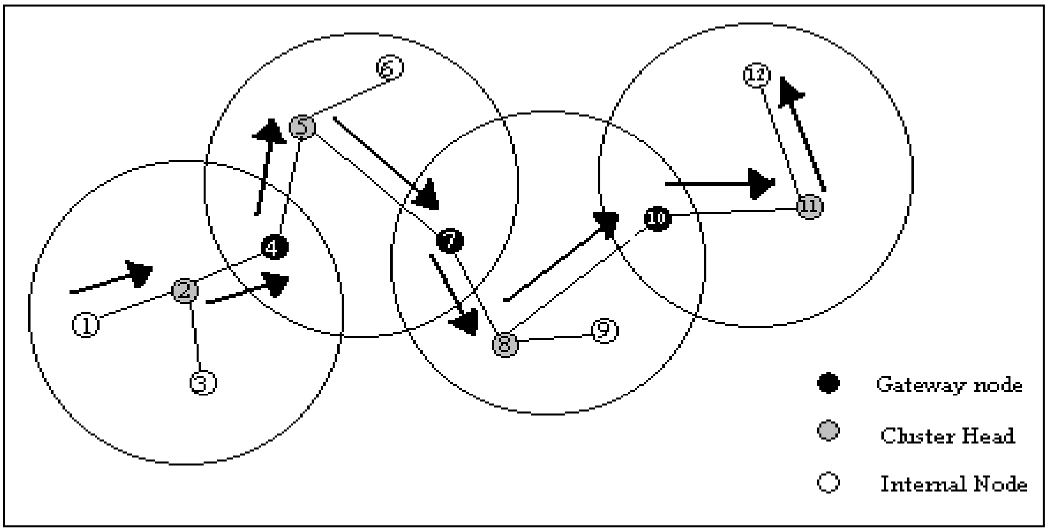

Figure 3.8 Example of CGSR routing from node 1 to node 12 [14].

Each node maintains a cluster member table that maintains a mapping from each node to its respective cluster-head. Each node broadcasts its cluster member table periodically and updates its table after receiving other node broadcasts using the DSDV algorithm. In addition, each node also maintains a routing table that determines the next hop to reach the destination cluster.

On receiving a packet, a node finds the nearest cluster-head along the route to the destination according to the cluster member table and the routing table. Then it consults its routing table to find the next hop in order to reach the cluster-head selected in step one and transmits the packet to that node.

The disadvantage of having a cluster head scheme is that some nodes, such as cluster heads and gateway nodes have higher computation and communication burden than other ones. The network reliability may also be affected due to single points of failure of these critical nodes.

3.4.2. Reactive routing protocols

Reactive routing has been proposed as a means of achieving a better balance between responsiveness and efficiency. The objective of reactive routing is to minimize the reaction of the routing algorithm to topology changes by maintaining a limited set of routes—those required for on-going communications. The idea is that by selectively limiting the set of destinations to which routes are maintained, less routing information needs to be routinely exchanged and processed. Consequently, less bandwidth is consumed by routing information, less computation is required to process routing information, and less memory is consumed by routing tables. Based on this technique, routing is expected to response more rapidly to topology changes, and additional network resources are expected to be available for the transmission and processing of application data.

In a reactive routing strategy, paths are maintained on a demand-basis using a query-response process. This involves a variation of controlled flooding referred to as a directed broadcast, in which a query, or route request packet is selectively forwarded along multiple paths toward a target destination. The search process dynamically constructs one, or multiple paths from the source node to the destination. This strategy limits the total number of destinations to which routing information must be maintained, and, consequently, the volume of control traffic required to achieve routing. The shortcomings of this approach include the possibility of significant delay at route setup time, the large volume of far reaching control traffic required to support the route query mechanism, and reduce path quality. Furthermore, despite the objective of maintaining only desired routes, the route query could propagate to every node in a network during the initial path setup causing each node to establish paths even when they are only required by certain sources. Finally, most reactive strategies do not discover optimal paths, and the paths typically become increasingly less optimal following each topology change [1-2, 10, 12, 23-28].

3.4.2.1. Gafni-Bertsekas (GB) protocol

The earliest fully reactive routing strategy was proposed for PR networks by Gafni-Bertsekas (GB) in 1981. The objective of GB is for PR nodes to maintain, on-demand, one or more loop-free routes to a desired destination after arbitrary link and/or node failures. No attempt is made to optimize the routes. The aim is simply to maintain connectivity in frequently changing topologies. The original GB paper remains one of the seminal papers in the field due to three very important contributions. Specifically, they were the first to:

1. Propose demand-based techniques to help minimize algorithm reaction to node mobility.
2. Advocate multi-path routing, rather than shortest path routing to achieve faster convergence and longer-lived *route*s. Hence, further reducing average algorithm reaction to node mobility.
3. Devised a novel distributed algorithm based on a total ordering strategy, which localizes reactions to link failures to those node materially affected by the failures.

Each of these ideas is important contribution, which have helped to shape the evolution of ad-hoc network routing.

Despite its advantages, the GB scheme has several shortcomings. Foremost among them are the possibility of instability during disconnected operation. In a very mobile environment, disconnection can occur very frequently. Furthermore, many nodes may be temporary visitors to a given wireless network. The GB protocol has no means of terminating the search for a new path when a network partition occurs or when a destination permanently leaves the network. The resulting instability affects all the nodes in the partition that have lost their routes. This can be viewed as a variation of the counting to infinity problem in DVA.

3.4.2.2. Dynamic source routing protocol

The Dynamic Source Routing protocol (DSR) [2, 23-24] is a pure on-demand routing protocol, which creates on-demand paths using a route query process based on directed broadcast of a route request packet. Once a route has been acquired by a source, the source caches that route locally until either it is informed that the route is no longer valid due to mobility along the path, or the node no longer requires the route and an inactivity timer has expired. DSR uses source routing to avoid the need for intermediate nodes to maintain up-to-routing information. Once a route has been discovered, there is no requirement for intermediate node routing tables. Routing information is contained entirely in the source routing header as a sequence of nodes over which the packet is to be forwarded.

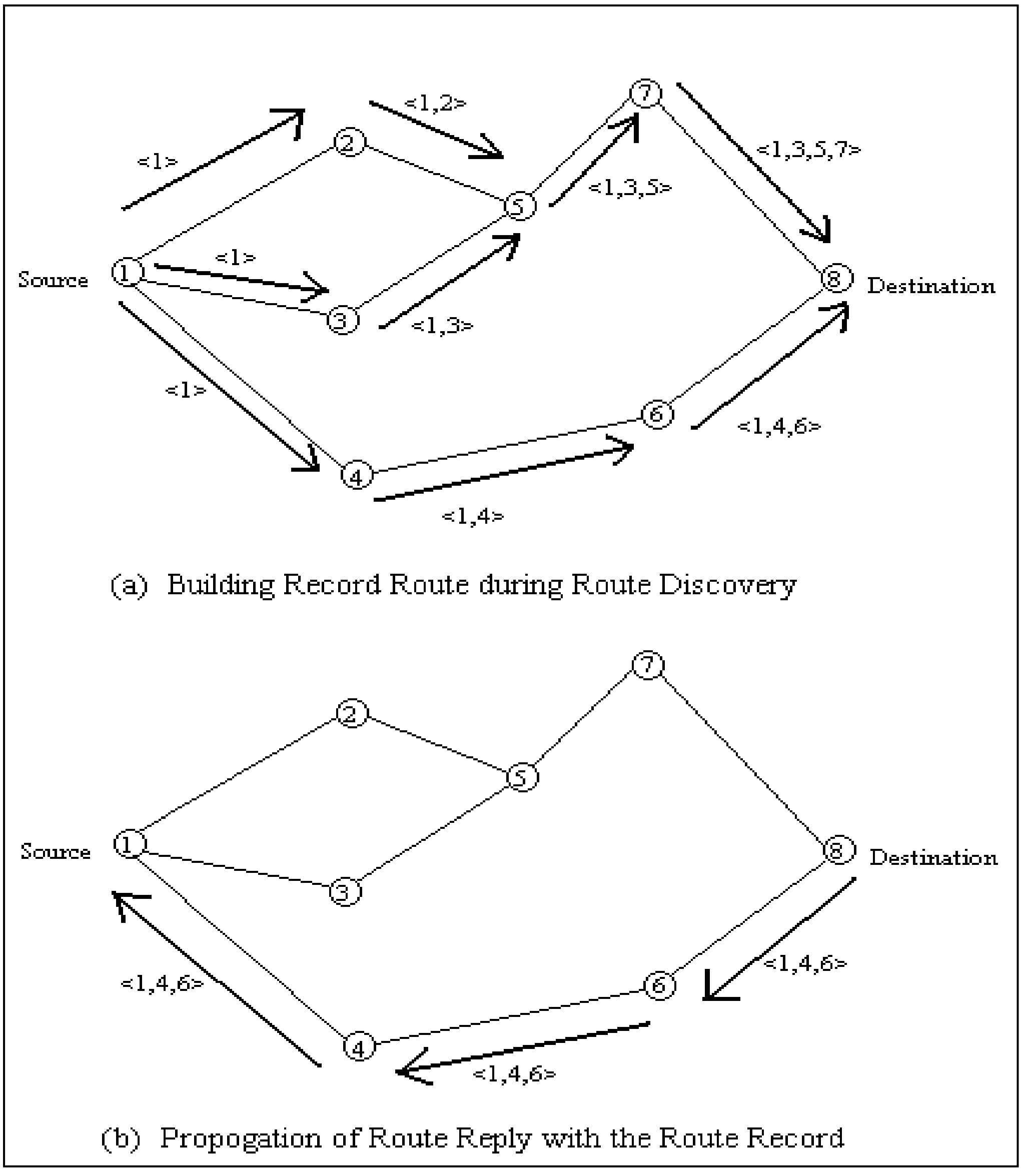

Figure 3.9 Creation of record route in DSR protocol.

The two major phases of the protocol are route discovery and route maintenance. When the source node wants to send a packet to a destination, it looks up its route cache to determine if it already contains a route to the destination. If it finds that an unexpired route to the destination exists, then it uses this route to send the packet. Nevertheless, if the node does not have such a route, then it initiates the route discovery process by broadcasting a route request packet. The route request packet contains the address of the source and the destination, and a unique identification number. Each intermediate node checks whether it knows of a route to the destination. If it does not, it appends its address to the route record of the packet and forwards the packet to its neighbors. In order to limit the number of route requests propagated, a node processes the route request packet only if it has not already seen the packet and its address is not present in the route record of the packet.

A route reply is generated when either the destination or an intermediate node with current information about the destination receives the route request packet. A route request packet reaching such a node already contains, in its route record, the sequence of hops taken from the source to this node.

As the route request packet propagates through the network, the route record is formed as shown in Figure 2.9 (a). If the route reply is generated by the destination, then it places the route record from route request packet into the route-reply packet. On the other hand, if the node generating the route reply is an intermediate node, then it appends its cached route-to-destination to the route record of route-request packet and puts that into the route-reply packet. Figure 2.9 (b) shows the route-reply packet being sent by the destination. In order to send the route-reply packet, the responding node must have a route to the source. If it has a route to the source in its route cache, it can use that route. The reverse of route record can be used if symmetric links are supported. In case symmetric links are not supported, the node can initiate route discovery to the source and piggyback the route reply on this new route request.

DSR protocol uses two types of packets for route maintenance: Route Error packet and Acknowledgements. When a node encounters a fatal transmission problem at its data link layer (the event that a next-hop along a source route is no longer reachable due to node mobility etc.), it generates a Route Error packet. When a node receives a route error packet, it removes the hop in error from its route cache. All routes that contain the hop in error are truncated at that point. Acknowledgment packets are used to verify the correct operation of the route links. This also includes passive acknowledgments in which a node hears the next hop forwarding the packet along the route.

The main benefit of DSR is that intermediate nodes do not need to respond at all to link failures unless a source directs them to—no routing information needs to be maintained at the intermediate nodes. However, DSR requires considerable overhead in each packet because the entire path must be recorder in the packet header.

The advantage of DSR over some of the other on-demand protocols is that DSR does not make use of periodic routing advertisements, thereby saving bandwidth and reducing power consumption. In addition, DSR allows nodes to keep multiple routes to a destination in their cache, so route discovery is faster than in many of the other on-demand protocols. On the other hand, because of source routing requirement, DSR is not scalable to large networks.

3.4.2.3. Ad hoc on-demand distance vector routing

Ad hoc On-demand Distance Vector Protocol (AODV) [25] was designed to address the major shortcomings of DSDV and concerns regarding the scalability of source routing. AODV uses pure on-demand route acquisition, whereby, only nodes that require a route to a given destination, or that lie along a path that is actually being used to route traffic to a destination, need actively maintain such routes.

AODV is very similar to DSR. The key difference is that AODV uses hop-by-routing versus source routing. AODV utilizes the same broadcast route discovery mechanism as DSR; however, avoids the overhead of source routing by building routes in the routing tables of the intermediate routes to each required destination. AODV limits the need for routing algorithm reaction to node mobility in two ways:

Routes are maintained only to a limited set of destination; consequently, only mobility which effects routes to those destinations will require routing algorithm reaction,

Intermediate nodes that are not used to route traffic will eventually purge inactive routes from their routing tables; consequently, only paths involving nodes actually routing traffic will require maintenance when affected by node mobility.

To find a path to the destination, the source broadcasts a route request packet. The neighbors in turn broadcast the packet to their neighbors until it reaches an intermediate node that has recent route information about the destination or until it reaches the destination (Figure 2.10 (a)). A node discards a route request packet that it had already seen. The route request packet uses sequence numbers to ensure that the routes are loop free and to make sure that if the intermediate nodes reply to route requests, they reply with the latest information only.

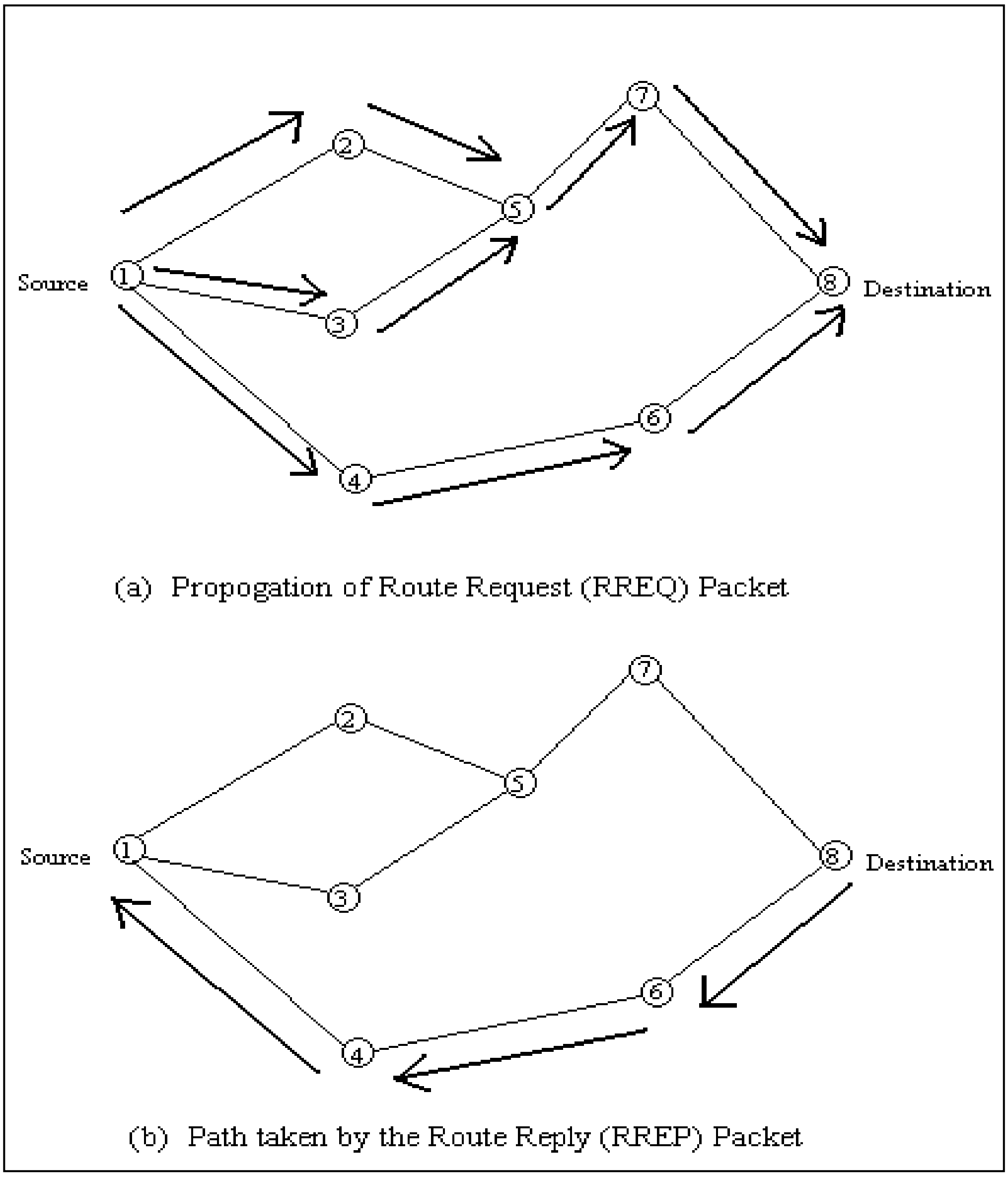

Figure 3.10 Route discovery in AODV.

When a node forwards a route request packet to its neighbors, it also records in its tables the node from which the first copy of the request came. This information is used to construct the reverse path for the route reply packet. AODV uses only symmetric links because the route reply packet follows the reverse path of route request packet. As the route reply packet traverses back to the source (Figure 2.10 (b)), the nodes along the path enter the forward route into their tables.

If the source moves, then it can reinitiate route discovery to the destination. If one of the intermediate nodes moves, then the moved node neighbor realizes the link failure and sends a link failure notification to its upstream neighbors and so on until it reaches the source. Then the source can reinitiate route discovery if needed.

AODV maintains routes for as long as the route is active. It uses bandwidth efficiently by minimizing the network load for control and data traffic, is responsive to changes in topology, is scalable and ensures loop free routing. In addition, AODV supports multicast.

3.4.2.4. Temporally ordered routing algorithm

The Temporally Ordered Routing Algorithm (TORA) [26] is a demand-based routing protocol, which maintains routes only to desired destinations. TORA is proposed for highly dynamic mobile, multihop wireless networks.

It is a source-initiated on-demand routing protocol. Route discovery is similar to the other reactive schemes; it relies on a directed broadcast approach to flood a route request message. It finds multiple routes from a source node to a destination node. The main feature of TORA is that the control messages are localized to a very small set of nodes near the occurrence of a topological change. To achieve this, the nodes maintain routing information about adjacent nodes. The critical point about TORA is it stems from its localized set of nodes. One cannot guarantee that these nodes will not be harmed. If this happens, most of the packets will not reach their destination without new path information. New path requests are triggered automatically but this process will require additional amount of time that, in turn, will hesitate the real-time applications.

The protocol has three basic functions: Route creation, Route maintenance, and Route erasure. Each node has five attributes: Logical time of a link failure, The unique ID of the node that defined the new reference level, a reflection indicator bit, a propagation ordering parameter, the unique ID of the node.

“The first three elements collectively represent the reference level. A new reference level is defined each time a node loses its last downstream link due to a link failure. The last two values define a delta with respect to the reference level [26].”

Route Creation is done using query (QRY) and update (UPD) packets. The route creation algorithm starts with the height (propagation ordering parameter in the quintuple) of destination set to 0 and all other node's height set to NULL (i.e., undefined). The source broadcasts a QRY packet with the destination node's id in it. A node with a non-NULL height responds to a UPD packet that has its height in it. A node receiving a UPD packet sets its height to one more than that of the node that generated the UPD. A node with higher height is considered as upstream and a node with lower height is as downstream.

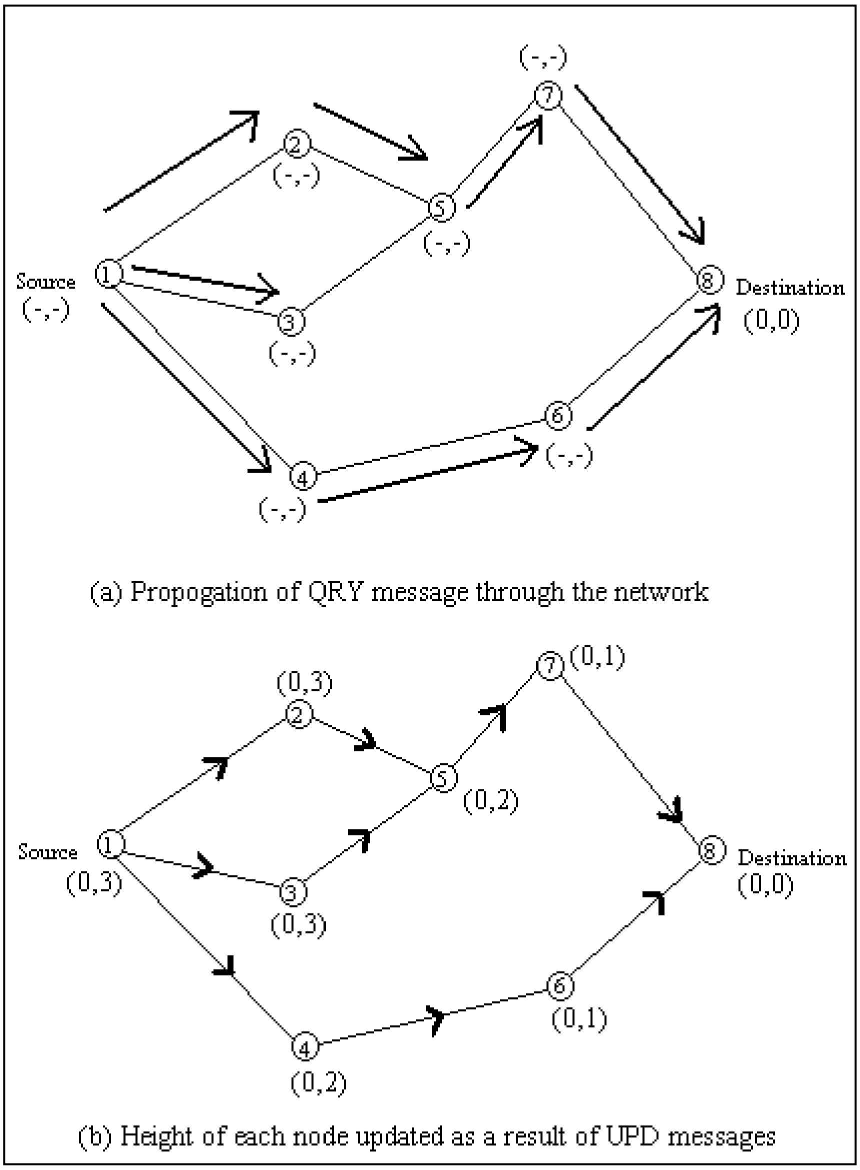

Figure 3.11 Route creation in TORA. (Numbers in braces are reference level, height of each node) [26].

In this way, a directed acyclic graph is constructed from the source to the destination. Figure 2.11 illustrates a route creation process in TORA. As shown in Figure 2.11(b), node 5 does not propagate QRY from node 3, as it has already seen and propagated QRY message from node 2. In Figure 2.11(b), the source (i.e., node 1) may have received a UPD each from node 2 or node 3, but since node 4 gives it lesser height, it retains that height.

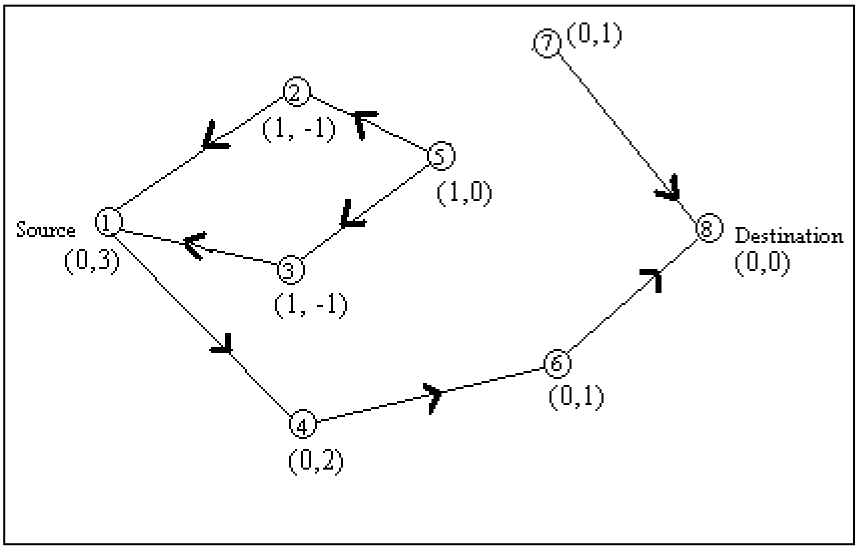

Figure 3.12 Re-establishing route failure of link 5-7. The new reference level node is 5.

When a node moves, the DAG route is broken and route maintenance is needed to reestablish a DAG for the same destination. When the last downstream link of a node fails, it generates a new reference level. This results in the propagation of that reference level by neighboring nodes as shown in Figure 2.12. Links are reversed to reflect the change in adapting to the new reference level. This has the same effect as reversing the direction of one or more links when a node has no downstream links.

In the route erasure phase, TORA floods a broadcast clear packet (CLR) throughout the network to erase invalid routes.

In TORA, there is a potential for oscillations to occur, especially when multiple sets of coordinating nodes are concurrently detecting partitions, erasing routes, and building new routes based on each other. Because TORA uses internodal coordination, its instability problem is similar to the "count-to-infinity" problem in distance-vector routing protocols, except that such oscillations are temporary and route convergence will ultimately occur.

TORA limits routing algorithm reaction in several ways: First, only routes to destinations required for on-going communications need to be maintained by the network. Next, a node does not need to respond to a topology change until it looses its last route to a required destination. Each link is assigned a direction during the route construction phase when a node looses its last downstream link (for a destination) it must initiate route maintenance. Finally, the actions required to perform route maintenance are designed to rapidly traverse only those nodes, which have been materially affected by the loss of path, in order to locate an alternate path maintained by another node. TORA addresses the problem of network partitioning in link-reversal algorithms. However, efficient operation of TORA requires high connectivity to ensure the availability of alternate routes and to increase the probability of greater link disjointedness among the paths.

TORA provides loop free paths at all instants. It provides multiple routes so that if one path is not available, other is readily available. It establishes routes quickly so that they may be used before the topology changes. It minimize algorithmic reactions/communication overhead and thus conserves available bandwidth and increases adaptability. It is also able to detect network partitions very quickly. Drawbacks are, since it uses internodal coordination it exhibits instability behavior similar to count-to-infinity problem in distance vector routing protocols. There is a potential for oscillations to occur, especially when multiple sets of coordinating nodes are concurrently detecting partitions, erasing routes, and building new routes based on each other. Though such oscillations are temporary and route convergence will ultimately occur.

3.4.2.5. Associativity based routing

A totally different approach in mobile routing is proposed in [27]. The Associativity Based Routing (ABR) protocol is free from loops, deadlock, and packet duplicates, and defines a new routing metric for ad hoc networks. The fundamental objective of ABR is to find longer-lived routes for ad hoc mobile networks. The three phases of ABR are Route discovery, Route reconstruction (RRC), and Route deletion.

In route discovery phase, the source broadcast a query packet. The intermediate node will forward the packet if it has not previously seen it. It will add its address and its associativity ticks with its neighbors in the query packet. The next succeeding intermediate node will erase its upstream neighbor' associativity tick entries and retain only those concerned with itself and its upstream node. The destination will wait for some time after receiving the first query. It then knows all the possible routes and it can thus select the best route (stable). The reply sent back to the source causes all the intermediate nodes to mark their routes to destination as valid.

RRC phase consists of partial route discovery, invalid route erasure, valid route updates, and new route discovery, depending on which node(s) along the route move. When a discovered route is no longer needed, the source node initiates a route delete (RD) broadcast (route deletion phase). All nodes along the route delete the route entry from their routing tables.

Associativity Based Routing [27] represents the first attempt to factor node mobility into the routing process by proposing a model based on the concept of node associativity. ABR builds routes on a demand-basis using basic techniques that are similar to those used by DSR and AODV. Specifically, routes are constructed using a controlled flooding process, which effectively searches the network for a stable route toward a desired destination. The novel aspect of ABR is that it attempts to select routes that are long-lived—that is, routes that are expected to survive longer than other routes. The objective is to reduce overall routing algorithm overhead by limiting the need to invoke route maintenance, which is normally required in response to node mobility. To achieve this, a new routing metric is proposed based on the concept of associativity - a measure of the duration of time that a radio link has been active between a pair of nodes. In proposing the associativity metric, ABR assumes that history provides a strong indication of the future stability of a link.

Associativity is proposed as a new metric specifically designed for ad-hoc networks. The objective is for nodes to choose paths that include stable links. In effect, associativity is a measure of how long a link has been active between a pair of nodes. According to ABR, the longer a pair of nodes has been associated, the better the link is for routing. There is no attempt in ABR to directly model node mobility in order to predict link stability. Consequently, the measure is based fully on history. It does not provide a quantitative measure that truly reflects node mobility.

The most important contribution of ABR is the idea of using relative location stability to help choose long-lived routes. The value of doing this in the decreased need to repair active routes that experience link failures due to node mobility.

Despite the novel approach advocated in ABR, there are serious shortcomings related to the associativity metric and the optimal path selection algorithm. Although the objective of the metric is to reflect how node mobility impacts link stability, it is not based on a well-defined model for node mobility. Instead, the metric relies entirely upon past link performance. Since node mobility and link characteristics are dynamic processes, the metric as defined is not a true predictor of future stability. It is merely a measure of past stability. Consequently, there is not quantitative basis for assessing the true stability of paths selected on the basis of this metric. Furthermore, the stability implied by longer associativity grows without bound. The best link available for routing to a rapidly moving node may not meet the associativity criteria which may operate well under less dynamic conditions. Finally, the method by which the metric is aggregated may not be the best way assess path stability.

3.4.2.6. Signal stability routing

The Signal Stability based Adaptive Routing (SSA) protocol. [28] proposes a simple framework for incorporating some limited physical-layer information directly into the route search and selection processes. SSA builds on the basic framework of location stability proposed in ABR, and is similar in many respects to the ABR protocol. However, the main difference, and hence the contribution of SSA is the incorporation of the signal-stability metric, which constrains the propagation of route requests during the route construction phase of the protocol. The objective of SSA is to build on-demand routes that traverse links with a high degree of signal strength stability and, whose incident nodes show a high degree of relative location stability. The purpose is to minimize the need to repair routes, which experience link failures due to poor or excessively variable signal quality, or due to node mobility.

Although SSA uses a demand-based directed broadcast to construct routes, the route construction process of SSA differs from ABR. Route requests propagate only over links that meet the signal strength stability criteria. SSA is comprised of two cooperative protocols: the Forwarding Protocol (FP) and the Dynamic Routing Protocol (DRP). Each node maintains two tables; Signal Stability Table (SST) and Routing Table (RT). Each host sends out a link layer beacon periodically to maintain connectivity. The receiving neighbors record the signal strength at which beacon is received (signal stability) and also the count on the number of times it is received (location stability). Based on this a link is assigned a status of strong channel (SC) or weak channel (WC).

Although SSA is the first ad-hoc routing protocol specifically designed to incorporate physical layer information into the routing decision process, the basic approach is very similar to ABR. In short, SSA relies on the average link strength as seen over time as a predictor for future link stability. The method that is proposed relies on static thresholds that are analogous to the thresholds in ABR; however, SSA also considers the average link strength as seen over time and uses this measure as the predictor for future link stability. In doing this, it may be possible to respond to changes in signal strength in an adaptive way. Specifically, although the thresholds remain static, the measure of stability can reflect some notion of the *direction* it is moving. In this way, it provides a more reasonable predictor of future performance. However, SSA continues to rely on the same location stability metric proposed in ABR, which relies on a static count of clicks.

3.4.3. Cluster-based routing in ad-hoc networks

Dynamic hierarchical techniques were designed for early PR network routing; however, they required substantial de-centralized control and were based upon the objective of minimizing routing table size—this assumed hierarchical table-based routing. More distributed approaches are advocated for ad-hoc networks, and more feasible objective is based upon generating a relatively stable cluster topology. In effect, where clustering in a fixed network is based upon making a large network appear much smaller from the perspective of the routing algorithm, the objective in an ad-hoc network should be to make a very dynamic network appear less dynamic.

Many researchers believe that clustering and dynamic hierarchical routing is too complex and cumbersome to be effective in ad-hoc networks. However, as new techniques have evolved, and the efficacy of hybrid routing strategies that can capture the benefits of both proactive and reactive routing become more apparent, the literature on the topic has become increasingly rich. In this section, several approaches to clustering and cluster-based routing for ad-hoc networks are discussed. Some of the schemes are fully proactive, whereas others combine proactive and reactive approaches into hybrid schemes that are either hierarchical or flat-routed. Although the techniques described in the literature reflect dynamic reorganization, and advocate the benefits of hybrid routing, two substantial things are missing: First, none of the schemes proposed thus far defines an adaptive strategy that can dynamically modify the cluster characteristics according to network conditions. Secondly, none of the clustering strategy -except VCL- use node mobility or location information to generate stable clusters, or provide quantitative information regarding the stability of clusters membership or paths.

Clustering strategies designed for ad-hoc networks differ in terms of their basic objectives, the criteria they use for managing the cluster organization, their cluster algorithms, and the routing strategies that are implemented on top of the cluster organization [1, 7, 12, 16, 17, 18, 29, 31, 33, 34, 36].

3.4.3.1. k-cluster-based routing

A cluster-based scheme proposed in [29] which dynamically organizes the topology into *k*-clusters, where nodes in a cluster are mutually reachable via k-hop paths. The algorithm considers k = 1, and reduces to finding cliques in the physical topology. Using a first-fit heuristic, the algorithm attempts to find the largest cliques possible. Although the algorithm does not form optimal clusters, it still requires a three-pass operation each time a topology change occurs: one for finding a set of feasible clusters; a second for choosing the largest of the feasible clusters that are essential to maintain cluster connectivity; and a third to eliminate any existing clusters which are made superfluous by the new clusters. The idea of this scheme is to use the cluster organization in order to manage the routing process efficiently. Specifically, the maintenance of clusters effectively generates a set of proactive routes to every destination in the network.

A de-centralized approach is used to create a set of overlapping clusters that cover all the nodes in the network. The algorithm is event driven. The actions depend on the specific topological change, which has occurred, namely, node activation, node failure, link activation or link failure. The basic cluster formation algorithm is executed whenever a node activates. The idea of the algorithm is to generate large clusters. Large clusters are desired because this will minimize the amount of routing update information. Despite the simplified heuristic, the algorithm remains too complex. Furthermore, it requires the node executing the cluster creation to have global topology knowledge. After the clusters have been created, the node must still disseminate the new cluster topology to the network nodes.

The network maintains routes proactively. Each node maintains a cluster membership list, which has the current cluster affiliation of every node in the network. Each node also maintains a boundary list, which specifies the designated boundary node between to each overlapping cluster. Boundary nodes belong to more than one cluster. Each time a new cluster is formed, the boundary nodes receive new cluster information and store the list of nodes in its cluster and a list of boundary nodes in the network. They then rebroadcast this information. Only boundary nodes disseminate routing information. The boundary nodes from all the clusters form a connected subgraph since all clusters are cliques; consequently, routing over multiple clusters is a matter of relaying a packet from the source to a sequence of boundary nodes, and finally from the last boundary node to the destination.

The k-cluster algorithm has several major shortcomings, namely, the algorithm is complex, it has far reaching effects whenever there is a topology change, and it relies on global topology information for routing, hence the cluster topology is not fully leveraged for the benefit of routing.

3.4.3.2. Adaptive clustering for mobile wireless networks

Their adaptive clustering scheme proposed in [30, 31] differs from the other schemes. Rather than using clustering to minimize the network's reaction to topological changes, their scheme is intended to provide controlled access to the bandwidth and scheduling of the nodes in each cluster in order to provide QoS support. Hierarchical routing and path maintenance were a secondary concern. The idea is to construct clusters of nodes that within one-hop of some *central* node, consequently, all nodes are within two-hops of each other. Cluster size is controlled through radio transmission power—assumed to be a fixed and uniform value.

A proactive global routing strategy is advocated that relies in DSDV to construct bandwidth constrained shorted-hop paths. No attempt is made to leverage the cluster structure to improve routing efficiency or path maintenance. Consequently, a completely flat-routed scheme is used. The clusters are used to control access to the transmission medium and reserve bandwidth for QoS constrained communications. The algorithm for organizing the clusters is not based on any quantitative performance criteria. A distributed algorithm based on lowest node identifiers deterministically partitions a network into two-hop clusters.

Whenever the movement of a clustered node causes it to become greater than two-hops from any other node in the cluster, cluster maintenance must be invoked. Specifically, some node must move from its current cluster to a neighboring cluster. The idea is to minimize the number of node transitions between clusters. The strategy is intended to let the highest degree node and its neighbors remain in their current cluster.

There are many undefined parts about the framework and many shortcomings. It relies on fully proactive routing. Furthermore, although the clustering algorithm itself is very simple, it has no ability to adapt to traffic or mobility conditions. The choice of which nodes to include in a cluster depends entirely upon the predefined assignment of node ID numbers and is then random according to the events which initiate cluster formation.

3.4.3.3. Routing using minimum connected dominating sets (Spine)

The objective of the spine [32, 33] is to provide an efficient and robust methodology for disseminating routing information in dynamic ad-hoc networks. The methodology is based upon the maintenance of a virtual backbone network, which consists of a relatively stable set of connected nodes, such that every node in the network is either in the set, or a one-hop neighbor of a node in the set. Such a set is referred to as a connected dominating set.

The spine framework does not completely specify the details of the routing algorithm. Instead, it presents a number of alternative strategies based upon the amount of topology information maintained at each node. The basic spine routing approach described above provide the maximum information at all nodes—global topology at the spine nodes, up-to-date routing tables at all the nodes. The problem with the basic spine approach is that it still requires a substantial amount of update traffic to maintain the global knowledge and keep the routing tables updated.

To address this problem two approaches are proposed. The first is called clustered spine routing (CSR), a two-level hierarchical routing scheme. The clustering algorithm is not specified; however, cluster membership is controlled via a set of predefined values, which bound the number of nodes, the diameter, and the maximum degree of each cluster. The objective is to maintain basic spine routing within the clusters, and to maintain a hybrid scheme for inter-cluster routing. The second approach for improving the performance of basic spine is called partial knowledge spine routing (PSR). In PSR, the spine nodes maintain knowledge of the spine structure, their local dominating set, and the domain membership table—that is, they must maintain knowledge of what domain every node is in. The route discovery process is a matter of sending a route request message on a spine-path from the dominator of the source node, to the dominator of the target destination node. The idea is to construct a path that avoids the actual spine path as much as possible—the spine must not be congested with user traffic since it provides what is effectively a dedicated backbone for control traffic. Unlike the CSR approach, which uses source routing, PSR routes are constructed as hop-by-hop routes.

The spine routing framework presents some important ideas. Specifically, it raises the question for the need to maintain a dedicated channel for control information. The basic spine involves multiple domains that are essentially clusters of nodes that are one-hop away from their cluster head—the dominator. The PSR uses a cluster organization that reduces the globally maintained information in favor of a reactive route construction process. Finally, CSR effectively represent a two-level of clustering strategy. Although these schemes reduce the amount of global information, in a highly dynamic environment it may be difficult to maintain the cluster or domain membership table at all the root or spine nodes.

3.4.3.4. Multimedia support for wireless network system (MMWN)

The Multimedia Support for Wireless Network (MMWN) [34] system is based upon a hybrid architecture, which includes the characteristics of ad-hoc and cellular networks. Its framework uses hierarchical routing over dynamic clusters, which are organized according to a set of system parameters that control the size of each cluster and the number of hierarchical levels. Aggregation of routing information is used to achieve scalability and limit the propagation of topological change information. A multilevel strategy is used to repair virtual-circuit (VC) connections, which have been disturbed due to node mobility.

The MMWN system presents a connection oriented, hierarchical routing strategy that proactively maintains routes to destinations at the same level of the hierarchy. For a two-level scheme, nodes proactively maintain routes to all destinations in the same cluster using a link-state protocol and source routing to setup connections. Border nodes are the nodes that have neighbors in other clusters. These are dynamically arranged into virtual gateways (VG), which provide the routes to remote destinations. The cluster topology is maintained proactively via a level-2 link-state protocol that uses VGs in place of physical links, thus cluster-by-cluster source routes can be constructed in a manner that is similar to the CSR scheme in spice routing. The problem of inter-cluster routing then reduces to a mobility management problem. Due to node mobility, it is possible for any node to dynamically change its cluster affiliation. Hence, in order to route to a remote node, a source must first identify the cluster in which the destination currently resides. The services of a location manager within each cluster are used to help locate the current cluster affiliation of a desired destination. The location managers participate in the dynamic maintenance of a distributed location database not unlike mobility management in cellular networks. Once a destination has been located, a virtual-circuit (VC) is setup using a cluster-by-cluster source routed connection setup message. At each successive cluster along the way, the route is expanded to reflect the topology information within the specific cluster and level of the hierarchy.

A multiple-level hybrid cluster architecture is defined in the MMWN system that is dynamically controlled de-centralized cluster leaders using a set of predefined parameter values. The hybrid aspect of MMWN is based upon the assignment of nodes as endpoints or switches. Switches have two tasks. They act as base-stations for end-point nodes which must affiliate with a local switch, and they must cooperate to organize the cluster topology and participate in the network routing protocols.

The clustering algorithm groups switches into cluster, and clusters into super-clusters. The number of levels depends on the system parameter values. A cluster leader is assigned using a lowest node id algorithm. The leader is responsible for managing the various clustering actions.

The MMWN system is a broad architecture that attempts to address a wide range of issues related to the management of ad-hoc networks. As such, the system is very complex and involves a large number of interrelated procedures that maintain the cluster topology, proactively maintain topology information, locate destination cluster affiliations, generate source routes, and manage virtual circuit construction and repair. The novel aspect of the scheme is use of the hybrid architecture. However, the use of proactive routing and static parameters in the definition and maintenance of the clusters call to question the potential for this architecture to support significant node mobility. Although routes are maintained proactively, the system suffers substantial overhead from the need to maintain the mobility management database. In a reactive scheme, mobility management is not used, or more accurately is embedded directly into the route acquisition process. In short, MMWN is a comprehensive approach to building scalable ad-hoc networks, but the multi-level hierarchy is probably too complex to maintain, and the routing scheme should better leverage the cluster topology to reactively discover routes to remote destinations.

3.4.3.5. Virtual subnet routing (VSN)

The Virtual Subnet Routing (VSN) [35] architecture uses a two-tier addressing structure to effectively construct a fixed hierarchical organization of the nodes in an ad-hoc network. According to the VSN strategy, each node is dynamically assigned an address consisting of two subnetwork identifiers, namely, a physical subnet (PSN), which depends on the relative spatial location of the nodes, and a virtual subnet (VSN), which depends on the assignment of addresses within the physical subnet. Packet routing is accomplished by forwarding within the physical subnet to the desired virtual subnet, and then across the virtual subnet to the desired physical subnet.

The idea of is for each node to join the a physical subnet which consists of a set of mutually reachable nodes—that is, each node within the physical subnet is assumed to be within bidirectional transmission range of each of the other nodes, consequently, multi-hop routing is not necessary within a physical subnet. Upon joining a new physical subnet, a node is assigned a unique address within the subnet that reflects membership in a virtual subnet. The number of possible virtual subnets is fixed, and only one node in each physical subnet can be assigned to each virtual subnet. Nodes in each physical subnet are assumed to be capable of adjusting their transmission power-levels, if necessary, in order to reach all the nodes in each adjacent physical subnetwork.

There are a several substantial shortcomings to the VSN architecture as it is proposed. First, there is no clear explanation of regarding the construction or maintenance of VSN routing, which is essentially a variant of inter-cluster routing. Clearly, this is a difficult challenge, which is central to the ad-hoc routing problem. Other serious problems that have not been addressed relate to the management of connected VSNs and the problem of locating destinations when the source node's VSN does not span the PSNs. Finally, the predefined addressing structure imposes constraints on the sizes of both the physical and logical subnet structures. Consequently, the strategy is not adaptive. Hence, cluster sizes may become very inefficient, both in terms of their ability to manage traffic at the MAC level within the PSN, and in terms of their ability to respond to node mobility.

3.4.3.6. Zone routing protocol (ZRP)

ZRP [36] attempts to balance the tradeoff between pro-active and reactive routing using a dynamically maintained zone topology. The objective of ZRP is to make ad hoc routing both scalable and robust in the face of significant node mobility, without saturating the network with routing update traffic or excessive route setup latency. ZRP makes several key contributions to ad-hoc routing, most important of which is the specification of a novel technique for improving the efficiency of the reactive route-search process using a hybrid routing scheme based upon proactive routing *zones*. Routing within a zone is managed by a proactive routing protocol, whereas, routing to destinations beyond a source node's routing zone is achieved via a reactive routing protocol.

Each node in an ad-hoc network maintains its own routing zone, a cluster of nodes which can be reached along paths that are less than or equal to ρ hops. The parameter ρ is known as the zone radius, it is a predefined value. Since each node is the center of its own routing zone, the zone topology consists of a set of fully overlapping dynamic clusters.

Intra-zone routing consists of a proactive routing protocol that maintains a hop-count limited routing table. Inter-zone routing is managed by a demand-based routing process, the inter-zone routing protocol (IERP). The idea is to attempt to effectively leverage the proactively maintained zone topology in order to make the reactive route search process faster and more efficient. Thus, making ZRP more scalable and better provides ability to handle mobility those monolithic schemes. ZRP does not use hierarchical routing—the inter-zone routes that are established are flat, hop-by-hop routes, which must be dynamically restored whenever node mobility leads to link failures along active paths.

The main contribution of ZRP is that they show hybrid routing to be a feasible alternative to pure proactive or demand-based approaches. Specifically, it is shown how using proactive routing in limited circumstances improves the reactive route search process. However, the use of a predefined parameter to bound the routing zone radius limits the adaptability of the strategy. Consequently, it is unable to leverage the zone topology as effectively as it could if the zone radius was adaptive. Furthermore, the continuously overlapping feature of the zones introduces serious complexities in managing the route search process as query explosion can result from multiple threads of the same query propagating in parallel through overlapping zones. Finally, by constructing flat-routed paths for inter-zone routing, ZRP fails to fully leverage the proactive route maintenance in the maintenance of inter-zone routes as it must respond to node mobility using reactive route repair. In summary, ZRP presents an effective hybrid strategy, however, it is limited by the three factors, namely, the zone characteristics are not dynamically adaptive, the zones overlap, and inter-zone routes do not leverage the existing zone topology as much as they could.

3.4.3.7. MANET cluster based routing (CBR)

In CBR [1], the network is dynamically partitioned into clusters of nodes that are reachable within two-hops. The basic clustering algorithm proposed in [30] is adapted for use in CBR; however, the objective of clustering in CBR is to maintain an effective topology that enables more efficient discovery and maintenance of routes. A dynamic source routing protocol is specified for the discovery of inter-cluster routes, which leverages the cluster topology in order to minimize flooding traffic. A simple lowest node id algorithm is used to construct the clusters. Each cluster is assigned a cluster head and a set of gateway nodes. Gateway nodes have links to nodes in adjacent clusters. Source routes are constructed on a demand basis through a modified flooding algorithm. Each route request message is relayed along a route that alternates between gateway nodes and cluster heads. The route request is propagated until it is received by the destination, or a node that can supply a partial source route to the destination. The route reply is returned along the reverse path, at each cluster, the cluster heads may modify the source route to reflect a more optimal path through the cluster, possibly avoiding the cluster head altogether. If the cluster topology is stable, CBR provides a more efficient alternative than DSR. However, topology dynamics may have a severe effect on the cluster topology, and it is not clear how well adapted the techniques are for cluster and route maintenance.

3.4.3.8. Virtual cell layout for mobile ad hoc networks (VCL)

Virtual Cell Layout [3, 4] differs from all other schemes since it use a cellular structure. It provides a efficient control over resources and presents new acceptable solutions for mobile ad hoc networks. It is a well-defined structure. The details of this scheme are in Section 3.1.

3.5. Overview

The discussion of which routing technique is better will not be our intention. There are papers dealing with this subject and they also compares these routing techniques [many ref]. Our discussion will be about the good and bad sides of the proposed routing techniques, since none of the proposed routing techniques saturate all the expected requirements of an ad hoc network.

4. ROUTING APPROACHES FOR SCALABLE AD HOC NETWORKS

4.1. Discussion

Many routing techniques and schemes are proposed for mobile ad hoc networks, but they fall into some drawbacks and do not satisfy all the requirements of ad hoc networks. On the other hand, all of them have some important features that may make one of them an acceptable solution for a special selected environment. In this section, we will discuss the advantages and disadvantages of proposed schemes, and will try to find out a scalable solution for mobile ad hoc networks.

Proactive routing techniques distribute routing information throughout the network periodically in order to pre-compute the routes to all possible destinations in the network. By this way, we can have high quality shortest routes to all possible destinations without any delay. But in a mobile network this approach causes resource overheads and more energy consumption.

To avoid overhead and energy consumption, reactive routing techniques were proposed. The objective of reactive routing is to make a better balance between responsiveness and efficiency. Reactive routing techniques manage it by maintaining a limited set of routes for just currently on-going communications, and by minimizing the reaction of the routing algorithm to topology chances for only these routes. Routes are not calculated priorly, so no route information is maintained. Routes are discovered on communication need. By this way, resource overhead and energy consumption are avoided. Nevertheless, the found route may not be the optimal route, and as the topology changes, the route becomes less optimal. In addition, there will be a significant delay at route setup phase, and in fact, the route may not be found. Finally, as the network size becomes greater, the scalability of this technique reduces.

To improve the performance of the routing in ad hoc networks, hybrid routing techniques have been proposed. The objective of hybrid routing is selectively using different routing strategies under different circumstances. The idea is to divide the routing task in order to get a better performance than a single algorithm's performance. In physically or logically different domains, or on different time-scales within the same network, multiple routing techniques – that those better satisfies the requirement of selection – can be implemented.

The main drawback common to all protocols is about scalability. Most of the proposed schemes are not scalable and simulated protocols do not satisfy the requirements of a network larger than 50 nodes (mobile terminals). In some simulations presented in [24], the network size reaches to 500 - 1000 nodes, but this size of network is also not the desired one. Their results also do not show the expected behavior of ad hoc networks.

To achieve network scalability, cluster–based routing techniques are proposed. In a cluster-based ad hoc network, the network is dynamically organized into partitions, called clusters, to maintain a stable and effective topology. The membership and characteristics of each cluster may change dynamically over time in response to node mobility and is determined by the criteria specified by the clustering algorithm. Clustering in ad hoc network can be used to support hierarchical routing (to make route search process more efficient for reactive protocols and for scalability), to support hybrid routing (in which different routing strategies operated in different domains, or levels or hierarchy) or to provide more control over access to transmission bandwidth. The criteria which determine the routing protocol to be used in a selected cluster or domain, depends upon the size of the clusters, the patterns and rates of node mobility, the distribution of the traffic among nodes, the efficiency of the route search process, the speed of the route repair following a node movement and other possible effects of routing algorithm processes. Therefore, an adaptive dynamic clustering strategy must be determined to specify the cluster size and the characteristics of the cluster.

Many cluster-based schemes are proposed as mentioned in Section 2.4.3. All of them have some drawbacks since they use a flat routing strategy and/or a pure proactive routing strategy. Although they presents us new applicable cluster-based routing schemes, they fall into the scalability problem since their tightly connection of flat routing or proactive routing strategy.

The question of which technique or scheme is better suited for ad hoc networks depends on a number of factors. Some of these factors described above. Each technique has some advantages under some certain circumstances, but may not perform well in other situations. All of the routing techniques fall into some drawbacks – mostly scalability problem - when they used purely alone. Except VCL, most of the proposed hybrid routing and cluster-based routing schemes are not well defined and have some drawbacks.

In this section, to the question “what must be done” will be answered as our approach to the routing and structure in mobile ad hoc networks.

To provide scalability for large networks, hierarchical structure can be used. The routing tables maintained in mobile nodes will be smaller, that makes the network appear smaller. There are at least two-level in a hierarchical structure. The lowest level is the physical level and consists of clusters of mobile nodes. Communications between the nodes in different clusters are established over cluster-heads by using virtual levels.

Hierarchical structure, however, is scalable for large sized networks, it is effected negatively from node mobility. Due to the node mobility, the membership of clusters in physical level (lowest level) will change continuously. It causes many process and computations at each level of the hierarchy reaching to the upper level that reduces the efficiency of the routing protocol implemented.

If the levels of the hierarchy are reduced to a small number, the disadvantages of the multi-level hierarchical structure are avoided. By organizing the mobile nodes at the lowest level (physical level) into clusters according to the characteristics of the mobile nodes, we can employ a hybrid routing strategy that use the advantages of both hierarchical structure and on-demand / table-based routing techniques. Our goal must be to construct such a kind of cluster-based structure. As stated above, most of the proposed cluster-based schemes and hybrid structures in literature have some drawbacks or are not well defined.

To take the advantage of both hybrid approach and hierarchical structure, and to get a better performance respect to a single routing algorithm, different multi-routing techniques can be implemented. The routing techniques, which will be implemented, can be selected according to number of nodes in clusters (size of clusters) and the number of clusters in the network. When cluster size remains small but the number of clusters is great, then a hybrid structure that uses a routing proactive technique in clusters and a reactive routing technique between clusters can be implemented. But when the cluster size becomes greater, the number of nodes in clusters becomes greater relatively. This causes the number of cluster in the network to reduce (less number of clusters). In this case, a hybrid structure that use a reactive routing technique in clusters and a proactive / reactive routing technique between clusters can be implemented.

To achieve this approach, an adaptive dynamic clustering strategy must be determined to specify the cluster size and to organize the clusters according to the characteristics of the mobile nodes. The hardness is to organize clusters dynamically with the changing conditions. VCL, which is the well specified of all schemes, proposes a semi-dynamic clustering strategy for mobile ad hoc networks.

In VCL, clusters are organized according to the characteristics of mobile nodes those are under coverage area of radio access points, which act as a base station. The other nodes outside of the range of the RAP, reaches to these access points via neighborhood nodes of the cluster over multi-hop paths and become a member of a cluster and provide connectivity, if they can. The other nodes that cannot reach to any radio access point anyway, organize into clusters dynamically as in ad hoc approach. Therefore, a dynamic clustering technique is used in VCL. The main contribution of VCL is about resource management over these organized clusters. Therefore, VCL provides a scalable solution for mobile ad hoc networks with efficient resource management.

4.2. Virtual Cell Layout Approach (VCL)

VCL differs from all other proposed schemes because of its structure. It is a well-defined scheme and is applicable to the mobile subsystem of the next generation tactical communications systems. The proposed approach enables the management of scarce resources efficiently in a mobile environment with a mobile infrastructure. Therefore, it is novel on this topic.

4.2.1. Structure of VCL

"In VCL, the area of communication is tessellated with regularly shaped, fixed size hexagons. Each hexagon represents a VCL cell to which the available spectrum is assigned according to the N=3 frequency reuse plan (Figure 3.1). The short codes that define the access points in UTRA, and the preamble codes used for random access are distributed among the fixed VCL cells, as well. Hence, the mobile access points can determine the most appropriate set of carriers and codes without a need for a central topology database or a central resource manager, if they can pinpoint their current geographic location [3]."

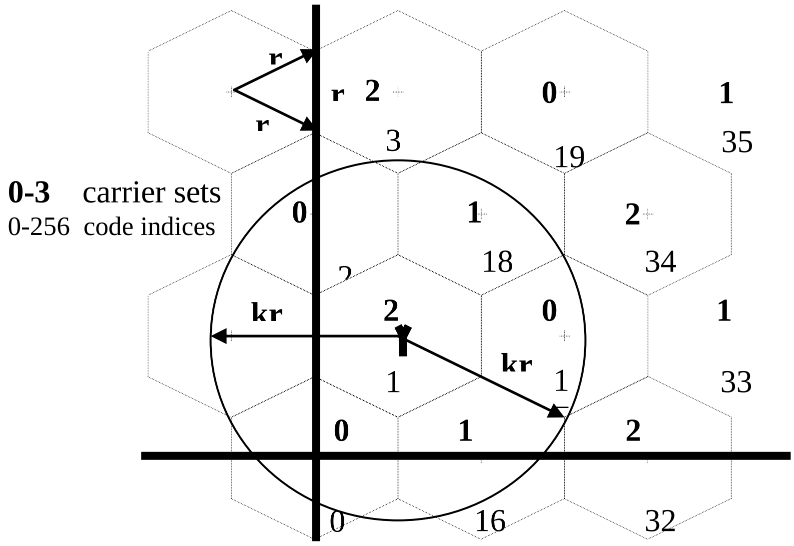

Figure 4.1 Virtual cell layout [3].

All the proposed algorithms and schemes used in VCL are in [3]. In [3, 4], a multi-tier, self-configuring system is presented as illustrated in Figure 3.2. It is assumed the availability of the following equipment in [3]:

- The availability of Man Packed Radios (MPR): the MPRs will be capable to communicate with n other radios, and even concentrate the traffic of n other radios into a single higher capacity channel which can be established with other MPRs, Radio Access Points (RAPs), Unmanned Aerial Vehicles (UAVs) and satellites. It is also essential that a man packed radio can communicate in two carriers simultaneously. Man Packed Radios are the radios that have the abilities of the Future Digital Radios explained in [3]. They are not essentially man packed. They may be mounted on the vehicles of different types.

- The availability of RAPs : RAPS will be capable of communicating with m MPRs simultaneously and concentrate their traffic into a single trunk that can be established with UAVs, satellites, Wide Area Subsystem (WAS) gateways or even with other RAPs.
- The availability of UAVs.
- The availability of satellites.
- The availability of WAS gateways.
- The availability of location tracking systems: it is assumed in [3] that every MPR, RAP and UAV is capable to find out its geographic location. This can be accomplished either with Global Positioning Systems (GPSs), which are ubiquitous, or with some other location finding systems. In the case that none of these systems are available or not functioning properly, it is assumed in [3] that the geographic locations can be entered into the communications devices manually by the operators.

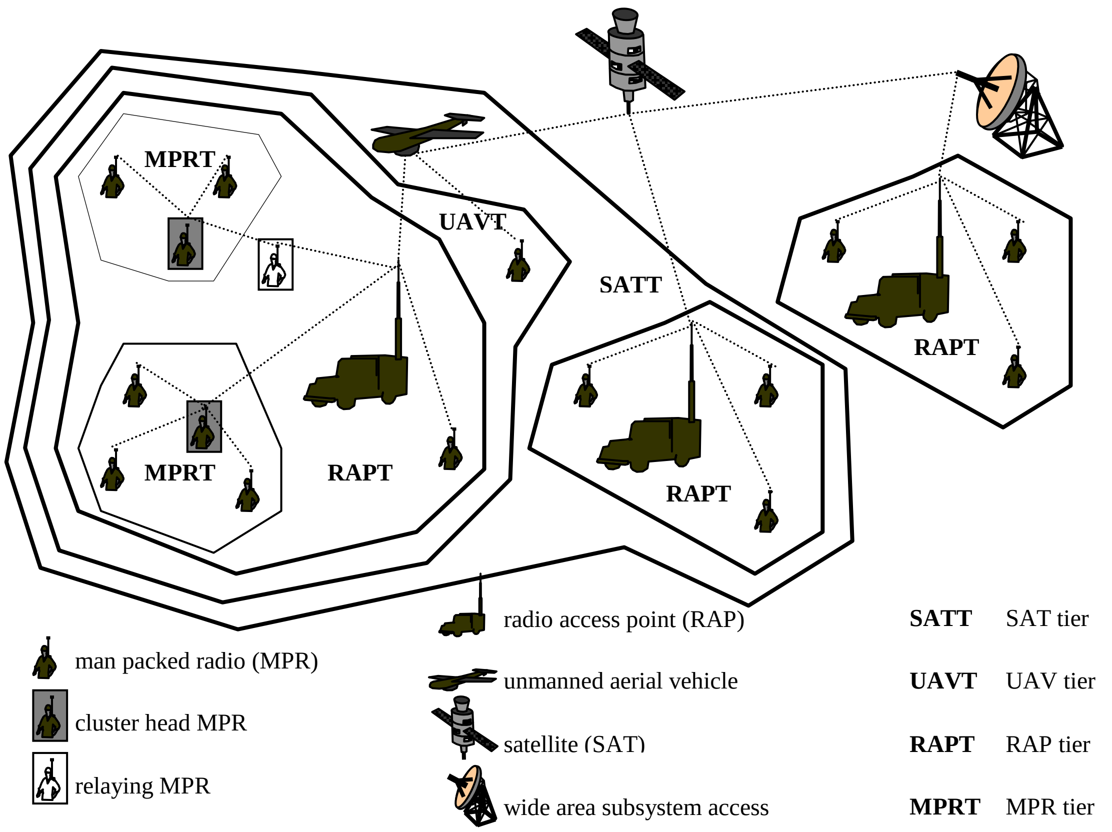

Figure 4.2 Multi-tier mobile subsystem [3].

And in [3] four tiers of Mobile Subsystem (MS) is proposed:

- “MPR Tier (MPRT) (microcell): this will be the low tier microcellular part of the MS. One of the man packed radios act as a cell head, and the cell head or one of the other man packed radios act as a gateway to other tiers, other cells or directly to a WAS access point.
- RAP Tier (RAPT) (macrocell): this will be the high tier macrocellular part of the MS. RAPs act as mobile base stations. RAPT cells may also construct underlay clusters, since MPRT and RAPT cells are perceived as underlay cells and a RAPT may include a number of MPRT cells.
- UAV Tier (UAVT): this is the first level overlay tier of the MS. UAVT cells cover the areas which are hidden for the lower tiers, and also help the lower tier cells to access the WAS and communicate with each other.
- Satellite Tier (SATT): this is the topmost overlay tier over the UAVT.”

“The system should be self configuring, since it is a very dynamic one. It is intended to use procedures similar to the mobility management functions employed in ordinary cellular networks with the following basic concepts to make the system self–configuring:

- MPRs are registered to RAPT cells whenever possible, if the tier to be registered is not set explicitly by the operators. If there is not a RAPT cell to be registered, MPRs try to register to an MPRT cell.
- If the MPRs cannot find an MPRT or a RAPT cell to be registered, they create a new MPRT cell and connect this new cell to the lowest possible overlay cell.
- MPRs handoff between the cells as they move and if it is required. If it is required and there are enough resources, they can handoff to the upper or the lower tiers.
- MPRs may reach to one of the tiers by multihop, which indicates that it is utilized ad hoc approaches especially in MPRT.
- All concepts, schemes and strategies are distributed.

- In the case of scarce resources or if needed, the network sometimes may be divided into smaller subnetworks that cannot communicate with each other. These smaller subnetworks can be as smaller as an MPRT cell. [3]"

The Virtual Cell Layout (VCL) is used for resource planning tasks, such as code, preamble code or carrier assignment. By the help of VCL, these tasks can be carried in a distributed way without relying on the existence of a central system or an accurate and timely topology database. If an access point knows its geographic location, this location information can be mapped into radio resources, which are a carrier set, a spreading code, and a preamble code, index for a UTRA based application.

The real cells are mobile and created by either RAPs or MPRs acting as cluster heads. The size of real cells may be different from the size of VCL cells. If we say the side length of a VCL cell is r, then the real cell radius becomes kr in which k is the multiplication factor to figure out the real cell radius from the VCL cell radius. When the multiplication factor is one, the real cell usually cannot cover the entire virtual cell, because the access points are mobile.

In [0, 2], only RAP and MPR tiers are presented. In RAP and MPR tiers, two basic types of nodes, namely RAPs and MPRs are used.

4.2.1.1. Radio access points

It is assumed that each RAP has a connection to the WAS through one of the access points in the higher tiers such as WAS edge nodes, satellites or UAVs. RAPs act similar to the base stations of UTRA. However, they do not rely on any other node to work.

They utilize a VLR in which it records the nodes registered to it and the other nodes registered to its children and grandchildren nodes in the hierarchy. If it can be registered to a node higher in the hierarchy, the nodes registered to it are also registered to this higher node. When a call request arrives, firstly the local VLRs are looked up, and if the destination for the call is not found in the VLR, the call is routed to the higher node in the hierarchy. Since in most of the cases, the subscribers try to communicate with the nodes close to them, the probability to find the destination for a call in the earlier VLR lookups is high.

4.2.1.2. Man packed radios

MPRs communicate with each other through the RAPs as they are subscribers of a digital cellular system. Actually, they are the terminal equipment are the equivalent of the mobile radios in UTRA with some additional abilities. First of these additional abilities is that they have the intelligence of knowing the possible codes and carriers that can be used in their current location, or the possible codes and the carriers to handoff. They do not need to retrieve this information from a base that they are registered. This is the result of VCL approach. Secondly, we assume that they can act as a base station with some limited capabilities when required. This means that they can relay the communications of other MPRs to the higher levels in the hierarchy. This is essential for the ad hoc approaches we devised in this level.

4.3. Scalability of VCL

VCL is a hierarchical cluster-based structure. Therefore, VCL has the properties of both the hierarchical architecture and cluster-based architecture. Because of that, there are many advantages of VCL architecture, but we will consider only the advantages about routing.

By using Radio Access Points (RAPs), almost all operation area are covered by these RAPs. Because RAPs move with mobile end terminals (Man Packed Radios-MPRs), connectivity is preserved. By the relay ability of MPRs, the MPRs those are not under coverage area of any RAP, can connect to RAPs via multi-hop MPRs. It results as almost full connectivity.

RAPs are responsible for resource and communication management. They are designed to act similar to the mobile base stations, and they are mobile like MPRs. They move as the network moves. Because the MPRs which have similar characteristics are registered to the same RAP, they almost show the same mobility patterns. RAPs move with MPRs, therefore, the membership of a RAP almost never changes in time. This means almost relatively stable mobile network.

Because of the property of relative stability of VCL, links used to communicate remains stable. This means fewer overheads, less information interchange, less bandwidth usage, less power consumption and less link failures. Therefore, the advantages of Associativity Based Routing (ABR) remain in VCL.

VCL is simulated with 18529 mobile terminals. Most of previously proposed schemes are simulated for 50 units, some reaching at most 500-1000 units. If we compare the number of simulated units, it is absolute that VCL provides a good solution to scalability in ad hoc networks.

If we summarize the advantages of VCL for a routing problem, they are:

1. Almost full connectivity is established because of RAPs usage.
2. VCL provides a relatively stable network, because clusters are organized according to MPRs characteristics. It benefits the associativity property, because of the clusters' member characteristics.
3. It is scalable for large size mobile networks.
4. On communications, short routes are established. This reduces the bandwidth and energy usage, route maintenance processes to a minimum level and minimizes the effects of nature.
5. It benefits the advantages of hierarchical structure.
6. It benefits the advantages of cluster-based routing.

4.4. Drawbacks of VCL

Structure of VCL and its advantages described in Section 3.2 and Section 3.3. VCL is a very well defined architecture, however, there are some points that may become an overhead in our proposed approach. In VCL, MPRs must register to clusters to become a member of a cluster. Registration and de-registration processes causes a knowledge exchange between MPRs and RAPs. Registering – deregistering processes cause an overload in bandwidth.

VCL uses hierarchical routing. At each level of hierarchy, nodes maintain VLR database. As levels of hierarchy become greater, the size of VLR database maintained becomes greater since all the down-level topology information is kept. At the upper level, HLR is maintained that keeps all the network information. This means large sized network information tables that require large memory size.

During a registration and deregistration process, HLR and all related VLRs are updated arbitrarily, which means a significant amount of computation on database. The drawbacks of multilevel hierarchical routing techniques may appear in VCL.

We propose a new scheme; VCL based Dynamic Source Routing (VB-DSR) that uses Dynamic Source Routing over VCL structure. In this approach, we benefit the advantages of VCL. By considering the subjects described above, we make some modifications on VCL structure for the nodes those remain out of coverage area of RAPs.

Table 4.1 Comparison of some of reactive routing protocols.

	AODV	DSR	TORA
Loop-free	Yes	Yes	No
Multiple routes	No	Yes	Yes
Distributed	Yes	Yes	Yes
Reactive	Yes	Yes	Yes
Unidirectional link support	No	Yes	No
QoS Support	No	No	No
Multicast	Yes	No	No
Security	No	No	No
Power conservation	No	No	No
Periodic Broadcasts	Yes	No	No
Requires reliable or sequenced	No	No	Yes

We implement Dynamic Source Routing protocol over VCL, because a table-based routing technique will be costly for VCL. Any of the reactive routing protocols may perform well over VCL structure. The most appropriate ones are AODV, DSR and ABR. However, we have selected DSR to implement on VCL for some reasons described below. Preliminary information of DSR is given in Section 2.4.2.2 and the structure of DSR will be explained in Section 3.6. Details of DSR are in [2].

A comparison of some reactive protocols is presented in Table 3.1.

4.5. Advantages of Dynamic Source Routing

DSR protocol has several advantages when it is used alone. Moreover, there are some other advantages when used with the VCL structure. Some of the properties of DSR protocol are very appropriate for VCL, and VCL makes DSR more powerful, and makes it able to implement for large sized networks. Therefore, a new powerful scheme appears for mobile ad hoc networks, DSR over VCL.

We described the advantages of DSR below:

1. The DSR algorithm is intended for networks in which the mobiles move at a moderate speed [2]. In VCL, clusters are organized according to the mobile node characteristics providing the mobile nodes move at moderate speed with respect to each other. Therefore, VCL is appropriate for DSR.

2. DSR is proposed for small sized networks that the diameter of the network often be a small number, 5 or 10 hops (number of hops a packet from any node located at one extreme edge to another node located at the opposite extreme edge of the ad hoc network). Network must be so small that all nodes must be reachable.

   This property may become a drawback for all schemes, but not for VCL. Because of VCL structure (most of the mobile nodes connected to a RAP via single hop path), the possible routes are as DSR protocol required. So, when VCL is used with DSR, the routes become very short, which is also a requirement for mobile ad hoc networks and all proposed schemes.

3. In DSR, mobile nodes has the ability to operate network interface in promiscuous receive mode. This ability causes mobile nodes to learn network topology faster than other proposed schemes.

4. In DSR, mobile nodes insert the learned routes to their route caches. Route cache usage let the mobile nodes first to look up and search routes in their route caches before beginning route discovery process on a communication need. When a hit occurs, the route search overhead and route search delay are avoided (no delay and overhead when a hit occurs in its own cache).

   Generally, communicating pairs again communicate with each others. The called destination is generally one of the called one before. According to this approach, keeping a route cache increase the route cache hits and avoids overheads mentioned above. After a while, a mobile node learns the routes of most of its possible destination and keeps them in its route cache.

   DSR allows mobile nodes to keep multiple routes to a destination in their route caches. Hence, when a link on a route is broken, the intermediate nodes and the source node, consecutively, can check their route caches for another valid route to the destination. If such a route is found, route reconstruction does not need to be invoked avoiding bandwidth usage and power consumption. Therefore, route recovery process is faster than in many of the other reactive protocols.

5. DSR supports unidirectional links and asymmetric routes. In wireless networks, it is possible that a link between two nodes may not work equally well in both directions due to the different antenna or propagation patterns or interference. DSR allows such uni-directional links to be used when necessary, improving overall performance and network connectivity in the system.
6. DSR has an unique advantage caused by source routing. Since the route is part of the packet itself, routing loops, either short- or long-lived cannot be formed. There is no possibility of route loops. Mobile nodes do not deal with detection and resolution of route loops.
7. Since the entire source route is inserted to the transmitted data packets, the intermediate nodes do not need to keep routing information for the currently carried on calls and do not try to route any packet. They just relay the received data packet to the next address of the source route.
8. The main advantage of DSR over almost all other proposed protocols is that DSR do not make use of periodic routing advertisements, link status sensing, or neighbor detection packets. Almost in all protocols, mobile nodes send periodic messages to gather information about their neighbors to build topology knowledge and to provide connectivity. There is no reason for periodic messaging in DSR. Packets are sent only when the mobile nodes had to. Thereby, wasteful usage of bandwidth and power is avoided, which is the major drawback of all proposed schemes. On the other hand, route discovery and route maintenance processes operate entirely on-demand. By this way, network overhead is decreased. Because the packets are sent when needed, firstly, the protocol is able to react quickly to topology changes. Secondly, updates are done for only those needed cases. There is no extra overhead.

   Because of on-demand behavior and lack of periodic activities, in the case when all nodes are relatively approximately stationary (like VCL), no packet is needed to send except the ones used in finding the route and carrying data.

When all needed routes are found, just the data packets are sent. No need to sent any other type of packet, because of the relatively stationary nodes. This means zero overhead. In the case when all nodes move, DSR protocol reacts to only the routes currently in use. Network topology changes that do not affect the routes currently in use, are not taken into consideration.

9. In a battlefield, military forces want to avoid unnecessary transmissions, and they usually establish “silence” for all units to reduce detection probability. Keeping silence is very important for forces in a battlefield, because the knowledge of position, size, density, structure, movement, and current operation of the units can be easily learned by listening their transmissions. To avoid the enemy to receive such information and to avoid the detection, silence is established. None of the units transmits unless they had to do.

   DSR has the advantage of providing this requirement. Since no periodic messaging in DSR, nodes send packets only when they need to. That reduces the detection / interception probability described above, as required in tactical operations. Therefore, there is an important reason to use DSR in military purposes, as the others do not provide this.

We want to emphasize why we selected DSR as routing protocol to implement over VCL. VCL manages resources efficiently. Most of the on-demand protocols except DSR rely on periodic message dissemination. Periodic messaging increases bandwidth usage. Therefore, other protocols except DSR bring no more benefit for bandwidth usage for VCL. In DSR, there is no periodic packet dissemination. Packets are sent only when needed. Therefore, only DSR fits to our goal of avoiding unnecessary bandwidth usage.

Silence for Electronic Counter Measures (ECM) in military, is very important in a battlefield as described above. None of the reactive protocols except DSR provides this.

As a result, DSR is implemented on VCL, and named “VCL based DSR (VB-DSR)”. The structure and algorithms of VB-DSR are described in Section 4.

4.6. Dynamic Source Routing Protocol (DSR)

The Dynamic Source Routing protocol (DSR) [2] allows nodes to dynamically discover a source route across multiple network hops to any destination in the ad hoc network. Each data packet sent then carries in its header the complete, ordered list of nodes through which the packet will pass, allowing packet routing to be trivially loop-free and avoiding the need for up-to-date routing information in the intermediate nodes through which the packet is forwarded. By including this source route in the header of each data packet, other nodes forwarding or overhearing any of these packets may also easily cache this routing information for future use.

The DSR protocol is composed of two mechanisms that work together to allow the discovery and maintenance of source routes in the ad hoc network:

Route Discovery is the mechanism by which a node *S* wishing to send a packet to a destination node *D* obtains a source route to *D*. Route Discovery is used only when *S* attempts to send a packet to *D* and does not already know a route to *D*.

Route Maintenance is the mechanism by which node *S* is able to detect, while using a source route to *D*, if the network topology has changed such that it can no longer use its route to *D* because a link along the route no longer works. When Route Maintenance indicates a source route is broken, *S* can attempt to use any other route it happens to know to *D*, or can invoke Route Discovery again to find a new route for subsequent packets to *D*. Route Maintenance for this route is used only when *S* is actually sending packets to *D*.

In DSR, Route Discovery and Route Maintenance each operate entirely "on demand". In particular, unlike other protocols, DSR requires no periodic packets of any kind at any level within the network. For example, DSR does not use any periodic routing advertisement, link status sensing, or neighbor detection packets, and does not rely on these functions from any underlying protocols in the network. This entirely on-demand behavior and lack of periodic activity allows the number of overhead packets caused by DSR to scale all the way down to zero, when all nodes are approximately stationary with respect to each other and all routes needed for current communication have already been discovered. As nodes begin to move more or as communication patterns change, the routing packet overhead of DSR automatically scales to only that needed to track the routes currently in use. Network topology changes not affecting routes currently in use are ignored and do not cause reaction from the protocol.

In response to a single Route Discovery (as well as through routing information from other packets overheard), a node may learn and cache multiple routes to any destination. This allows the reaction to routing changes to be much more rapid, since a node with multiple routes to a destination can try another cached route if the one it has been using should fail. This caching of multiple routes also avoids the overhead of needing to perform a new Route Discovery each time a route in use breaks.

The operation of both Route Discovery and Route Maintenance in DSR are designed to allow uni-directional links and asymmetric routes to be easily supported to improve overall performance and network connectivity in the system.

4.6.1. DSR route discovery process

When some source node originates a new packet addressed to some destination node, the source node places in the header of the packet a source route giving the sequence of hops that the packet is to follow on its way to the destination. Normally, the sender will obtain a suitable source route by searching its "Route Cache" of routes previously learned, but if no route is found in its cache, it will initiate the Route Discovery protocol to dynamically find a new route to this destination node.

To initiate the Route Discovery, source node transmits a "Route Request" as a single local broadcast packet, which is received by (approximately) all nodes currently within wireless transmission range of source node *S*. Each Route Request identifies the initiator and target of the Route Discovery, and also contains a unique request identification, determined by the initiator of the Request. Each Route Request also contains a record listing the address of each intermediate node through which this particular copy of the Route Request has been forwarded. This route record is initialized to an empty list by the initiator of the Route Discovery.

When another node receives this Route Request, if it is the target of the Route Discovery, it returns a "Route Reply" to the initiator of the Route Discovery, giving a copy of the accumulated route record from the Route Request. When the initiator receives this Route Reply, it caches this route in its Route Cache for use in sending subsequent packets to this destination.

Otherwise, if this node receiving the Route Request has recently seen another Route Request message from this initiator bearing this same request identification and target address, or if this node's own address is already listed in the route record in the Route Request, this node discards the Request. Otherwise, this node appends its own address to the route record in the Route Request and propagates it by transmitting it as a local broadcast packet (with the same request identification).

For MAC protocols such as IEEE 802.11 that require a bi-directional frame exchange as part of the MAC protocol, this route reversal is preferred, as it avoids the overhead of a possible second Route Discovery.

When initiating a Route Discovery, the sending node saves a copy of the original packet (that triggered the Discovery) in a local buffer called the "Send Buffer". The Send Buffer contains a copy of each packet that cannot be transmitted by this node because it does not yet have a source route to the packet's destination. Each packet in the Send Buffer is logically associated with the time that it was placed into the Send Buffer and is discarded after residing in the Send Buffer for some timeout period; if necessary for preventing the Send Buffer from overflowing, a FIFO or other replacement strategy may also be used to evict packets even before they expire. While a packet remains in the Send Buffer, the node should occasionally initiate a new Route Discovery for the packet's destination address. However, the node must limit the rate at which such new Route Discoveries for the same address are initiated, since it is possible that the destination node is not currently reachable.

In order to reduce the overhead from such Route Discoveries, a node must use an exponential back-off algorithm to limit the rate at which it initiates new Route Discoveries for the same target.

4.6.2. Route maintenance process of DSR

When originating or forwarding a packet using a source route, each node transmitting the packet is responsible for confirming that the packet has been received by the next hop along the source route; the packet should be retransmitted (up to a maximum number of attempts) until this confirmation of receipt is received.

If no receipt confirmation is received after the packet has been retransmitted the maximum number of attempts by some hop, this node should return a "Route Error" to the original sender of the packet, identifying the link over which the packet could not be forwarded. Node *A* then removes this broken link from its cache. For sending such a retransmission or other packets to this same destination *E*, if *A* has in its Route Cache another route to *E* (for example, from additional Route Replies from its earlier Route Discovery, or from having overheard sufficient routing information from other packets), it can send the packet using the new route immediately. Otherwise, it should perform a new Route Discovery for this target (subject to the exponential back-off).

4.6.3. Replying to route requests using cached routes

A node receiving a Route Request for which it is not the target, searches its own Route Cache for a route to the target of the Request. If any route is found, the node generally returns a Route Reply to the initiator itself rather than forwarding the Route Request. In the Route Reply, this node sets the route record to list the sequence of hops over which this copy of the Route Request was forwarded to it, concatenated with the source route to this target obtained from its own Route Cache.

However, before transmitting a Route Reply packet that was generated using information from its Route Cache in this way, a node must verify that the resulting route being returned in the Route Reply, after this concatenation, contains no duplicate nodes listed in the route record.

If the Route Request does not meet these restrictions, the node discards the Route Request rather than replying to it or propagating it.

4.6.4. Route request hop limits

Each Route Request message contains a "hop limit" that may be used to limit the number of intermediate nodes allowed to forward that copy of the Route Request. This hop limit is implemented using the Time-to-Live (TTL) field in the IP header of the packet carrying the Route Request. As the Request is forwarded, this limit is decremented, and the Request packet is discarded if the limit reaches zero before finding the target.

A node may send its first Route Request attempt for some target node using a hop limit of 1, such that any node receiving the initial transmission of the Route Request will not forward the Request to other nodes by re-broadcasting it. This form of Route Request is called a "non-propagating" Route Request. It provides an inexpensive method for determining if the target is currently a neighbor of the initiator or if a neighbor node has a route to the target cached (effectively using the neighbors' Route Caches as an extension of the initiator's own Route Cache). If no Route Reply is received after a short timeout, then a "propagating" Route Request (i.e., with no hop limit) may be sent.

Another possible use of the hop limit in a Route Request is to implement an "expanding ring" search for the target. For example, a node could send an initial non-propagating Route Request as described above; if no Route Reply is received for it, the node could initiate another Route Request with a hop limit of 2. For each Route Request initiated, if no Route Reply is received for it, the node could double the hop limit used on the previous attempt, to progressively explore for the target node without allowing the Route Request to propagate over the entire network. However, this expanding ring search approach could have the effect of increasing the average latency of Route Discovery, since multiple Discovery attempts and timeouts may be needed before discovering a route to the target node.

4.6.5. Structure of nodes in DSR

Each node in DSR, maintains;

- A route cache; to store all routing information learned. A node adds information to its Route Cache as it learns of new links between nodes in the ad hoc network and removes information from its Route Cache as it learns that existing links in network have broken. The Route Cache supports storing more than one route to each destination. Route caches can be implemented alternatively as described in [3, 42].
- a Route Request Table; to record information about Route Requests that have been recently originated or forwarded by this node. The Route Request Table on a node records the following information about nodes to which this node has initiated a Route Request:
  - The time that this node last originated a Route Request for that target node.
  - The number of consecutive Route Requests initiated for this target since receiving a valid Route Reply giving a route to that target node.

    - The remaining amount of time before which this node may next attempt at a Route Discovery for that target node.
    - The Time-to-Live (TTL) field used in the IP header of last Route Request initiated by this node for that target node.

  In addition, the Route Request Table on a node also records a FIFO cache of size REQUEST_TABLE_IDS entries containing the Identification value and target address from the most recent Route Requests received by this node from that initiator node.

- A Send Buffer; as a queue of packets that cannot be sent by that node because it does not yet have a source route to each such packet's destination. Each packet in the Send Buffer is logically associated with the time that it was placed into the Buffer, and should be removed from the Send Buffer and silently discarded SEND_BUFFER_TIMEOUT seconds after initially being placed in the Buffer. If necessary, a FIFO strategy should be used to evict packets before they timeout to prevent the buffer from overflowing. According to the retransmission rate, a Route Discovery should be initiated as often as possible for the destination address of any packets residing in the Send Buffer.

- A Retransmission Buffer; as a queue of packets sent by this node that are awaiting the receipt of an acknowledgment from the next hop in the source route.

  For each packet in the Retransmission Buffer, a node maintains (1) a count of the number of retransmissions and (2) the time of the last retransmission. Packets are removed from the Retransmission Buffer when an acknowledgment is received or when the number of retransmissions exceeds DSR_MAXRXTSHIFT. In the later case, the removal of the packet from the Retransmission Buffer should result in a Route Error being returned to the original source of the packet.

5. PROPOSED SCHEME: VCL BASED DYNAMIC SOURCE ROUTING

5.1. The System Description

In VCL, the communication area is tessellated with virtual cells for resource planning tasks such as code, preamble code, or carrier assignment. Resource management is accomplished by RAPs or HEADMPRs (HEADMPRs are similar to the cluster-head for clustered MPRs those are not in range of any RAP, and even have no connection to an MPR registered to RAP in some way.

In VB-DSR, as in VCL, only a small number of MPRs remain out of range of all RAPs. Opposite of VCL, these MPRs do not attempt to organize into a cluster. They act in pure ad hoc manner. Its reason is that when one of gathered MPRs had a neighbor MPR that has a connection to a RAP over a single hop path (it is in the range of RAP) or multi-hop path (it is not in the range of RAP, but has neighbor MPR that has a connection to a RAP over a single-hop or multi-hop path), all these MPRs will already have been connected to RAP via this neighbor MPR. The remaining processes are the same in VCL. However, by this way, the cumbersome of HEADMPR is prevented and many processes for selecting a HEADMPR (cluster-head) are avoided since no HEADMPR is used. This also avoids many state transitions of MPRs and the caused overload of them as described in Section 3.3.

Since there is no HEADMPR in VB-DSR, no MPR is responsible for resource management, even if they have no connection to a RAP anyway. Only RAPs are responsible for resource management.

In VB-DSR, RAPs will use the resource of the virtual cells as in VCL. Therefore, RAPs will require their geographic location information to locate themselves in virtual cell layout. This gives the advantage of that resources will be allocated to RAPs dynamically, which is an important expected behavior of ad hoc networks.

In our thesis, we only work on RAP and MPR tiers. Studies of higher tiers such as UAVT and SATT, are left as future work. In RAP and MPR tiers, we have two basic types of nodes, namely RAPs and MPRs. In the following sections, we define the algorithms and approaches used by these components, and some possible cases.

5.2. Radio Access Points

RAPs have the same functionality and similar responsibilities as in VCL, except that no VLR is kept and no registration /de-registration process is done. The states of RAPs and their procedures are almost the same as in VCL with a few modulations because of DSR protocol used in VB-DSR (Figure 4.1).

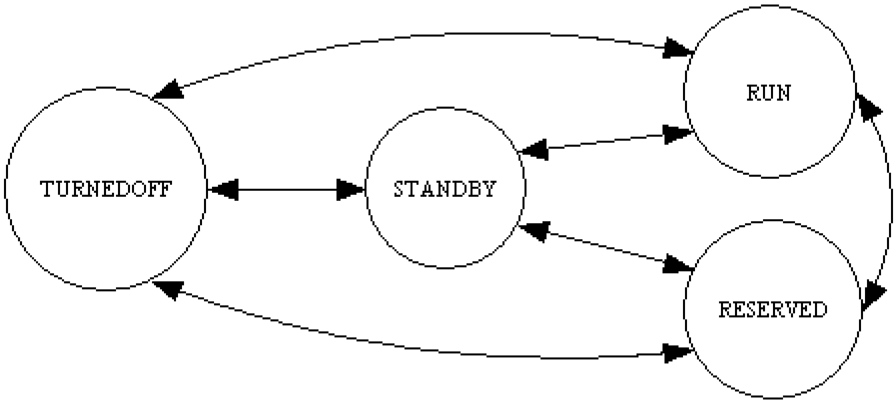

Figure 5.1 State transition diagram of RAP [3].

5.3. Man Packed Radios

As mentioned before, all the components of VB-DSR are the same as in VCL. They have the same abilities. MPRs in VB-DSR communicate with each other through the RAPs while they were in the range of RAPs. Properties of MPRs are the same as in VCL. The only difference is that they do not need to act as a base station because of the VB-DSR approach.

An MPR can relay communications of other MPRs as expected in ad hoc approach. “By ad hoc approach what we mean is not an approach which ensures that each component has a communication path with each other component in the network. Our approach is a fully distributed one, which is devised to make the components communicate terrestrially even if they do not have an access to a RAP. As we stated before, we envisioned a multitier system in which UAVs and satellites provide umbrella cells. However, we prefer to communicate terrestrially by using multi-hop approaches compared to communicating through UAVT and SATT cells. Hence, we lower the emissions and power consumption, and our design becomes more robust by avoiding from the usage of more vulnerable components, namely satellites and UAVs. [3]"

MPRs use the procedures related to their current status. They can work distributedly without a central authority. The basic design principle is having a system, in which two MPRs can communicate with each other even if they cannot communicate with anybody else.

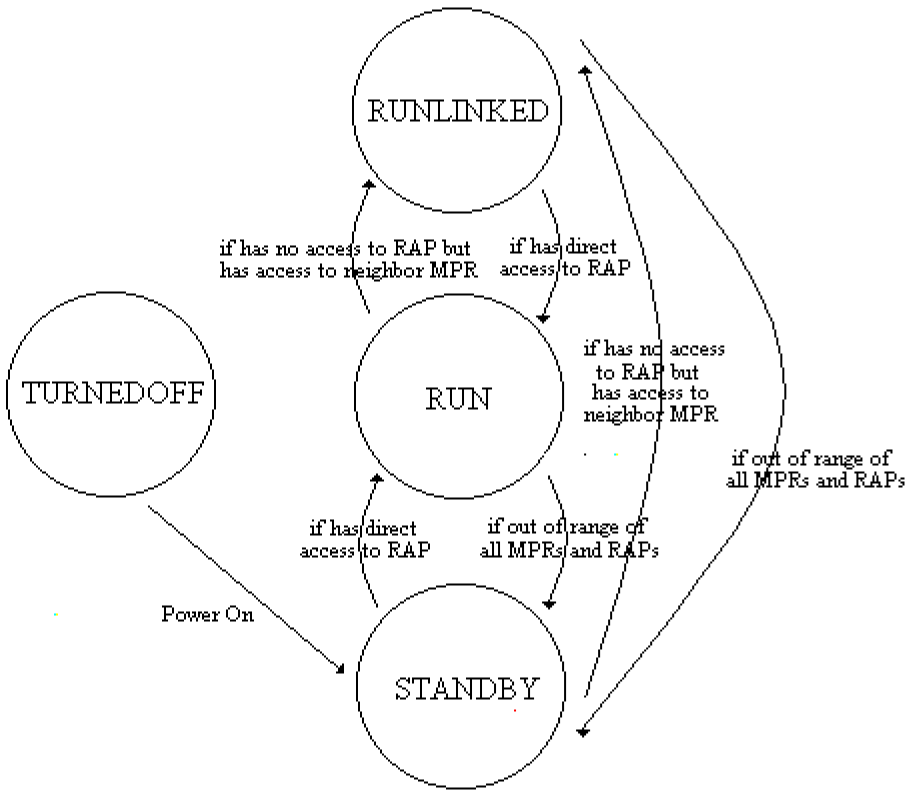

Figure 5.2 State transition diagram of MPR.

The state transition diagram of an MPR is illustrated in Figure 4.2. When an MPR can communicate with an access point, it behaves like a terminal equipment in a cellular network. The MPRs that cannot access to RAPs anyway, communicate each other by ad hoc approach.

There are five states of MPRs and in all states the same transition algorithms are used very simple than VCL. TURNEDOFF and STANDBY states are initial states. The other states are the ones an MPR can be in while operating.

When an MPR powers on, it changes its status to STANDBY. In all states - STANDBY, RUN and RUNLINKED– the same status checking procedure is executed by MPRs illustrated in Figure 4.3.

```
if a primary SYNC channel is detected
    Accomplish slot synchronization with the strongest primary SYNC channel.
    Demodulate the secondary SYNC channel to learn the code group.
    Carry an exhaustive search on the code group to learn which code is used
      as the scrambling code by the synchronized RAP.
    Change status to RUN.
else if a transmission is detected on contention-based channel
    Change status to RUNLINKED
else
    Change status to STANDBY.
```

Figure 5.3 State transition algorithm of MPR.

5.4. Multiple Access (MA) Schemes

VCL has a cellular structure. In cellular networks, mostly, time Division Multiple Access (TDMA), Frequency Division Multiple Access (FDMA) and Code Division Multiple Access (CDMA) techniques are used as multiple access schemes. These schemes require an access point for resource management. TDMA also requires time synchronization. FDMA has low utilization with respect to other schemes. In VCL, therefore, CDMA is used to achieve a good resource management.

Radio Access Points (RAPs) broadcast the access channel information to the mobile terminals (MPRs) in the range of them. The MPRs not in the range of RAPs, try to access RAPs indirectly via relay MPRs, if they can. The MPRs which cannot access RAPs directly or indirectly, organize into clusters. The cluster-heads are then charged as access points. Therefore, all MPRs which access RAPs or cluster-head MPRs, are able to use CDMA technique.

In VB-DSR, MPRs access RAPs if and only if they were in the range of RAPs. Therefore, CDMA technique is used only to access RAPs within coverage area of RAPs. Different from VCL, MPRs out of range of RAPs, do not attempt to organize into clusters. They act in a distributed manner. Therefore, CDMA technique cannot be used for these MPRs. FDMA, TDMA, PRMA, RAMA techniques also cannot be employed because they require an access point and/or time synchronization.

In a distributed environment, we can employ contention-based techniques. In contention-based protocols, the transmission is not guaranteed to be successful. Whenever two or more users transmit on the shared channel simultaneously, a collision occurs and the packets cannot be received correctly. Aloha protocols are the simplest protocols to employ. Generated packets are transmitted immediately without checking the transmission medium, hoping for no interference by others. On collision, every colliding user retransmits their packets after an independent random delay. Achieving a successful transmission may take a long time on collisions because of random delays. To reduce the collisions, thereby, to reduce the successful packet transmission latency, Slotted Aloha can be employed. However, slotted aloha requires time synchronization. Besides, Aloha Type techniques cannot be employed in a battlefield because of bursty traffic. In a bursty traffic, collisions increase, therefore the successful packet transmission latencies relatively more increase.

Hence, there remains only carrier sensing protocols. Carrier sensing protocols have better performance than Aloha type protocols, because they listen to the transmission medium before transmission attempt. By this way, collisions are decreased with respect to the Aloha type protocols.

We employed CSMA/CA technique for MPRs those have not access RAPs. CSMA/CA has better performance than other carrier sensing protocols because of collision avoidance method. However, it does not relieve us from collisions.

Carrier sensing protocols perform well in Ethernet technologies, and can be used in small sized wireless LANs, where the distances between nodes remain short (less than 100m.). As the distances or the number of nodes in the network increase, the collisions increase. Moreover, in our case, burty traffic in battlefield reduces the performance of the carrier sensing protocols. However, we used CSMA/CA technique for some acceptable reasons. MPRs those have not access RAPs remains very small in number. The statistics are given in Section 5.3.1. Therefore, CSMA/CA techniques used for a small number of nodes those remain in short distances between themselves. On the other hand, DSR protocol avoids unnecessary transmissions preventing the possible collisions.

5.5. Call Management

We used the call model depicted in VCL [3]. However, there is a slight difference in the call breaking phase. In VB-DSR, on call breakings, the source node continuously attempts to construct a valid route to destination. A call is terminated only when there is no valid route anymore to the destination.

5.6. Behavior of the System

In proposed approach (VB-DSR), there are two main components, RAPs and MPRs. MPRs those are under coverage area of a RAP send their packets via RAP, do not propagate the packets. MPRs those are not under the coverage area of any RAP, can send their packets to neighbors (propagate the packets) and use pure DSR. Because of these, a node can be in one of these three states while operating (The states are named so that they resemble the states in VCL) (Figure 4.4):

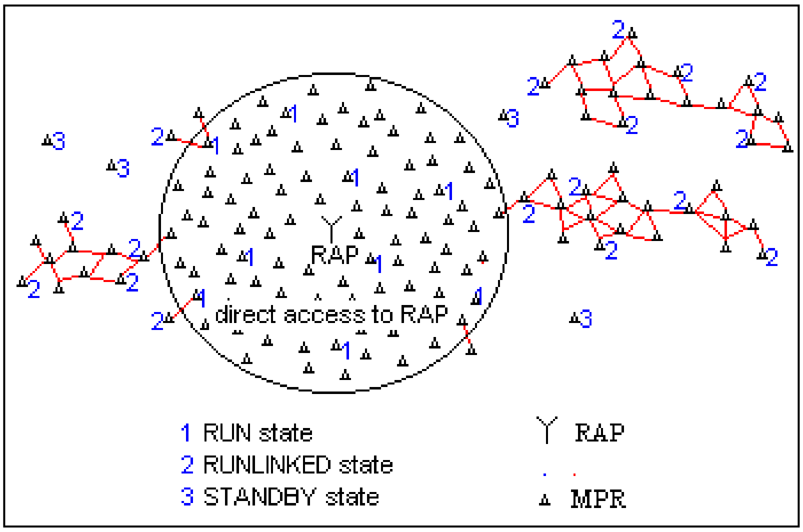

Figure 5.4 Network topology.

- in state RUN : an MPR is under the coverage area of a RAP.
- in state RUNLINKED : an MPR is not under the coverage area of any RAP but has neighbor MPRs.
- in state STANDBY : an MPR is not under the coverage area of any RAP nor has any neighbors.

There are six type of packets in DSR protocol. One of them is the packet that carries communication data. The others are the control packets used in discovering and maintaining routes, etc. Packet types used in VB-DSR is defined in Section 4.7.

RAPs use the downlink and uplink communication channels with MPRs (Figure 4.5). The MPRs in the coverage are of a RAP send their control packets on uplink channel, and RAPs send their control packets on downlink channel to the MPRs in the coverage area of itself. When communication is established (route is found), MPRs send their packets on data channel.

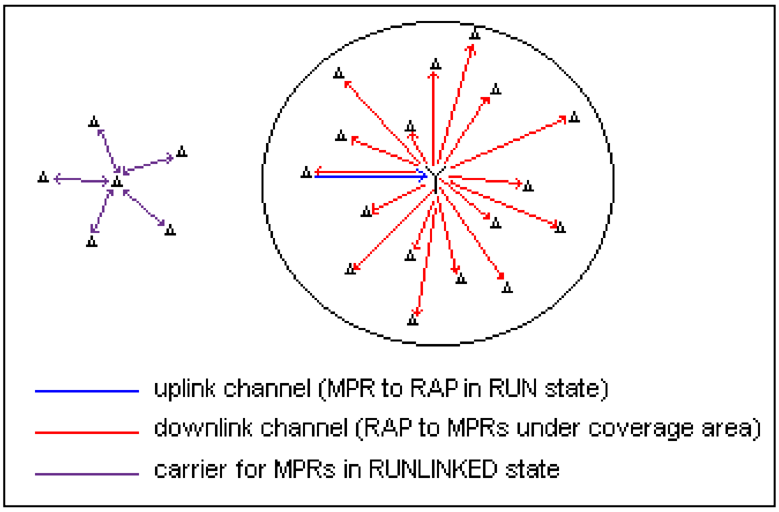

Figure 5.5 Carriers used in VB-DSR.

The MPRs not in the coverage area of any RAP use only one communication channel. All packets (both data and control packets) is sent on this channel (Figure 4.5).

5.7. Packet Types in VB-DSR

We used the packet types of DSR protocol in VB-DSR. These packets are:

- Route Request Packet; generated by the initiator of the call and is sent to discover a route to destination.
- Route Reply Packet; generated by either the destination node or intermediate nodes if they have a route to destination as a reply to the route request packets. Route reply packets follow the reverse route of the accumulated route of the received route request packet.
- Route Error Packet; generated by the nodes that experience error to send data packets to the next node. It is sent to the source node to inform the source and on-route nodes about the error experienced.

- Data Packets; generated by both the source and destination nodes and is sent over the constructed route. They carry the data of the conversation.

Route Acknowledgement and Acknowledgement Request packets dissemination is not implemented. In VB-DSR, RAPs act as base stations, so CDMA is used in coverage area of RAPs. We assume that acknowledgement requests and acknowledgements are sent on Forward Paging Channel and Reverse Access Channel of the RAPs. Outside of the RAPs, because there are a small number MPR, we assumed that they use passive acknowledgement technique for acknowledgement process.

Table 5.1 Packet lengths in VB-DSR.

	Fixed Part (byte)	**Option Part**	**Total (byte)**
Route Request Packet	4	8 + (4 x n)	12 + (4 x n)
Route Reply Packet	4	12 + (4 x n)	16 + (4 x n)
Route Error Packet	4	16	20
Data Packet	4	8 + (4 x n)	12 + (4 x n) + PAYLOAD = 512

In addition to these packet types, we used two more packet types in VB-DSR. They are:

- Busy Packets; generated by the destination on a call request and is sent to the source to refuse its call request. It follows the reverse of the accumulated route of the route request packet. In fact, Busy packets are the same as route reply packets. The only difference is they refuse the call request. Therefore, the destination must change only the option type of the reply packet from REPLY to BUSY. The whole packet is the same as route reply packet.

- Hello Packets; generated by RUNLINKED state MPRs if they cannot receive any packet within a time interval or if they enter a new VCL cell. It is broadcasted as described in Section 4.9. In fact, hello packets are the same as route request packets. The difference is that their TTL value is predetermined and no route reply packet is sent as a reply to a hello packet. Their mission is to inform neighbors about the originator’s existence. Therefore, the originator must only set the option type of route request packet to HELLO.

The length of the used packets are the same as in DSR and is presented in Table 4.1. n is the number of addresses inserted in the address part of the packets' header. In DSR, each address is 4 byte long. Therefore, addresses are multiplied with 4 in Table 4.1.

5.8. Route Discovery Process

Route discovery in VB-DSR is consisted of two phases as in DSR: (1) Non-propagating route request phase, (2) propagating route request phase. Therefore, we calculated the latency according to these route discovery phases. Route recovery phase is not calculated since route recovery is not a function of VB-DSR. We use the first replies of route requests to calculate the latency. DSR assumes that the first route reply is received from the shortest route.

The latency between the transmission of the route request packet and the arrival of route reply packet includes propagation delay, queuing delay and retransmission delay. Hence, the latency is affected by some controllable parameters such as buffer size and buffer update interval, selected medium access scheme, and by some uncontrollable parameters such transmission rate. Latency increases as the number of hops or packet size increases. We assume that the latency for any packet between two RAPs is 2 ms., because we assumed that they can communicate with each other with high rates such as bursty traffic.

We classify the route replies as:

- ***Neighbor replies:*** In route discovery phase, the source node first sends a route request packet with a maximum propagation hop limit one (TTL=1). The neighbors received this packet do not propagate it. Any of them returns route reply packet if the destination is itself or if it has a route to the destination, else discards the packet.
- ***Cache replies:*** These route replies are produced by the nodes between the source and the destination, i.e., intermediate nodes. The route is constructed with the route information in the route cache of the intermediate node and the accumulated route of the received route request packet.
- ***Destination replies:*** These are the route replies sent by the destination. They are the most reliable routes since they indicate the most up-to-date topology.

The route reply statistics and the latencies occurred in route search processes are calculated according to this classification.

At a call request, an MPR in the coverage area of a RAP first checks its route cache to find a route to the destination. If it has, it uses that route without initiating route search process. If it cannot find a route to the destination, it sends a route search packet addressed to the attached RAP. The source MPR iterates route search process while the route is not found and the iteration number remains below a predefined value. At iterations, time to live value (TTL) of the route search packet and reply waiting time of route request table are exponentially incremented. Therefore, a controlled route search process is executed avoiding message flooding throughout the network. The RAP receiving the route search packet searches its route cache. If it finds a route, it sends back a route reply packet addressed only to the source MPR, else broadcast the search packet to other RAPs and to all MPRs under its coverage area. Other RAPs receiving the packet searches their route caches, if one of them finds a route to destination, that one sends a route reply back to the source MPR by using the accumulated route on search packet.

The RAPs those have not a route to destination broadcast the search packet to the MPRs under their coverage area. Hence, all the MPRs under the coverage area of a RAP will receive the search packet. Since almost all MPRs are under the coverage area of RAPs, or at least one of them has a path to destination, the destination is found.

5.8.1. Possible cases for the network components

5.8.1.1. Case 1 (simple case):

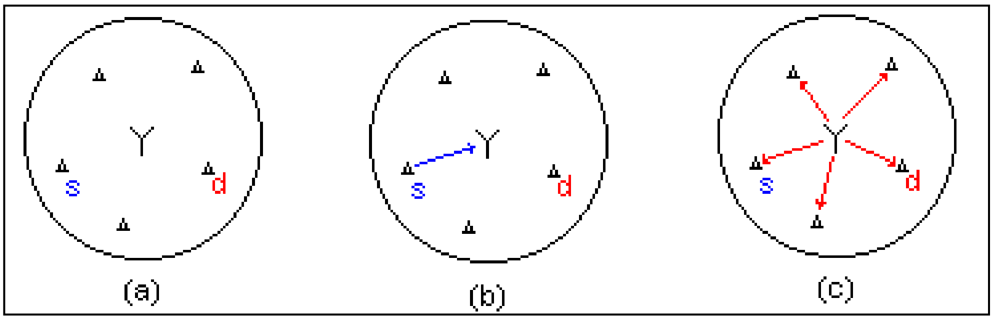

Figure 5.6 Route search for case 1.

(a) Source *s* and the destination *d* MPRs are under the same RAP's coverage area (Figure 4.6).

(b) Source needs to communicate with the destination. First, it checks its route cache. We assume that no route is found to *d*. So, it prepares a route request packet and sends it to RAP on uplink channel.

(c) RAP receives the packet and makes controls (whether received the packet before or not, route cache check, route request table check, etc.), founds no route to *d*. (in VB-DSR, MPRs do not register to RAP. By this way, on demand is preserved and needless registry updates are avoided.)

i. Inserts the accumulated address of the received route request packet to its (RAP's) route cache.

ii. Sends (propagates by broadcasting) the route request packet (to all MPRs). (RAP transmits the packet by broadcasting, so all nodes in coverage area of it receive the packet. But the packet's destination address is MPR *d*. Since all MPRs those receiving this packet are in the coverage area of RAP (so received this packet) and since this packet is sent by RAP (on downlink channel and the last address inserted to accumulated address part of the packet's header is the RAP's address), they check the destination address, and all of them except node *d* discards the packet. But all nodes insert the accumulated address to their route caches.)

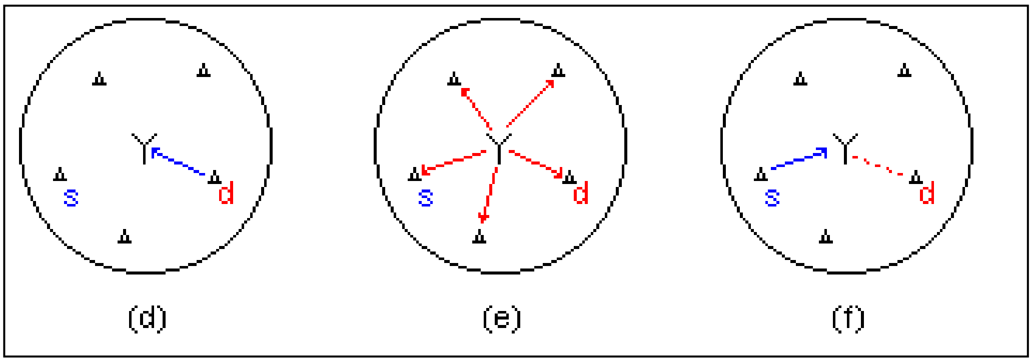

Figure 5.7 Route reply for case 1.

(d) MPR *d* sends route reply packet. It just reverses the accumulated route that came with the route request packet. And sends its route reply packet by using this route (Figure 4.7).

(e) RAP receives the route reply packet.

i. Caches the accumulated address.

ii. Sends the route reply packet to the next address defined on the route reply packet address part (accumulated and reversed route).

(f) All the nodes under the coverage area of RAP receive the packet.

i. All of them insert the route of route reply packet to their route caches.

ii. All of them, except node *s*, discard the packet since the route reply packet is not addressed to them.

iii. Node *s* receives the packet. If not received any other route reply packet previously, uses this route for communication and begins sending data carrying packets.

The found route is: node *s* – RAP – node *d*. Used channels are uplink channel and downlink channel = 2 transmissions for finding the route. For finding the route (to distribute route request packet), 2 transmissions is needed. In addition, for route reply, 2 transmissions is needed. So, for all control packets, only 4 transmissions is needed. The rest of them are to send the data packets. No more control packet is needed to send while a route error will not occur.

5.8.1.2. Case 2

Communicating MPRs are in different RAPs coverage area (Figure 4.8).

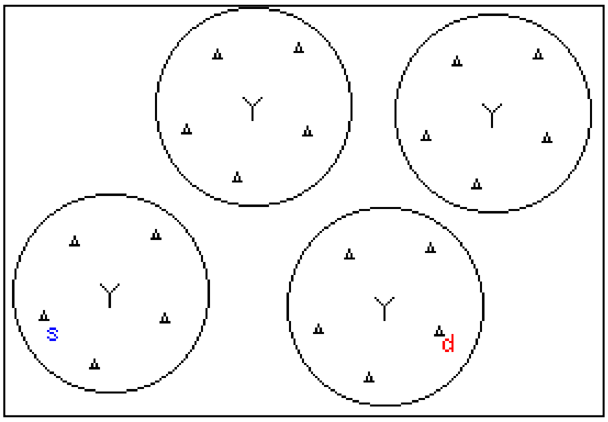

Figure 5.8 Case 2.

All steps are the same with Case 1 except step (c).

(In fact, step (c) is the same with Case 1 step (c) in Section 4.8.1.1. To make it simple and understandable, this part is not written in Case 1.

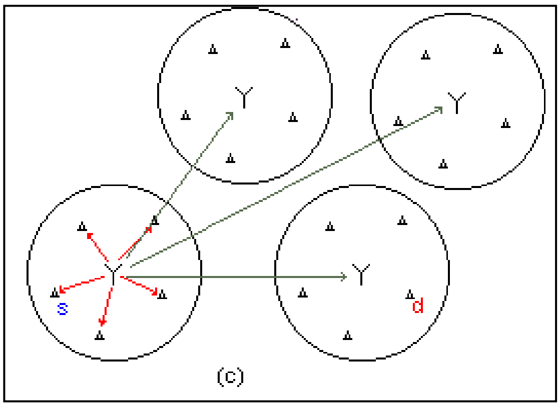

Figure 5.9 Case 2, dissemination of route request packet.

When the RAP, that the MPR *s* is in the coverage area, receives the route request packet, it sends the packet to all other RAPs and to all MPRs those in its coverage area (Figure 4.9). The other RAPs those receive this route request packet also sends the packet to the MPRs in their coverage area. Therefore, all MPRs and RAPs receive this route request packet (this also happen in case 1, but not written for simplicity).

5.8.1.3. Case 3

One of the communicating MPRs is under the coverage area of a RAP, the other is not. We can consider it in two sub cases.

<u>*Case 3-a :*</u> Source (s) is not under coverage area of any RAP, but destination (d) is (Figure 4.10).

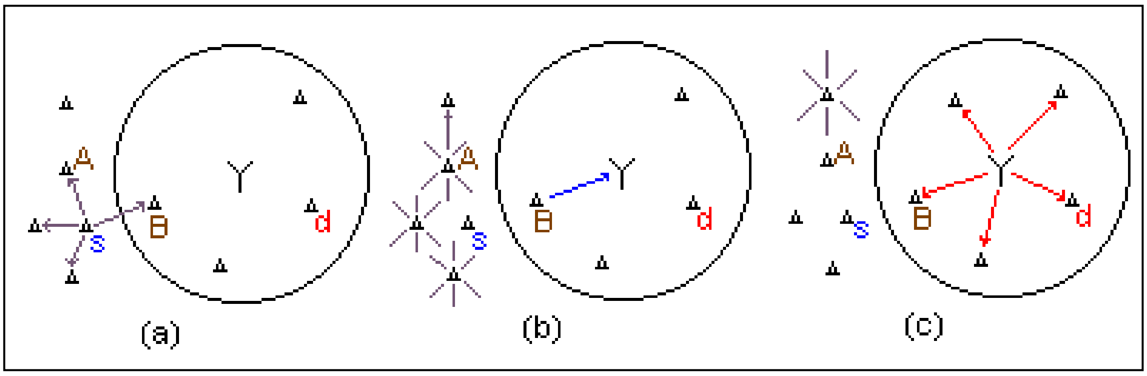

Figure 5.10 Route search process for case 3-a.

(a) Since source cannot hear any transmission of any RAP (it is not in the coverage area of any RAP), it knows its own states as RUNLINKED (but hear transmissions of neighbor MPRs). So, it uses the carrier defined for RUNLINKED MPRs. It sends the route request packets to its neighbors (broadcasts – addressed to all neighbors)(it may have no info about its neighbors if it didn't receive any packet from neighbors).

(b) All the neighbors receive the packet and process it.

i. Each of them looks at the destination address, searches its route cache, and tries to find any existing route to the destination (if the destination is not itself, as in Figure 4.10).

ii. If no route is found, they propagate it:

- If the neighbor MPR (e.g. MPR *A*) is not in the coverage area of any RAP (in RUNLINKED state), it propagates the packet as in step (a).
- Else, if the neighbor MPR (e.g. MPR *B*) is in the coverage area of a RAP (in RUN state), it sends the route request packet only to RAP (addressed only to the RAP being under coverage area) via uplink channel.

(c) This step is the same as in Case 1.

(d) This step is the same as in Case 1.

(e) This step is the same as in Case 1.

(f) The route reply packet follows the route inserted into the header. All the MPRs which are on the route receive the route reply packet. MPR *B* just forwards the packet to the next node in the route. It knows the existence of that node, because it received the route request packet from it. Since next node (or progressive nodes) in source route is *s* (source), it begins to send data carrying packets by using this route if it does not receive any route reply before.

Case 3-b:

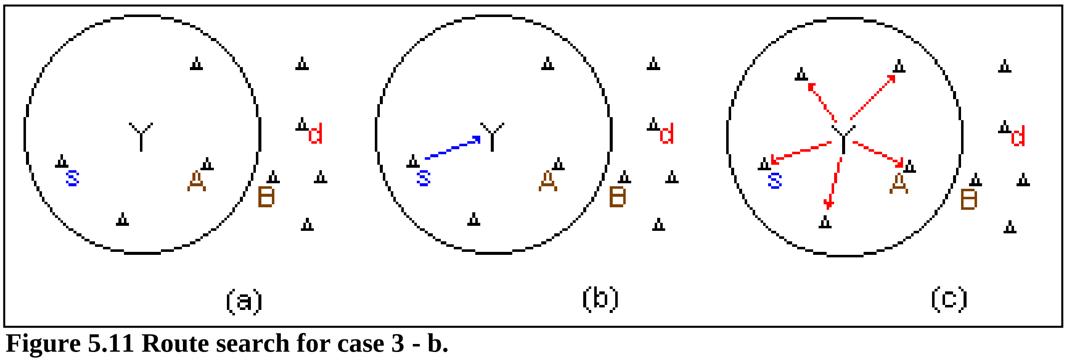

Figure 5.11 Route search for case 3 - b.

(a) Source *s* is in the coverage area of a RAP, but the destination *d* is not (Figure 4.11).

(b) Source prepares a route request packet and sends it to RAP.

(c) RAP receives the packet and broadcasts it (addressing to all nodes with the destination address is MPR *d*) as in previous cases.

(d) All the MPRs in the coverage area of the RAP receive the packet. They examine the destination address, and look up their route caches for a route to destination (as in previous cases). All the nodes, except node *A*, discard the route request packet. Node *A* prepares a route reply packet by concatenating the accumulated route of the received route request packet with the route found in its route cache to node *d*. It has a route to node *d*, because it received any kind of packet from node *d* over single hop or multi-hop path (Figure 4.12).

(e) MPR *A* sends the route reply packet back to the MPR *s* over reverse path inserted in route reply packet.

The other steps are as previous cases.

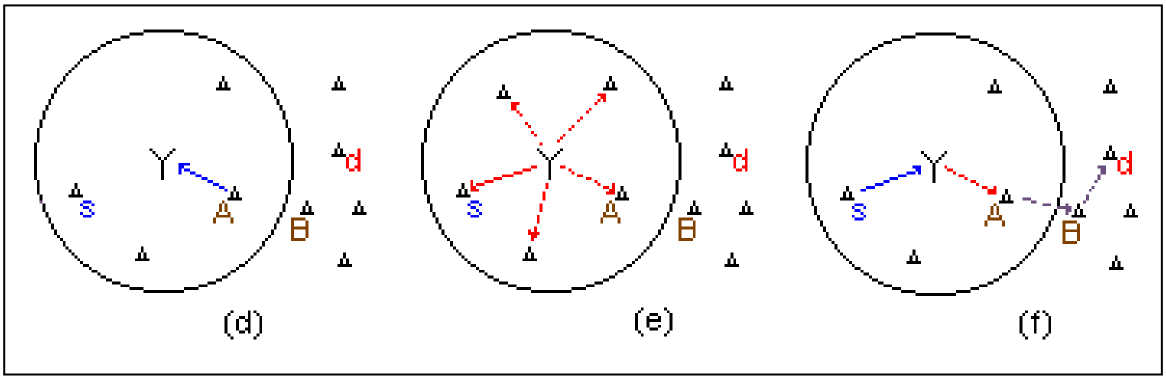

Figure 5.12 Route reply process for case 3–b (MPR A has a route to destination d in its route cache).

The worst case in VB-DSR is case 3-b. In given case above, we assume that the MPR *A* has information about node d in its route cache, because it previously received a packet from MPR *d* via a single hop or multi-hop path. In VB-DSR, the nodes do not send periodic messages. They send messages only when they need to. MPRs in RUNLINKED state, cannot send any packet unless they get any route request packet from any of their neighbors or unless they have a communication need. In such a case, none of them will be aware of each other. Awareness of each other happens only when one of them needs to communicate and sends a route request packet by broadcasting. All the neighbors those are in RUNLINKED state rebroadcast the route request packet; therefore, all the MPRs those are in RUNLINKED state receive the packet.

In case 3-b, if no transmission is received from the RUNLINKED MPRs or from MPR *d*, MPR *A* will not have a path to MPR *d*. In this case, MPR *A* will discard the route request packet addressed to MPR *d*, as the other MPRs in RUN state do. Therefore, the route from source MPR *s* to destination MPR *d* cannot be found because of this extreme situation. To avoid this extreme situation, we employed the approaches described in Section 4.9.

5.8.1.4. Case 4

Source and the destination is in RUNLINKED states (they are not in the coverage area of a RAP or RAPs) and do not have a path to each other (but may have via a RAP) (Figure 4.13).

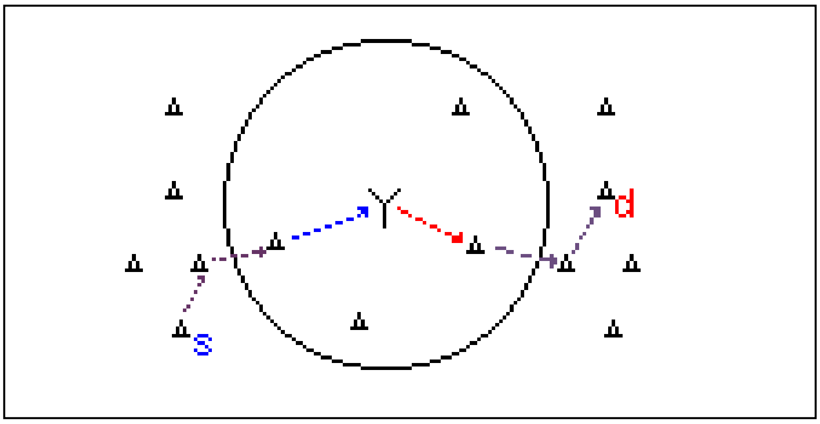

Figure 5.13 Flow of the packets for case 4.

This case is the combination of Case 3 – a and Case 3 – b. Therefore, route search and route reply processes are executed as in these cases.

5.8.1.5. Case 5

Source or destination or both of them are in RUNLINKED state but have no route to each other (but a path can be found by multiple RAPs (Figure 4.14)).

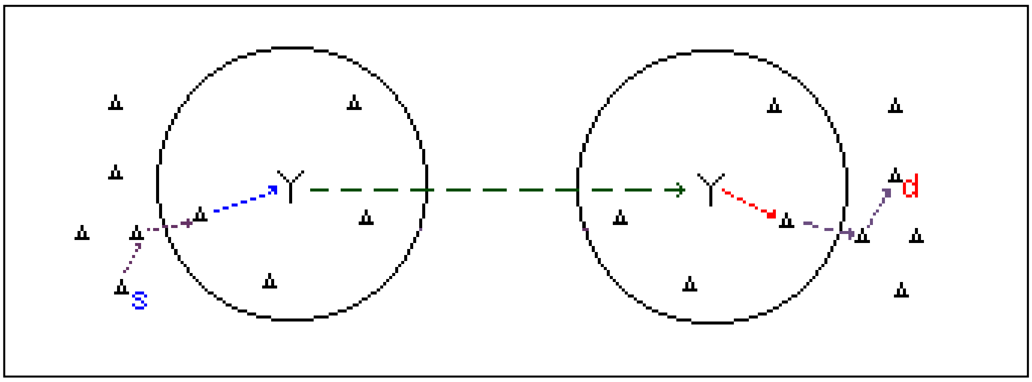

Figure 5.14 Flow of the packets for case 5.

This case is the combination of Case 2 and Case 4. Therefore, route search and route reply processes are executed as in these cases.

5.8.1.6. Case 6

Both source and destination are in RUNLINKED state and may have a path without a RAP (Figure 4.15).

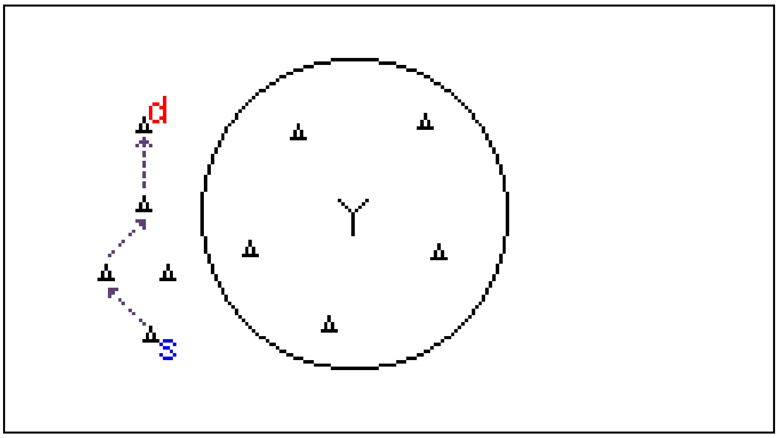

Figure 5.15 Flow of the packets for case 6.

In this case, pure DSR is used in route search and route maintenance. Route search is done as described in Case 3 – a step (a) and (b). When route is found, route reply is sent back to the source.

5.8.1.7. Overview of cases

In all cases, the found route is not more than a few hop. Since RAPs are involved in search processes, a route search packet is disseminated to all RAPs' coverage area. Therefore, all MPRs in the coverage area of a RAP will be aware of both the MPR requesting the call and the MPRs accumulated in the address part of the packet header. In addition, each MPR will be aware of which RAP it is under coverage area.

Since the MPRs under coverage area of a RAP do not propagate the route search messages between themselves (they address the route request packets only to the RAPs), the overhead in DSR is avoided. Since the RAP propagates the route search messages to all MPRs in its coverage area and to all other RAPs, all the MPRs those are in the coverage area of any RAP, receive the message and be aware of the RAP and the MPR (or RAPs and MPRs) because of the accumulated addresses in the header part of the message.

If the source and destination MPRs are under coverage area of a RAP (or different RAPs)(Figure 4.16 (a) and (b)), the route is found at most 2 or 3 transmissions (2 for the same RAP, 3 for different RAPs); 1 transmission from source to RAP, (1 transmission from RAP to RAP), 1 transmission from RAP to destination.

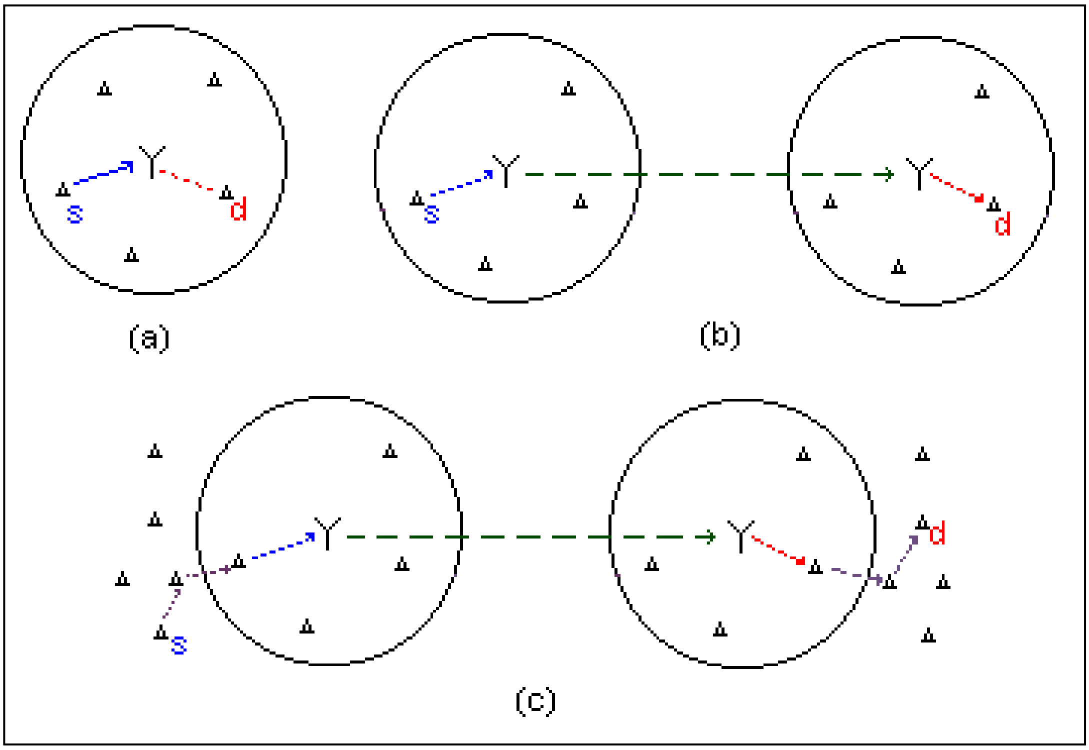

Figure 5.16 Hop counts for different cases.

In the worst case (Figure 4.16 (c)), the found route hop count is; 2 (+1(for different RAPs)) + count of the links between RUNLINKED MPRs from source to the MPR that is in the coverage area of the RAP + count of links between RUNLINKED MPRs from destination to the MPR that is in the coverage area of the RAP. Because we use VCL, all MPRs except a small number of MPRs, are in the coverage area of a

RAP. There remain a small number of MPRs in RUNLINKED state. This means, a RUNLINKED MPR can reach to a RUN state MPR with one or two hops.

5.9. Route Cache Updating Process

In the worst case, as described in Section 4.8.1.3, a destination may not be found, though it can be. To avoid such a situation, we employed Route Cache Update Process for RUNLINKED MPRs. In this process, RUNLINKED MPRs transmit hello packets to their neighbor nodes on occurrence of one of two cases. By this way, packet dissemination is executed for only a small number of nodes by avoiding the possible overhead.

In the first case, Route Cache Update Process is executed on time basis. If a node that is out of coverage area of RAPs, does not receive any type of packet from other nodes, or does not receive any access information from any RAP within a predefined time interval, then it broadcasts a hello packet to inform its neighbors about its existence.

In the second case, Route Cache Update Process is executed on location change. An MPR in battlefield shows the same mobility pattern with the others. However, there may be some distinctions. A node may keep its position while the others move, or the opposite, it may move continuously while the others keep their locations or move slowly. In this case, this node's neighbor list and its route cache must be updated, while the neighbor nodes' route caches must also be updated. To do so, nodes entering a new VCL cell, inform its neighbors by broadcasting a hello packet.

Hello packets are processed like Route Request packets. Sender inserts its address to the address part of the header of the hello packet. By this way, receiving nodes gets entire route information of the packet passed through. Only RUNLINKED state MPRs can rebroadcast the received hello packets. When a RUN state MPR receives a hello packet, it just updates its route cache.

6. PERFORMANCE EVALUATION OF THE PROPOSED SYSTEM

In order to evaluate the performance of the proposed system, Computer Aided Exercises Interacted Tactical Communications Simulation (CITACS) [5] is used. CITACS is a simulation system developed in the Network Laboratory (NETLAB) at Bogazici University. CITACS interacts with real computer aided military exercises to obtain data related to the movement and the posture of the military units. We also enhance the simulation approach by implementing the dynamic source routing protocol and the CSMA/CA access scheme for ad hoc nodes., and use it for the evaluation of tactical communication systems. In this approach, the commands entered during the military computer aided exercises are replayed by running a constructive (combat) model which generates mobility, posture and status data for a number of units, then these data are enhanced and drive a simulation which produces the data related to the performance metrics. This data is used to generate the mobility and call patterns.

6.1. Translator

Translator interacts with Joint Theater Level Simulation (JTLS), converts the results of the wargame run with JTLS into the format defined for the implemented simulation system. Each tuple in the database has nine fields:

- Unit name: this field represents nine character long unit names.
- Unit type: units are classified into 10 broad categories; non-applicable (an entity which is not a unit), headquarter, infantry, artillery, armor, special force, squadron, support unit, signal unit, others.
- Unit size: unit sizes are classified into 10 categories; squad, section, platoon, company, battalion, regiment, brigade, division, headquarter, others.
- Latitudes and longitudes: this is the geographic location of the units in degrees. The accuracy is up to second level.

- Posture: unit postures are classified into 10 categories; attack, defend, delay-withdraw, move, air operations, amphibious, formation, incapable, inactive, wiped out.
- Current power: this field indicates the current power to full power ratio.
- Direction: this is the direction that the unit is facing. Six directions are used; north, north-east, north-west, south, south-east, south-west.
- In-combat: whether the unit is in combat or not in combat information is recorded into this field.

There is a separate record related to each unit for each minute in this database. The simulation time information is recorded in separator records with minutes resolution. Between two time records, the records related to each unit for that minute is recorded.

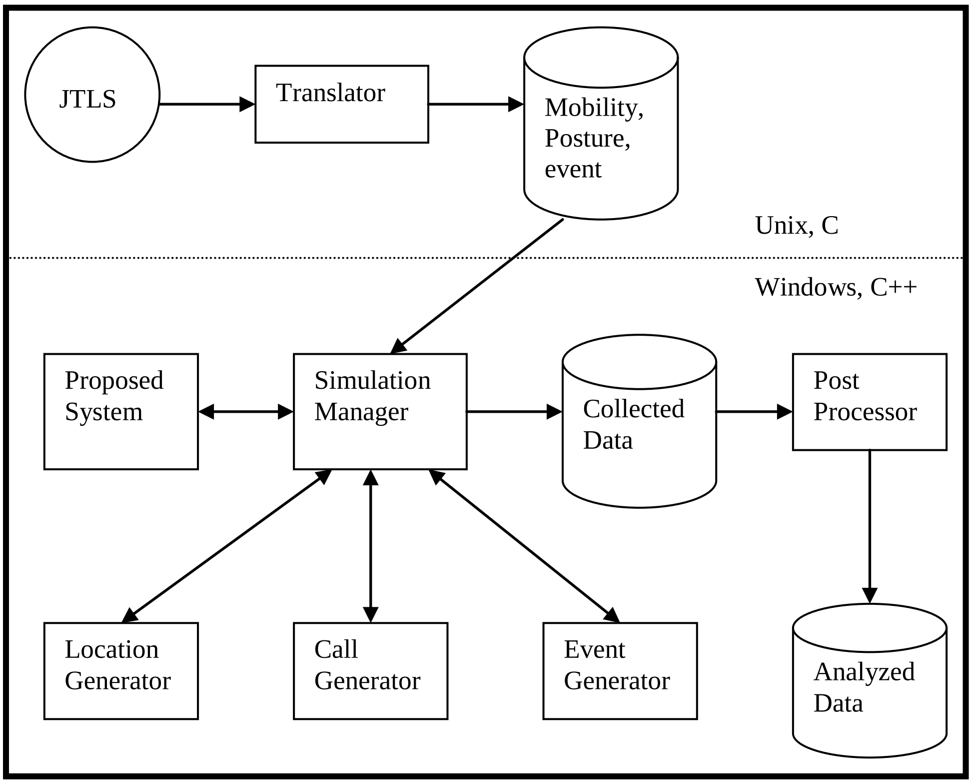

Figure 6.1 The layout of the developed simulation system [3].

6.2. Simulation Manager, Location, Call and Event Generators

The block diagram of Simulation Manager is illustrated in Figure 5.1. It reads the records related to each unit minute by minute from the database created by the translator. After reading the data of the next minute, the speeds and the lost power of units in the next minute is calculated, and the simulation is run until reaching the time read.

Since the resolution of data created by translator is in unit size and in minutes, this resolution should be enhanced up to the radio level and seconds. This is done by the location manager.

At the initial phase, the generic unit organization illustrated in Figure 5.2 is used to determine the number of radios within a unit. It is assumed in [3] that each unit has one radio for the Commanding Officer (CO), four radios for the headquarter, and four subordinate units which has the same organization. This standard organization is kept down to the squad level in which we have three radios in total. These radios are deployed uniformly to the area whose center is determined from the translator database. The size of different units under different postures is given in Table 5.1. These values are taken from [3] and were determined according to the generic organization. If a unit has a RAP, the RAP is located randomly with uniform distribution within a circle whose center is the center is the center of the unit. The radius of this circle is equal to 10 per cent of the unit front size.

Table 6.1 The front and depths of the simulated generic units (in meters) [3].

Unit[1]	**Bran.**	**Att.**		**Def.**		**With.**		**Move**		**Amp.**		**Other**	
		Front	Dept	Front	Dept	Front	Dept	Front	Dept	Front	Dept	Front	Dept
Co	Inf.	1000	500	1500	1000	2000	2000	400	4000	1000	500	1500	1000
	Art.	500	250	500	250	500	250	500	250	500	250	500	250
	Tank	1000	500	1500	1000	2000	2000	400	400	1000	500	1500	1000
	SOF	1000	500	1500	1000	2000	2000	400	400	1000	500	1500	1000
	Sup.	500	250	500	250	500	250	500	250	500	250	500	250
	Hq.	50	50	50	50	50	50	50	50	50	50	50	50
Bat	Inf.	1500	1000	3000	2500	6000	5000	400	2000	1500	1000	3000	2500
	Art.	3000	1000	3000	1000	3000	1000	3000	1000	3000	1000	3000	1000
	Tank	1500	1000	3000	2500	6000	5000	400	2000	1500	1000	3000	2500
	SOF	1500	1000	3000	2500	6000	5000	400	2000	1500	1000	3000	2500
	Sup.	1500	1000	1500	1000	1500	1000	1500	1000	1500	1000	1500	1000
	Hq.	100	100	100	100	100	100	100	100	100	100	100	100

[1] Bran.: Branch, Att.: Attack, Def.: Defense, With.: Withdraw, Amp.: Amphibious, Co.: Company, Bat. : Battalion, Reg.: Regiment, Brig.: Brigade, Inf.: Infantry, Art.: Artillery, SOF : Special Operations Force, Sup.: Support, Hq.: Headquarter.

Reg. Brig.	Inf.	4000	3000	8000	16k	16k	20k	1000	7000	4000	3000	8000	16k
	Art.	3000	1000	3000	1000	3000	1000	3000	1000	3000	1000	3000	1000
	Tank	4000	3000	8000	16k	16k	20k	1000	7000	4000	3000	8000	16k
	SOF	4000	3000	8000	16k	16k	20k	1000	7000	4000	3000	8000	16k
	Sup.	1500	1000	1500	1000	1500	1000	1500	1000	1500	1000	1500	1000
	Hq.	200	200	200	200	200	200	200	200	200	200	200	200
High	Hq.	200	200	200	200	200	200	200	200	200	200	200	200

"After the initial deployment, the simulation manager reads the data for the next time period. According to the locations that the units will be in the next period and their current locations, speed and direction of movement for each unit is calculated in degrees per second. Then simulation is forwarded second by second, and in each second, location manager adds the speed of the units into the locations of the radios and the RAPs owned by the unit. [3]"

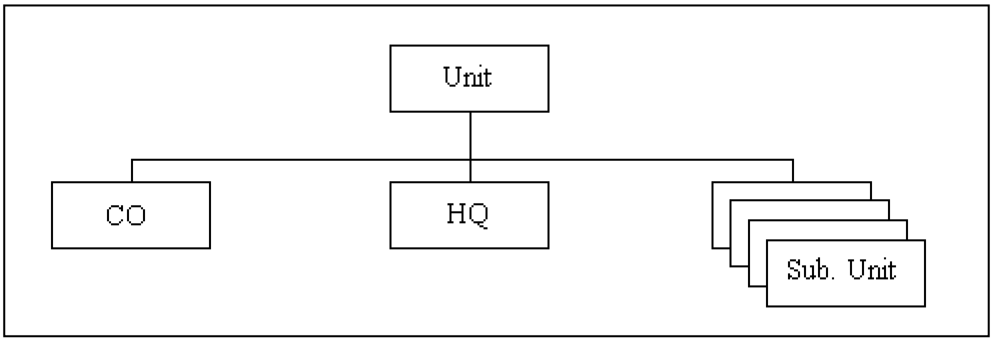

Figure 6.2 The generic unit organization [3].

"Call generator generates the calls according to the unit type, unit posture and whether the unit is in combat or not. The data related with these parameters come from the translator database. Using these data together with the average arrival rates coming from a statistical work, calls are generated. The call generator decides for each radio whether the radio initiates a call, the destination, type and duration of the call. [3]"

The call generation statistics tables are taken from [3]. These tables are a result of a statistical study with 20 officers who have at least 10 years of experience on leading a combat unit [3].

In Table 5.2, the call destination statistics are summarized. It is asked to the officers that with what probability a call is destined to where. As expected, in most of the cases the subordinate units or the COs of the leading units are called. This is very important, because it indicates that the subscribers in vicinity of the calling subscriber are called most of the time.

Table 6.2 Call destination statistics [3].

The destination of a call	(%)
To a subordinate unit	61.3
To the leading HQ or CO	22.4
The other units attached to the same higher unit	7.8
The neighbor units attached to other units	4.6
Anyone, any unit	3.9

Table 6.3 The number of calls in an attack within one hour [3].

Branch	In contact			Without contact		
	Minimum	*Average*	*Maximum*	*Minimum*	*Average*	*Maximum*
Infantry	10.1	16.8	30.5	6.4	9	16.5
Artillery	12.5	18.1	26.8	4.6	8.4	12.4
Tank	18.7	21.1	30.2	6.6	12.1	16.6
Special	16.4	17	24.4	4.3	7.5	10.8
Signal	9.4	14.3	19.4	5.4	13.2	18.1
Headquarter	12.3	15.6	21.8	7.5	12.7	16
Other	9	13.3	18	7.6	11.6	15.7

Table 6.4 The factor to normalize the call rates for the other postures [3].

Posture	In contact	Without contact
Defense	0.79	0.53
Withdraw-delay	1.01	0.68
Move	0.73	0.53
Amphibious	1.05	0.76
Other	0.74	0.56

“To decide on the call rate, the simulation manager provides the information of the type and posture of the unit together with whether it is in contact or not. Then the call manager looks up the call rate from Table 5.3 according to the unit type and in-contact information. If posture is not attack, this value is multiplied with the factor read from Table 5.4 according to the posture and in-contact information. The result is the call rate for that unit for that time period. [3]”

It is assumed in [3] that the call arrivals are Poisson. “This assumption is acceptable. The exponential distribution for the call interval times is a good approximation in the battlefield, because war fighters try to communicate with short time intervals in certain period of times, and if the time intervals between the calls get larger than the mean intervals, they get much larger than the mean. Since this is the same in call duration times, it is assumed call duration times are exponentially distributed, too. The call duration and call rate distributions are also factorized. When the uniform distribution is tried, it is found that there is not a meaningful difference in call blocking and call termination rates from the results of the simulation in which the exponential distribution is used. [3]“

Table 6.5 Call duration times [3].

	Duration in seconds
Minimum	6.7
Average	19.3
Maximum	41.3

The expected call duration for the calls are generated with exponential distribution whose mean value is read from Table 5.5.

The last thing to be determined about the calls is the type of the call. Six types of multimedia calls envisioned in [3], and distributed the calls among these types uniformly with the percentages shown in Table 5.6. The rates are determined intuitively in [3], not a determined after a statistical study, since multimedia is a new concept for tactical communications. However, we factorize the ratio of multimedia calls to ordinary voice calls in our simulation studies as illustrated in Table 5.6.

Table 6.6 Types of calls.

Multimedia to total call	**%37**	**%63**	**%74**	**%98**
Voice	63	37	26	2
Teleconference	18	30.6	36	47.7
Videophony	7.2	12.3	14.4	19.1
Videoconference	1.8	3.1	3.6	4.8
High priority data	1	1.7	2	2.6
Data	9	15.3	18	23.8

6.3. Performance of the Proposed System

To see the effect on the performance of the system, we studied different factors. These factors are in Table 5.9. While evaluating the performance, the following metrics are used;

- Route acquisition latency (in milliseconds)
- Routing overhead (in bytes and in number of packets)
- Packet loss rate
- Route robustness (in number of hops)
- Benefit of route caches (number of hits)
- Load of components
- Call blocking ratio
- Call termination ratio
- Partially connected MPR Ratio
- Not-connected MPR Ratio

Table 6.7 Scenarios used in simulations.

Scenario #	# of Units	# of RAPs	# of MPRs
1	6	4	694
2	15	11	1895
3	28	20	3452

Table 6.8 VCL parameters used in all simulations.

VCL Cell Radius	2000 meters
Real Cell Multiplication Factor (k)	1
Number Of Available Channels Per Each Cell	3
Eb/N0	5
Call Rate Factor	1
Duration of Simulations	30 minute
Size of Simulation Area	85 km. x 40 km.

We experimented with 3 different scenarios to run 23 simulations. Most of the simulations are done with the Scenario-3. Scenario-1 and Scenario-2 are derived from Scenario-3. The least unit number is in Scenario-1, and the greatest unit number is in Scenario-3. The number of units used in scenarios is defined in Table 5.7. In all scenarios, the VCL parameters defined in Table 5.8 are used and RAPs are deployed with battalions.

Table 6.9 The factoring parameters used in the simulation studies.

Test #	Scenario #	TTL Value For Hello Packets	Time Interval For Hello Packets (sec)	# of VCL Cells to Be Passed to Send Hello Packet	Collision Probability For CSMA/CA	Channel Bitrate (kbps)
1	3	No Hello Packet	No Hello Packet	No Hello Packet	0.1	32
2	3	2	60	1	0.1	8
3	3	2	60	1	0.1	16
4	3	2	60	1	0.1	32
5	3	1	60	1	0.1	32
6	3	4	60	1	0.1	32
7	3	1	90	1	0.1	32
8	3	2	90	1	0.1	32
9	3	4	90	1	0.1	32
10	3	1	120	1	0.1	32
11	3	2	120	1	0.1	32
12	3	4	120	1	0.1	32
13	3	1	240	1	0.1	32
14	3	2	240	1	0.1	32
15	3	4	240	1	0.1	32
16	3	2	120	1	0.1	16
17	1	2	120	1	0.1	32
18	2	2	120	1	0.1	32
19	3	2	120	1	0.2	32
20	3	2	120	1	0.3	32
21	3	2	120	1	0.4	32
22	3	2	120	1	0.5	32
23	3	2	120	2	0.1	32

6.3.1. RUNLINKED state MPR ratio

Table 6.10 RUNLINKED state MPR ratios.

Scenario #	# of RAPs	# of MPRs	# of RUNLINKED state MPRs	Ratio of RUNLINKED state MPRs
1	4	694	31	0.044
2	11	1895	34	0.018
3	20	3452	50	0.014

Before testing the system, we want to show RUNLINKED state MPR ratios. RUNLINKED state MPR ratio decreases, as the scenario size gets larger as shown in Table 5.10. Its reason is that the numbers of RUNLINKED state MPRs (MPRs those remain out of coverage area of RAPs) decreases, as they become in coverage area of RAPs when more RAPs are added to the system in scenario 2 and scenario 3. This ratio is important for VB-DSR, because in VB-DSR, MPRs those remain out of coverage area of RAPs use CSMA/CA technique. CSMA/CA is contention-based technique. In contention-based techniques, collisions increase if the network size increase. Hence, contention-based techniques are not scalable for large sized networks. However, as shown in Table 5.10, the ratio of the MPRs that had to use CSMA/CA technique remains very low. For scenario 3, RUNLINKED state MPR ratio is 1.4%. 50 of 3452 MPR use CSMA/CA technique. These 50 MPRs are not gathered, they are distributed over the operation area. Therefore, their locations can be away enough to collide. Secondly, DSR protocol avoids nodes from unnecessary transmissions that decrease the collision probability. As a result, for these reasons and low RUNLINKED state MPR ratio, we can use CSMA/CA technique for the MPR out of coverage area of RAPs.

6.3.2. Parameter decisions

Firstly, we analyze for determining the channel bit rate that will be used in the proposed system. Therefore, we test the system with different channel bit rates and examine the latency and the overhead occurred on route search processes. We compare them in Figure 5.3 – Figure 5.6.

Figure 5.3 – Figure 5.5 show the latency distributions. Figure 5.6 is the combination of Figures 5.3 – Figure 5.5 The x-axis shows the latency on route acquisition in milliseconds. The y-axis shows the number of routes constructed by the route reply

messages. These replies include only the replies that their accumulated source route is used as the route for data packets. During route search process, route reply packets may be sent by several nodes for the same destination, each may carry the same or different routes; but only one of them is selected by the source to construct the route. In DSR, the first reply is selected to construct the route and the others are thrown.

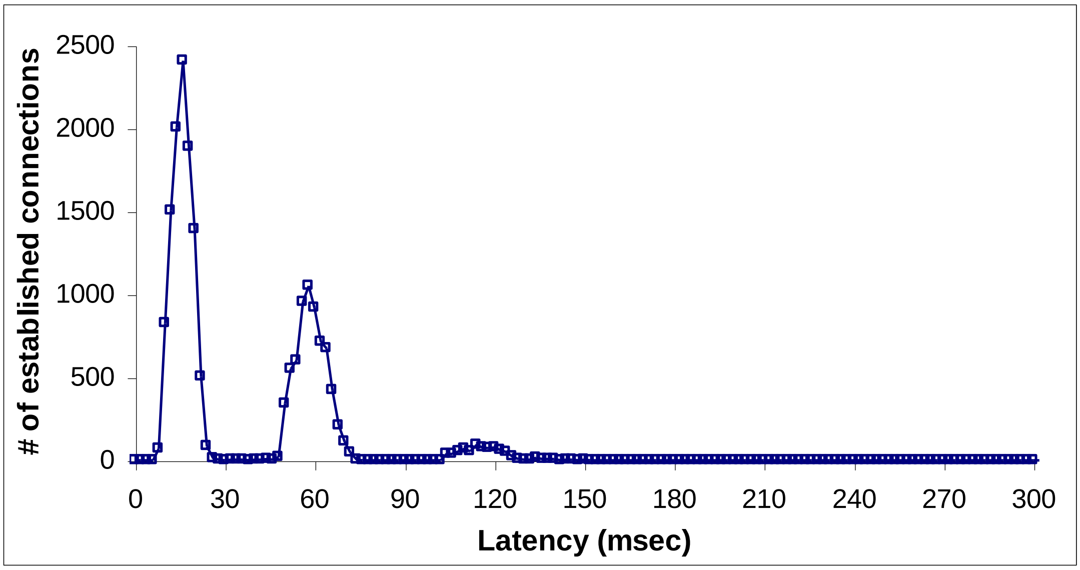

Figure 6.3 Latency distribution for 32 kbps.

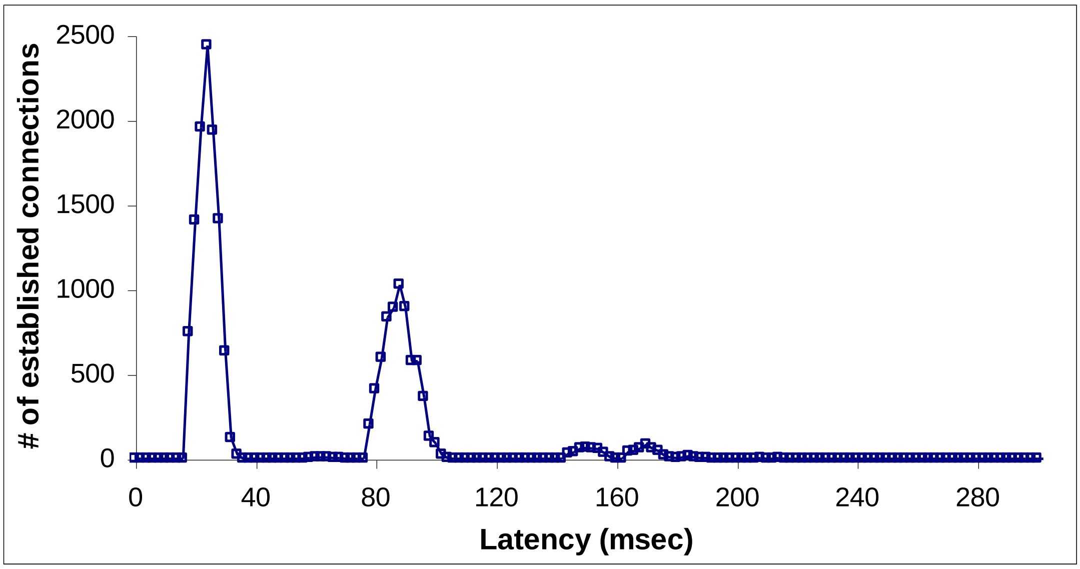

Figure 6.4 Latency distribution for 16 kbps.

In Figure 5.3 – Figure 5.5, the replies become dense in two parts by showing sharp increases and decreases. The reason of increases and decreases will be explained ahead. It is seen in figure 5.3 that a huge portion of routes is constructed within 30 ms, the second portion of the routes is constructed between 30 ms and 90 ms. Only a small number of routes are constructed between 90 ms and 150ms.

Each channel bit rate has similar latency distribution curves (Figure 5.6). However, as the bit rate decreases, the latency increases. It is clearer to see the decrease on Table 5.11. 8 kbps bit rate has 76 ms average route acquisition latency. As the bit rate increases the latency decreases, while getting its lowest value 37 ms at 32 kbps bit rate.

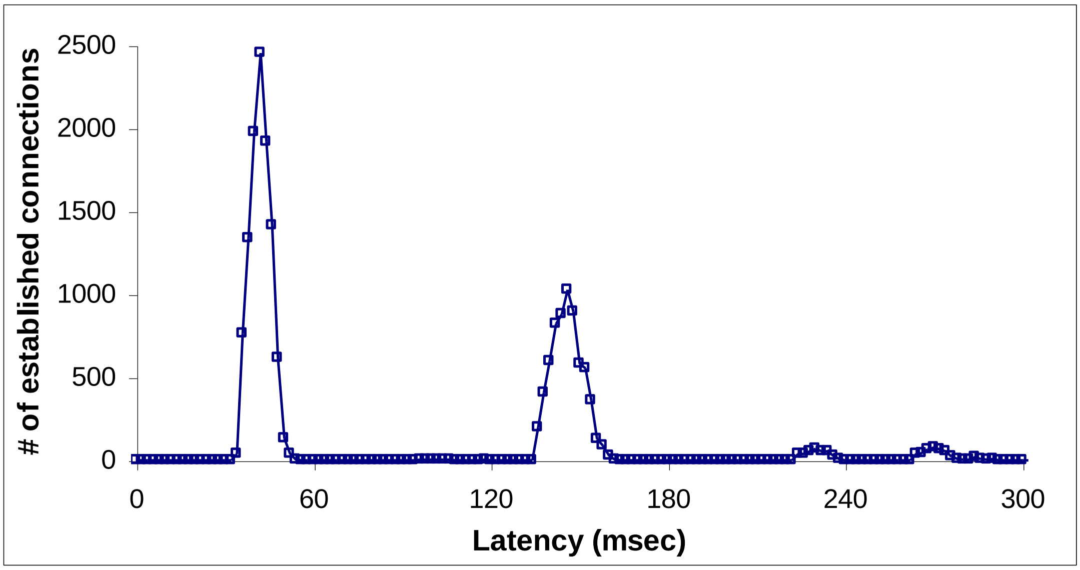

Figure 6.5 Latency distribution for 8 kbps.

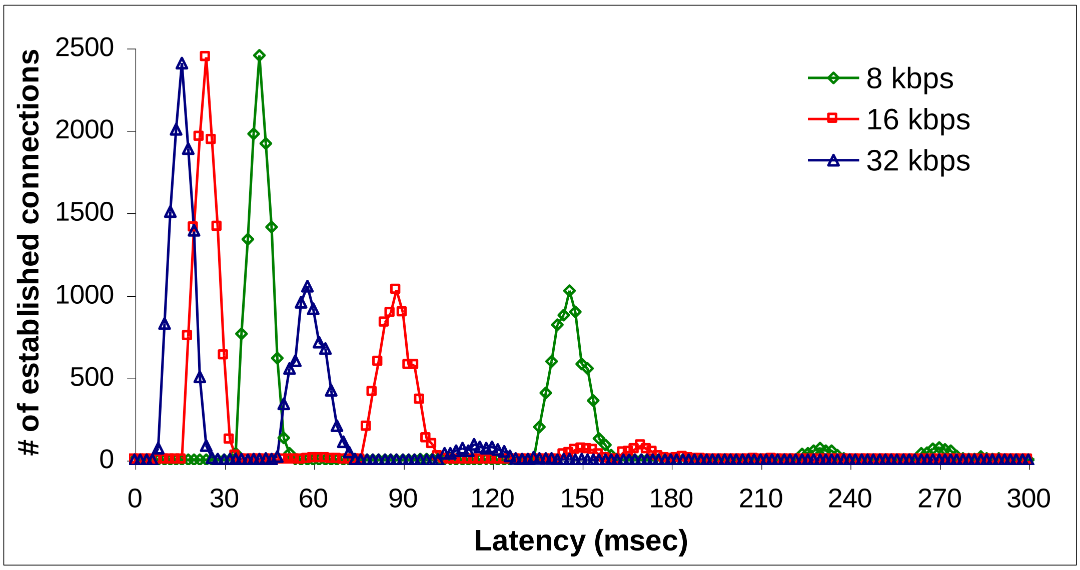

Figure 6.6 Latency distributions for all bit rates.

The decrease is related with the transmission delay of the packets. At low bit rates, the packets arrive the receiver later with respect to high bit rates. Therefore, route construction phase takes longer time at low bit rates and the latency decreases as the channel bit rate increase. We can explain the distributions become dense in two parts and the reason of sharp decreases and increases with the route search attempt processes (Table 5.12).

Table 6.11Latency statistics.

Channel Bit rate	Latency (msec)		
	Minimum	*Maximum*	*Average*
8 kbps	35	641	89
16 kbps	18	478	54
32 kbps	9	287	36

Table 6.12 Route search attempt statistics.

Channel Bit rate	# of Generated Calls	Own Cache Hit Ratio	Route Search Attempt					
			1	2	3	4	5	Total
			Ratio of Found Routes (%)					
8 kbps	18885	0.0738	0.5453	0.3364	0.0419	0.0004	0	0.9978
16 kbps	18882	0.1246	0.4940	0.3375	0.0413	0.0004	0	0.9978
32 kbps	18896	0.0736	0.5456	0.3366	0.0416	0.0004	0	0.9978

In DSR, route search attempts continue iteratively until the route is found. The first route search process is non-propagating route search process. The route search packets of the first route search attempts are sent only to the neighbors. If the route is not found by route cache information of neighbor nodes or the destination is not one of the neighbor nodes, a new route search attempt is executed by exponentially incrementing the TTL value of route search packets and reply waiting time of route request tables. If the route is not found at most 16 attempts, the destination is assumed unreachable. In VB-DSR, we reduce the number of attempts to 5 with the algorithms we enhanced. We can find a destination in VB-DSR at most five attempts.

For the calls that their routes remain in the route caches of the source nodes, no route search process is initiated. As seen in Table 5.12, half of the routes are found in the first attempt. This means that half of the routes are found by the route information of neighbor nodes. For the routes that are not found, second route search attempt is carried on. Most of the remaining unfound routes are found in the second route search attempt. After the second attempt, only a small number of routes remain unfound. Therefore, attempts 3, 4, and 5 are carried on. After the fifth attempt, for the unfound routes, the destination assumed to be unreachable. It is seen in Table

5.12 that after the third attempt, *total attempts* changes slightly. At the first attempt, time to live (TTL) value of the route search packets are set to 1, while route reply waiting time is set to 30 ms, 40 ms and 60 ms for channel bit rates 32 kbps, 16 kbps and 8 kbps respectively. For the routes that are not found by the aid of neighbor nodes' route caches within 30 ms, 40 ms and 60 ms for channel bit rates 32 kbps, 16 kbps and 8 kbps respectively, consecutive route search attempts are carried on by increasing exponentially the TTL value of the route search packets and the route reply waiting time of new route search attempts. In Figure 5.3 – Figure 5-6, the latency distributions become dense in two parts, because most of the routes are found at first and second route search attempts.

Table 6.13 Routing overhead ratios for different channel bitrates.

Channel Bit	Routing Overhead Ratio
8 kbps	0.0018684
16 kbps	0.0018502
32 kbps	0.0018525

The routing overhead ratios for different channel bit rates are shown in Table 5.13. It is seen that each channel bit rate has approximately equal routing overhead ratios. Routing overhead may change because of different route acquisition latencies that cause reattempts in route search processes. However, in Table 5.13, each channel bit rate has almost equal routing overhead ratios. Therefore, we may choose 32 kbps as channel bit rate since it cause less latency than others. On the other hand, we use CDMA technique as MA scheme in range of RAPs. To increase the soft CDMA capacity (in other words, to serve more users), we choose 16 kbps channel bitrate as in VCL.

Secondly, we analyze the parameters, which will be used in hello packets, affecting the performance of the system. These parameters are; (1) Time To Live (TTL) value, that determines the hop limit of the hello packet to transmit, (2) Time interval for a RUNLINKED state MPR must wait before sending a hello packet on case of no packet is received. As mentioned in Section 4.9, the hello packets are sent to reduce the number of unreachable destinations by forcing the neighbors to update their route caches. It is certain that with a high TTL value and short wait interval, route caches will be updated more frequently, and more accurately. However, high TTL value and short wait interval increase the routing overhead. Therefore, we analyze these parameters to determine their optimal values.

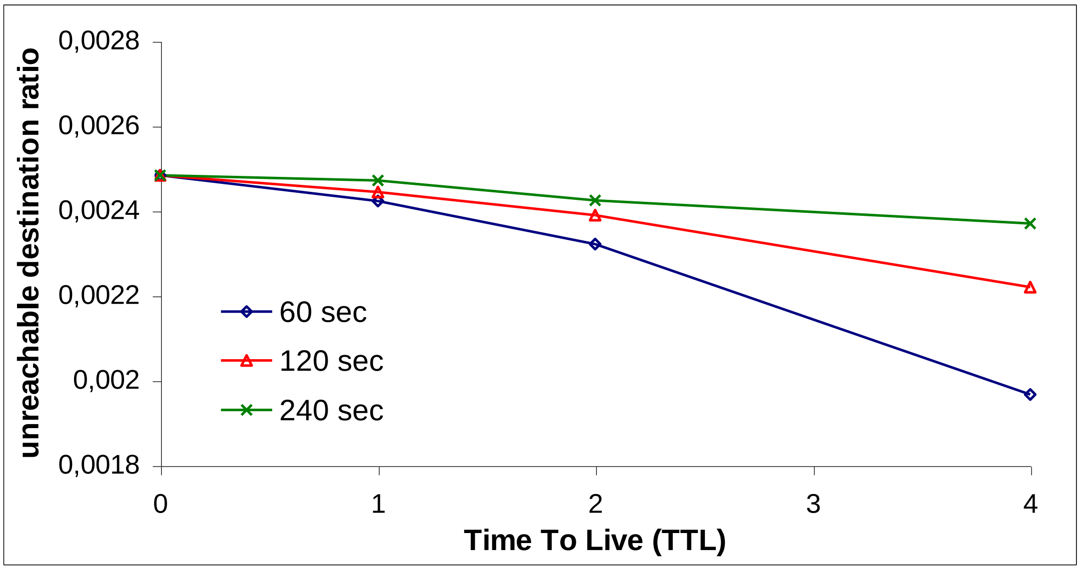

Figure 6.7Ratio of unreachable destinations for different TTL values and time intervals.

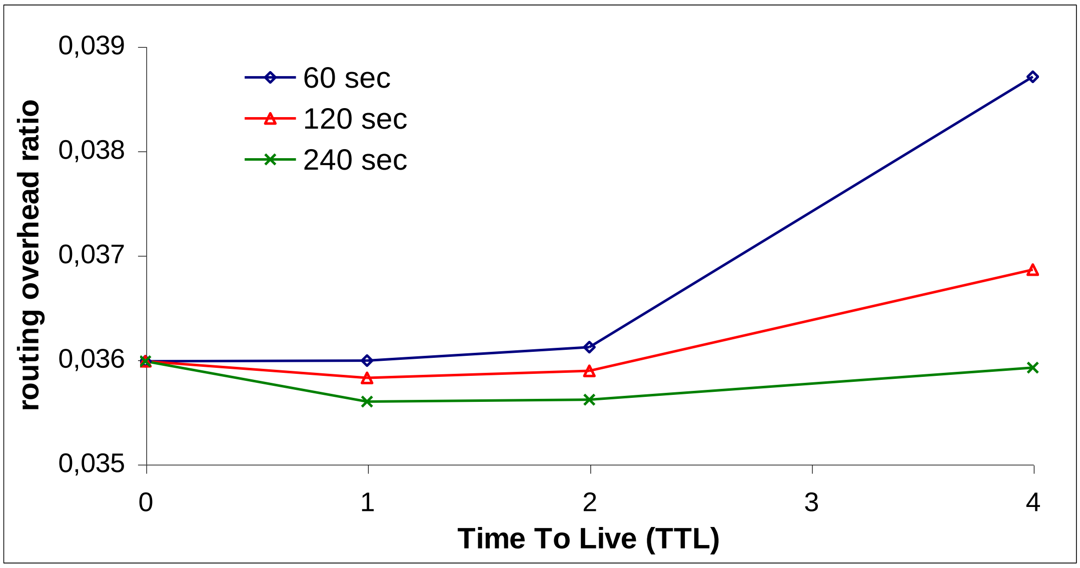

Figure 6.8 Routing overhead for different TTL values and time intervals.

The ratio of unreachable destinations decreases as the TTL value increases and the time interval decreases (Figure 5.7). The x-axis represents the TTL value of hello packets. Values with TTL value 0 at the intersection of the axis belong to the tests when no hello packet is sent. The lowest value is reached at TTL = 4 with time interval 60 second.

As a preliminary decision, we may get TTL value as 4 with time interval 60 sec, however, we must observe the caused overhead of hello packets.

In Figure 5.8, it is obvious that, as the TTL increases, the routing overhead increases at every time interval. The greatest routing overhead is caused by the lowest unreachable destination ratio parameters. To reduce unreachable destination ratio 0.0005, approximately 0.003 routing overhead is caused. For commercial use, the parameters TTL = 4 and time interval 60 sec may not be appreciate because of their caused overhead. However, in tactical operations, every component is important and must be reachable, if they can. Therefore, extra overhead can be affordable to reach these hardly reachable destinations. Therefore, we use TTL =4 and time interval 60 sec values in further studies

6.3.3. Route acquisition latency

Table 6.14 Route acquisition latency statistics.

Scenario #	Latencies (msec)			Total Calls Generated	Own Cache Hits	# of Replies		
	Min	*Max*	*Ave*			*Neigh.*	*Intermediate*	*Dest.*
1	18	209	53.88	5841	630	3339	3666	1591
2	18	210	57.52	13076	1194	7260	8134	3827
3	18	251	55.48	19829	1654	11728	12533	6026

In Table 5.14, we summarize the results from the latency tests. Table 5.14 shows that a significant amount of route is constructed by the *neighbor replies*, which indicates only a limited number of nodes are involved in most of the route discovery processes.

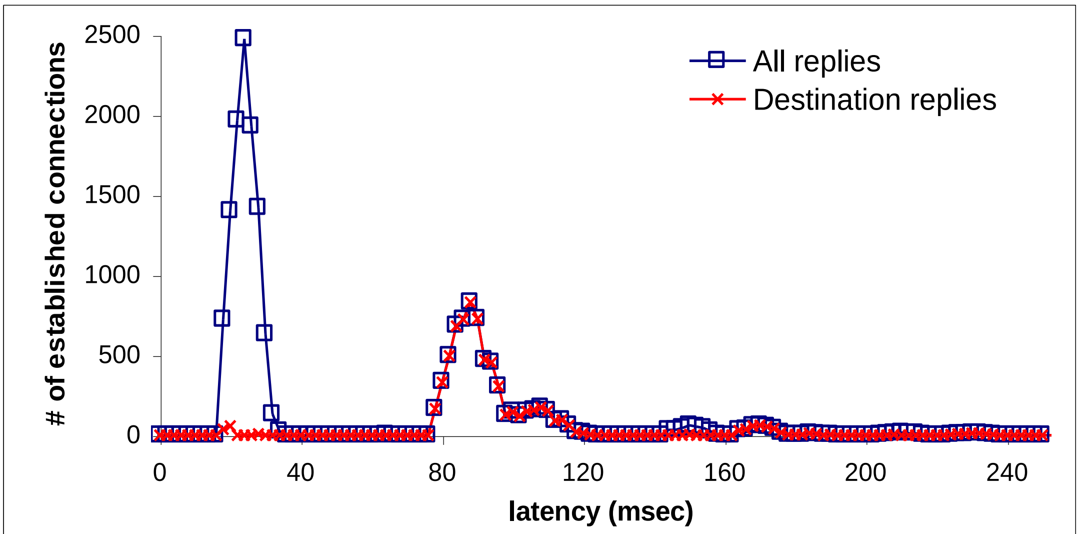

Figure 6.9 Latency distribution for scenario-3.

The latency distribution for Scenario-3 is shown in Figure 5.9. Other scenarios also have the similar latency distributions. In Figure 5.9, a*ll replies* contains all the replies returned from neighbor nodes, intermediate nodes and destination nodes, while the *destination replies* contains only the replies returned from the destination nodes. Latencies become dense in two parts. In the first part, *destination replies* are very low. On the other hand, *all replies* are very high. The difference gives the number of routes constructed by the replies returned from neighbor or intermediate nodes. As seen, almost all of the routes in the first part are constructed by the route cache information of neighbor nodes and intermediate nodes within 40 ms. After 40 ms, in the second part, *all replies* overlaps *the destination* replies. It means that all the replies in the second part are returned from the destination nodes. None of them is returned from the neighbor or intermediate nodes. Therefore, we can say that most of the routes are constructed by the route information of the neighbor and the intermediate nodes within 40 ms, and the remaining routes are constructed, at worst, by the replies of destination nodes within 120 ms.

Table 6.15 Route length statistics.

Scenario #	**Route Length (# of hops)**							
	Min	*Max*	*Average*	**1**	**2**	**3**	**4**	**5**
				Ratio of Routes				
1	1	5	2.019	0.0253	0.9325	0.0401	0.0017	0.00035
2	1	5	2.055	0.0135	0.9197	0.0654	0.0013	0.00015
3	1	5	2.041	0.008	0.9434	0.0479	0.0007	0.00005

Low latency is caused by short routes. As shown in Table 5.15, routes are at most 5-hops long with an average 2-hops long. Length of most of the routes are 2 hops, a small number is 1 hop and 3 hops length. The number of route in 4 and 5 hops lengths are negligible since they are very low in number. Short routes are constructed because of the VCL structure. Short routes reduce the route construction latency. On the other hand, most of the routes are constructed by neighbor and intermediate nodes' route replies within 40 ms (in Figure 5.9). Nodes in VB-DSR has ability to learn network topology. They cache the routing information in their route cache, and construct the routes by this topology information. Learning topology decreases the route construction latency and increases cache hit ratio while avoiding the routing overhead.

6.3.4. Routing overhead

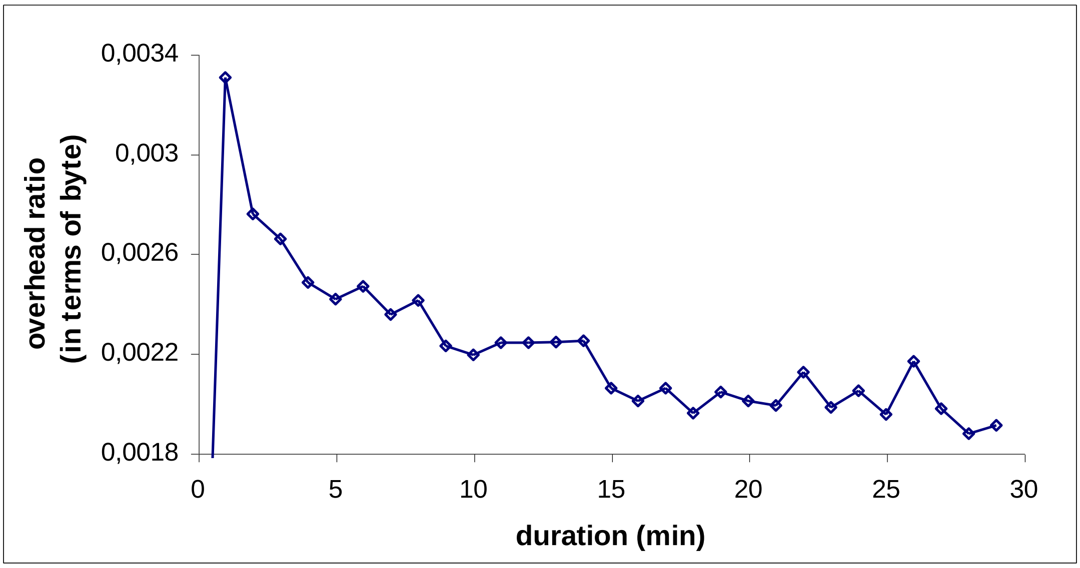

Figure 6.10Routing overhead ratio for Scenario-3.

Routing overhead ratios for Scenario-3 are shown in Figure 5.10. The x-axis represents the simulation time in minutes. Duration 0 (zero) is the beginning of the simulation and 30 is the 30th minute in the simulation. There is a sharp routing overhead increase in the beginning of the simulation. However, as the simulation time passes, the routing overhead decreases. At the beginning of the simulation, none of the nodes has any topology information. Therefore, more routing packets are sent to discover routes to unknown destinations. We consider the first 15 minutes as warm up period. As the time passes, nodes learn the topology by caching the captured route information to their route caches. This avoids the nodes to initiate a route search process if they would find a route to destination in their route caches. Therefore, routing overhead decreases as the simulation time passes. The decreases and increases at every minute is related with the calls generated. Because the number of generated calls differs at every minute, their caused overhead shows decreases and increases during the simulation.

If we get rid of the warm up period, the average routing overhead ratio remains approximately 0.0020 as seen in Figure 5.10.

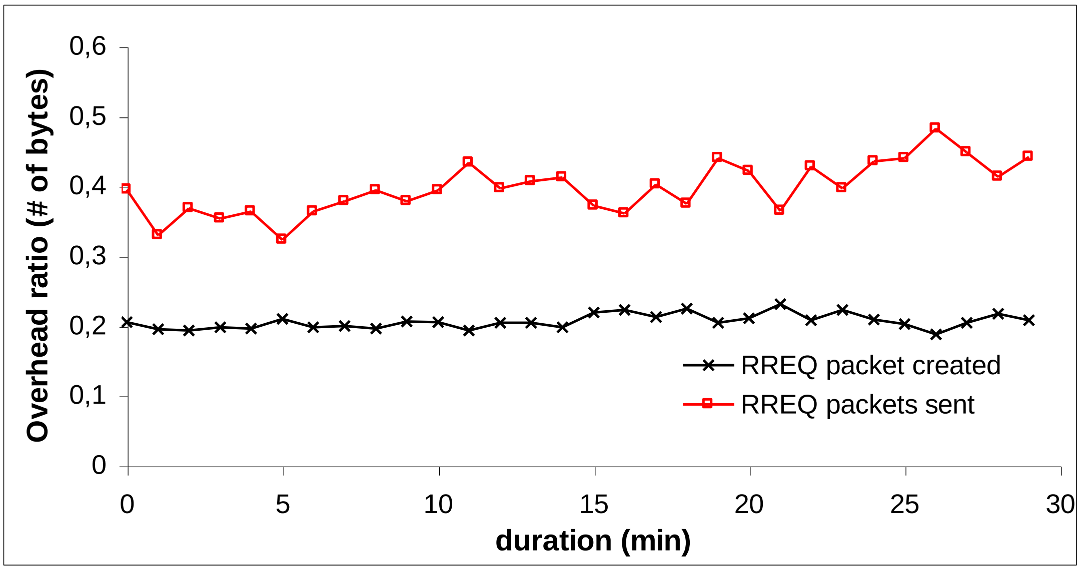

Figure 6.11 Partial overhead ratios for Scenario-3.

The route request packets, route reply packets, busy packets and hello packets cause routing overhead in the network. In Figure 5.11, partial overhead ratios of these packet types are shown. Route request packet ratio includes hello packets and route reply packet ratio includes busy packets. In Figure 5.11, *RREQ packets created* curve represents the ratio of packets created at the source nodes to discover a route and *RREQ packets sent* curve represents the ratio of the whole route request packets disseminated throughout the network. It is seen that sent packets ratio is two times greater than the created packets ratio. It shows that route search packets are not much disseminated throughout the network. Because most of the routes are found at first attempts, packet dissemination throughout the network remains low. On the other hand, packet dissemination is executed in a controlled manner in VB-DSR by the help of VCL. Route search packets are addressed only to RAPs. RAPs disseminate packets if it is required to in further route search attempts. Therefore, there is low routing overhead.

The difference of the ratio of *RREQ packets sent* from overall overhead gives the ratio of the route reply packets' overhead. Overhead ratio of route reply packets in terms of byte is greater than the overhead ratio of route request. Its reason is that on transmissions of route reply packets, the entire route between source and destination is inserted to header part of the packets. On the other hand, route request packets' headers carry only the route information of currently passed route. Secondly, each route reply packets' header fixed parts is 4 byte longer than route request packets' header fixed part. Therefore, route reply packets overhead ratio exceeds the route request packets' overhead ratio.

6.3.5. Cache usage

Low route discovery latency and significant amount of neighbor and cache replies are caused by the complete topology knowledge of RAPs and VCL structure. RAPs have almost full topology knowledge, because most of the packets are passed through them and they are the neighbor nodes of all RUN state MPRs.

Table 6.16 Cache hit statistics.

	Total Calls Generated	Cache Hits			
		Total	*Own*	*Neighbor*	*Intermediate*
Scenario-1	5841	4296	630	3339	3666
Scenario-2	13076	9378	1194	7260	8134
Scenario-3	19829	14187	1654	11728	12533

The statistics about route cache hits is given in Table 5.16. It is shown that a large amount of route to destination is found in nodes route caches. Cache usage avoids unnecessary route search process decreasing the routing overhead. The latency can be decreased and the cache hits can be increased by optimizing the size of the route caches of RAPs, and higher cache hit rates can be achieved by using better route caching strategies [42].

6.3.6. Packet loss rate

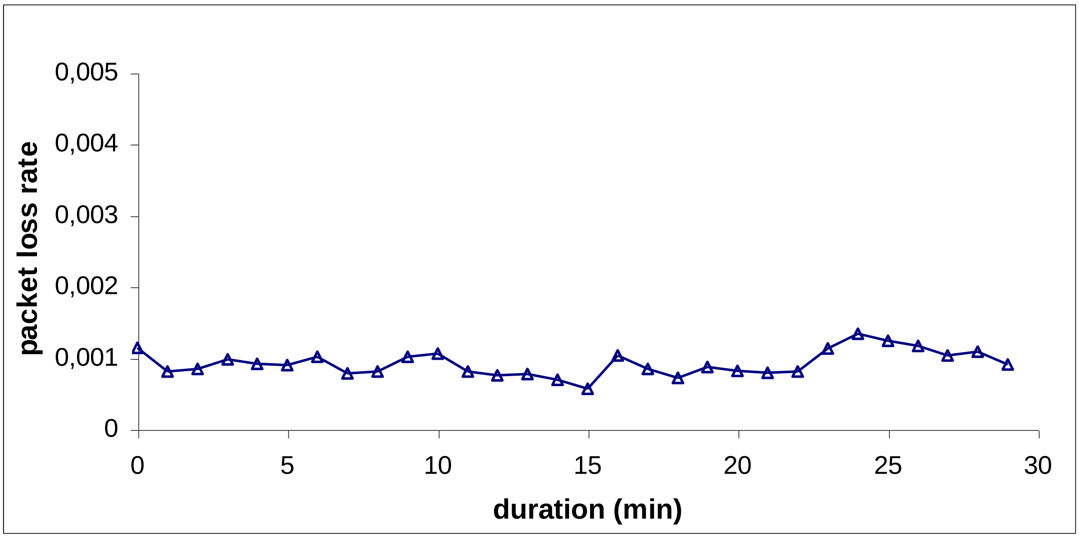

Figure 6.12 Packet lost rate for Scenario-3.

In Figure 5.12, packet loss rates of the simulated scenarios are illustrated. All scenarios have similar packet loss rate curves. It is seen that almost every packet reaches to the destination with a delivery ratio over 99.9%. Packet loss rates are high at initial minutes of the scenario, and decrease as the simulation time passed. Its reason is that at start time, none of the nodes has any topology information. Therefore, more routing packets are sent on initial minutes that cause more collisions. However, as the time passed, nodes learn more and more topology information avoiding the routing packet transmissions. Therefore, collisions decrease as nodes learn the topology while the time proceeds.

Table 6.17 Average packet loss rates.

Scenario #	Average Packet Loss Rate
1	0.00619
2	0.00175
3	0.00125

The average packet loss rates remain very small for all scenarios (Table 5.17), because collisions occur only between RUNLINKED state MPRs. In coverage are of RAP, nodes use CDMA as multiple access scheme and only a small number of nodes remain out of coverage area of RAP. Therefore, only a small number of nodes use CSMA/CA technique. On the other hand, DSR protocol avoids nodes from unnecessary transmissions. As a result, packet loss rates remain very low. No packet is dropped due to the buffer timeout. In addition, no packet is dropped between RAPs and RUN state MPRs since they use CDMA technique.

6.3.7. Unreachable destinations

Table 6.18 Unreachable destination ratios.

Scenario #	Ratio of Unreachable Destinations
1	0.0176
2	0.0055
3	0.0020

In addition to the performance of the system, we observe the ratio of the unreachable destinations (Table 5.18). Ratio of unreachable destinations gets lower as the size of scenario increases. Its reason is related with connectivity. The unreachable destinations in the first scenario become reachable as more MPRs and more RAPs are added to the network in Scenario-2 and Scenario-3. In other words, the unreachable nodes in the first scenario become connected in later scenarios. Therefore, they become reachable reducing the ratio of the unreachable destinations. For scenario-3, unreachable destination ratio is 0.2%.

6.3.8. Load of components

The reason of the system perform well is due to the VCL structure and due to the RAP usage. Routes become very short by VCL structure. Short routes cause the route search latency to remain very low in VB-DSR. VB-DSR controls the packet dissemination throughout the network. Short routes and controlled packet dissemination reduce the routing overhead. However, there are short routes, at each route search process and at each constructed route, RAPs are involved. Every packet is first sent to RAPs, and then RAPs disseminate the packets if required. Therefore, RAPs have great load reducing the load of MPRs.

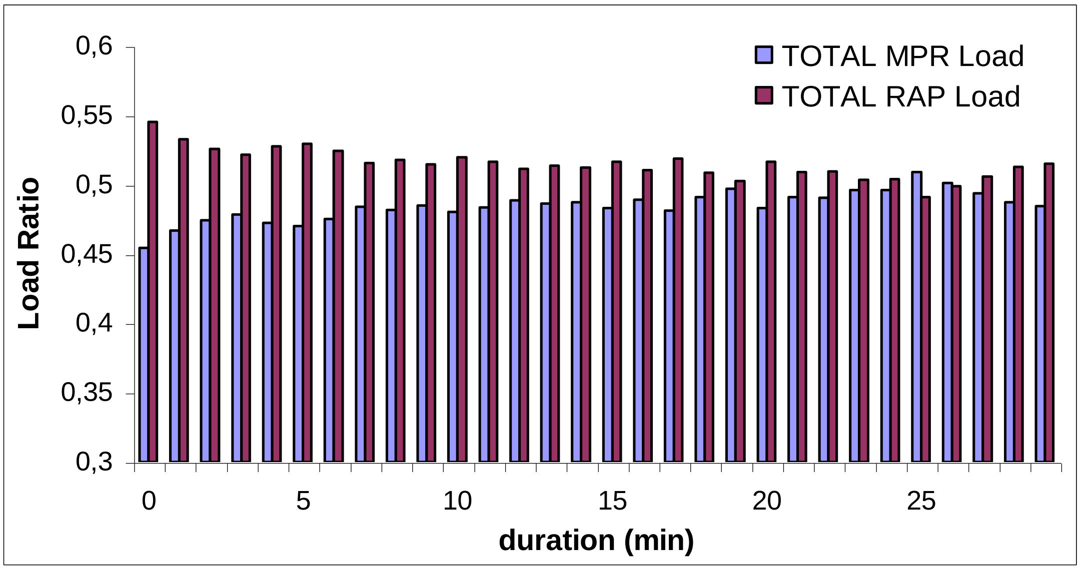

Figure 6.13 Load ratios of components.

In Figure 5.13, the ratio of loads of MPRs and RAPs are shown. When considered with unit numbers, a single MPR has very low load with respect to a RAP.

Load of a single MPR is shown in Table 5.19. The load decreases, as the scenario size gets larger. If we compare an MPR load with a RAP load, a RAP has load approximately 200 times greater than an MPR. By this way, MPRs energy consumption is reduced to a minimum level.

Table 6.19 Average load of components.

Scenario #	Average Load of an MPR (# of packets)	Average Load of a RAP (# of packets)
1	170	17622
2	94	16116
3	76	14023

6.3.9. Different collision probabilities

We use CDMA technique for the MPRs in the coverage of RAPs and CSMA/CA technique for the MPRs out of coverage area of RAPs. CSMA/CA is a contention-based technique. We assumed the collision probability for CSMA/CA as p=0.1. However, this value may not be realistic for military tactical operations, because calls are usually generated bursty. In other words, all the nodes may attempt to communicate at the same time. This will increase the collision probability. Secondly, distances between units may vary in tactical operations. As the distance increases, the collision probability increases. For these reasons, using the same collision probability during the simulation may not realistic. Therefore, we test the system with different collision probabilities; increasing the collision probability from 10 % to 50 %. The results are shown in Figure 5.14 and Figure 5.15.

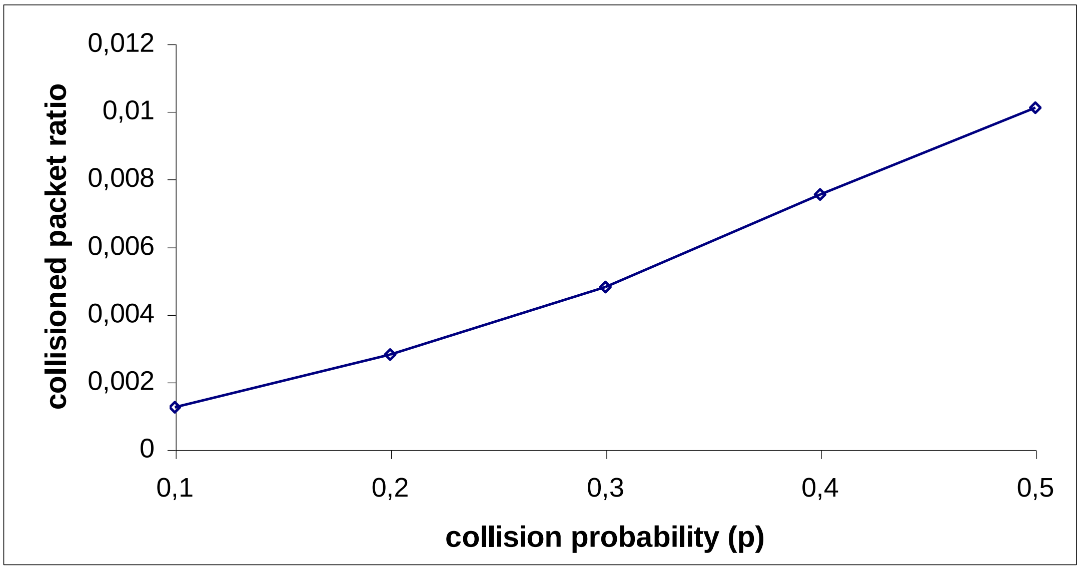

Figure 6.14 Packet loss rates for different collision probabilities.

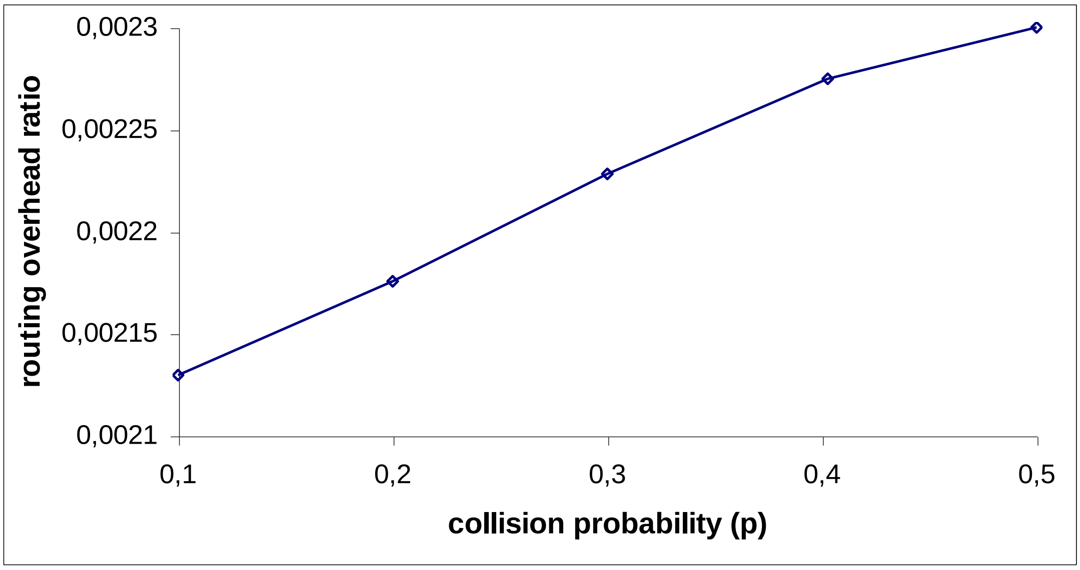

Figure 6.15 Routing overhead ratios for different collision probabilities.

Packet loss rate increases as the collision probability increases as expected. However, at even with the collision probability 50%, packet loss ratio remain below 1%. 99% of the all packets are transmitted successfully with the collision probability 50%. There is a very low packet loss ratio. Only a small number of MPRs remains out of RAPs, hence, only a small number of MPRs use CSMA/CA technique. On the other hand, DSR avoids nodes from unnecessary transmissions. Therefore, packet loss rate remains very low at even high collision rates.

Since collisioned packets are retransmitted, they cause overhead in the network. In Figure 5.15, routing overhead ratio increases as the collision probability increases. However, the caused overhead difference is below 0.1% between collision probabilities 10% and 50%. There is not a significant increase. Therefore, routing overhead is not affected by high collision probabilities.

7. CONCLUSION

As there becomes improvements in the computing industry and as the changing requirements of users grow, demands for a flexible, cost efficient networking alternatives that support terminal mobility, which can be deployable rapidly and be managed easily, make ad hoc networks a very important research topic.

Traditional routing techniques designed for infrastructured networks are infeasible for this highly dynamic mobile network. Although, new routing strategies are proposed to support node mobility in wireless networks while attempting to minimize routing overhead, most of the proposed scheme does not fully address the ad hoc routing problem. Most of the proposed schemes neither are scalable nor effectively balance the competing objective of responsiveness and efficiency. It is not certain that any of the currently proposed strategies are able to meet the future demands of wireless mobile users and provide scalability. Virtual Cell Layout appears as well-defined structure - applicable to next the generation tactical communication systems – behind of all other proposed schemes. Virtual Cell Layout is novel as its efficient resource management and mobile network structure.

We propose a new approach named as VCL based DSR (VB-DSR), that addresses the challenges meet in mobile ad hoc networks. In VB-DSR approach, Man Packed Radios (MPRs) are organize into clusters under the coverage area of Radio Access Points (RAPs), as in VCL approach. Differently than VCL, in VB-DSR, nodes do not attempt to register / deregister while there have access to RAPs. MPRs under coverage area of a RAP send their all packets and carry their all communications over RAPs. The MPRs those are not under coverage area of any RAP, disseminate their packets in ad hoc manner. The Dynamic Source Routing (DSR) protocol is implemented as routing technique. As in DSR, all of the mobile nodes (MPRs and RAPs) attempt to send packets only when they have to. There is no periodic message dissemination or beaconing. Therefore, routing overhead is minimized and needless traffic is avoided.

In VB-DSR, RAPs are used as access points as in VCL. Since RAPs have large transmission range, they can cover most of the operation area. Only a small number of MPRs may remain out of range of RAPs. Therefore, almost fully connected network topology is provided.

VB-DSR is a cluster-based two-level hierarchical structure. The constructed routes are range between 1-5 hops length with an average of 2-hops length, as obtained from results. This is a very short and stable route as demanded in mobile ad hoc networks. Short route, also, provides power savings in communications. With the two-level hierarchy and short length routes, the delay for a route search and the latency of a packet to reach its destination remains very small.

Route acquisition latency remains very low since most of the routes are constructed by the cache information of neighbor nodes. From the latency and cache-hit results, we conclude that the nodes have the ability to learn the network topology. Results show that VB-DSR has a low packet loss rate because of the efficient resource management of VCL approach. RAPs have great importance by affecting the performance of the system.

We evaluate the performance of our design with different metrics. The simulation software interacts with a constructive combat model, namely Joint Theater Level Simulation (JTLS), which runs by applying the commands entered during the previous Computer Aided Exercises (CAX), and retrieve the mobility, status, posture and other related information for a number of units. These data is very realistic, since they are retrieved from the real exercises. Then the software enhances the resolutions and generates the calls by using the retrieved information, and runs the designed system.

In our studies, we make some modifications on DSR protocol that provides us better results then pure DSR. We leave the work on optimizing the DSR protocol for VB-DSR as a future work.

7.1. Future Work

We implement DSR protocol over VCL structure. We make some modifications on VCL and DSR. We enhance new algorithms for their application.

In our study, we used CSMA/CA technique for the nodes that are out of coverage area of RAPs. We assume that RAPs exist during the simulation. However, RAPs can break down to failure or may be destroyed. In such a case, performance of the system will decrease. Because the number of MPRs out of coverage area of RAPs will increase, collisions will increase. Therefore, VCL's clustering algorithms can be employed for the MPRs that are out of coverage area of RAPs, enabling the implementation of CDMA technique throughout the whole network. This will increase the robustness of the system.

We assume that the system work properly in satellite and UAV tiers. Therefore, we assume that RAPs can access each other without any obstruction or limitation. Implementation of these tiers and realizing connections between RAPs complete the system.

DSR protocol performs well in VCL structure by some modifications and approaches we make. These increase the performance of the system. Performance comparison of VB-DSR with the implementation of other routing techniques over VCL is left as future studies. We assume that mobile terminals have bi-directional links. VB-DSR supports uni-directional links and asymmetric routes. Implementation of the physical layer is left as future work.

REFERENCES

[1] **Perkins, C.R.**, 2000. Ad Hoc Networks, Addison-Wesley, Boston.

[2] **Johnson, D.B. and Maltz, D.A.**, 1996. Dynamic Source Routing in Ad Hoc Wireless Networks, *In Mobile Computing, edited by T.Imielinski and H. Korth,* Chapter 5, Kluwer Publishing Company, pp. 153-181.

[3] **Çayırcı, E.,** 2000. Application of the 3G PCS Technologies to the Mobile Subsystem of the Next Generation Tactical Communications Systems, *Ph.D. Thesis*, Boğaziçi University.

[4] **Çayırcı, E. and Ersoy, C.,** 2001. A PCS based architecture for tactical mobile communications, *Elsevier Computer Networks,* Vol. 35, Issues 2-3, pp. 327-350..

[5] **Cayirci, E. and Ersoy, C.**, 2001. CAX Interacted Tactical Communications Simulation, *Proceedings of Communications Networks and Distributed Systems Modeling and Simulation Conference (WMC'2001)*, pp. 123-128.

[6] **Johnson, D.B**, 1994. Routing in Ad Hoc Networks of Mobile Hosts, *IEEE Workshop on Mobile Computing systems and Applications*.

[7] **Pei, G. and M. Gerla**, Mobility Management in Hierarchical Multi-Hop Mobile Wireless Networks, *Internet Draft*.

[8] **Pei, G., Gerla, M., Hong, X. and Chiang, C-C.** A Wireless Hierarchical Routing Protocol with Group Mobility. *Internet Draft*.

[9] **Royer, E.M. and Toh, C-K.**. A Review of Current Routing Protocols For Ad Hoc Mobile Wireless Networks, *Internet Draft*.

[10] **Iwata, A.. Chiang, C-C., Pei, G., Gerla, M. and Chen, T-W.**, 1999. Scalable Routing Strategies for Ad Hoc Wireless Networks, *IEEE Journal on Selected Areas in Communications*, Vol: 17, No: 8.

[11] **Hatzis, K.P., et al.,** 1999. Fundamental Control Algorithms in Mobile Networks, *SPAA'99*.

[12] **Soytürk, M., Harmancı, A.E. and Çayırcı, E.,** 2001. Mobile Ad Hoc Networks and Routing, *Bilişim'01*.

[13] **Johansson, P., and et al.**, 1999. Scenario-based Performance Analysis of Routing Protocols for Mobile Ad Hoc Networks, *Mobicom'99*.

[14] **Leeuwen, B.V., Espinoza, J. and Sholander, P.**, 1998. Effective Protocols for Mobile Communications and Networking, *Sandia National Laboratories*, 2753.

[15] **Rom, R. and Sidi, M.,** 1990. Multiple Access Protocols, *Springer-Verlag*, New York.

[16] **Ramanathan, S. and Steenstrup, M.**, 1996. A Survey of Routing Techniques for Mobile Communications Networks. *ACM Baltzer Mobile Networks and Applications,* 1(2),.

[17] **Chen, G. and Stojmenovic, I.**, Clustering and Routing in Mobile Wireless Networks, *Internet Draft.*

[18] **Ko, Y-B. and Vaidya, N.H.**, 1997. A Routing Protocol for Physically Hierarchical Ad Hoc Networks, *TAMU Technical Report*, 97-010.

[19] **Perkins, C. R. and Bhagwat, P.,** 1994. Highly Dynamic Destination Sequenced Distance Vector Routing (DSDV) for Mobile Computers. *In ACM SIGCOMM*, pages 234–244.

[20] **Murthy, S. and Garcia-Lunes-Aceves, J.J.,** 1996. An Efficient Routing Protocol for Wireless Networks. *ACM Balzer Mobile Networks and Applications Journal,* Special Issue on Routing in Mobile Communications Networks.

[21] **Chen, T-W and Gerla, M.,** 1998. Global State Routing: A New Routing Scheme for Ad Hoc Wireless Networks. *In Proc. ICC*.

[22] **Chiang, C-C.**, 1997. Routing in Clustered Multihop Mobile Wireless Networks with Fading Channel, *Proc. IEEE SICON'97,* pp197-211.

[23] **Maltz, D.A., Broch, J., Jetcheva, J. and Johnson, D.B.,** 1999. The Effects of On-Demand Behavior in Routing Protocols for Multi-Hop Wireless Ad Hoc Networks, *IEEE Journal on Selected Areas in Communications special issue on Mobile and Wireless Networks*.

[24] **Johnson, D.B. and Maltz, D.A.**, 1996. Protocols for Adaptive Wireless and Mobile Networking, *IEEE Personal Communications,* 3(1).

[25] **Perkins, C. and Royer, E.,** 1999. Ad Hoc On-Demand Distance Vector Routing. *In Proceedings 2nd IEEE Workshop on Mobile Computing Systems and Applications*, (WMCSA'99).

[26] **Park, V. D. and Corson, M. S.,** 1997. A Highly Adaptive Distributed Routing Algorithm for Mobile Wireless Networks. *In IEEE Infocom 97.*

[27] **Toh, C-K**., 1997. Associativity-Based Routing for Ad-Hoc Networks, *Wireless Personal Communications,* 4(2).

[28] **Dube, R. et al**., 1997. Signal Stability Based Adaptive Routing (SSA) for Ad-Hoc Networks. *IEEE Personal Communications*.

[29] **Krishna, P. et al**., 1997. A Cluster-Based Approach for Routing in Dynamic Networks. *ACM Computer Communications Review*, 27(2).

[30] **Lin, C.R. and Gerla, M.,** 1997. Adaptive Clustering for Mobile Wireless Networks, *IEEE Journal on Selected Areas in Communications*, 15 (7).

[31] **Gerla, M. and Tsai, J.,** 1995. Multicluster Mobile Multimedia Radio Network, *Wireless Networks*, 1.

[32] **Das, B. and Bharghavan, V.,** 1997. Routing in Ad-Hoc Networks Using Minimum Connected Dominations Sets, *In IEEE International Conference on Communications* (ICC'97),.

[33] **Das, B., Sivakumar, R. and Bharghavan, V.,** 1997. Routing in Ad-Hoc Networks Using a Spine, Internet Draft.

[34] **Ramanathan, R. and Steenstrup, M.,** 1998. Hierarchically-Organized, Multihop Mobile Wireless Networks for Quality-of-Service Support. *Mobile Networks and Applications,* 3.

[35] **Sharony, J.**, 1996. An Architecture for Mobile Radio Networks with Dynamically Changing Topology Using Virtual Subnets, *In ACM Baltzer Mobile Networks and Applications Journal*, 75-86.

[36] **Haas, Z**., 1997. A New Routing Protocol for Reconfigurable Wireless Networks. *In Proc. of IEEE International Conf. on Universal Personal Communications (ICUPC).*

[37] **Das, S.R., Castaneda, R. and Yan, J.**, 2000. Simulation Based Performance Evaluation of Routing Protocols for Mobile Ad Hoc Networks, *ACM Baltzer Mobile Networks and Applications*, 179-189, (5).

[38] **Broch, J., Maltz, D., Johnson, D., Hu, Y-C. and Jetcheva, J.** A Performance Comparison of Multi-Hop Wireless Ad Hoc Network Routing Protocols, *Internet Draft*.

[39] **Das, S.R., Perkins, C.E. and Royer, E.M.**, Performance Comparison of Two On-Demand Routing Protocols for Ad Hoc Networks, *Internet Draft*.

[40] **Lee, S-J., Gerla, M. and Toh, C-K.** Performance Evaluation of Table-Driven and On-Demand Routing Protocols for Mobile Ad Hoc Networks, *Internet Draft*.

[41] **Aron, I.D. and Gupta, S.K.S.**, 2000. Analytical Comparison of Local and End-to-End Error Recovery in Reactive Routing Protocols for Mobile Ad Hoc Networks, *MSWIM 2000*.

[42] **Hu, Y-C. and Johnson, D.B.** Caching Strategies in On-Demand Routing Protocols for Wireless Ad Hoc Networks, *Internet Draft*.

[43] **Hong, X., Gerla, M., Pei, G. and Chiang, C-C.**, A Group Mobility Model for Ad Hoc Wireless Networks. *Internet Draft*.

[44] **Saas, P.F. and Gorr, L.**, 1995. Communications for the Digitized Battlefield of the 21st Century, *IEEE Communications Magazine*, 86-95, October.

[45] **Saas, P**., 1999. Communications Networks for the Force XXI Digitized Battlefield, *In ACM Baltzer Mobile Networks and Applications Journal*, 139-155, (4).

[46] **Pei, G., Gerla, M., Hong, X. and Chiang, C-C.** A Wireless Hierarchical Routing Protocol with Group Mobility. *Internet Draft*.

[47] **Singh, S. and Raghavendra, C.S.**. PAMAS – Power Aware Multi-Access Protocol with Signaling for Ad Hoc Networks, *InternetDraft.*

[48] **Singh, S., Woo, M. and Raghavendra, C.S.**. Power Aware Routing in Mobile Ad Hoc Networks, *InternetDraft.*

[49] **Behrens, J. and Garcia-Luna-Aceves, J-J.** Distributed Scalable Routing Based on Link-State Vectors, *Internet Draft*.

[50] **Prakash, R.**, 1999. Unidirectional Links Prove Costly in Wireless Ad Hoc Networks, *Internet Draft*.

[51] **Ko, Y-B. and Vaidya, N.H.**, 1997. Using Location Information to Improve Routing in Ad Hoc Networks, *TAMU Technical Report*, 97-013.

ATTACHMENTS

ATTACHMENT A

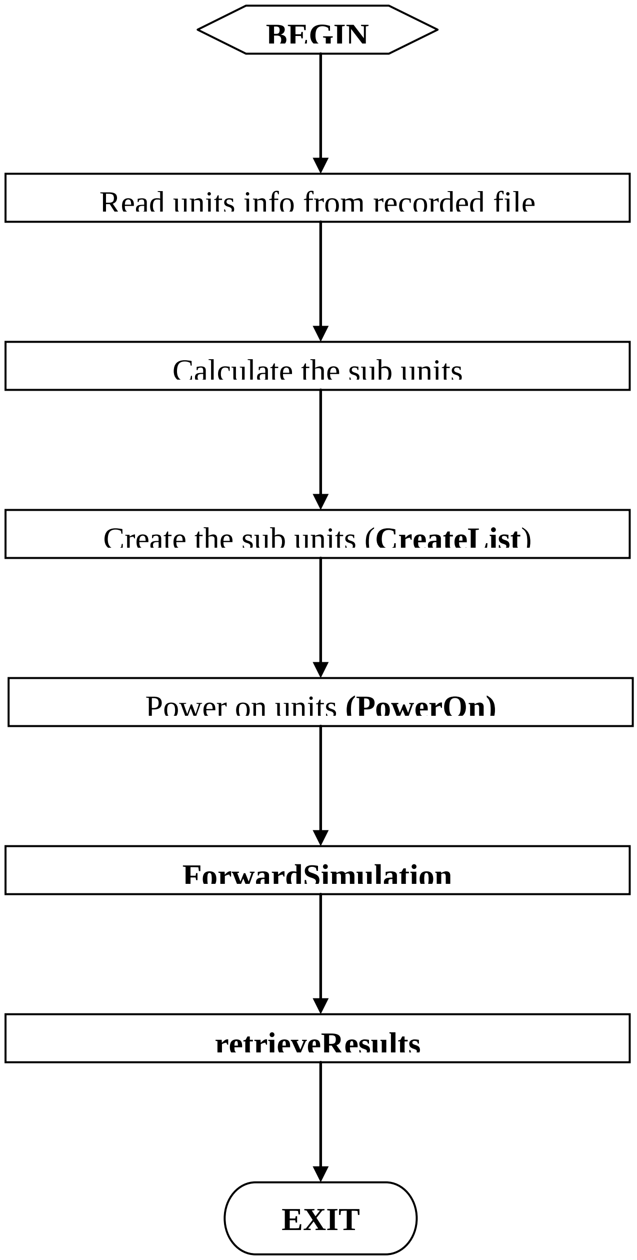

Figure A. 1 Flow diagram of Procedure runSimulation.

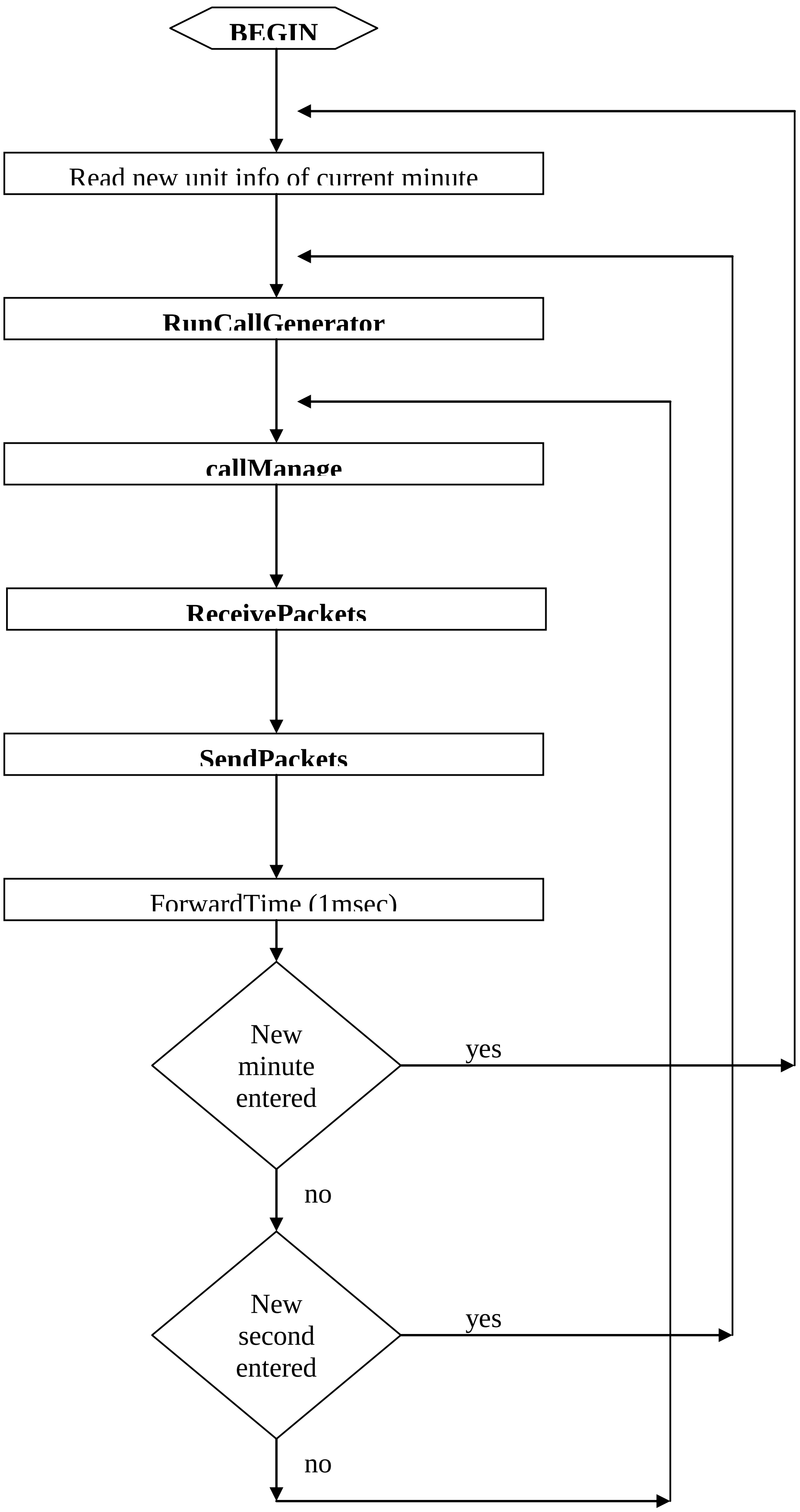

Figure A. 2 Flow diagram of Procedure forwardSimulation.

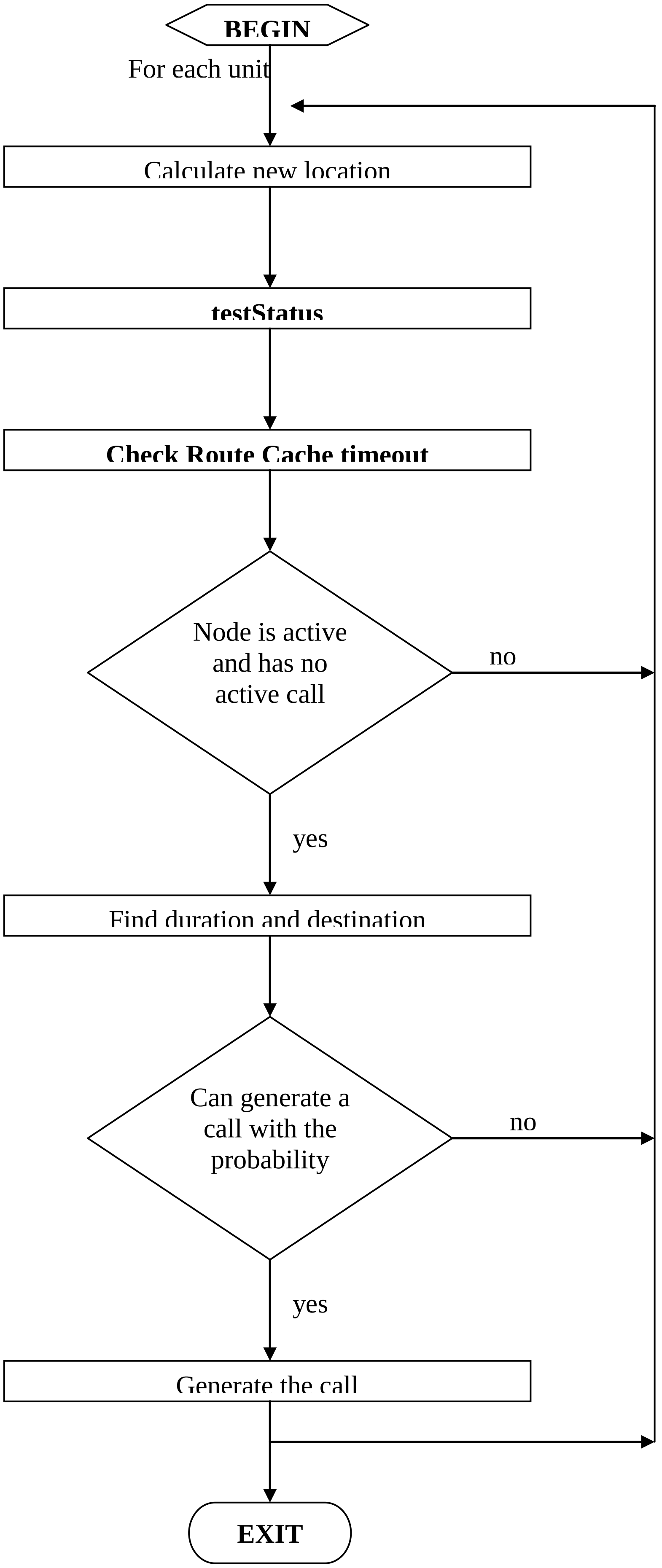

Figure A. 3 Flow diagram of Procedure runCallGenerator.

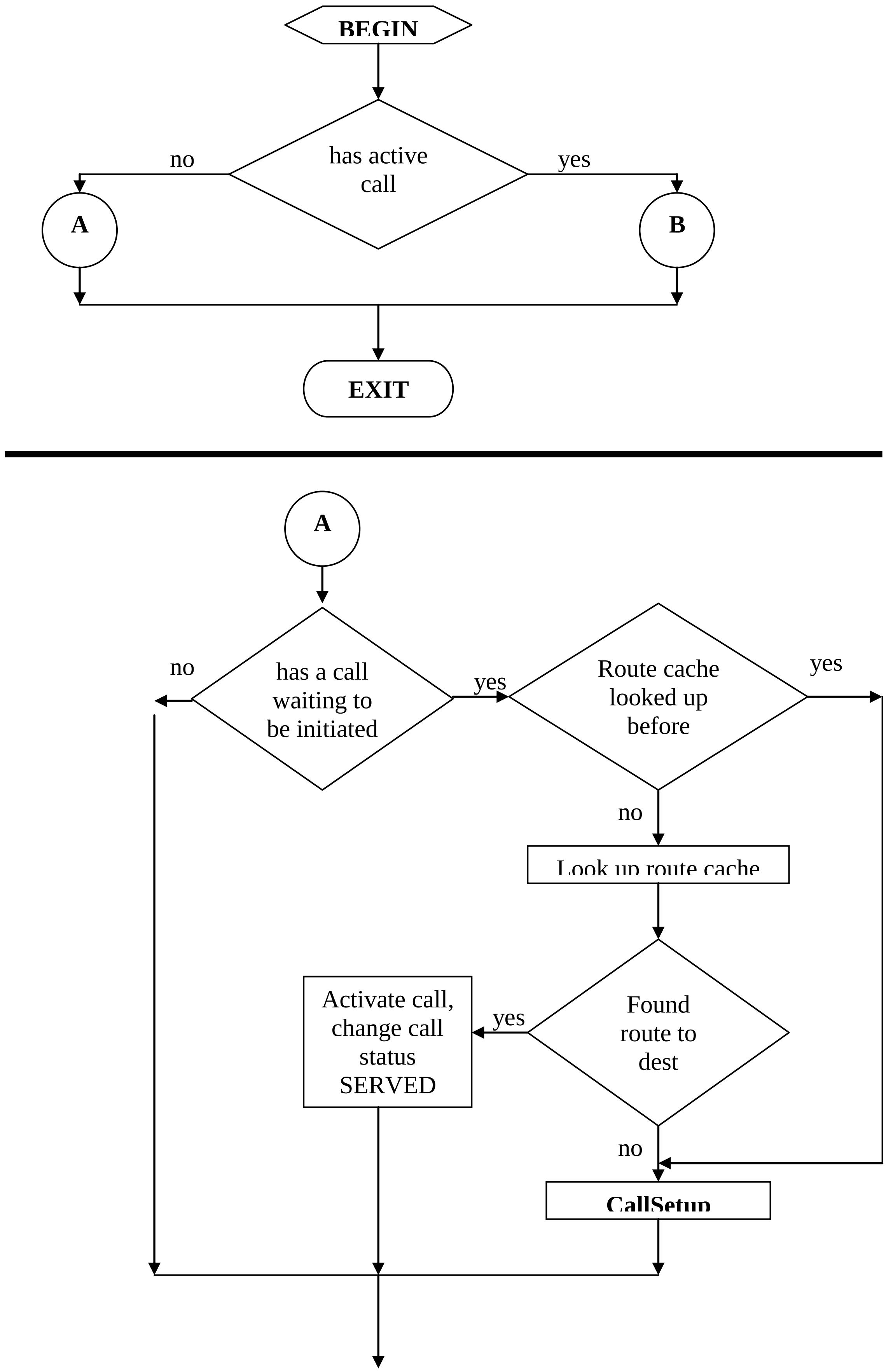

Figure A. 4 Flow diagram for Procedure callManage

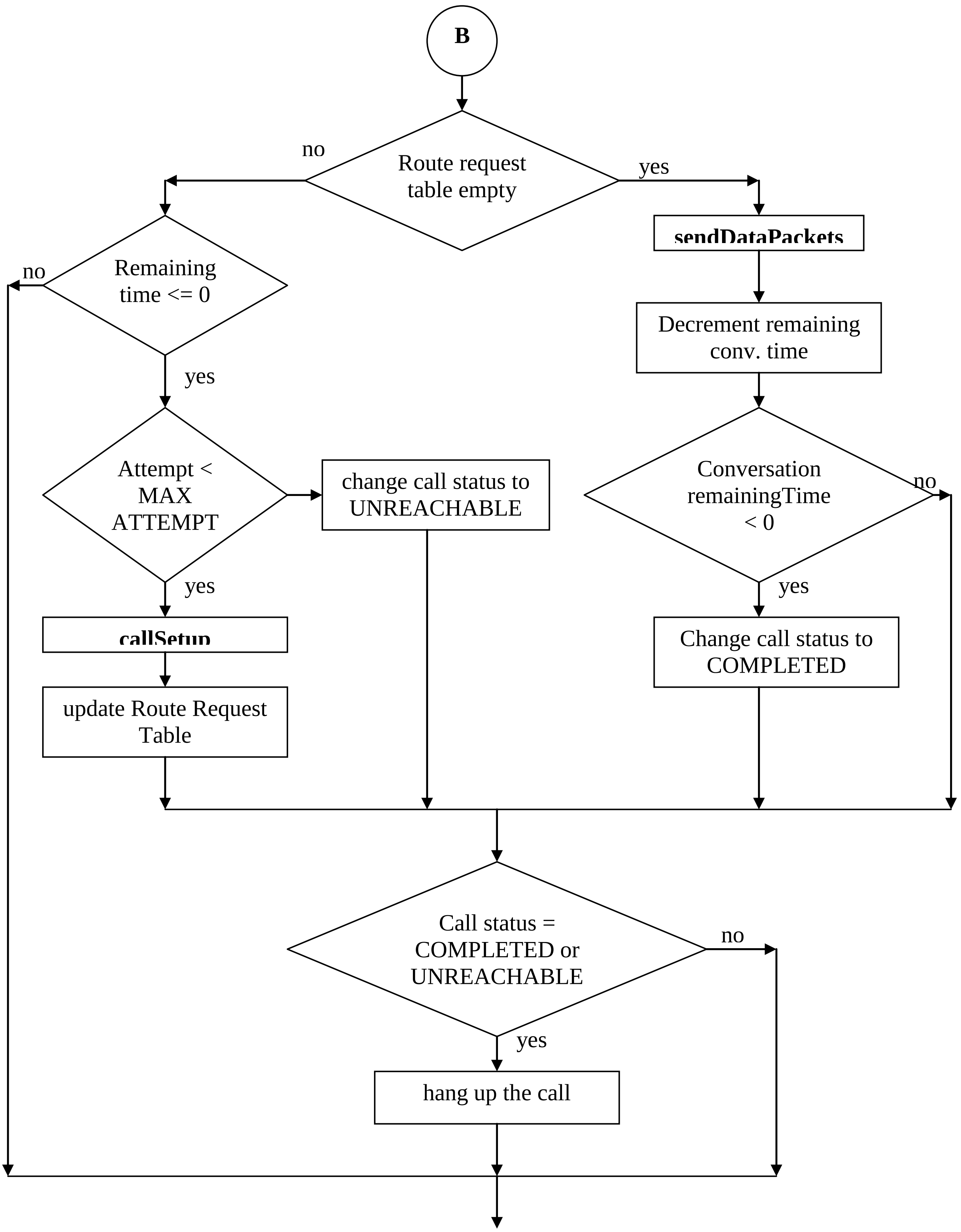

Figure A. 5 Flow diagram of Procedure callManage (continue).

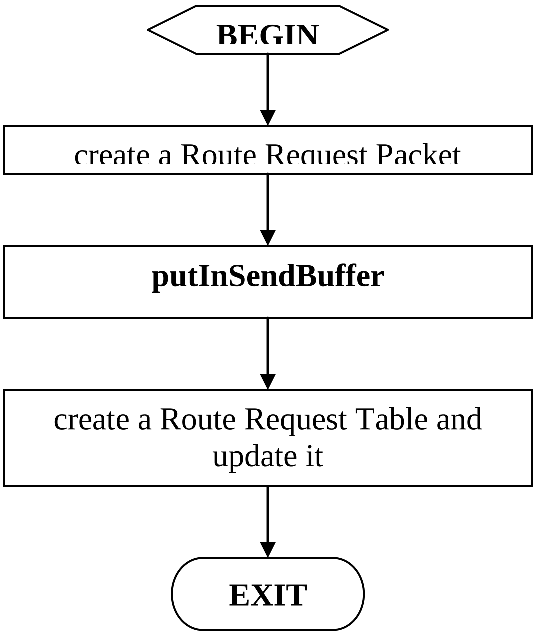

Figure A. 6 Flow diagram of Procedure callSetup.

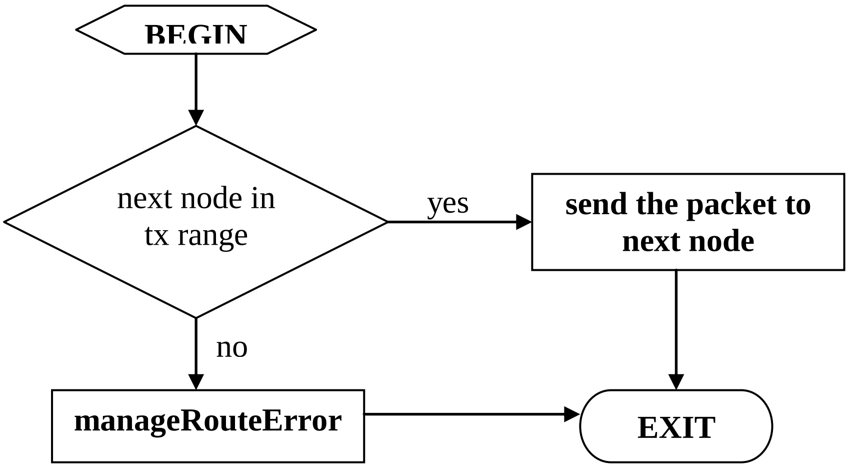

Figure A. 7 Flow diagram of Procedure sendToNext.

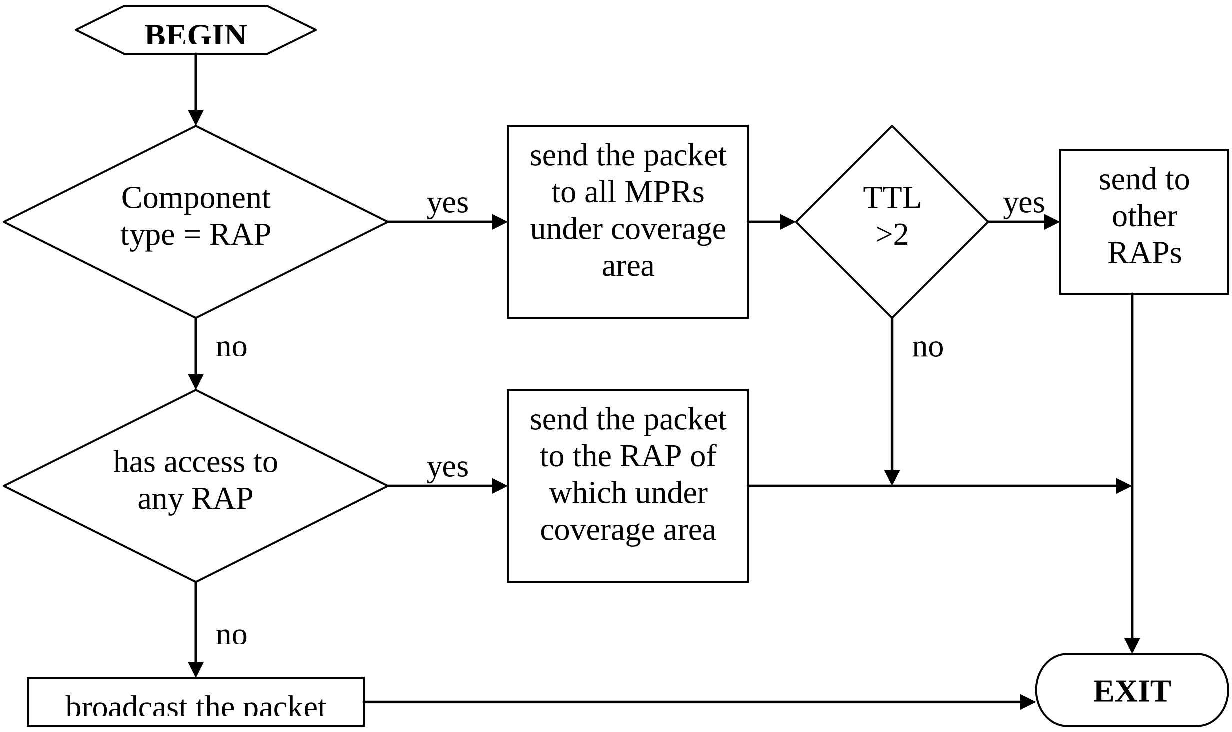

Figure A. 8 Flow diagram of Procedure manageRouteRequest.

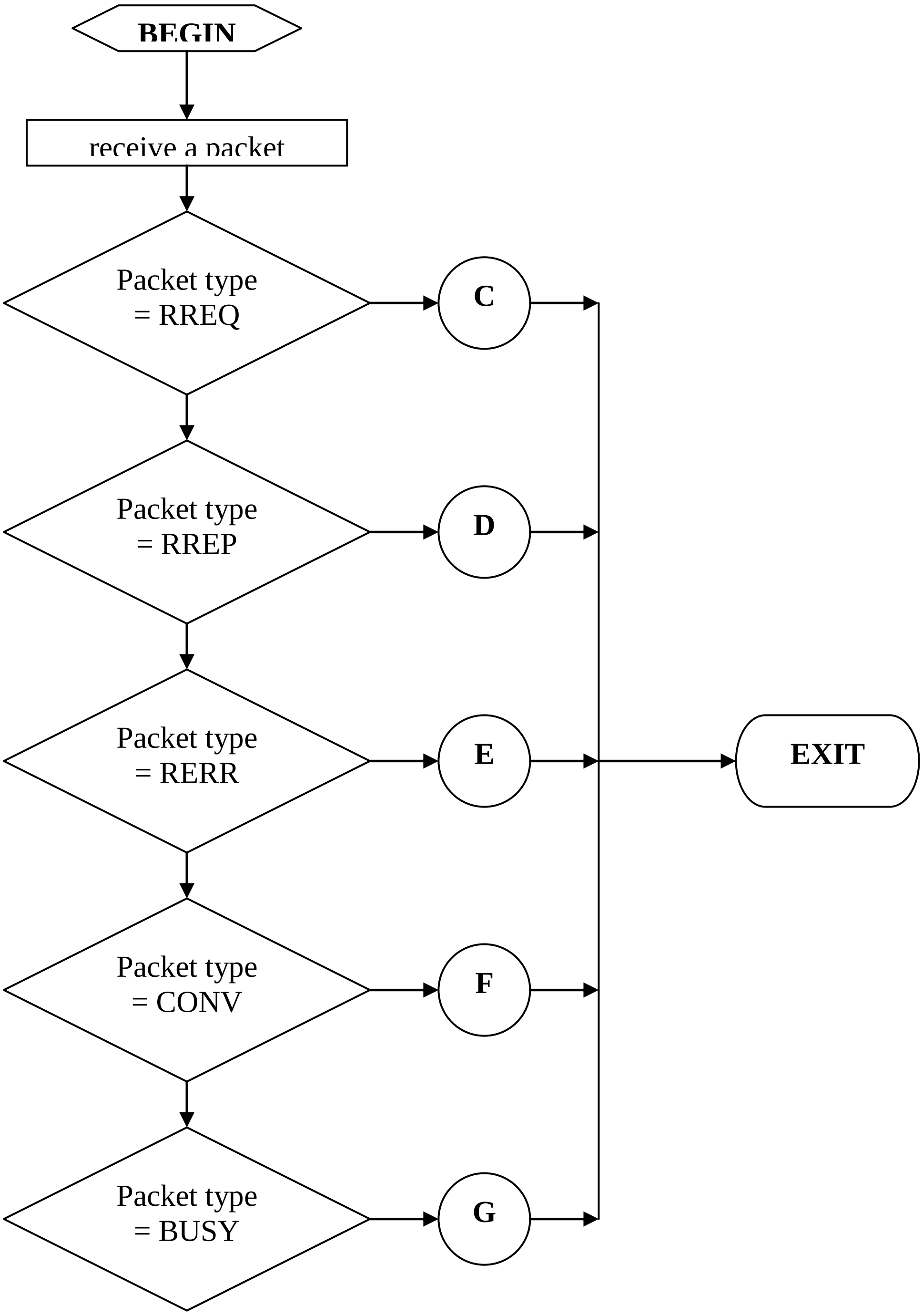

Figure A. 9 Flow diagram of Procedure receivePackets.

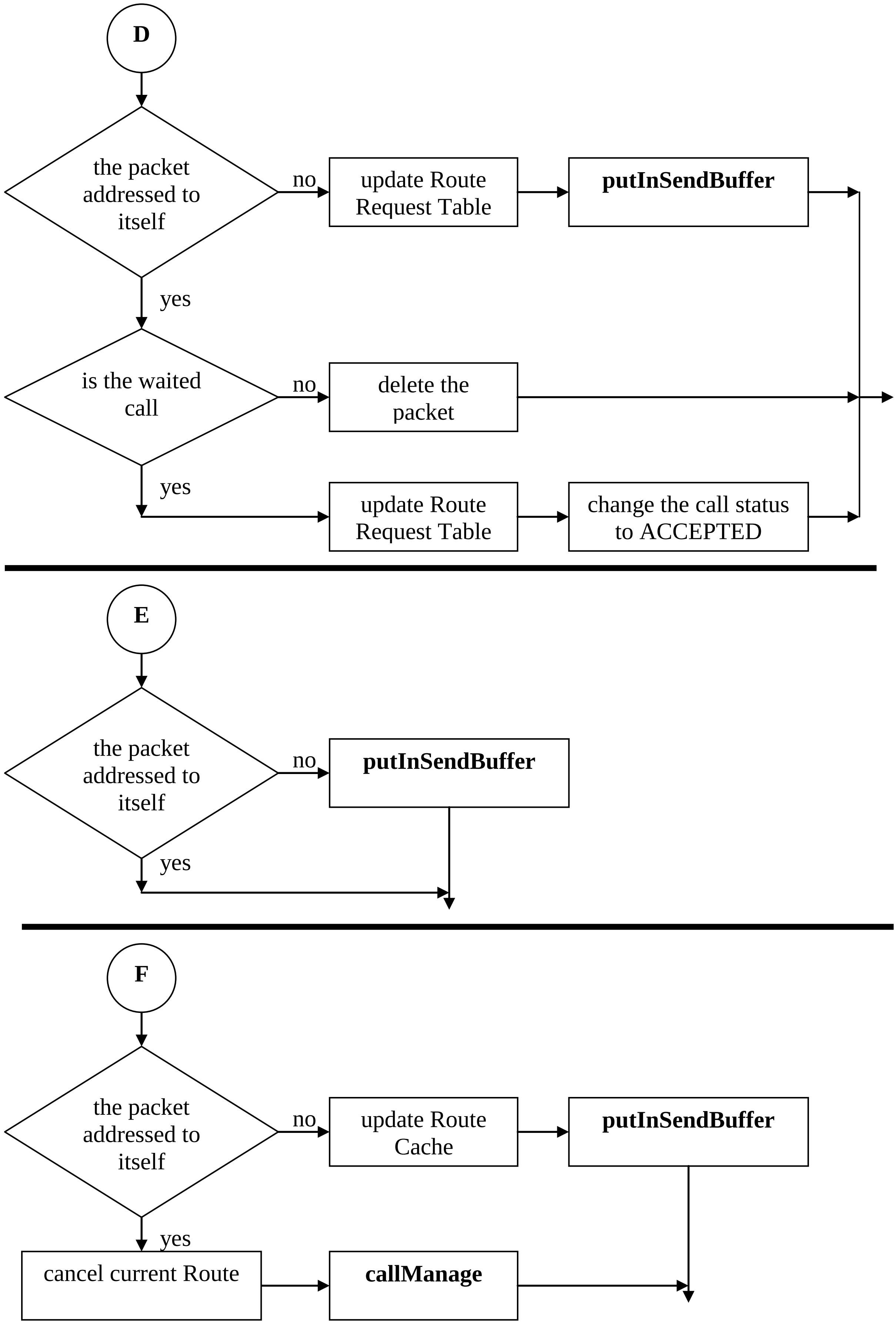

Figure A. 10 Flow diagram of Procedure receivePackets (continue).

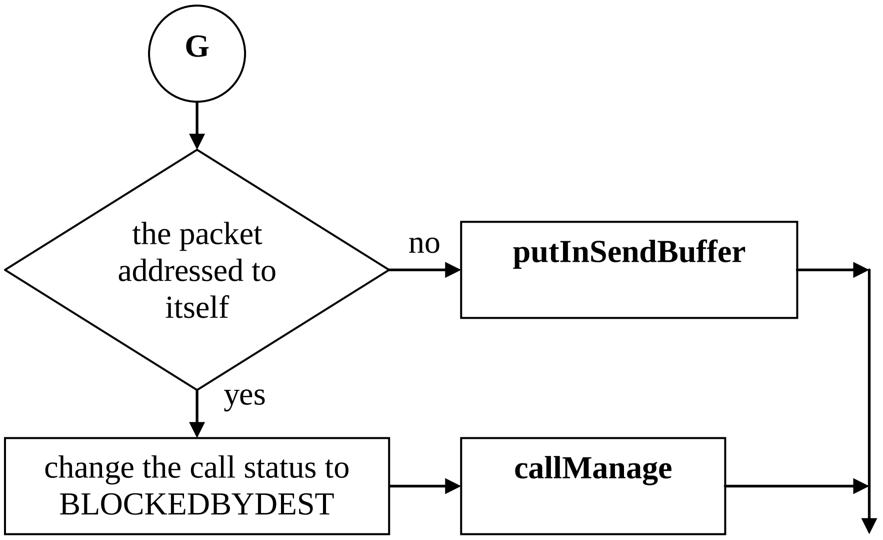

Figure A. 11 Flow diagram of Procedure receivePackets (continue).

BIBLIOGRAPHY

Mujdat Soyturk graduated from Turkish Naval Academy in Industrial Engineering / Operation Research program in 1994. He worked in different duties in Turkish Navy for four years. He graduated from OBI Subay Temel Egitimi in Boğaziçi University in 1999. He is continuing MS education in computer science department in Istanbul Technical University. His research area is Wireless Mobile Networks.