Nagasai Panchakarla

Adaptive Voice Applications over Delay Tolerant Networks

School of Electrical Engineering

Thesis submitted for examination for the degree of Master of Science in Technology.

Espoo 25.10.2012

Thesis supervisor and instructor:

Prof. Jörg Ott

Aalto University
School of Electrical Engineering

AALTO UNIVERSITY
SCHOOL OF ELECTRICAL ENGINEERING

ABSTRACT OF THE
MASTER'S THESIS

Author: Nagasai Panchakarla

Title: Adaptive Voice Applications over
Delay Tolerant Networks

Date: 25.10.2012 Language: English Number of pages:8+55

Department of Communications and Networking

Professorship: Network Technology Code: S-38

Supervisor and instructor: Prof. Jörg Ott

Internet is predominantly based on best effort packet transmission. Performance of applications over internet suffer due to disconnections, delays, losses and dynamic nature of elements in the network. Voice communications, such as Voice over Internet Protocols (VoIP), over mobile networks has to deal with technical barriers such as delays and temporary disconnections. Delay tolerant networks provides communication based on asynchronous messaging that deals with delays and disconnections; which provides a mechanism to deliver the messages irrespective of instantaneous end-to-end path connectivity.

In the thesis, delay tolerant adaptive media is proposed to allow DTN-based communication as a fall-back if real time end-to-end voice communication fails. We designed a system which adapts to delays and losses by switching between RTP/UDP and RTP/DTN-based voice packets transmission. The real time communication works fine as long as continuous end-to-end path exists. The continuous path might not exist when there are changes in the network topology of mobile users. So in the case of non availability of end-to-end path, we swiftly adapt to RTP/DTN-based voice with variable length messaging mechanism. To assess the call quality in different modes of operation, we used R values of E model specified by ITU-T. The results show that the proposed delay tolerant adaptive media for adaptive voice over delay tolerant networks achieves better utility for the users when end-to-end connectivity is not available or when delays are higher.

Keywords: DTN, voice communication, adaptive

Preface

First and foremost, I am indebted to my Prof. Jörg Ott, whose vision is crucial for this thesis. This thesis would not be possible without his knowledge and amiable guidance. I would like to thank him for providing me the opportunity and his guidance will continue to inspire me for years.

I would like to thank Jani Lakkakorpi for the help and support with NS2DTN package. Furthermore, I would like to thank Satya, Varun, all my friends and colleagues for all the insightful discussions and suggestions. It is my pleasure to be in such a wonderful atmosphere.

Last but not least, I would like to express gratitude to my parents, Madan Kumar Panchakarla and Nagamani Panchakarla for their constant love and encouragement.

Otaniemi, 25.10.2012

Nagasai Panchakarla

Contents

List of Figures

List of Tables

Symbols and abbreviations

Abbreviations

AVPF	Audio Visual Profile Feedback
AODV	Ad-hoc On-Demand Distance Vector
BP	Bundle Protocol
CAPEX	Capital Expenditure
DAM	Delay-tolerant Adaptive Media
DTN	Delay-tolerant Networking
DTNs	Delay-tolerant Networks
DT-SIP	Disconnection Tolerant SIP
EID	Endpoint Identifier
FEC	Forward error correction
FTP	File Transfer Protocol
GoB	Good or Better
IP	Internet Protocol
IPC	Inter Process Communication
ITU	International Telecommunication Union
MANET	Mobile Ad hoc Networks
MOS	Mean Opinion Score
NS2DTN	Network Simulator 2 DTN
OPEX	Operational Expenditure
PoC	Push-to-talk over Cellular
PoW	Poor or Worse
PTT	Push-to-talk
P2P	Peer-to-Peer
QoS	Quality of Service
RERR	Route Error
RREP	Route Reply
RREQ	Route Request
RTP	Real-time Transport Protocol
RTCP	Real-time Transport Control Protocol
RTT	Round Trip Time
RWP	Random Waypoint
SCTP	Stream Control Transmission Protocol
SIP	Session Initiation Protocol
TCP	Transmission Control Protocol
TLS	Transport Layer Security
TTL	Time-To-Live
UAC	User Agent Client
UAS	User Agent Server
UDP	User Datagram Protocol
URI	Uniform Resource Identifier
URL	Uniform Resource Locator
VoIP	Voice over Internet Protocols

1 Introduction

Applications over existing communications networks assume continuous end-to-end connectivity. This assumptions which had been carried over well connected wired networks may differ significantly for the mobile and wireless communication networks. Wireless networks might experience disconnections, delays and losses because of coverage gaps, interference, attenuation, congestion etc. In the case of short disconnections and packet losses, the applications could deal those situations by queuing the packets and thus losses could be reduced by retransmissions. But queuing and retransmission of the packets induces further delay. During congestion in the network, transport protocol such as TCP uses congestion control mechanisms that scale back the transmission rate. Varying transmission rate and throughput also induces further delays. Long network disconnections halts the communication until the time of disconnection. So delay is the key metric for many applications when trading off losses, data rate, retransmissions etc. But having a well connected wireless networks comes with heavy capital expenditure for the operators or users. Mobile Ad hoc Networks (MANETs) is a network formed on the fly by a group of mobile nodes. In the case MANETs as well, if the node density (number of mobile nodes in a given area) is low and the mobility of the nodes is high, the continuous end-to-end connectivity is not guaranteed. So a new network paradigm is required to bridge the gap for unavailability of end-to-end connectivity.

Delay or Disruption tolerant networking (DTN) assumes point to point connections at any given instance of time and thus provide interoperable communications for networks that possess challenging environments [2]. In DTN, the messages are stored in the mobile nodes, carried by the mobile nodes in the network and then forwarded when connectivity is established between two nodes. It provides communication mechanism to transfer data between the end nodes. Thus with store, carry and forward paradigm, DTN provides end-to-end data transfer, irrespective of the presence of end-to-end connectivity. DTN is for challenged network scenarios that has delays, bandwidth limitations, losses, node mobility etc. in which present internet fares badly. The increased computation power of the mobile devices provide opportunity to create applications over delay tolerant networks.

VoIP is real time voice communications that is typically based on best-effort UDP/IP protocol stack. Mobile users using VoIP solutions may experience dropped calls or lost speech because of packet loss, varying throughput and delays during network congestion. This could happen either in mobile network infrastructure or mobile ad-hoc networking environments. This problem could be addressed by regularly upgrading the infrastructure or increasing the utility of the existing infrastructure. Utility of the network infrastructure could be maximized by utilising the available resources fairly [5].

DTN-based voice messaging and half duplex asynchronous push-to-talk provide a recent paradigm for improving the utility of the applications. In the next subsection, we discuss how adaptivity increases the utility of the applications. As explained before, delay is the key metric to trade off with losses, congestion and varying data rate. DTN-based voice messaging provides communication in networks with delays and for networks with unavailable end-to-end connectivity. In the thesis, we present a system and algorithm design that dynamically adapts the voice data units size as a function of the observed connectivity characteristics such as delays and losses. The system swiftly adapts between synchronous VoIP and DTN-based asynchronous voice messaging.

1.1 Application Classification

Depending on time criticality, the applications are classified into elastic (non-real time) and inelastic (real time) applications, as shown in the figure 1.1 given by Larry Peterson and Bruce Davie [45]. The classification is based on Integrated Services in the Internet Architecture[RFC 1633] [4] by R. Braden, D. Clark and S. Schenker. In the elastic applications such as FTP, email and instant messaging etc. the operation range in terms of time are broad and thus the adaptivity to delays of few seconds or minutes are acceptable. The Quality of Service (QoS) requirements like delays, data losses and data rate are flexible in elastic applications. In the elastic applications with acceptable delays, the adaptivity is achieved by transport protocols such as TCP but on the other hand the packet losses are not acceptable in the elastic applications.

Inelastic or real time applications are tolerable to the loss of packets. Depending on the tolerance to the packet loss, real time applications are classified into tolerant and intolerant applications. On-line stock trading, remotely controlling a car etc.

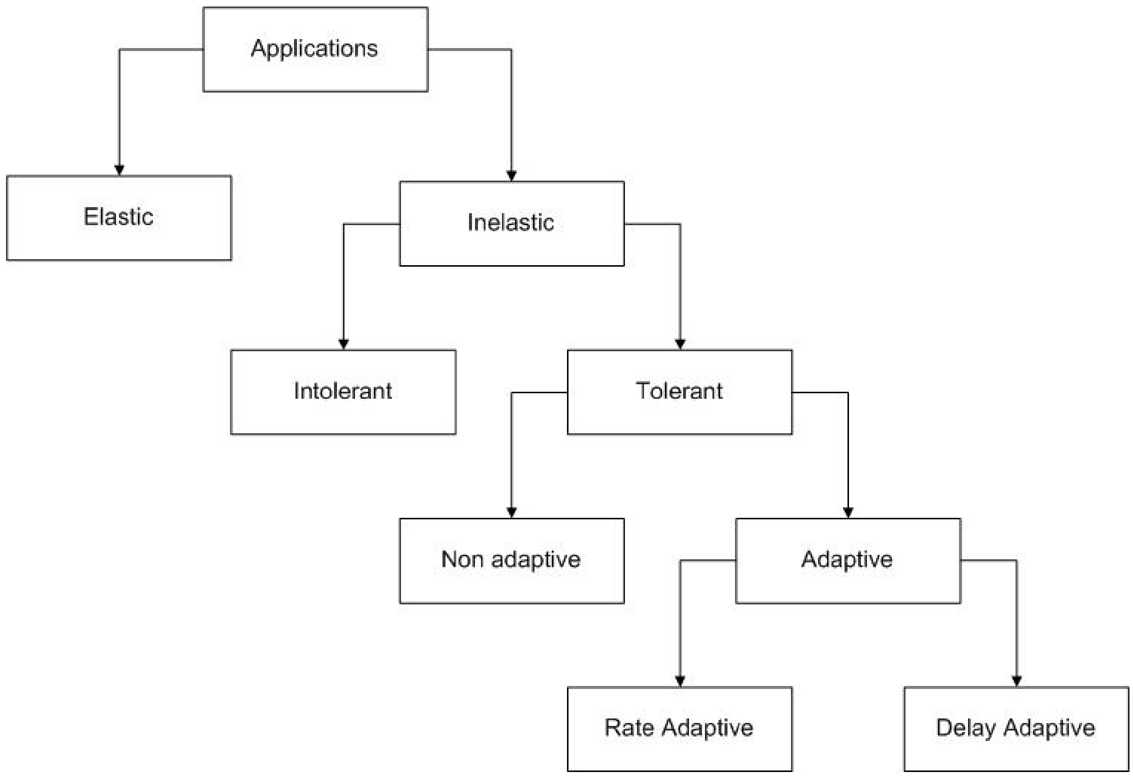


Figure 1.1: Classification of Applications

are intolerant real time applications in which packet delivery should be guaranteed with stringent constraints on bandwidth, delay and jitter. The applications such as VoIP, audio/video streaming are sensitive to delays but they are tolerable to the limited packet losses. These applications are tolerant real time applications and they usually run over UDP. The tolerant real time applications are classified into two types depending on their adaptability to delay and data rate. The applications such as VoIP call which are adaptive to the playback time point are delay adaptive real time applications; where as applications such as video streaming which are adaptive to data rate by changing the coding scheme are rate adaptive real time applications.

1.2 Utility of Applications

The utility of an application is the perceived performance of the application in varying network conditions. In this section, we assume that the varying network conditions are proportional to the bandwidth and we discuss the utility of elastic, inelastic and adaptive applications. Utility functions are used to evaluate perceived user experience for the applications and also to quantify the fairness in usage of resources. As we have seen in previous sections, the applications are divided into elastic and inelastic applications.

The utility function in terms of bandwidth for elastic traffic is

$$u(b) = 1 - e^{\frac{kb}{b_{max}}}$$

$b =$ allocated traffic
$k =$ positive constant
$b_{max} =$ maximum bandwidth required

As shown in figure 1.2 [35], the utility curve for elastic applications is concave with no minimum bandwidth requirement. The elastic applications can tolerate larger delays and during congestion periods the packets could be cached at the source; so no minimum bandwidth requirement. And so, as the bandwidth increases then the curve is strictly concave which shows decrease in marginal utility.

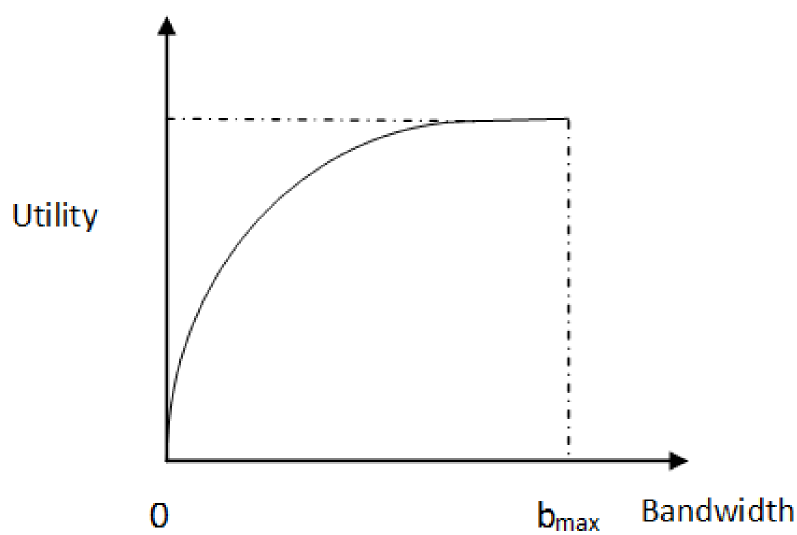


Figure 1.2: Utility of Elastic Applications Traffic

The utility function for intolerant real time applications is

$$u(b) = \begin{cases} 1, & \text{when } b \geq b_{min} \\ 0, & \text{when } b < b_{min} \end{cases}$$

$$b_{min} = \text{minimum bandwidth required}$$

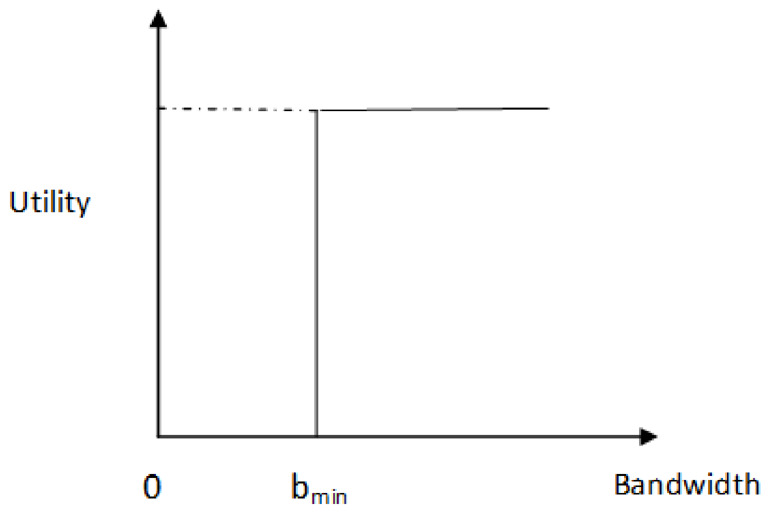


Figure 1.3: Utility of Inelastic Applications Traffic

The intolerant real time applications have stringent constraints on bandwidth and delays, as shown in figure 1.3 [35]. So in the case of a voice call, the call is permitted only if minimum bandwidth(b_{min}) is available. And once minimum bandwidth is available then the utility of the application is maximum. In case of real time applications in network conditions where availability of end-to-end path varies, that effects the bandwidth, the utility of the applications might fluctuate effecting the cumulative utility. In case of voice call, it could be complete incomprehensible voice quality. So it might be useful to broaden the scope of intolerant real time applications by introducing adaptability.

The utility function for adaptive real time applications is

$$u(b) = 1 - e^{\frac{k_1 b^2}{k_2 + b}}$$

b = allocated traffic
k_1, k_2 = positive constant
b_{max} = maximum bandwidth required

As shown in the figure 1.4 [35], the utility in the adaptive traffic has an intrinsic value(b_{intr}), below which the application performance drops sharply. The performance is unacceptable below a minimum bandwidth(b_{min}). The utility curve is convex before b_{min} and becomes concave after b_{intr}. The marginal utility is low at very small bandwidth and at very high bandwidths. At very small bandwidths, the perceived quality is completely unacceptable and at very high bandwidths, the perceived quality is better than needed; thus the marginal utility is low.

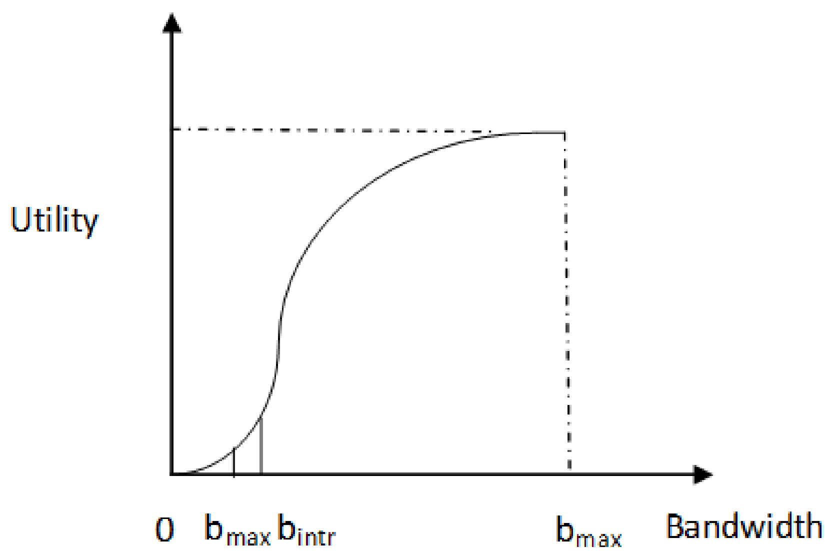


Figure 1.4: Utility of Adaptive Applications Traffic

And also studies on Skype users [9] shows that users are more tolerable to delays than the standards suggested by ITU-T. Studies shows that voice quality is more important than the communication delay.In the case of voice applications, especially if there is failure in end-to-end communication path or high node mobility, having an adaptive approach might increase the cumulative utility for the user.

1.3 Benefits of Ad-hoc and Opportunistic Networks

Both mobile ad-hoc networks and opportunistic networks are decentralized, self configurable wireless networks that works with out the support of infrastructure. The main difference between them is routing in opportunistic networks do not expect end-to-end path and works in the case of high node mobility as well. Due to increase in mobile device intelligence and pace of innovation in the mobile devices, the real applications over ad-hoc networks and opportunistic networks could be poised to take off in coming years [10]. The application markets places such as Android play, Apple App Store, Windows Phone App Store etc. are providing a good ecosystem for the development of applications.

Moreover, the ad-hoc and opportunistic networks provides communication during a crisis situations such as natural disasters. These networks provide communication link between crisis hit area and the internet. This would greatly improve the efficiency of emergency rescue teams. There are situations where access to certain services/content are blocked in the form of censorship by the organizations. These networks enable people to communicate in such situations. These networks provide wide range of applications such as Rural tele-medicine networks [50], new type of content sharing that is geographical bounded but decoupled with time; namely floating content [30], voice message phone [20] and many others. Certain amount of altruism [24] from the users is required as they are sharing their own resources, but there are not much studies on this to analyse and enable users to be part of these networks. In economic benefits point of view, the main advantage of ad-hoc and opportunistic networks is negligible capital expenditure(CAPEX). It provides business opportunity for new entrants as there is no huge investment to set up a network. As the network is self organized, self healing and easy to maintain; the operational expenditure(OPEX) could also be low for the operators.

1.4 Problem Statement

The end-to-end performance of voice applications is dependant on delay, losses and jitter. Jitter, which is variances in delay, could be handled by having play-out buffer which induces more delay. Packet losses could be reduced by retransmissions or by Forward error correction(FEC) mechanisms which induces more delay. Thus delay is the key metric for the voice applications.

In case of lack of end-to-end path or high delays, the utility of the interactive real time voice applications is very low and often it is zero. The utility could be improved by introducing the adaptivity in interactive real time voice applications. Presently, most of the adaptivity in real time applications are rate adaptation by switching between coding schemes, FEC and retransmission procedures. But all these procedures induce more delay and might not work in stringent delay requirement for real time voice communications. Delay ranges specified by ITU-T are; less than 150ms one way delay is preferable, 150-400ms one way delay as tolerable and above 400 ms as unacceptable. But adaptivity could be extended beyond the stringent operating conditions of inelastic applications and pushing the boundaries into elastic applications. Thus increasing the cumulative utility for the user by enabling more flexible communication. If the synchronous voice communication is not possible then switching to asynchronous voice messaging would provide better utility for the user.

1.5 Goal and Scope of the Thesis

Currently voice applications are either real time voice applications or asynchronous push-to-talk styled voice messaging applications. The objective of the thesis is two fold. First objective is to present a delay tolerant adaptive media algorithm that swiftly switches between synchronous real time voice communication and asynchronous voice messaging according to varying network conditions. The second objective is simulation studies that evaluates the performance of the adaptation algorithm. The performance of the algorithm is evaluated in two different network conditions; one is 3G scenario by collecting Helsinki traces and the other is ad-hoc communications scenario. The performance metrics to evaluate the algorithm are R values of the E-model calculations and their differences with a reference.

The thesis is focussed on providing voice communications with better utility for the user in the varying network conditions. Specially, we focussed on designing an algorithm to provide DTN-based voice messaging as fall back procedure when real time communication fails. The main focus of implementing a real world prototype model is left for the future work.

1.6 Thesis Outline

In this introductory chapter, we have discussed about the classification of applications and utility of the applications. It discussed the utility of elastic and inelastic applications, with further discussion on how adaptation could be helpful in broadening the application utility. It introduced socio-economic benefits of ad-hoc and opportunistic networks, problem statement, goal and scope of the thesis.

Chapter 2 presents overview of the background of the thesis. It introduces Delay Tolerant Networking architecture and protocols involved. It gives overview of synchronous and asynchronous voice communications. It briefly explains about the related work on DT-Talkie and Disconnection Tolerant SIP. At the end, it discusses about the VoIP quality measurements and E model.

Chapter 3 proposes Delay Tolerant Adaptive Media algorithm to adaptively switch between real time communication and message based half duplex communication in varying network conditions. It proposes application level network independent algorithm.

Chapter 4 presents the results of the Delay Tolerant Adaptive Media algorithm in 3G scenario and Mobile ad hoc network scenario. It analyses the results and presents how the algorithm performs in different scenarios.

Chapter 5 draws final conclusion of the thesis, summarizes the results and discusses the future work.

2 Background and Related Work

In this chapter, we give an overview of relevant technology background and related work in the context of the thesis. We introduce Delay and Disruption Tolerant Networks and discuss its underlying store-carry and forward principle, architecture and packet structure. After that we present the protocol stack related to the Synchronous VoIP. Then, we present Asynchronous Voice Messaging options like Push-to-talk, Push-to-talk over Cellular, Peer-to-Peer Push-to-talk. The relevant Disconnection Tolerant SIP (DT-SIP) and DT-Talkie are discussed briefly. Finally, we introduced voice quality measurement techniques (R Vaues of E model) that we have used for the evaluation, before summarizing this technology background and related work section.

2.1 Delay and Disruption Tolerant Networks

Today's Internet operates poorly in challenged networks that are characterized by long delays and frequent disruptions. Delay/Disruption tolerant networking is an emerging interoperable communications that works in the case of networks with inadequate performance characteristics. It is particularly useful when there is no end-to-end path availability and long delays (delays of minutes, hours and even days). It uses asynchronous messaging with hop-by-hop reliability and works across different networks. Some of the factors for which existing Internet probably fails and DTN is viable options are summarized below.

- Transmission, propagation, and queuing delays are affected by the network conditions. A network with high latency and queuing delays.
- A network with frequent disconnections/partitioning and end-to-end path is unavailable between the source and destination.
- A network with asymmetric and low data rates, in which return path might be unavailable.
- A network with significant node mobility, in which hop-by-hop connections between nodes and contact times are unpredictable.

2.1.1 Message Forwarding

Routing or message forwarding in Mobile Ad-hoc Network (MANET) assumes that there is end-to-end path connectivity between the source and the destination. MANETs uses routing protocols in which messages are relayed if there is availability of end-to-end path, otherwise the message forwarding is postponed. The routing protocols in MANETS are devided into three types. They are

- Proactive protocols such as Dynamic Destination Sequenced Distance Vector (DSDV), Wireless Routing Protocol (WRP), Clusterhead Gateway Switch Routing protocol (CGSR), Optimized Link State Routing Protocol (OLSR) etc. are table driven protocols; in which each node maintains the routing table and updates the table even when messages are not being forwarded.
- Reactive protocols such as Ad hoc On-Demand Distance Vector (AODV), Temporally Ordered Routing Algorithm (TORA), Dynamic Source Routing (DSR) are demand driven protocols; in which routes are calculated when needed.
- Hybrid protocols such as Zone Routing Protocol (ZRP) are combination of table driven and demand driven protocols. They dynamically calculate routes for far-away nodes and maintain the routes for near-by nodes even if not needed.

Consider Ad-hoc On-Demand Distance Vector (AODV) routing protocol in ad-hoc networks. It is an on-demand reactive routing protocol. The three message types used by AODV are Route Request (RREQ), Route Reply (RREP) and Route Error (RERR) messages. Each node broadcast the HELLO messages at regular intervals of time and thus nodes keep track of the neighbouring nodes. Route Request (RREQ) messages are broadcasted by the originator node for the request of the route. The RREQ message contains information such as source address, destination address, sequence number, message lifetime, hop count, unique id etc. The nodes that receives RREQ messages updates the information of the originator in their routing tables, broadcast the message to neighbouring nodes and sets a reverse path. When the destination receives the RREQ message, it replies with RREP message to the originator. Thus the route gets activated. If an intermediate node disappears or there is a break in the link, then Route Error (RERR) messages are sent to the originator.

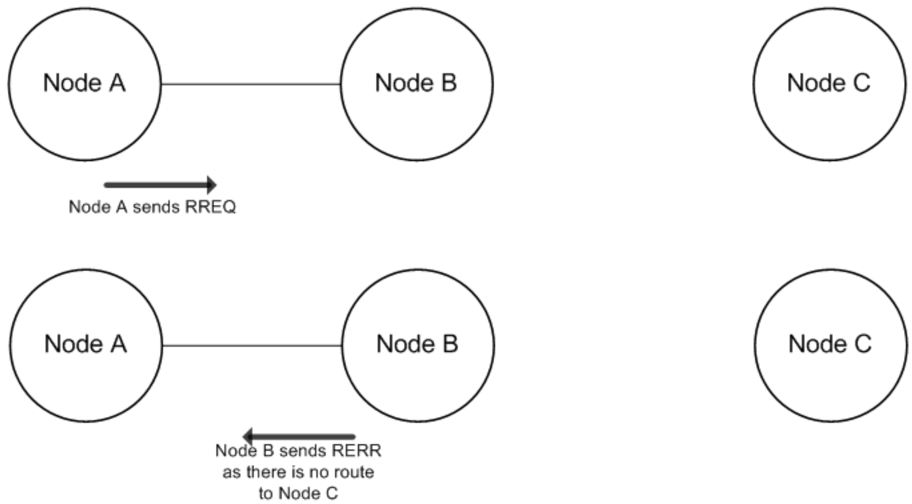


Figure 2.1: AODV Routing

As shown in the figure 2.1 [1], in case of network disruptions, the nodes initiate the route discovery mechanisms and data could be transferred only when entire end-to-end route is possible. Consider node A and node C are not in the communication range of each other. If the node B is mobile and connected to either node A or node C at any given point of time, AODV might never form a route. Thus in case of mobile nodes, AODV does not ensure route and further data delivery in MANETs is not ensured. In the case of DTN, it uses store-carry and forward principle for message forwarding. The messages in DTNs are called bundles. As we can see in the figure 2.2, node A has to send a bundle to node C. Node A and node B are within the range at time t=0, node A is within the communication range of B, but node B is not in the communication range of C. So A sends the bundle to B. At time t= T, B moved away from communication range of A and node B is within the communication range of C. Reaching with in the communication range of C is called contact. When node B is in contact with node C, the bundle is forwarded to the node C. Thus the bundle is forwarded from the source to the destination by store, carry and forward principle. The reliability of data is ensured to sender via bundle received acknowledgements from the receiver. This ensures the reliable data delivery in DTN. A DTN node can accept a bundle and promise to store it in its data storage until the bundle is forwarded. Node's willingness to store the bundle is custody of the bundle and taking responsibility to transfer is custody transfer.

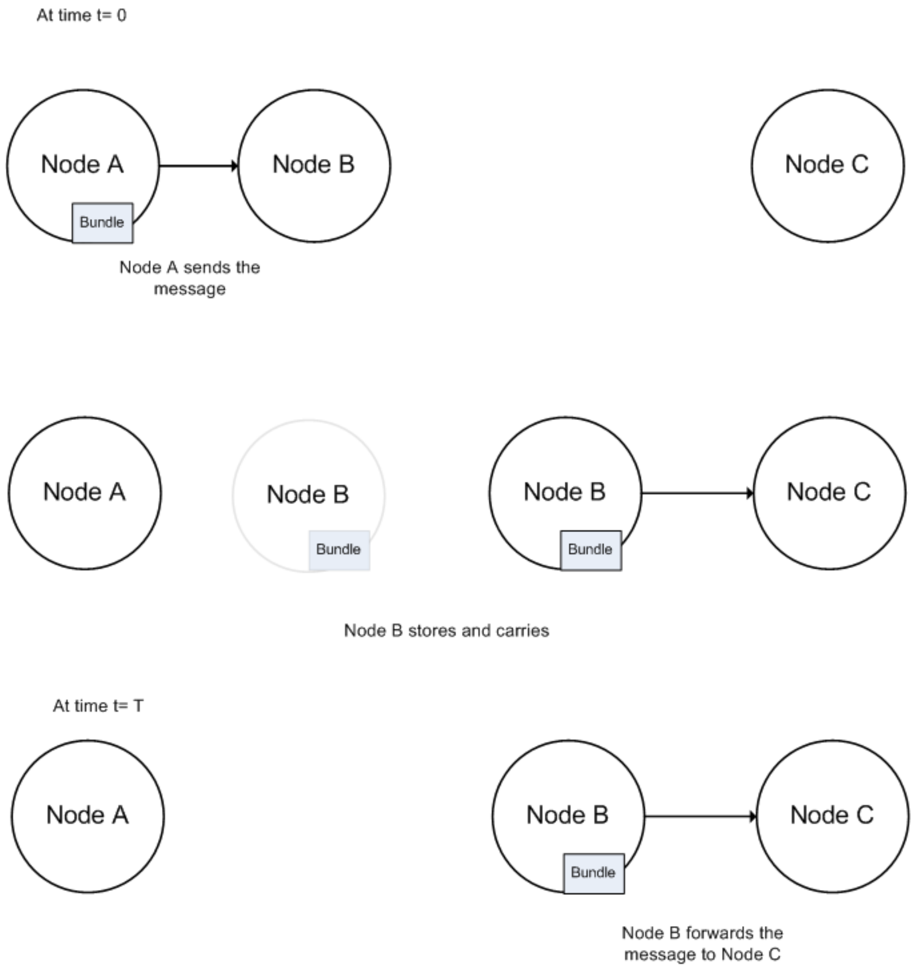


Figure 2.2: Store, carry and forward messaging

2.1.2 DTN Architecture

The DTN architecture [16] [7] has been developed to tolerate long delays and disruptions. It is described in RFC 4838 and the architecture was emphasized from Interplanetary Internet. In addition to TCP/IP, the architecture provides support for different networks such as raw Ethernet, serial port, usb storage, blue-tooth etc. The interoperability between different networks is provided by convergence layer adapters (CLAs). The DTN messages or protocol data units are called DTN bundles. During the transmission, the bundles could be fragmented into fragment bundles [46]. As shown in the figure 2.3, the network layer above the convergence layer is the bundle protocol layer. In order to carry the bundles, the DTN nodes uses persistent storage to save the bundles in a node.

The source and destination of the bundle application are identified by endpoint

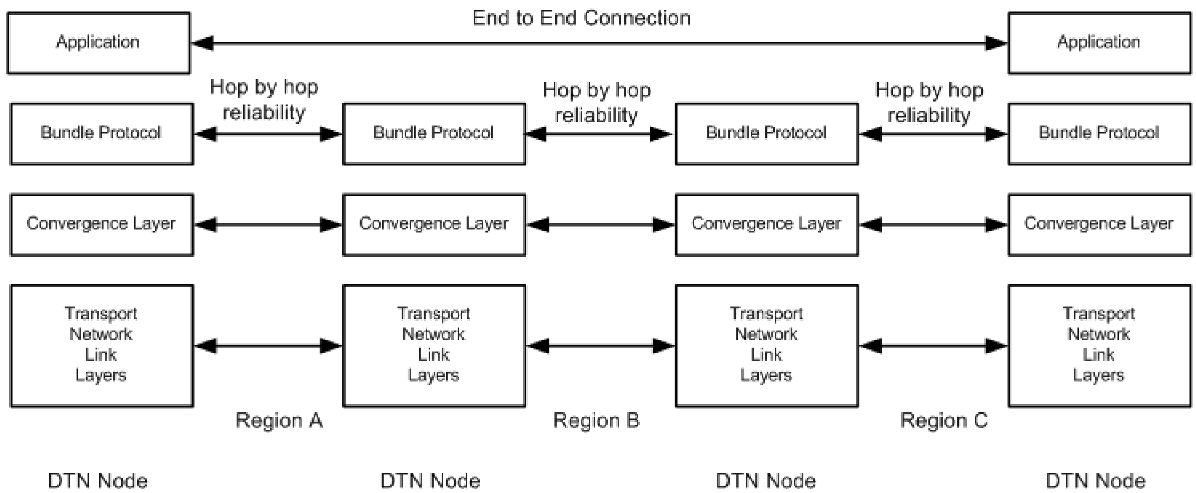


Figure 2.3: DTN Architecture

Identifiers. Endpoint Identifier (EID) is unique for every DTN node and it follows Uniform Resource Identifier (URI) syntax. The form of URI is ⟨scheme name⟩ : ⟨scheme specific part⟩. If a DTN node is willing to receive the bundles, it performs a registration of its EID. EIDs are not mandatory for routing, but they do provide convenient interpretation and process of mapping EIDs to network address is called binding. The example EID for DTN is

dtn://node-id/audio-id

Bundle protocol layer implements bundle protocol described in RFC 5050 [54]. The node that offers bundle protocol services is called bundle protocol agent. The bundles are comprised of blocks. Figure 2.4 shows the primary bundle block. The primary block consists of version, flags to process the bundle, block length that contains the aggregate length remaining fields in the block, source and destination offsets that contains dictionary byte array of the scheme name of the endpoint identifiers, source and destination SSP offsets that contains dictionary byte array of the scheme-specific part of the endpoint identifiers, report-to scheme and SSP offsets for status reports pertaining to forwarding and delivery of the bundle, custodian scheme and SSP offsets for current custodian endpoint identifier, creation timestamp which is a pair of SDNVs that consists of bundle's creation time and bundle's creation time-stamp sequence number, lifetime of bundle, length of the dictionary byte array, dictionary, fragment offset that indicates start of the original application data unit and total length of the original application data unit. The fields with (*)

in the block are of variable length and represented using self delimiting numerical values (SDNVs).

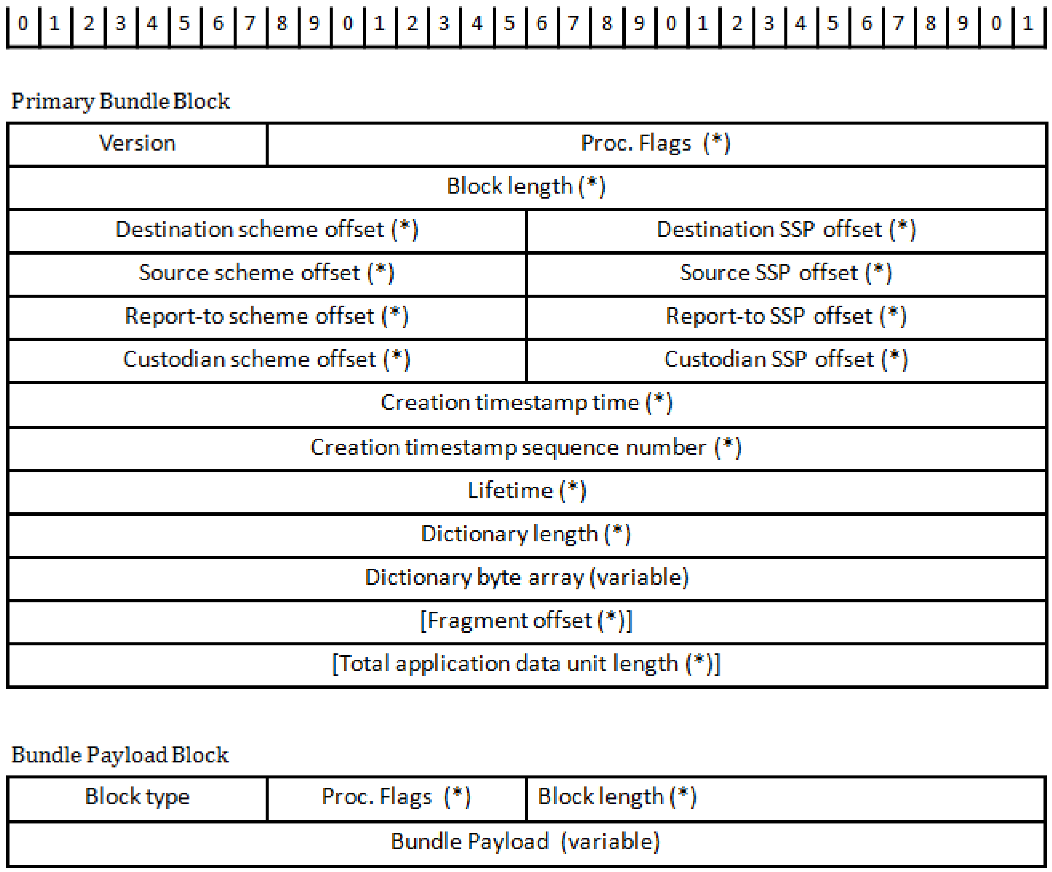


Figure 2.4: Primary Bundle Block

2.2 Synchronous VoIP

Synchronous VoIP traffic is typically a real time voice communication based on RTP/UDP/IP protocol stack. Call Processing of a VoIP call between two end points involves series of steps and multiple protocols are involved. The call control and signaling methods that establish, modify, and terminate a call is based on signaling protocols like Session Initiation Protocol (SIP), H.323 [57] etc. Once the call session is established, the voice traffic is transported via media transport protocols of which most used one is Real-time Transport Protocol (RTP). During RTP traffic flow, the quality of voice is monitored and reported via Real Time Control Protocol (RTCP) reports. For better voice quality and network utilization, various codecs such as

G.711, G.729, G.723 etc. are used.

2.2.1 Session Initiation Protocol

Session Initiation Protocol (SIP) [49] is the text based application level signaling protocol. SIP is used to establish, modify and tear down the media sessions between two or more users. VoIP calls, multimedia distribution and conferences make use of SIP. It is based on request/response based model. The basic functionality of the SIP includes locating the endpoints to initiate the session, inviting the other end user agent to start the session, negotiating the capabilities of each end points, establishing the session and tear-down. The typical components of the SIP are SIP user agents and SIP servers. User agent client (UAC) is the end point that initiates the SIP transactions and User agent server (UAS) handle the SIP requests by response. Registrar, proxy, location and redirect servers are the types of SIP servers and their functionality is described in the table 1.

Server	**Functionality**
Registrar	It accepts register requests from the UAC. UAC provides IP address and SIP URI to the Registrar. Registrar stores and keeps track of registered user addresses.
Proxy	It acts as an intermediate message router between the UAC and UAS
Redirect	It accepts SIP INVITE request from the caller, retrieves the address of the callee and replies to the caller with the callee's correct SIP address
Location	It provides the callees correct SIP address to the Proxy and redirect servers.

Table 1: SIP Server Elements

Figure 2.5 shows the SIP call flow between two user agents A and B. Both user agents A and B should be registered to a SIP registrar which helps in node discovery. To being the call, user agent A sends an INVITE request to user agent B via SIP Proxy Server. SIP Proxy server forwards the messages to respective user agents. Before forwarding the invite request to user agent B, the proxy server send 100 Trying message to user agent A. The invite request could probably have the list of parameters like codec type, payload type, port etc supported by the caller. Session description protocol (SDP) is used to convey the list of parameters in the invite request. SDP is part of SIP message body and is used for capability negotiations between the SIP user agents. The user agent B responds to the request with 183 session progress response code, conveying that call has reached the user and waiting for the user to pick the call. When the user picks the call, the user agent B responds to the request with 200 OK response code and may also contains parameters it supports. The user agent A replies to the user agent B with ACK, which acknowledges that the 200 OK response has been received. That completes the three way handshake and establishes the dialog. The the media session consists of Real Time Protocol (RTP) data that encapsulates the real time data as payload with in the RTP packets. The RTP packets are sent between both the user agents. After the media session ends, either of the user agents can start the session termination by sending the BYE message and the other end user agent responds to the BYE by 200 OK response code.

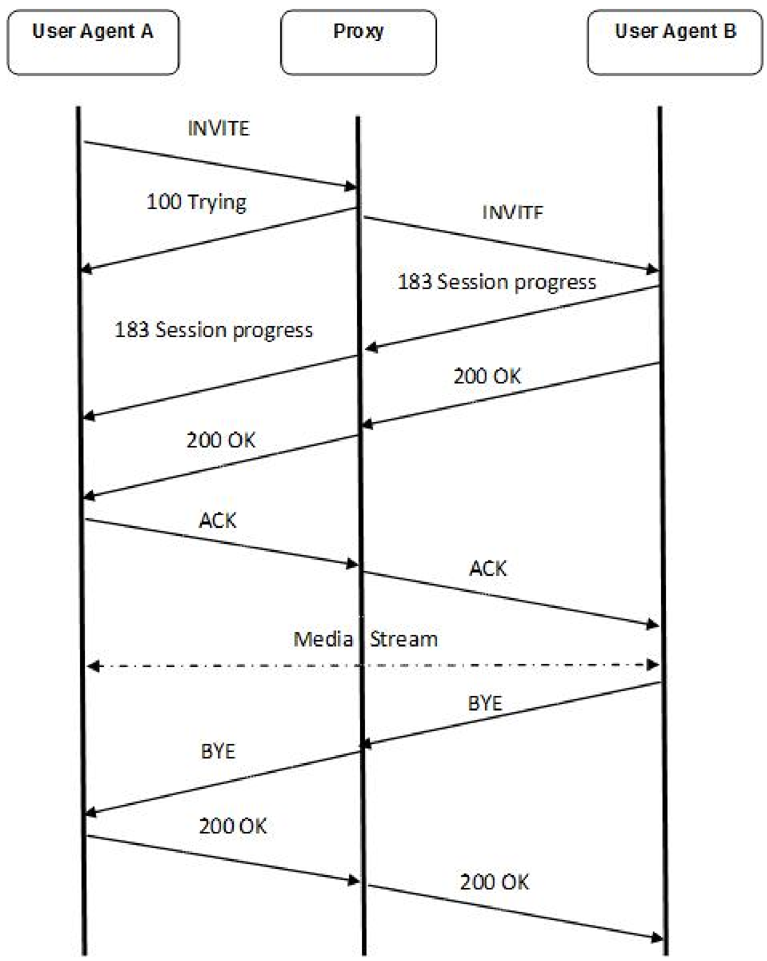


Figure 2.5: SIP Call Flow

The typical SIP request message consists of start line, message headers and message body. The start line consists of information whether it is a request or response, SIP URI and version. The message headers consists of control information and associated variables. The message body typically consists of Session Description Protocol (SDP) messages that provides the media capabilities of the user agents. The following table 2 lists the typical SIP request messages.

SIP request method	**Functionality**
INVITE	Request to establish a call session
REGISTER	Request by user agent to register its location with registrar server
BYE	Request to terminate a call session
CANCEL	Request to cancel pending requests
ACK	Acknowledges the INVITE/200 OK handshake
NOTIFY	Notification about the state changes

Table 2: SIP Request Messages

The user agents are identified by the SIP URIs. The SIP URI format is as follows sip/sips scheme:user:password@host:port;parameters?headers
The SIP requests are responded by SIP response codes. They are similar to HTTP response codes. SIP response codes are of six categories and they shown in the table 3

SIP response code	**Functionality**
1xx	Informational response
2xx	Success response
3xx	Redirection response
4xx	Client error response
5xx	Server error response
6xx	Global failure response

Table 3: SIP Response Codes

2.2.2 Session Description Protocol

Session Description Protocol (SDP) is used for session characterization, time description and media description. It is used by endpoints to negotiate the capabilities like ports, timings, media type, media formats etc by offer/answer exchanges. Using offer/answer model, the user agents are agreed up on their respective capabilities, media formats, timings etc. The message format is ⟨type⟩=⟨value⟩.⟨type⟩ is exactly one character. The table 4 shows the SDP message types.

Type	Functionality
v=	Version
o=	Originator + unique identifier
u=	URL
e=	email
b=	bitrate
z=	time zone adjustment
a=	attribute lines (extensions)
m=	media and port attribute lines
c=	connection information (IP address specification)
t=	Start, end time (NTP seconds, special case: 0, 0)

Table 4: SDP Message Types

The connection field "c=⟨network type⟩⟨address type⟩ ⟨connection address⟩" contains connection information and the session announcement should consists of one "c=" field either at session level or in each media description. The field network type is a string for type of network (IN of Internet), address type is a string to specify the type of address (IP4 for IPv4 addressing) and connection address is typically class D multicast address for IP4. The attributes "a=⟨attribute⟩:⟨value⟩" are used to extend the SDP. Typically media descriptions may have multiple media level attributes to convey additional information regarding the media. The field "m=⟨media⟩ ⟨port⟩ ⟨transport⟩ ⟨fmt list⟩" consists of media descriptions. The sub-field media type is used to define the media such as "audio", "video", "application", "data" and "control" etc. The port sub-field specifies transport port to which media should be sent. The transport protocol sub-field specifies the type of protocol being supported(RTP/AVP for Realtime Transport Protocol with Audio/Video profile carried over UDP). And the final sub-field specifies the media formats defined in RTP Audio/Video Profile.

2.2.3 Real Time Protocol (RTP)

Real time protocol (RTP) [52] is real time end-to-end data transport protocol for interactive audio and video applications. It is application level protocol and typically runs on the top of UDP transport protocol for multiplexing and checksum. RTP consists of real time information that would help to adjust the jitter, to measure the packet losses and to rearrange the out of order packets. The RTP header is shown in figure 2.6.

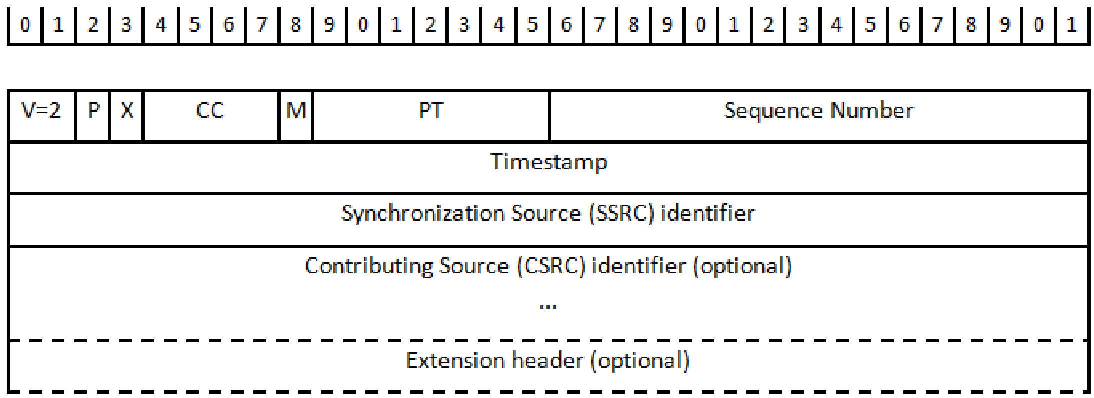


Figure 2.6: RTP Header

The fields are explained in table 5. The RTP header extensions are used to include custom individual implementations as a part-of to the RTP data header.

Field	**Functionality**
Version (V)	Version of the RTP
Padding (P)	Padding bit is set if payload consists of additional padding octets
Extension (X)	It is set if fixed extension header follows
CSRC count (CC)	Number of CSRC identifiers
Marker (M)	Frame boundaries bit
Payload type (PT)	Format of the RTP payload
Sequence number	Increment of sequence number by one for each RTP packet
Timestamp	Sampling instant of the first octet
SSRC	Synchronization source identifier
CSRC list	Contribution source identifiers list

Table 5: RTP Header Fields

RTP Control Protocol (RTCP)

RTP Control Protocol (RTCP) is used to exchange the control information regarding the RTP session. It is the periodic transmission of the control information. It is used to monitor the quality of RTP session like jitter, packet loss, round trip time etc. Different types of RTCP message types [17] are shown in the figure 2.7 and explained in table 6.

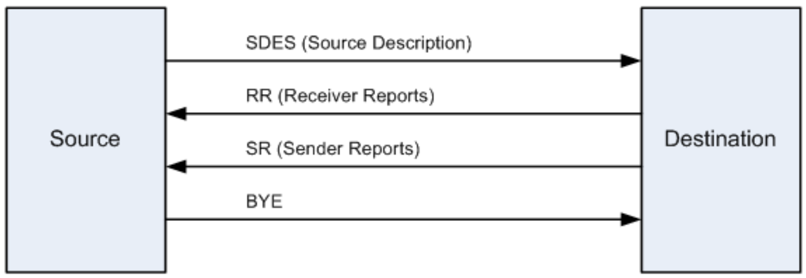


Figure 2.7: RTCP Messages

RTCP Report Type	**Functionality**
SR	It is sender report,that consists of transmission and reception statistics from the active senders
RR	It is receiver report that consists of reception statistics from the passive participants.
SDES	Source Description that consists of identity of the users.
BYE	End of participation

Table 6: RTCP Message Types

The RTCP SR report consists of SSRC of the sender to identify the source from which the receiver receives the report. It contains RTP timestamps for media synchronization of playback and NTP timestamps for calculation of round trip times. The packet count and octet count of the sender are used to calculate average data packet rate and average payload data rate respectively. The RR block starts with the sender identifier. It contains the cumulative number of packets lost and fraction lost since the previous RR report. They are calculated with the help of sequence numbers. The receiver calculates the jitter and sends it to the sender via RR report block. The RR report block also consists of highest sequence number received and sender report information. The RR report block is shown in figure 2.8

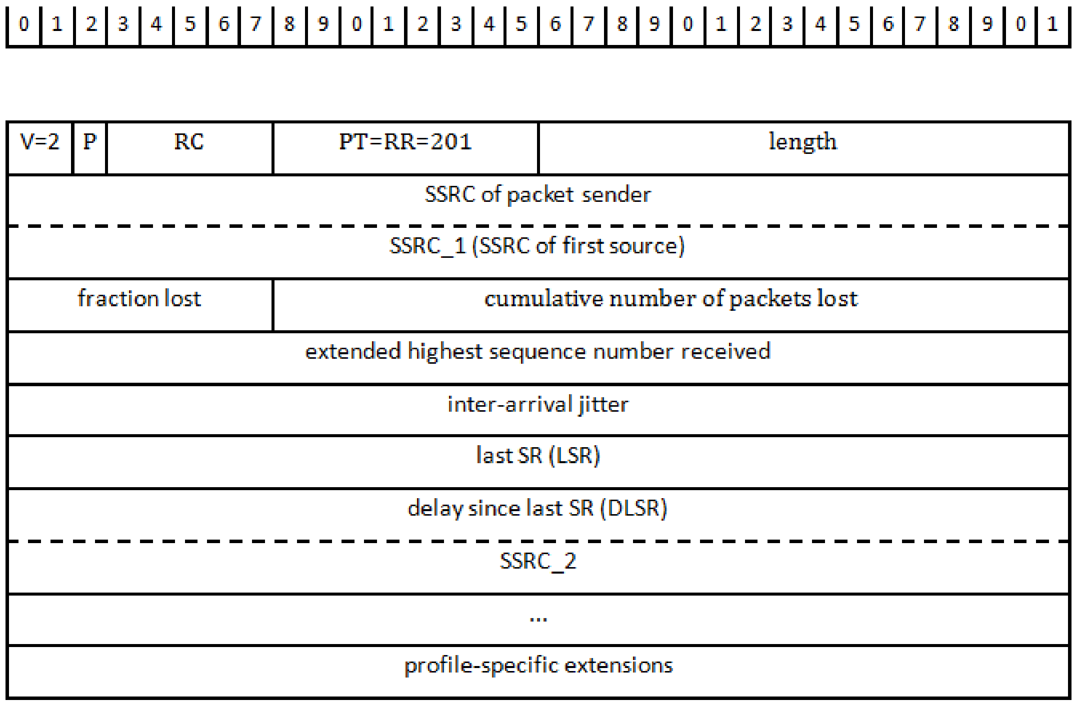


Figure 2.8: RTCP Receiver Report

2.2.4 RTP Audio Video Profile

RTP/AVP [51] provides the information about the audio and/or video profiles of data present in the RTP packets. In the case voice data transmission over network, the voice is first sampled, quantized and coded using a codec. Then the coded samples are buffered into audio frame data. Then the audio data is used as payload for the RTP and sent over the IP network. The RTP audio frame is shown in figure 2.9

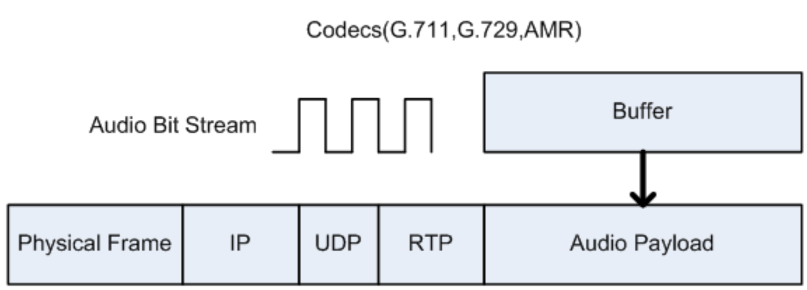


Figure 2.9: RTP Audio Frame

The receiving end receives the RTP packets and places the packets into the play-out buffer. Play out buffer helps the receiver to collect minimum number of packet to play the voice stream. The receiver may provide feedback to the sender using RTP/AVPF which is Audio Visual Profile with Feedback (AVPF). The RTP/AVPF is extended RTP profile for RTCP based feedback mechanism to the sender and for feedback-based error repair though retransmissions, FEC etc. The receiver collects the statistics and reports those reception statistics to the sender using RTCP.

2.3 Asynchronous Voice Messaging

Asynchronous voice communication does not have a timing relationship between the two end points that are involved in a voice call. It consists of sending/receiving voice messages between the endpoints and provides acceptable time delay in between. Typical examples of asynchronous voice communication are voice mail/messaging, Push to talk (PTT). The major drawback of Asynchronous voice communication is not being synchronous. But if synchronous voice communication fails then it provides an excellent fall-back option. Voice messaging is voice data sent as messages to the destination.

2.3.1 Push-to-Talk

Push to talk [60] [44] is walkie talkie style communication, that provides half duplex voice communication between two end points or within a group. The original push to talk radio communications are geographically limited because of direct communication between the radios. PTT session could be between two users or among a group of users. During group conversation, one user talks at a time while rest of the group members listen and take turns one after the other. It consumes significant amount of radio resources due to involvement of multiple users and half duplex voice transmission.

2.3.2 Push-to-Talk over Cellular

Push-to-talk over Cellular (PoC) [38] provides walkie-talkie like services over cellular wireless networks. PoC works over the top of IP protocols and provides half duplex voice communication. The service is not geographically limited but they are dependant on the service provider. Open Mobile Alliance (OMA) provides standardization of PoC under IP Multimedia Subsystem (IMS).

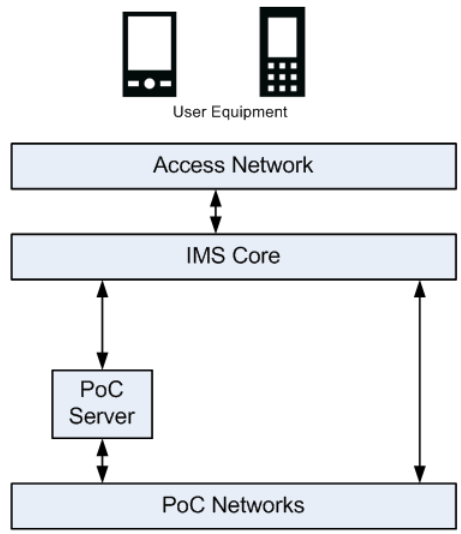


Figure 2.10: PoC Architecture

The users of the PoC (PoC client) are identified by unique SIP URI. As given in OMA PoC specialisations, it uses centralized PoC server for coordination among the users, Session Initiation Protocol (SIP) for call signaling and RTP/RTCP for media transport. As shown in figure 2.10, OMA PoC architecture is centralized architecture with single point of failure. All the responsibilities of call signalling and voice transport are on the central PoC server. The issues regarding the architecture are service provider dependency, scalability, reliability and cost for both users and the operator.

2.3.3 Peer-to-Peer Push-to-Talk

Peer-to-Peer Push-to-Talk is decentralized system that is cost effective and scalable. Lin et al. have proposed iPTT [34] that is decentralized and service provider independent architecture. iPTT uses P2P Session Initiation Protocol (SIP) defined by Internet Engineering Task Force (IETF) for call signaling, RTP for media transport and RTCP for floor control. It does not depend on the cellular wireless networks and works over mobile ad-hoc networks. As shown in the figure 2.11 [34], iPTT architecture make use of super-nodes that coordinates the PTT clients and call handling for ordinary nodes.

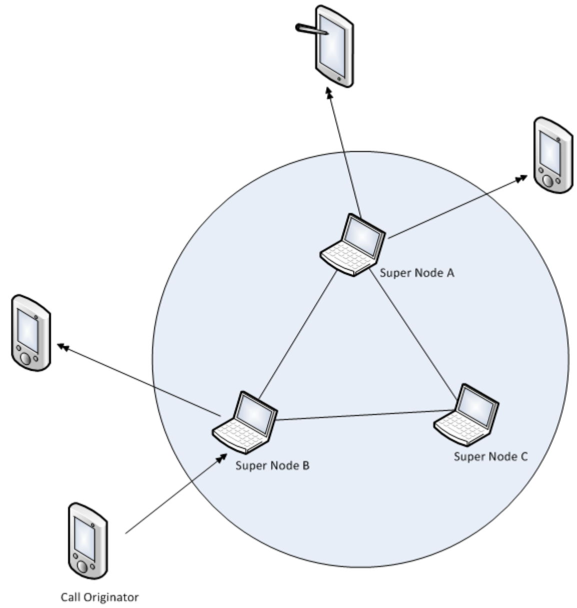


Figure 2.11: iPTT architecture

But the drawback of these methods are they are dependant on end-to-end path connectivity for the call establishment and maintenance. End-to-end path might not be possible in case of higher delays, presence of disruptions in the network and node mobility. This would seriously effect the PTT services over IP networks. The significant work related to overcome this are Disconnection tolerant SIP(DT-SIP) and Delay Tolerant Talkie (DT-Talkie). The following subsection discuss about DT-SIP and DT-Talkie.

2.3.4 Disconnection Tolerant SIP (DT-SIP)

Disconnection Tolerant SIP [43] [61] is an enhancement to the SIP-based communication to be able to work for challenged wireless networks. In the presence of frequent network disconnections during a VoIP call, the session gets disconnected. During the network disconnections, Disconnection Tolerant SIP would keep the session context active during the call. It uses both SIP-based detection and media based detection to discover the network disconnections. And it uses proactive audio recording to overcome audio loss during the session break. In this DT-SIP, both sender and receiver continuously monitor the changes in network conditions with the help of RTP and RTCP reports. The disconnections are classified into short, medium and long disconnections. The short disconnections are recovered by repeating the last talk spurt, medium disconnections are recovered by automatic redialling and for long disconnections voice messages are recorded to send as voice mail.

2.3.5 DT-Talkie

DT-Talkie [28] [29] is asynchronous voice messaging application similar to Walkie-talkie-style interaction between peers or groups, the underlying technology for it is Delay tolerant networks. It uses mobile to mobile communication without any infrastructure and end-to-end path. It stores the voice messages and carries it forward to the receiver or the group. It ensures hop-by-hop reliability using DTN2 protocol stack. It assumes non availability of network connections.

DT-Talkie application was implemented on Maemo-based Nokia Internet tablets using DTN Reference Implementation(DTN2), in which voice messages are captured, encoded and transmitted as DTN bundles. Instead of directly converting voice messages to bundles, the application uses MIME application-layer framing mechanism that allows additional content such as images, vCard etc. to be transmitted in-addition to the voice messages. In order to avoid unnecessary round trips, the application uses different approaches for the codec negotiation. One approach is sending same voice message in three different codecs(G.729, MP3 and Speex) for the first bundle message. The receiver picks the most preferable codec and replies with the selected codec. Other approach is transferring the first voice message in uncompressed PCM format and using the first exchange of voice messages for codec negotiation. Due to delays, the application provides less interactive communication but maintains the quality of speech.

2.4 Voice Quality Measurement

The quality of voice traffic is measured on human Quality of Service(QoS) perception [21] and network conditions. Human perception is evaluation of perceived voice quality by users. Network conditions like delay, packet loss, jitter and the most importantly network design are the parameters that could be measured. The metrics that are used to measure quality of voice according to network conditions are not precise. So integrating the QoS based on perception and network condition provides better metrics to measure the QoS for voice. Quality models integrate the human ratings and networking QoS. ITU-T G.107 [26] recommendation provides E-model as quality model.

2.4.1 E Model

E model provides the quality model for end-to-end telephone call with considerations to various impairments like delay and echoes. It consists of scalar rating of transmission quality. The scalar rating value is R value. The assumption for calculation of R value is that different transmission factors(impairment factors) that are not correlated to each other could be added to psychological factors. The calculation of R value is as follows

$$R = R_o - I_s - I_{d1} - I_{e,eff} + A$$

where as

R_o = signal-to-noise ratio including the noise sources from circuit and room
I_s = Impairments factors with voice signal
I_s = Impairment factors by delay
$I_{e,eff}$ = Effective equipment impairment factor
A = Advantage factor

The R value ranges between 0 to 100. As we can see from the table, R value of 100 is indication of very high quality whereas R value of less than 60 is of poor quality. Good-or-better (GoB) or Poor-or-worse (PoW) are calculated from R value and Gaussian error function.

$$erf(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2}\,\mathrm{d}t$$

$$\text{GoB} \;= 100E\left(\frac{R-60}{16}\right)\%$$

$$\text{PoW} \;= 100E\left(\frac{46-R}{16}\right)\%$$

R-value Rating	MOS Score	GoB	PoW	User satisfaction
90 to 100	4.34	97	0	Very satisfied
80 to 90	4.03	89	0	Satisfied
70 to 80	3.60	73	6	Some users dissatisfied
60 to 70	3.10	50	17	Many users dissatisfied
50 to 60	2.58	27	38	Nearly all users dissatisfied

Table 7: R-Values and User Satisfaction

The R values are mapped to tradition MOS (Mean Opinion scores) as shown in the table 7.

$$\begin{aligned}
&\text{For } R < 0: && \text{MOS} = 1 \\
&\text{For } 0 < R < 100: && \text{MOS} = 1 + 0.035\text{R} + \text{R(R-60)(100-R)} \times 0.000007 \\
&\text{For } R > 100: && \text{MOS} = 4.5
\end{aligned}$$

The drawback of E model is R value calculation does not consider the impairments caused by dynamic adaptations.

2.5 Summary

Delay and Disruption Tolerant Networks provides asynchronous message communication with hop-by-hop reliability in challenged networks. MANETs perform poorly, if end-to-end connectivity is not available. With its store-carry and forward paradigm, DTN delivers messages even in case of end-to-end path unavailability. In the case of both synchronous VoIP and asynchronous voice messaging, typically SIP is used for signaling, SDP for session description/negotiation and RTP/UDP/IP protocol stack for packet transmission. The drawback of these methods is dependency on end-to-end path connectivity for the call establishment and maintenance. As we have seen in the case of challenged networks, having an end-to-end path might not be possible. DT-Talkie and DT-SIP solves the problem by providing communication, with no dependency on end-to-end path connectivity. Due to unpredictability of the network conditions, swiftly switching between synchronous VoIP and DTN-based asynchronous messaging would provides better communication for the user. In the next chapter we present the algorithm for Delay Tolerant Adaptive Media and in the further chapters we present its evaluation in different scenarios.

3 Delay Tolerant Adaptive Media

As we have seen in the previous sections, the various options for the voice communications are synchronous real time interactive voice, half duplex voice messaging and voice-mail. All these are disjoint options for the user. In each of the option, user has to choose any one service at a given point of time. But as the network conditions changes, user has to disconnect one mode of communication to switch to other mode of communication. Delay-tolerant Adaptive Media (DAM) provides solution for the user by switching to most suitable form of communication as the network conditions changes.

As shown in figure 3.1, initially when a synchronous voice call is taking place, the audio packets are sent as RTP/UDP packets with regular RTCP reports which provides the information regarding network characteristics. In asynchronous communications such as Push-to-Talk and DT-Talkie, longer delays are allowed. So when the network characteristics are not suitable for real-time audio, then the system swiftly switches to DTN-based asynchronous voice messaging and reverts back to real time voice when everything gets normal. In the case of DAM, RTCP reports which provides the information regarding network characteristics are embedded inside the DTN bundles and transmitted.

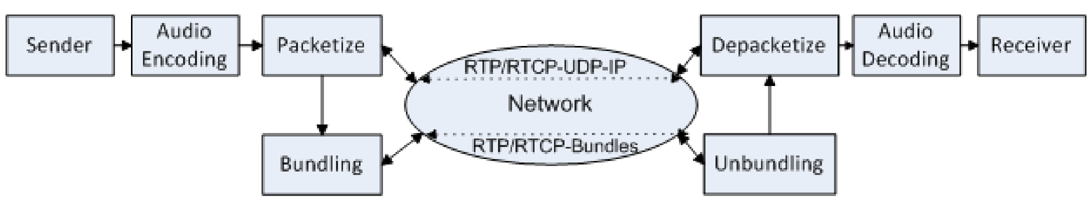


Figure 3.1: System Overview

3.1 DAM Adaptation Algorithm

In this chapter, we discuss about the sender side and receiver side algorithms. Both the algorithms are application level adaptation algorithms and they are based on RTP/RTCP reports. The state diagram for the DAM algorithm is shown in the figure 3.2. If N_ζ consecutive RTCP reports are not received, then it signifies that the network conditions are not normal and possible disconnections in the network. So the mode is switched to DTN. If RTCP reports are received, but if packet losses are greater than p_ν, it signifies that the speech quality is poor. As we have explained in previous chapters, delays are acceptable to the user but not the degradation of speech quality. So in the case of losses greater than threshold value, the DAM algorithm switches to DTN mode. If the one way delay is less than T_κ, then the network is suitable for interactive voice communications. But in case of DTN we calculate average delays in a RTCP interval, so there might be cases in which some packets might not be arrived but average delays would be less than T_κ. In order to avoid false positives, we used probe packets over UDP. If all the probe packets are received with out losses, then it is an additional indicator that the network is suitable for interactive voice communications. The following subsections discuss the sender side and receiver side algorithms in detail.

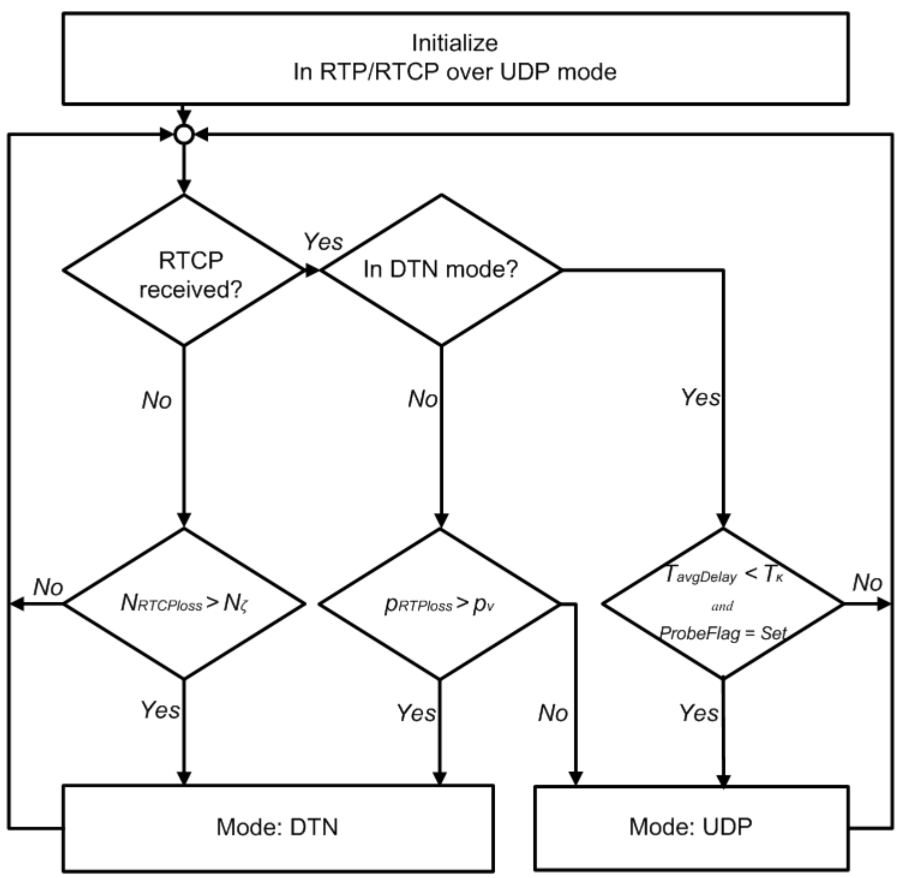


Figure 3.2: DAM State Machine

3.1.1 Sender Side Algorithm

When the audio call session is established and started, the audio packets are sent as RTP packets over UDP. The receiver receives the packets and regularly sends RTCP receiver reports to the sender. The RTCP receiver reports consists of observed packet loss and RTT. The sender checks the packet losses and if it exceeds the threshold (p_ν), then it switches to audio bundles over DTN. Otherwise the mode is UDP. If the sender does not receive (N_ζ) consecutive RTCP reports, then audio packets transfer is over DTN. For transmission in DTN mode, a group of RTP packets are bundled together to form a DTN bundle. The bundle size is adaptive to the RTT values received via RTCP reports. For the simulations we used static message sizes of 1 second, 5 seconds and also adaptive message sizes. Additionally, sender sends the probe packets over UDP at regular intervals of time to check the network characteristics over UDP during DTN mode. The probe flags are to check the end-to-end path availability over UDP.

3.1.2 Receiver Side Algorithm

The receiver regularly checks for the probe packets and if (N_ι) consecutive probe packets are received then it notifies the sender via probe flag in RTCP receiver reports. In DTN mode, the receiver sends RTCP reports as DTN bundles. The sender checks the RTCP reports and if the average delays are less than the accepted threshold (T_κ) with probe flag set then it swiftly switches back to RTP over UDP real time communication. Based on our measurements and simulations, we set the parament to following values: $T_\kappa = 500\,\mathrm{ms}$, $p_\nu = 0.1$, $N_\zeta = 3$, and $N_\iota = 5$.

3.2 Call Signaling and Media Transport

For call signaling, SIP is typically used to establish, modify and tear down the media sessions. We assume that the three way SIP handshake was carried out before the voice call starts. The negotiation of session capabilities are carried out via SDP based offer/answer exchanges in which two end points negotiate ports, media type, media codecs etc. To shift to DTN-based asynchronous voice messaging using DAM algorithm, the two end points should support DAM. The two user agents could exchange messages using their DTN EIDs. The SDP message could be as follows.

```
m=audio 53333 RTP/AVPF 0
a=rtcp:53334
c=IN IP4 example.operator.com
a=dam:dtn://dtn-URI/audio-id mtime=10s pt=0
```

The SDP attribute of a=dam attribute indicates the support for the DAM and the receiver enables the DAM option if it supports it. The a=dam should be exchanged between both the parties in offer and answer in-order to be able to fall-back the call to DAM mode. The m, a attributes signifies that the sending node is ready to receive RTP packets with payload type 0 (PCM) on port 53333 using AVPF and on ready to receive RTCP packets on port 53334. The DTN EID of sender is dtn://dtn-eid/audio-id which can receive maximum voice packet size of 10 seconds

Algorithm 1 Sender

Send packets via: DTN **OR** UDP **mode**

```
Intialize: Mode := UDP
if RTCP received then
  if RTCP received via UDP then
    if p_RTPloss > p_ν then
      Mode := DTN
    end if
  end if
  if RTCP received via DTN then
    if ProbeFlag = Set then
      if T_avgDelay < T_κ then
        Mode := UDP
      end if
    end if
  end if
else
  if N_RTCPloss > N_ζ then
    Mode := DTN
  end if
end if
if Mode = DTN then
  PacketizeRTPPacketsInBundle
  BundleSizeWrtRtt
  sendBundleOverDTN
  sendProbePacketsOverUDP
else
  sendPacketsOverUDP
end if
```

Algorithm 2 Receiver

```
Receive packets via: DTN OR UDP
  Initialize RTP/RTCP
  if RTPreceivedoverUDP then
    if RTPType = Probe then
      CheckProbeSequence
      Update ProbeFlag
    else
      UpdateRTCPReceiverReportBlock
      sendRTCPViaUDP
    end if
  else
    ExtractRTPpacketsFromBundle
    UpdateRTCPReceiverReportBlock
    UpdateRTCPProbeFlag
    sendRTCPViaDTN
  end if
```

with payload type 0(PCM). The user agents can start the call session termination by sending the BYE message as in synchronous SIP. In case of challenged wireless networks, Disconnection Tolerant SIP [43] could be used.

The media transport in both UDP-based transmission and DTN-based transmission uses RTP transport protocol. But in DTN mode the RTP packets are carried inside the DTN bundles which is specified in Bundle Protocol. The exchange of the control information regarding the media transport is carried out using RTCP receiver reports which carries RTT and loss measurements. We used audio visual profile feedback profile (AVPF) for supporting multiple packets per second in RTCP rate. The normal audio visual profile (AVP) does not support frequent RTCP in 5 seconds. In DTN mode, the RTCP reports are bundled inside DTN bundles using Bundle Protocol.

3.3 Summary

In this chapter, we described the sender side and receiver side algorithms for DAM. The algorithms are based on RTP/RTCP and they are application level network independent algorithms. We used packet losses and delays as the metrics to switch to DTN messaging. And also, we used delays and probes packets to switch back to real time voice communications. In the next chapter, we evaluate the performance of the DAM algorithms in the case of 3G network and Mobile ad-hoc network scenarios.

4 Evaluation

In this chapter, we present the evaluation of Delay Tolerant Adaptive Media. We used two scenarios for our evaluation of the algorithm. The two scenarios are 3G network scenario and a more challenging scenario of mobile ad-hoc networks. The first scenario investigates the voice communication between two nodes over 3G network in Helsinki area. The second scenario investigates the voice communication between two nodes in a ns-2 set-up of 40 mobile nodes with Random waypoint mobility model and ad-hoc mode. The evaluation consists of generation of rtpdump traces of voice traffic over an actual VoIP call, ns-2 simulations for DTN and MANET routing, using rtpdump traces for voice traffic simulations over ns-2.

4.1 NS2DTN

NS2DTN [32] is the simulation of DTN routing and Bundle protocol using ns-2 simulator. It uses more accurate dei80211mr [37] radio link model. In NS2DTN the bundles are fragmented into 1500 bytes of IP packets and then sent to the lower layers. The reliability of the packet delivery is implemented using bundle return receipts. The three main components of NS2DTN are hello messages, checking storage and receiving bundles. Hello messages are broadcast periodically at every 100 ms and they contains the bundle identifiers of the bundles stored in the node. Each node periodically checks the storage for the bundles and return receipts to be forwarded. Each node receives the bundles and process them accordingly depending on the message type i.e. Hello message, Return receipt, Nack. The following table 8 describes the bundle node parameters used for the simulations

Parameter	**Value**
Hello message interval [ms]	100
Bundle retransmission timeout [s]	1000
Bundle routing protocol	1: Epidemic
AntiPacket [Delete forwarded bundles]	1: Yes
Bundle drop strategy	0: Drop-tail
Bundle buffer size [bytes]	100000

Table 8: Bundle Node Parameters

4.2 Voice Traffic Generation

The voice traffic patterns are generated by capturing RTP packets in rtpdump format of a voice call. The voice call is between two SIP nodes with one node in Finland and other in France. The VoIP software client used for RTP packet generation is Linphone and packets are captured using Wireshark, open source packet analyser tool. Each raw packet captured is 20ms of voice data, which is 214 bytes of packet size. The packet structure is shown in table 9.

14 bytes Ethernet	20 bytes IP	8 bytes UDP	12 bytes RTP	160 bytes payload

Table 9: Packet Structure

4.3 NS-2 Voice Traffic Simulation

For ns-2 simulations, we used application level RTP Traffic generator which is an extension to ns-2. The RTP traffic generator at a node reads the voice traffic patterns in rtpdump file format and generates a stream of RTP packets according to the timestamp in the rtpdump traces. For the evaluation of the DAM mode using NS2DTN, we used RTP Traffic generator for Voice Traffic Simulation in ns-2.

4.4 Metrics: R-Value difference

The performance of the DAM algorithm is evaluated using the R values and their difference in different modes of operation. Firstly we created a perfectly connected network between two nodes in NS2DTN and generated the voice traffic. When the nodes are well connected, then the mode is RTP/UDP for the entire simulation and we calculated the R Value which is R_{ref}. Then we calculated the R value in actual evaluation scenarios which is R. The performance indicator is ΔR [36]
where as

$$\Delta R = R_{ref} - R$$

If smaller the difference between R_{ref} and R, then better the quality of the audio. As the difference increases then the quality degrade. ΔR =0 indicates maximum audio quality achievable and ΔR =93 indicates the poorest audio quality.

4.5 Evaluation 1: Helsinki 3G Traces

The first evaluation is carried out using a mobile node with UMTS data connection moving on a bus and a fixed node. Both the nodes are situated in Helsinki metropolitan area, Finland and the bus route is between Espoo and Helsinki. The mobile node is an HP Probook 4320s 3G notebook running MS Windows 7 as operation system. The fixed node is Intel Xeon E5420 server and records the incoming RTP packets using rtpspy. The mobile node uses rtpsend that generated a stream of RTP packets to the fixed node at a steady state. The RTP packets consisted of 160 bytes G.711 payload with additional RTP/UDP/IP headers and they were sent at every 20 ms interval. In order to measure accurate packet delay there is a need for clock synchronization between both the nodes. Both the nodes are synchronized using Network Time Protocol (NTP).

For the simulation, we used NS2DTN set-up with two well connected nodes. The reception characteristics of each node modulates according to the delay and loss patterns measured during UMTS data collection of mobile and fixed node. The application level RTP Traffic generator is used to generate the collected voice patterns from one node to the other node. The voice patterns are generated for around 12 minutes with 214 bytes of packets sent at every 20ms interval. The traffic generation is in line with RTP stream sent over bus route using rtpspy. The receiving node regularly generate RTCP reports at every 2 second interval. IN DTN mode, RTCP reports are sent as DTN bundles. As the receiving node modulates according to the delay and loss patterns, the sending node adapts according to the DAM algorithm.

4.5.1 Simulation Observations

To assess the DAM algorithm in 3G network scenario, we took two distinct traces in Helsinki metropolitan area. We used these two traces to evaluate the algorithm in the simulation environment explained in the previous section. As shown in figure 4.1, the plots are based on R value differences (ΔR) which conveys the voice quality difference over time span for which traces were taken. As explained above, R value difference of zero indicates maximum audio quality achievable and higher the R value difference lower the voice quality. The crosses indicates the ΔR values for UDP-based transmission and the dots indicates ΔR values for DAM-based transmission.

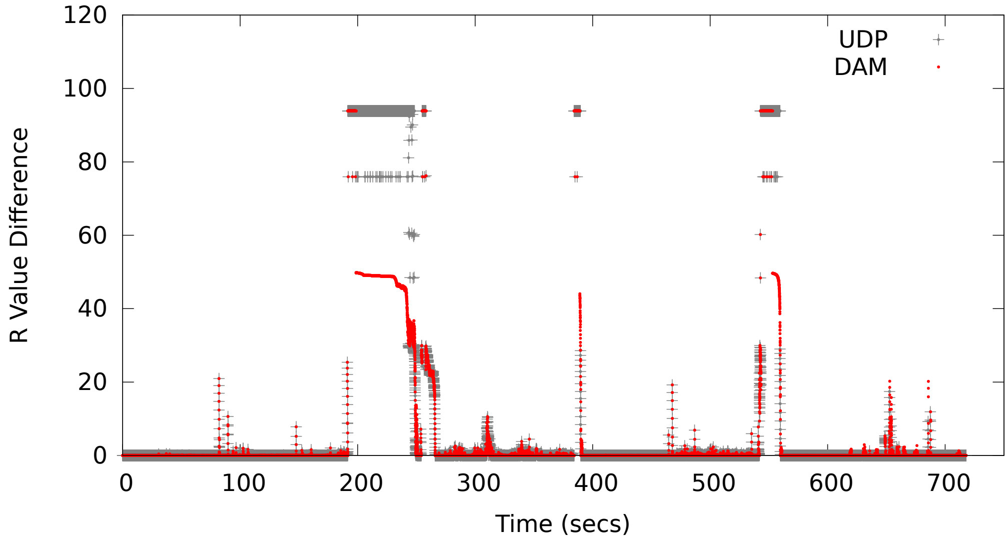
As we can see in figure 4.1, DAM algorithm performs better when the conditions of the network are worse for sufficiently long time. In figure 4.1a the DAM algorithm shifts to DTN-based messaging at around 200 to 270 seconds and 550 to 570 seconds. In figure 4.1b the DAM algorithm shifts to DTN-based messaging at around 340 to 370 seconds and 620 to 6500 seconds. In both the cases, the UDP based transmission is almost negligible with almost no voice transmission but in DAM-based transmission there is around 50% increase in voice quality perception. But for short network connectivity problems the quality of UDP-based transmission and DAM-based transmission are equal. For rest of the time, there is good connectivity for more than 90% of the time, so DAM algorithm runs in UDP mode and thus the performance is same for the both the modes of transmission. So when we consider the total sample size of 71,100 R values, the DAM algorithm perform better in 4 to 8% of the time with quality reduction of about 0.5% only and with same performance rest of the time.

4.6 Evaluation 2: Ad-hoc Communications

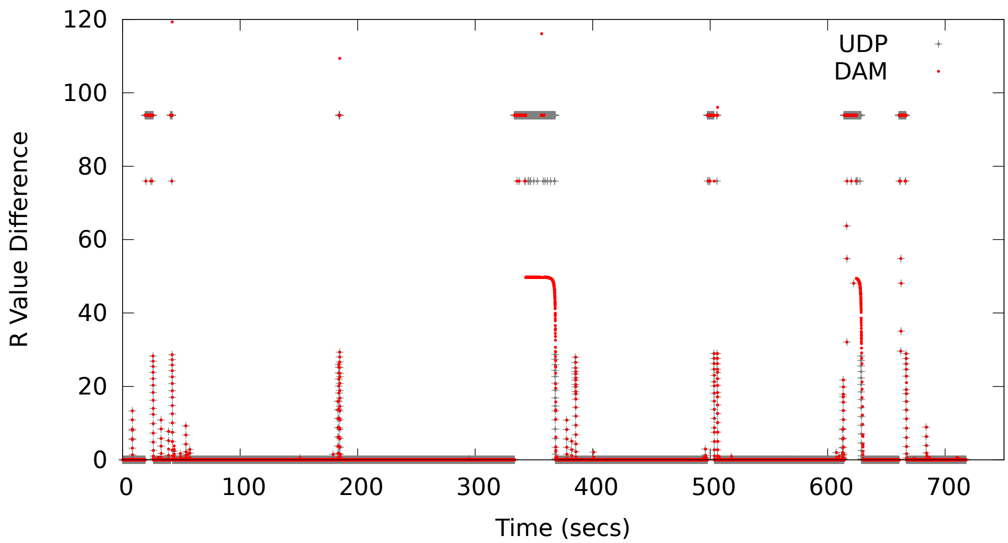
The second evaluation is carried for Ad-hoc network scenario of 40 mobile nodes using AODV or DTN routing [41] [32]. Two nodes out of 40 mobile nodes are chosen for real time communication. In the case of 3G scenario, even though we could observe the improvement in the R values, the real need for DTN-based communications is limited as the coverage of the 3G network is good enough with rare disconnections. Even though there are disconnections which are rare, the timespan of disconnections are small enough so that they could be bridged by other reliable stream oriented services like TCP [6], SCTP etc instead of UDP. So we further extended our evaluation towards Ad-hoc Communications with 40 mobile nodes. Jani Lakkakorpi et al discussed in their work on Adaptive routing in Mobile Opportunistic Networks [32] that epidemic routing gives best performance with respect to delivery ratio and delay in sparse networks. Thus we used epidemic routing with bidirectional real time communication between two nodes.

4.6.1 Mobility Model

For the node mobility simulation, we chose Random Waypoint mobility model (RWP) [25]. In Ad-hoc networks, RWP is widely used simulation model. In RWP



(a)



(b)

Figure 4.1: Helsinki Traces

nodes consists of straight legs between the waypoint and they move in zigzag line. All the nodes are uniformly distributed over a convex area A. The waypoints are denoted as $P_i(P_i \sim \mathrm{U(A)})$, nodes move in a straight leg from P_{i-1} to P_i with a velocity of Pv_i. The RWP process sequence is defined as

$(P_0; P_1; v_1)$, $(P_1; P_2; v_2)$...

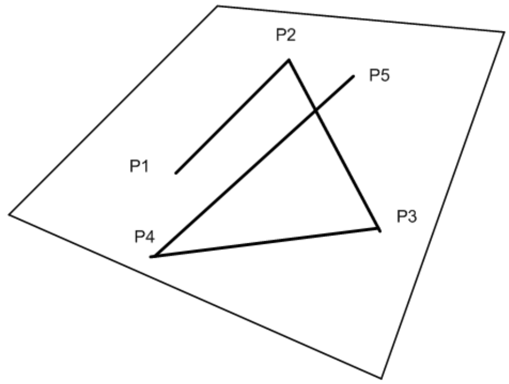


Figure 4.2: Random Waypoint

As shown in figure 4.2, each node has a pause time at the waypoint P_i and then move towards the next waypoint. The drawbacks of the model are human contact points are not appropriate and low pause times. But the low pause times could be used in favour of simulations for frequent disconnections in the network. With RWP it is easy to generate different scenarios needed for the simulations. We used setdest program of ns-2 (ns-allinone) package to generate a 40 mobile nodes RWP simulation scenario. The nodes move at a random velocity with a maximum speed of 20m/s and pause time of 2 seconds. Area sizes for the simulation ranges from $50m \times 50m$ to $2000m \times 2000m$.

4.6.2 IEEE 802.11g Wireless channel Simulation

To simulate the wireless channel model, we used dei80211mr library of ns-allinone package. It is realistic wireless channel model with respect to default model present in the ns-2. We used this library to simulate wireless channel between the nodes of the ad-hoc network. As shown in the table 10, it provides various options to choose the parameter values for 802.11 b/g standards. It also provides various options to support transmission rates, modulation and coding methods defined in 802.11b/g standards. The library consists of Signal to Interference and Noise Ratio(SINR)-based packet level error model that uses Packet Error Rate (PER) for determining random packet losses instead of R_x Threshold variable that has been removed. PER calculation uses pre-determined curves provided by the user via TCL. Received strength, noise and interference are used to calculate the SINR.

Gaussian model is used to calculate the interference. And finally, noise strength could be set via TCL. The noise is calculated according to

P_n = kTB;

k = Boltzmann's constant (1.38e-23 J/K)
T = Room temperature (290 K)
B = Bandwidth (2.437 GHz)

The parameters are set as shown in the table 10 and they model 802.11g wireless channel. As per parameters set, the transmission range between the nodes varies from 66 to 130 meters [32].

Parameter	**Value**
noise_	9.75e-12 W
CSThresh_	1e-10 W
Pt_	0.0178 W
freq_	2.437e6 Hz
L_	1.0
useShortPreamble_	true
gSyncInterval_	0.00001 s
CWMin_	16
CWMax_	1024
RTSThreshold_	0 B
ShortRetryLimit_	8
LongRetryLimit_	5
SlotTime_	0.000009 s
SIFS_	0.000016 s

Table 10: 802.11g Wireless Channel Parameters

4.6.3 Simulation Observations

The simulation evaluation is started with 40 mobile nodes and 1 second message size with RWP mobility. Then we evaluated the algorithm with 40 mobile nodes and 5 seconds message size with Random Way Point (RWP) mobility model described in previous subsection. Finally, we evaluated the algorithm with 40 mobile nodes and adaptive message sizes between 500ms and 5 second message size, in which message size changes with respect to round trip time (RTT). The areas sizes are varied from $50m \times 50m$ to $200m \times 2000m$, for which 16 simulations were run at different area sizes between $50m \times 50m$ to $200m \times 2000m$ and significant results were presented. As shown in figures 4.3, 4.4 and 4.5; we calculated R value differences (ΔR) which conveys the voice quality difference and plotted them against the time for which the simulations were run.

For larger areas greater than $750m \times 750m$, the achieved gain with the help of DAM algorithm decreases as the network gets sparser with less node interactions. At the area size of $2000m \times 2000m$, the connectivity between the nodes is almost negligible and so for most of the time no messages were delivered. And for smaller areas lesser than $150m \times 150m$, the connectivity between the nodes is highest and in almost all the cases there is either direct connectivity or continuous end-to-end connectivity between any two nodes. So in the smaller areas DAM algorithm is always in UDP-based transmission mode with AODV routing and thus the gain of DAM is near to zero. For the medium sized networks between $250m \times 250m$ to $750m \times 750m$ area, the DAM algorithm showed significant gain in the voice packets transmission and voice quality.

Figures 4.3 and 4.4 shows the plots of R value differences (ΔR) versus time of bidirectional communication between two nodes in the scenario of 40 mobile nodes with RWP mobility with DTN static message sizes of 1 second and 5 seconds respectively. Figure 4.5 represent the the plots of R value differences (ΔR) versus time in the scenario of 40 mobile nodes with RWP mobility with DTN messages adaptively changing between 500 ms and 5 seconds with respect to round trip time (RTT). In all the three figures, sub-plot (a) shows the results from small area size ($250m \times 250m$), sub-plot (b) shows the results from medium area size ($500m \times 500m$) and sub-plot (a) shows the results from larger area size ($750m \times 750m$). As explained above ΔR=0 indicates maximum audio quality achievable and higher the R value difference lower the voice quality of which ΔR=93 is complete loss of voice quality. The crosses indicates the ΔR values for UDP-based transmission and the dots indicates ΔR values for DAM-based transmission. The DTN and UDP modes in case of DAM are represented in the graphs as toggle on/off plot with values 120 and 110 respectively.

As shown in all the three figures 4.3, 4.4 and 4.5; RTP/UDP-based communication suffers significantly when there is no availability of end-to-end connectivity and performs well when sufficient connectivity is available in the network. As shown in sub-plot (a) of all the three figures 4.3, 4.4 and 4.5, from around 25 seconds to 150 seconds and 370 seconds to 430 seconds there is voice message delivery is almost nil or completely incomprehensible voice for RTP/UDP-based communication. Where as, for DAM-based communication, the algorithm shifts to DTN mode and thus there is significant voice message delivery. This is possibly due to node mobility in which continuous end-to-end path is not available but DTN-based store, carry and forward of the voice messages is possible. Even though the voice quality of messages over DTN are not full duplex supported, but this assures some form of communication when there is no continuous real time communication possible. While switching from RTP/UDP mode to RTP/DTN mode and then back, the performance of DAM suffers due to latency in switching to other modes. In the case of network with small area size, there is occasional toggling of mode between RTN/UDP and RTP/DTN due to latencies in bundle delivery.

As shown in sub-plot (b) of all the three figures 4.3, 4.4 and 4.5, for UDP-based communication in medium sized network area, the ΔR is very high, almost 78 to 93, in the entire time period with occasional possibility of voice message transfer. This shows that its neither possible to have a real time voice communication nor any form of intelligible voice communication over UDP. In this case DAM-based communication results shows that there is significant gain in the R values and thus having a half duplex communication is very much helpful. The results show that the DAM achieves better utility for the users in this case where UDP does not support message delivery.

As shown in sub-plot (c) of all the three figures 4.3, 4.4 and 4.5, in the entire time span of 700 seconds, the UDP-based communication could transfer data only for 10 seconds and thus complete packet loss for rest of the voice data. In contrast, DAM shifts to DTN mode and could able to deliver voice messages from 5 seconds to 300 seconds and from then there is not route between the nodes. So the rest of the voice messages are stored in the cache, waiting for data mule or route connectivity. These voice messages could be delivered as voice mail when ever connectivity possible with in the bundle lifetime. As the network gets sparser, the UDP-based communication is virtually impossible and DAM-based communication supports occasional voice message delivery.

Figure 4.3 shows that there are occasional oscillations between DTN and UDP mode in DAM-based communication with 1 second message sizes. For message sizes of 5 seconds as shown in figure 4.4, the oscillations are partly reduced but there is increase in latency and lower R values. As we can see that during 5 second message size, the switching line to RTP/DTN mode is visible with a slanted line because of latency to wait for 5 second voice message. This might effect the user perception sharply whenever disruption occurs in addition to the detection time of failure in RTP/UDP communication. This could be partly solved by using adaptive voice bundle messages between 500 ms and 5 seconds as shown in figure 4.5. In case of adaptive message size, when ever there is mode switch from RTP/UDP to RTP/DTN the initial voice message size is of 500 ms. And then voice message size could be adaptively changes to the round trip time. So the visible slant mode shift in 5 second messages is not present in adaptive message plot. Adaptive message size provides user perception to adapt smoothly providing better utility.

Table 11 is the R Value Comparison table between DAM-based communication and UDP-based communication. The comparison table is based on the following relation between R_{dam} and R_{udp}, in which R_{dam} is R value in DAM mode and R_{udp} is R value in UDP mode.

$$\begin{array}{ll} \text{DAM is better if:} & R_{dam} < R_{udp} - \delta \\ \text{DAM is similar if :} & R_{udp} - \delta < R < R_{udp} + \delta \\ \text{DAM is worse if :} & R > R_{udp} + \delta \end{array}$$

As shown in the comparison table, the DAM method shows significant increase in R values in almost all cases ranging from 21.3% to 42.6%. The chart shows that R values are worse as well, but it is due to shift from one mode to other. The highest gain in performance is with adaptive message size in medium sized area ($500m \times 500m$). The performance gain in medium sized area for which DAM is better is 42.6%. DAM performance with R values similar or better follows a inverted U curve, it increases as the network topology changes from small to medium size and it decreases as the network size increases. For the simulations, we used $\delta = 5$. Investigating the algorithm to message sizes of talk-spurts and making a working prototype model is not in the scope of this thesis.

(a) 1 second message size

Area Size	**DAM Better**	**DAM Similar**	**DAM Worse**
$250m \times 250m$	7533	26593	1228
$500m \times 500m$	14550	14873	5931
$750m \times 750m$	7917	26124	1313

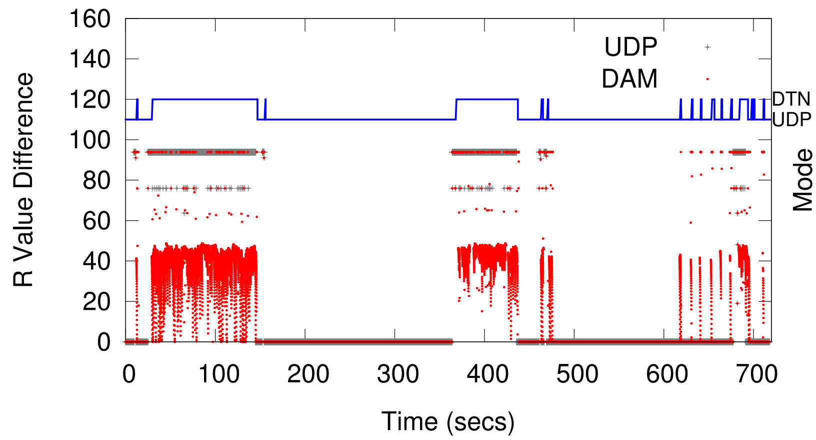
(b) 5 second message size

Area Size	**DAM Better**	**DAM Similar**	**DAM Worse**
$250m \times 250m$	8395	23126	3833
$500m \times 500m$	10460	17861	7033
$750m \times 750m$	3718	30676	960

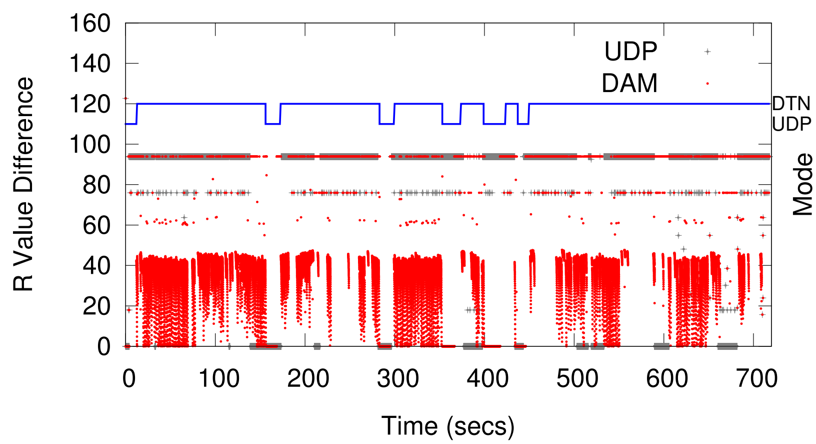
(c) Adaptive message size

Area Size	**DAM Better**	**DAM Similar**	**DAM Worse**
$250m \times 250m$	5789	27621	1944
$500m \times 500m$	15061	16697	3596
$750m \times 750m$	7969	26169	1216

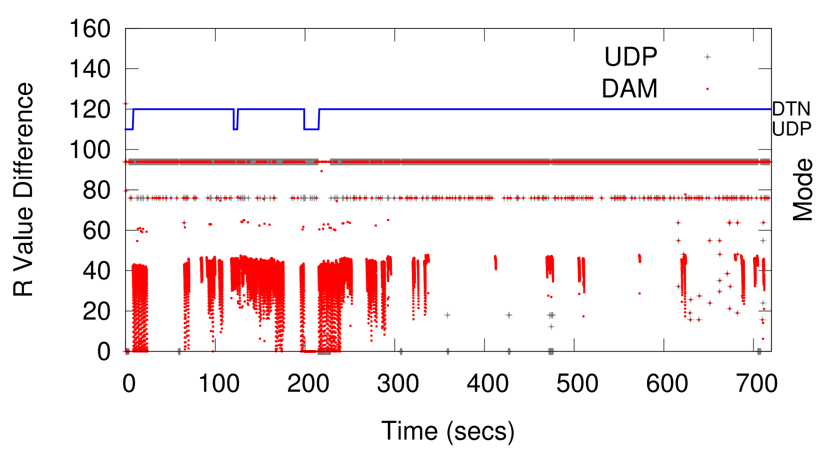
Table 11: R Value Comparison



(a) 250m by 250m area size(small)

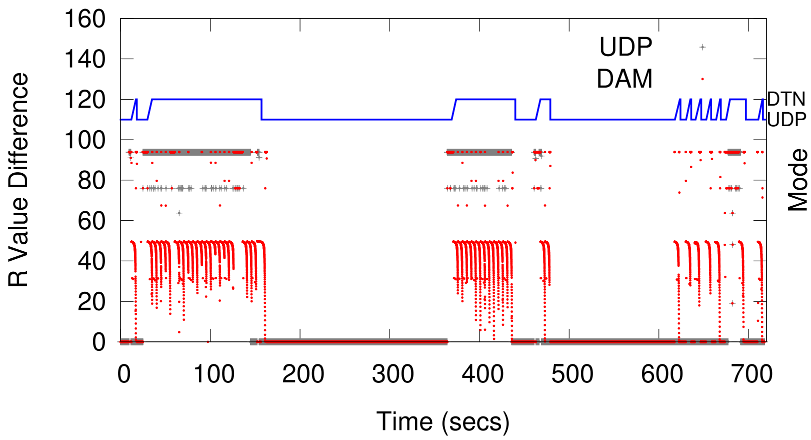


(b) 500m by 500m area size(medium)

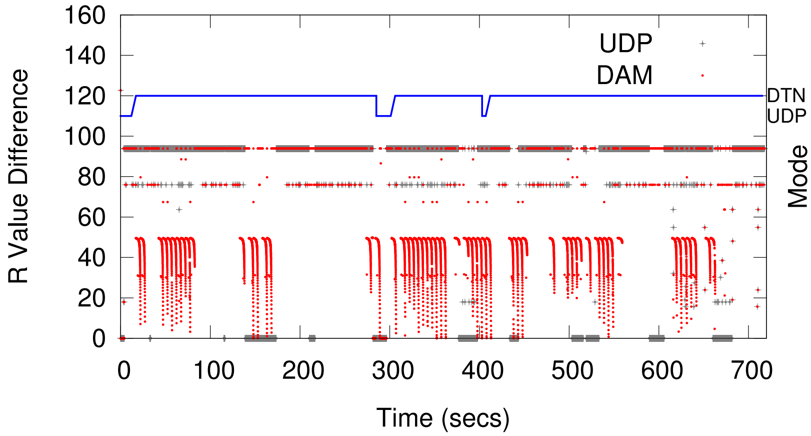


(c) 750m by 750m area size(large)

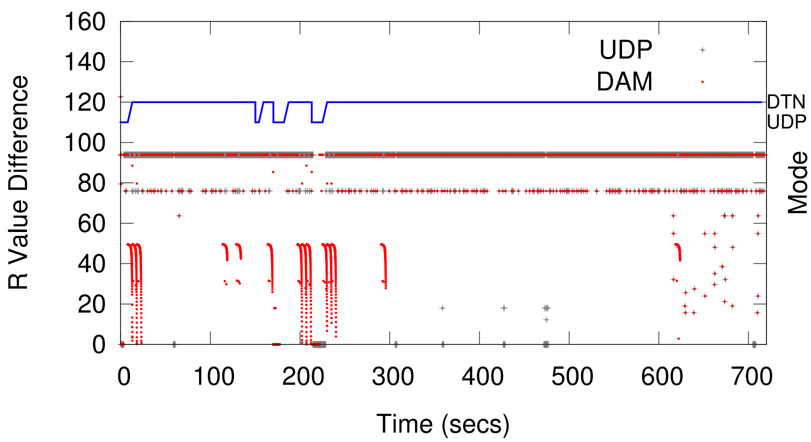
Figure 4.3: Ad-hoc communications: Random waypoint with 40 nodes, 1s messages



(a) 250m by 250m area size(small)

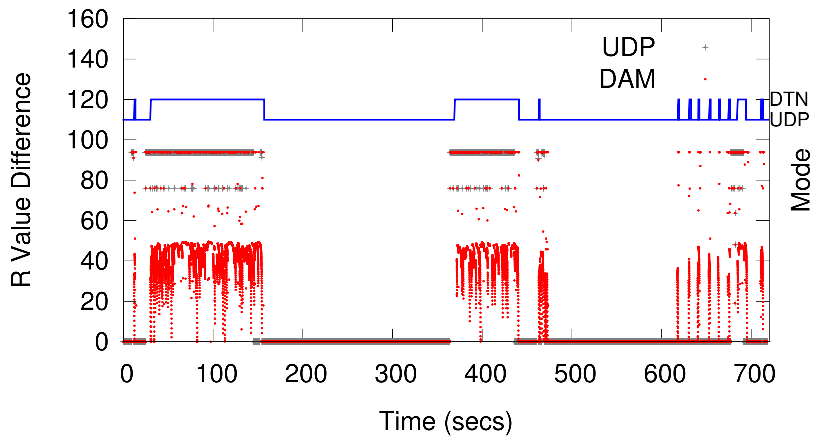


(b) 500m by 500m area size(medium)

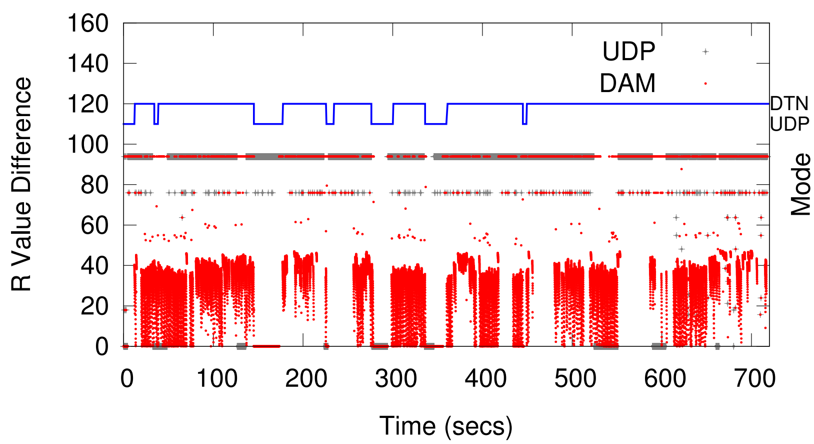


(c) 750m by 750m area size(large)

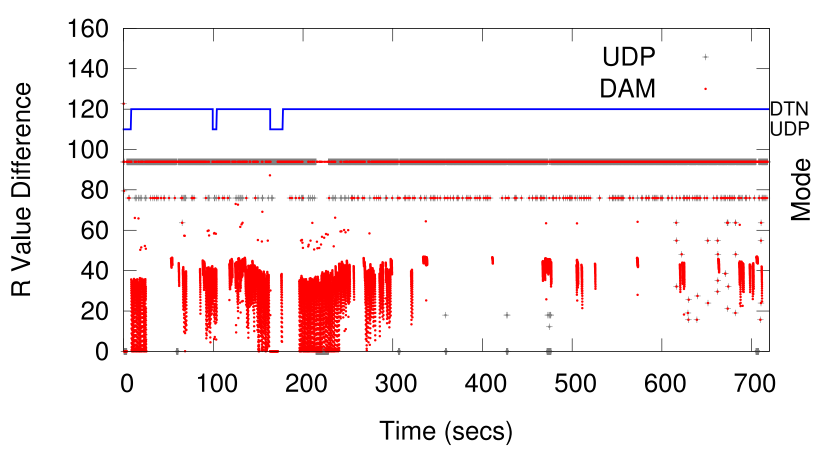
Figure 4.4: Ad-hoc communications: Random waypoint with 40 nodes, 5s messages



(a) 250m by 250m area size(small)



(b) 500m by 500m area size(medium)



(c) 750m by 750m area size(large)

Figure 4.5: Ad-hoc communications: RWP with 40 nodes, Adaptive messages

5 Conclusion

In this thesis, we investigated voice communications in challenged networks and presented a system and algorithm design for the adaptive real time communication. We discussed the limitations of voice applications over existing network infrastructure, their limitations especially during disruptions and high node mobility scenarios. We developed delay tolerant adaptive media algorithm that adapts to the packet losses and delays. It adaptively selects the most suitable communication between real time packet based on RTP/UDP and message based DTN bundles containing RTP packets.

We have shown that the proposed delay tolerant adaptive media for adaptive voice over delay tolerant networks achieves better performance and utility than the pure RTP/UDP-based transmission. The performance of the DAM algorithm is significant in ad-hoc communications with random waypoint scenario, but in the Helsinki 3G trace scenario it is not significant due to better connectivity. The results show that the delay tolerant adaptive media performs exceptionally well when the nodes in the network are sparsely connected. With the advent of increased device intelligence, with support to ad-hoc networking, we expect the increased usage of ad-hoc communications. We used R value of E model and its difference with a reference R value from an ideal call, to evaluate the simulations. From the simulation based evaluations, it shows that the future work is to build a prototype application. One limitation for building a scalable prototype model could be lack of ad-doc networking capabilities enabled in the present mobiles phones.

Building a prototype application for DAM would bring the non technical dimension, usability, into consideration. While building an adaptive application that shifts from real time voice to voice messaging, the application should notify the user in some form, so that user expectations changes smoothly. Especially when the quality of the voice is dropped when there is a shift to DTN-based voice messaging, it is important to have a responsive user interface that helps in easy usability for the user and conveys the changes to the user. Until now, most of the existing user interfaces/instructions are created according to immediate instantaneous response expectations by the user. With the advent of delays, the user interfaces/instructions should convey the user that the response is time varying in nature. Therefore, the application user interface or instructions should adapt according to the quality of the network.

Last but not least, almost all of the present voice communications are dependant on a central server and what if existing voice communications are unavailable due to failure or unavailability of the infrastructure? The failure or unavailability of the infrastructure could be due to disasters. But also these communication services are vulnerable to the censorship by government or service providers. In those extreme conditions, DTN-based communication would be a viable option.

6 References

References

[1] Ahmed, S. Message Forwarding in People-centric Delay Tolerant Networks, 2010.

[2] Birman, K. P., and Joseph, T. A. Reliable communication in the presence of failures. *ACM Transactions on Computer Systems 5*, 1 (February 1987), 47–76.

[3] Blum, N., and Magedanz, T. PTT + IMS = PTM - towards community/presence-based IMS multimedia services. In *Proc. 7th IEEE International Symp. on Multimedia* (2005).

[4] Braden, R., Clark, D., and Shenker, S. Integrated Services in the Internet Architecture: an Overview. RFC 1633, June 1994.

[5] Briscoe, B. Flow Rate Fairness: Dismantling a Religion. *ACM Computer Communications Review 37*, 2 (April 2007), 65–74.

[6] Brosh, E., Baset, S. A., Rubenstein, D., and Schulzrinne, H. The Delay-Friendliness of TCP. In *Proc. of ACM SIGMETRICS* (2008).

[7] Cerf, V., Burleigh, S., Hooke, A., Torgerson, L., Durst, R., Scott, K., Fall, K., and H.Weiss. Delay-Tolerant Network Architecture. RFC 4838, 2007.

[8] Chang, L.-H., Sung, C.-H., Chu, H.-C., and Liaw, J.-J. Design and implementation of the push-to-talk service in ad hoc VoIP network. *IET Communications 3*, 5 (2009), 740–751.

[9] Chen, K.-T., Huang, C.-Y., Huang, P., and Lei, C.-L. Quantifying skype user satisfaction. In *Proceedings of ACM SIGCOMM* (2006).

[10] Chlamtac, I., Conti, M., and Liu, J. J.-N. Mobile ad hoc networking: imperatives and challenges. *Ad Hoc Networks 1*, 1 (2003), 13–64.

[11] Clark, D. D., and Tennenhouse, D. L. Architectural Considerations for a new Generation of Protocols. In *Proceedings of the ACM SIGCOMM* (1990).

[12] Cruz, R. S., Nunes, M. S., Varatojo, G., and Reis, L. Push-to-Talk in IMS Mobile Environment. In *ICNS* (2009), J. L. Mauri, V. C. Giner, R. Tomas, T. Serra, and O. Dini, Eds., IEEE Computer Society, pp. 389–395.

[13] Dang, T. D., Sonkoly, B., and Molnar, S. Fractal Analysis and Modeling of VoIP Traffic. In *Proceedings of Networks* (2004).

[14] (ED.), A. H. L. RTP Payload Format for Generic Forward Error Correction. RFC 5109, December 2007.

[15] EKMAN, F., KERANEN, A., KARVO, J., AND OTT, J. Working Day Movement Model. In *Proc. 1st SIGMOBILE Workshop on Mobility Models for Networking Research* (2008).

[16] FALL, K. A Delay-Tolerant Network Architecture for Challenged Internets. In *Proceedings of the ACM SIGCOMM* (2003).

[17] FRIEDMAN, T., CACERES, R., AND CLARK, A. RTP Control Protocol Extended Reports (RTCP XR). RFC 3611 (Proposed Standard), November 2003.

[18] GAN, C.-H., AND LIN, Y.-B. Push-to-Talk Service for Intelligent Transportation Systems. *IEEE Transactions on Intelligent Transportation Systems 8*, 3 (2007), 391–399.

[19] HAFSLUND, A., HOANG, T. T., AND KURE, O. Push-to-talk applications in mobile ad hoc networks. In *Proceedings of the 61st IEEE Vehicular Technology Conference* (2005).

[20] HEIMERL, K., HONICKY, R., BREWER, E., AND PARIKH, T. Message phone: A user study and analysis of asynchronous messaging in rural uganda. In *Proc. ACM Workshop on Networked Systems for Developing Regions* (2009).

[21] HOENE, C., KARL, H., AND WOLISZ, A. A perceptual quality model intended for adaptive VoIP applications. *International Journal of Communication Systems 9*, 19 (2006), 299 – 316.

[22] HONICKY, R., BAKR, O., DEMMER, M., AND BREWER, E. A message oriented phone system for low cost connectivity. In *Proc. of ACM HotNets VI* (2007).

[23] HSU, J.-M., LAIN, W.-B., AND LIANG, J.-C. A Context-Aware Push-to-Talk Service. In *Proc. 2nd International Conference on Multimedia and Ubiquitous Engineering* (2008).

[24] HUI, P., XU, K., LI, V. O. K., CROWCROFT, J., LATORA, V., AND LIO, P. Selfishness, altruism and message spreading in mobile social networks.

[25] HYYTIA, E., AND VIRTAMO, J. Random Waypoint Mobility Model in Cellular Networks. *WIRELESS NETWORKS 13*, 2 (2007).

[26] INTERNATIONAL TELECOMMUNICATION UNION, TELECOMMUNICATION SECTOR (ITU-T). The E-model: a computational model for use in transmission planning. Recommendation G.107, 2011.

[27] ISLAM, M. T., TURKULAINEN, A., , AND OTT, J. Voice Messaging for Mobile Delay-Tolerant Networks. In *Proceedings of ACM COMSNETS* (2011).

[28] Islam, M. T., Turkulainen, A., Kärkkäinen, T., Pitkänen, M., and Ott, J. Practical Voice Communications in Challenged Networks. In *Proceedings of the ExtremeCom workshop* (2009).

[29] Islam, T. Voice Communication in Mobile Delay-Tolerant Networks, November 2009.

[30] Kangasharju, J., Ott, J., and Karkulahti, O. Floating content: Information availability in urban environments.

[31] Kim, P., Balazs, A., van den Brock, E., Kieselinann, G., and Bohm, W. IMS-based push-to-talk over GPRS/UMTS. In *Proc. of IEEE WCNC* (2005).

[32] Lakkakorpi, J., Pitkänen, M., and Ott, J. Adaptive Routing in Mobile Opportunistic Networks. In *Proc. of ACM MSWiM* (October 2010).

[33] Lazzaro, J. Framing Real-time Transport Protocol (RTP) and RTP Control Protocol (RTCP) Packtes over Connection-Oriented Transport. RFC 4571, July 2008.

[34] Lin, J.-R., Pang, A.-C., and Wang, Y.-C. iPTT: peer-to-peer push-to-talk for VoIP. *Wireless Communications and Mobile Computing 8*, 10 (2008), 1331–1343.

[35] Lu, N. Utility-based Bandwidth Adaptation for QoS Provisioning in Multimedia Wireless Networks, 2007.

[36] Mazurczyk, W., and Kotulsk, Z. Adaptive VoIP with Audio Watermarking for Improved Call Quality and Security. *Information Assurance and Security 2*, 3 (2007), 226–234.

[37] of Padova, U. dei80211mr: a new 802.11 implementation for NS-2. http://www.dei.unipd.it/wdyn/?IDsezione=5090.

[38] Open Mobile Alliance. Push to talk over Cellular (PoC) Architecture. OMA-AD-PoC-V1 0-20060609-A, 2006.

[39] O'Regan, E., and Pesch, D. Performance Estimation of a SIP based Push-to-Talk Service for 3G Networks. In *Proceedings of the 5th European Wireless Conference* (2004).

[40] Ott, J. Towards more adaptive voice applications. In *Proc. of the International Symposium On Leveraging Applications of Formal Methods, Verification and Validation* (2010).

[41] Ott, J., Kutscher, D., and Dwertmann, C. Integrating DTN and MANET Routing. In *Proceedings of the ACM SIGCOMM Workshop on Challenged Networks* (2006).

[42] Ott, J., Wenger, S., Sato, N., Burmeister, C., and Rey, J. Extended RTP Profile for Real-time Transport Control Protocol (RTCP)-Based Feedback (RTP/AVPF), July 2006.

[43] Ott, J., and Xiaojun, L. Disconnection Tolerance for SIP-based Real-time Media Sessions. In *Proc. of ACM MUM* (December 2007).

[44] Parthasarathy, A. Push to talk over cellular (PoC) server. In *Proceedings of the IEEE International Conference on Networking, Sensing and Control* (2005).

[45] Peterson, L. L., and Davie, B. S. Computer Networks: A Systems Approach, 2007.

[46] Pitkänen, M. J., Keränen, A., and Ott, J. Message Fragmentation in Opportunistic DTNs. In *Proc. 2nd WoWMoM Workshop on Autonomic and Opportunistic Communications* (2008).

[47] Raktale, S. 3PoC: an architecture for enabling push to talk services in 3GPP networks. In *Proc. IEEE International Conference on Personal Wireless Communications* (2005).

[48] Ronnholm, V. Push-to-Talk over Bluetooth. In *Proceedings of the 39th Annual Hawaii International Conference on System Sciences* (2006).

[49] Rosenberg, J., Schulzrinne, H., U, C., Camarillo, G., Johnston, A., Peterson, J., Sparks, R., and Handley, M. Sip: session initiation protocol. In *IETF RFC 3261* (2002).

[50] Scholl, J., Lambrinos, L., and Lindgren, A. Rural telemedicine networks using store-and-forward Voice-over-IP. *Stud Health Technol. Inform. 150* (2009), 448–452.

[51] Schulzrinne, H., and Casner, S. RTP Profile for Audio and Video Conferences with Minimal Control. RFC 3551 (Standard), July 2003.

[52] Schulzrinne, H., Casner, S. L., Frederick, R., and Jacobson, V. RTP: A transport protocol for real-time applications. IETF Request for Comments: RFC 3550, 2003.

[53] Scott, J., Hui, P., Crowcroft, J., and Diot, C. Haggle: A Networking Architecture Designed Around Mobile Users. In *Proceedings of the IFIP WONS* (2006).

[54] Scott, K., and Burleigh, S. Bundle Protocol Specification. RFC 5050, November 2007.

[55] Shalunov, S., Hazel, G., Iyengar, J., and Kuehlewind, M. Low Extra Delay Background Transport. Internet Draft draft-ietf-ledbat-congestion-09, Work in Progress, 2011.

[56] Spyropoulos, T., Psounis, K., and Raghavendra, C. S. Spray and wait: an efficient routing scheme for intermittently connected mobile networks. In *Proceeding of the ACM SIGCOMM workshop on Delay-tolerant networking* (2005).

[57] Toga, J., and Ott, J. Itu-t standardization activities for interactive multimedia communications on packet-based networks: H.323 and related recommendations. *Comput. Netw. 31*, 3 (Feb. 1999), 205–223.

[58] Vahdat, A., and Becker, D. Epidemic routing for partially connected ad hoc networks. Technical Report CS-200006, Duke University, April 2000.

[59] Whitbeck, J., and Conan, V. HYMAD: Hybrid DTN-MANET Routing for Dense and Highly Dynamic Wireless Networks. In *Proceedings of the 3rd IEEE WoWMoM Workshop on Autonomic and Opportunistic Communications* (2009).

[60] Wu, L.-Y., Tsai, M.-H., Lin, Y.-B., and Yang, J.-S. A client-side design and implementation for push to talk over cellular service. *Wireless Communications and Mobile Computing 7*, 5 (2007), 539–552.

[61] xiaojun, L. Disruption Tolerance for SIP, May 2008.