BIO-INSPIRED QUALITY OF SERVICE AWARE ROUTING IN MOBILE AD HOC NETWORKS

Thesis

Submitted in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

by

KIRAN. M

DEPARTMENT OF INFORMATION TECHNOLOGY
NATIONAL INSTITUTE OF TECHNOLOGY KARNATAKA
SURATHKAL, MANGALORE -575 025

JANUARY, 2014

CERTIFICATE

This is to certify that the Research Thesis entitled **BIO-INSPIRED QUALITY OF SERVICE AWARE ROUTING IN MOBILE AD HOC NETWORKS** submitted by **Mr. Kiran M** (Register Number: 090717IT09F01) as the record of the research work carried out by him, is *accepted as the Research Thesis submission* in partial fulfillment of the requirements for the award of degree of **Doctor of Philosophy**.

Prof. G Ram Mohana Reddy

(Research Guide)

Prof. Ananthanarayana V S

(Chairman-DRPC)

Acknowledgements

I would like to express my sincere gratitude to my research guide Prof. G. Ram Mohana Reddy, Department of Information Technology, NITK, Surathkal, for his guidance, support and encouragement throughout my research work.

I express heartfelt thanks to my Research Progress Assessment Committee (RPAC) members Prof. D Venkata Reddy and Dr. Ashvini Chaturvedi, NITK, Surathkal, for their valuable suggestions and constant encouragement to improve my research work. I also thank Prof. Ananthanarayana V S, Chairman, DRPC and Head, Department of Information Technology, NITK, Surathkal, for his timely suggestions and support.

I sincerely thank all teaching, technical and administrative staff of the Information Technology Department for their help during my research work. I also extend my thanks to The Management, JNN College of Engineering, Shimoga for giving me an opportunity to pursue Doctoral Degree on deputation basis. I am grateful to Dr. Srikanth C, Mr. Hemanth Kumar M L, Dr. Manu A P and Mrs. ShreePrajna K J of JNN college of engineering for their support.

I would like to thank my parents and inlaws for their exhaustive support and encouragement and I owe this work to my wife Dr. Karishma Kiran and my daughter Manasvi Kiran. Without them, surely, this research work would not have been possible.

Finally, I would like to express my gratitude to my friends, Mr. Venkata Ramana, Mr. Srinivasa Murthy T, Mr. Ganesh K. Reddy, Dr. Shivmurthy, Mr. Nagaraj K. Kodihalli, Mr. Melwyn D'souza, Mr. Santhosh Achar, Mr. Sandeep and all other research scholars for their encouragements.

This work is supported by RGNF grant funded by the Indian Government, GoI order No. F. 14-2 (SC) / 2009 (SA III) and this support is greatly acknowledged.

Kiran M

ABSTRACT

In recent years a lot of work has been done in an effort to incorporate Swarm Intelligence (SI) techniques in building an adaptive routing protocols for Mobile Ad Hoc Networks (MANETs). As centralized approach for routing in MANETs lack in scalability and fault-tolerance, SI techniques provide natural solutions through distributed approach to the adaptive routing for MANETs. The mobile nodes found in MANETs are capable of monitoring the network status as well as data processing. Thus the MANETs can be made Context Aware with the help of mobile nodes local monitoring capability. In this thesis work, a novel mobility aware bio inspired routing protocol for MANETs referred to as *Mobility Aware Termite (MA-Termite)* is proposed by inheriting the hill building nature of social inset *Termite*

MA-Termite will find the reliable path between the source and destination node based on the stable nodes in terms of its mobility with the help of the local monitoring capability of nodes. Further, analytical model is also proposed for studying an asymptotic pheromone behavior of MA-Termite using two different parameters *(decay rate and pheromone sensitivity)* over both single and double links. The results depict how individual parameters are correlated and how they affect the global performance of MANETs. The best possible parameter values are determined for optimal performance for MA-Termite.

Recently, several telecommunication applications of bio-inspired algorithms achieved remarkable success. In SI techniques, the captivating features of insects or mammals are correlated with the real world problems to find solutions. The natural question is whether it is possible to develop a new hybrid algorithm by combining the distinguishing features of these insects or mammals? In this regard, the salient features of mammals such as *bats* are combined with the proposed MA-Termite algorithm to come up with a new hybrid routing algorithm referred to as *Bat-Termite* for MANETs. Bat-Termite improved the backup route maintenance and also exhibited superior routing features such as quick route discovery, high robustness with efficient management of multiple routes and rapid route repair.

One of the features of both MA-Termite and Bat-Termite algorithms is they always exclusively choose the highest pheromone link thus *congests* the highest pheromone link over a period of time. This undesirable behavior is referred to as *stagnation*. Further, MA-Termite lags in load balancing and fails to take the full benefit of multipath environment. One of the methods to avoid the stagnation problem is *pheromone heuristic control*. Thus, a novel heuristic hybrid Load Balanced Quality of Service (QoS) aware routing protocol referred to as *Load Balanced-*

Bat-Termite (LB-Bat-Termite) is proposed for MANETs in order to solve the *stagnation problem* of both MA-Termite and hybrid Bat-Termite algorithms. The LB-Bat-Termite algorithm with its context awareness, QoS awareness and load balancing features exhibited considerable performance gain due to load balancing. LB-Bat-Termite produces additional control packets in order to maintain all possible paths to the destination node and thus mobile nodes spends most of its time in route maintenance than data transfer; hence causes performance degradation under high node density conditions.

In prder to improve the scalability and to reduce the control packet overhead, a novel heuristic Load Balanced Termite based QoS aware routing protocol is referred to as Load Balanced-Termite (LB-Termite) is proposed for MANETs. LB-Termite exhibited considerable performance gain under both scalability and mobility factors. The proposed bio-inspired QoS aware routing algorithms in this thesis work could be used for applications such as university or campus settings, data sharing during lecturing or meeting or data sharing during virtual classrooms. The proposed algorithms for MANETs in this thesis namely MA-Termite, Bat-Termite, LB-Bat-Termite and LB-Termite are compared with the state-of-the-art bio-inspired (Simple Ant Routing Algorithm and Termite Algorithm) and non bio- inspired routing algorithms (Ad Hoc On demand Distance Vector Routing algorithm) for its performance evaluation and results are encouraging in terms of QoS parameters (Throughput, Total Packet Drops, End to End Delay and Control Packet Overhead).

Keywords: MANET Routing, Swarm Intelligence, Load Balancing, Node Mobility, Bat, Termite, Quality of Service, Cross Layer Design.

Contents

List of Figures v

List of Tables ix

List of Abbreviations x

Nomenclature xii

1 Introduction 1

1.1 Routing in Mobile Ad Hoc Networks 1
1.2 Quality of Service 3
1.3 Cross Layer Design 4
1.4 Bio-Inspired Computing 5
1.4.1 Swarm Intelligence Techniques 5
1.4.2 Non-Swarm Intelligence Techniques 7
1.5 Motivation 8
1.6 Outline of the Thesis 8
1.7 Summary 10

2 Literature Survey 11

2.1 Legacy Routing Protocols for MANETs 11
2.2 Swarm Intelligence Based Routing in MANETs 13
2.2.1 ACO Based Bio-Inspired Routing in MANETs 14
2.2.2 Bees Based Routing Algorithms for MANETs 26
2.2.3 Flocks of Birds Based Routing in MANETs 27
2.2.4 Termite Based Routing Algorithms for MANETs 28

2.3 Non-SI Based Bio-Inspired Routing Algorithms 32
2.4 Hybrid Bio-Inspired Routing Algorithms 32
2.5 Outcome of The Literature Review 39
2.6 Problem Statement 40
2.7 Research Objectives 40
2.8 Summary 42

3 Intelligent Cross Layer Models 43

3.1 False Route Failure in MANETs 43
3.2 Working Principle of IEEE 802.11 MAC 44
3.3 Hidden Node Problem at the MAC Layer 44
3.4 A Service Driven Cross Layer Model 46
3.5 Prioritized QoS Based Cross Layer Model 47
3.6 Results and Discussion 51
3.6.1 Simulation Results of Service Driven Cross Layer Model 51
3.6.2 Simulation Results of Prioritized QoS Cross Layer Model 54
3.7 Summary 56

4 Mobility Aware Termite:
A Novel Bio-Inspired Routing in MANETs 57

4.1 Motivation for Development of MA-Termite Algorithm 57
4.2 Working Principle of MA-Termite 58
4.2.1 Cross Layer Design Model 59
4.2.2 Trail Pheromone Update-Decay Function 60
4.2.3 Forwarding Function of MA-Termite Algorithm 61
4.2.4 Node Structure 61
4.2.5 Pheromone Bounds 62
4.2.6 Trail Pheromone Table 62
4.2.7 Local Status Table 62
4.2.8 Route Setup and Maintenance 63
4.3 Asymptotic Behaviour of MA-Termite 63
4.3.1 Node Separation Distance Distribution 64
4.3.2 Expected Pheromone Decay 66

4.3.3 Analytical Justification for Pheromone Update-Decay Function 67
4.3.4 Average Expected Amount of Pheromone Over Single Link 67
4.3.5 Average Expected Amount of Pheromone Over Double Links 69
4.4 Performance Evaluation of MA-Termite 72
4.4.1 Experimental Setup 73
4.4.2 Scenario A 74
4.4.3 Scenario B 79
4.5 Summary 82

5 Bat-Termite:
A Novel Hybrid Bio-Inspired Routing 84
5.1 Motivation for Development of Bat-Termite Algorithm 84
5.2 Echolocation in Bats 86
5.3 Echolocation in Bat-Termite 87
5.4 Route Maintenance Phase of Bat-Termite 88
5.5 Packet Structure of Bat-Termite 89
5.6 Performance Evaluation of Bat-Termite 90
5.6.1 Experimental Setup 91
5.6.2 Scenario A 92
5.6.3 Scenario B 97
5.7 Summary 100

6 Load Balanced-Bat-Termite:
A Novel Hybrid Bio-Inspired Routing 102
6.1 Motivation for Development of LB-Bat-Termite 102
6.2 Load Balancing Approach of LB-Bat-Termite 105
6.3 Alarm Pheromone Update Function of LB-Bat-Termite 106
6.4 Trial Pheromone Vs Alarm Pheromone 107
6.5 Forwarding Function of LB-Bat-Termite 107
6.6 Node Structure of LB-Bat-Termite 108
6.6.1 Local Status Table 108
6.6.2 Trail Pheromone Table 109
6.6.3 Route Discovery Phase of LB-Bat-Termite 110

6.6.4 Route Maintenance Phase of LB-Bat-Termite 111
6.7 Performance Evaluation of LB-Bat-Termite 113
6.7.1 Experimental Setup 113
6.7.2 Scenario A 114
6.7.3 Scenario B 120
6.8 Summary 124

7 Load Balanced-Termite:
A Novel Load Aware Bio-Inspired Routing 125
7.1 Motivation for Development of LB-Termite Algorithm 125
7.2 Working Principle of LB-Termite Algorithm 127
7.3 Performance Evaluation of LB-Termite 127
7.3.1 Experimental Setup 127
7.3.2 Scenario A 128
7.3.3 Scenario B 134
7.4 Comparative Study of MA-Termite, Bat-Termite, LB-Bat-Termite and LB-Termite Algorithms 138
7.5 Summary 146

8 Conclusions and Future Work 148

References 153

Publications 156

List of Figures

1.1 Example for CLD - A Three Layer Model 4

2.1 Hill Building Nature of the Termite at Different Time Intervals 28

2.2 ACO Taxonomy based on Agents and Heuristics used in Routing 38

3.1 Working Principle of IEEE 802.11 DCF MAC 44

3.2 Hidden Node Problem 45

3.3 Cross Layer Model Across Application, Network and MAC Layers 46

3.4 Cross Layer Model Across Application, Network, MAC and Physical Layers . 48

3.5 Secure and Warning Range of the Nodes 49

3.6 Simulation of Service Driven Cross Layer Model Using AODV Routing Protocol 52

3.7 Simulation of Service Driven Cross Layer Model Using DSR Routing Protocol 54

3.8 Throughput comparison of PrQoS and Traditional Method 55

3.9 Control Packet Overhead Comparison of PrQoS and Traditional Method 55

3.10 Route Re-Discovery Attempts Comparison of PrQoS and Traditional Method . 56

4.1 Principle of Computing Inter Packet Distance Between the Nodes 59

4.2 Cross Layer Design Model for Finding the Distance Between the Mobile Nodes 60

4.3 Node Structure of MA-Termite 61

4.4 MANET Model for Studying Asymptotic Behavior of MA-Termite 65

4.5 Double Link System for studying Asymptotic Behaviour of MA-Termite 69

4.6 Expected Link Pheromone Vs Distance β = 0.9357, τ=0.01, r = 10, F=1, K = 0 70

4.7 Expected Link Pheromone for different F Values, β = 0.9357, r = 10, K=0 . . . 71

4.8 Expected Link Pheromone for different decay rate F=1, K = 0, r = 10 72

4.9 Throughput Vs Simulation Time in Sec (500m x 500m Sim. Area) 74

4.10 Throughput Vs Simulation Time in Sec (1000m x 1000m Sim. Area) 75

4.11 Packet Drop Vs Simulation Time in Sec (500m x 500m Sim. Area) 76
4.12 Packet Drop Vs Simulation Time in Sec (1000m x 1000m Sim. Area) 76
4.13 End-to-End Delay Vs Simulation Time in Sec (500m x 500m Sim. Area) 77
4.14 End-to-End Delay Vs Simulation Time in Sec (1000m x 1000m Sim. Area) . . 77
4.15 Control Packet Overhead Vs Simulation Time in Sec (500m x 500m Sim. Area) 78
4.16 Control Packet Overhead Vs Simulation Time in Sec (1000m x 1000m Sim. Area) 79
4.17 Throughput Vs Node Velocity 80
4.18 Packet Drop Vs Node Velocity 80
4.19 End-to-End Delay Vs Node Velocity 81
4.20 Control Packet Overhead Vs Node Velocity 82

5.1 Asymptotic Behavior of MA-Termite on a Two Link System 85
5.2 Experimental Results of Asymptotic Behavior of MA-Termite on a Two Link System 85
5.3 Echolocation Features of Bats 86
5.4 Echolocation Feature of Bat-Termite 87
5.5 Throughput Vs Simulation Time in Sec (500m x 500m Sim. Area) 92
5.6 Throughput Vs Simulation Time in Sec (1000m x 1000m Sim. Area) 93
5.7 Packet Drop Vs Simulation Time in Sec (500m x 500m Sim. Area) 94
5.8 Packet Drop Vs Simulation Time in Sec (1000m x 1000m Sim. Area) 94
5.9 End-to-End Delay Vs Simulation Time in Sec (500m x 500m Sim. Area) 95
5.10 End-to-End Delay Vs Simulation Time in Sec (1000m x 1000m Sim. Area) . . 95
5.11 Control Packet Overhead Vs Simulation Time in Sec (500m x 500m Sim. Area) 96
5.12 Control Packet Overhead Vs Simulation Time in Sec (1000m x 1000m Sim. Area) 97
5.13 Throughput Vs Node Velocity 98
5.14 Packet Drop Vs Node Velocity 98
5.15 End-to-End Delay Vs Node Velocity 99
5.16 Control Packet Overhead Vs Node Velocity 100

6.1 Two Path System for Studying the Asymptotic Behavior of MA and Bat-Termite 103
6.2 Experimental Results of Asymptotic Behavior of MA and Bat-Termite on a Two Link System . 104
6.3 Illustration of Nodal Load Balancing of LB-Bat-Termite Algorithm 106
6.4 Throughput Vs Simulation Time in Sec (500m x 500m Sim. Area) 114
6.5 Throughput Vs Simulation Time in Sec (1000m x 1000m Sim. Area) 115

6.6 Total Packet Drops Vs Simulation Time in Sec (500m x 500m Sim. Area) . . . 116
6.7 Total Packet Drops Vs Simulation Time in Sec (1000m x 1000m Sim. Area) . . 117
6.8 End-to-End Delay Vs Simulation Time in Sec (500m x 500m Sim. Area) 118
6.9 End-to-End Delay Vs Simulation Time in Sec (1000m x 1000m Sim. Area) . . 118
6.10 Total Control Packets Vs Simulation Time in Sec (500m x 500m Sim. Area) . . 119
6.11 Total Control Packets Vs Simulation Time in Sec (1000m x 1000m Sim. Area) 120
6.12 Throughput Vs Node Velocity 121
6.13 Total Packet Drops Vs Node Velocity 122
6.14 End-to-End Delay Vs Node Velocity 122
6.15 Total Control Packets Vs Node Velocity 123

7.1 n path system for studying the asymptotic behavior of LB-Bat-Termite 126
7.2 Throughput Vs Simulation Time in Sec (500m x 500m Sim. Area) 129
7.3 Throughput Vs Simulation Time in Sec (1000m x 1000m Sim. Area) 130
7.4 Total Packet Drops Vs Simulation Time in Sec (500m x 500m Sim. Area) . . . 130
7.5 Total Packet Drops Vs Simulation Time in Sec (1000m x 1000m Sim. Area) . . 131
7.6 End-to-End Delay Vs Simulation Time in Sec (500m x 500m Sim. Area) 132
7.7 End-to-End Delay Vs Simulation Time in Sec (1000m x 1000m Sim. Area) . . 132
7.8 Control Packet Overhead Vs Simulation Time in Sec (500m x 500m Sim. Area) 133
7.9 Control Packet Overhead Vs Simulation Time in Sec (1000m x 1000m Sim. Area)134
7.10 Throughput Vs Node Velocity 135
7.11 Total Packet Drops Vs Node Velocity 136
7.12 End-to-End Delay Vs Node Velocity 136
7.13 Control Packet Overhead Vs Node Velocity 137
7.14 Throughput Vs Simulation Time in Sec (500m x 500m Sim. Area) 138
7.15 Throughput Vs Simulation Time in Sec (1000m x 1000m Sim. Area) 139
7.16 Throughput Vs Node Velocity 139
7.17 Packet Drop Vs Simulation Time in Sec (500m x 500m Sim. Area) 140
7.18 Packet Drop Vs Simulation Time in Sec (1000m x 1000m Sim. Area) 141
7.19 Packet Drop Vs Node Velocity 141
7.20 End-to-End Delay Vs Simulation Time in Sec (500m x 500m Sim. Area) 142
7.21 End-to-End Delay Vs Simulation Time in Sec (1000m x 1000m Sim. Area) . . 143
7.22 End-to-End Delay Vs Node Velocity 143

7.23 Control Packet Overhead Vs Simulation Time in Sec (500m x 500m Sim. Area) 144
7.24 Control Packet Overhead Vs Simulation Time in Sec (1000m x 1000m Sim. Area) 145
7.25 Control Packet Overhead Vs Node Velocity 145

List of Tables

1.1 QoS at Different Layers 3

2.1 Congestion Aware ACO Based Bio-Inspired Routing Algorithms 33

2.2 Load and Mobility Aware ACO Based Bio-Inspired Routing Algorithms 34

2.3 CLD and Hybrid ACO based Bio-Inspired Routing Algorithm 34

2.4 Bees Based Bio-Inspired Routing Algorithms 34

2.5 Location Aware ACO Based Bio-Inspired Routing Algorithms 35

2.6 Flocks of Birds Based Bio-Inspired Routing Algorithms 35

2.7 QoS Aware ACO Based Bio-Inspired Routing Algorithms 36

2.8 Termite Based Bio-Inspired Routing Algorithms 37

2.9 Other ACO Based Bio-Inspired Routing Algorithms 37

3.1 Simulation Parameters For Evaluation of Intelligent Cross Layer Models 51

4.1 Parameter Values used for MA-Termite 73

5.1 Parameter Values used for Bat-Termite 91

6.1 Parameter Values used for LB-Bat-Termite 113

7.1 Parameter Values used for LB-Termite 128

7.2 Protocol Behavior Under Increasing Node Density (100 to 600 Nodes) 146

7.3 Protocol Behavior Under Increasing Node Velocity(2 - 10 m/s) 146

List of Abbreviations

ACO	: Ant Colony Optimization
AODV	: Ad Hoc On demand Distance Vector
BT	: Basic Termite
Bat-T	: Bat-Termite
CLD	: Cross Layer Design
CTS	: Clear To Send
DCF	: Distributed Coordination function
FRF	: False Route Failure
LB-Termite	: Load Balanced-Termite
LST	: Local Status Table
MA-T	: Mobility Aware-Termite
MANET	: Mobile Ad Hoc Networks
MAC	: Medium Access Control
PrQoS	: Prioritized QoS
QoS	: Quality of Service
RREQ	: Route REQuest
RREP	: Route REPly
RERR	: Route ERRor
RTS	: Request To Send

SARA	: Simple Ant Routing Algorithm
SI	: Swarm Intelligence
SRL	: Short Retry Limit
TPT	: Trail Pheromone Table

Nomenclature

P	: Pheromone On link
p	: Probability of using neighbor link
α	: Pheromone carried by the packet
d_c	: Current Distance
d_p	: Previous Distance
τ	: Decay rate
P_r	: Received Signal Strength
PH_BEGIN	: Initial Value of The Pheromone Over The Link
PH_MIN	: Minimum value of The Pheromone Over The Link
PH_MAX	: Maximum Value of The Pheromone Over The Link
$f(d)$	: Node Separation Distance Distribution
$E(e^{-d\tau})$	: Expected Pheromone Decay
$EP(n)$	: Expected Amount of Pheromone for *n* arrived Packets
$EP(d)$	: Expected amount of pheromone on a link over node separation distance *d*
μ	: The average value of the received pheromone
F	: Pheromone Sensitivity
K	: Pheromone Threshold
ϕ	: Alarm Pheromone Over the Neighbor Link

Chapter 1

Introduction

This chapter gives an overview of Mobile Ad Hoc Networks (MANETs) and the corresponding challenges in designing routing protocols for MANETs in general and in particular the techniques such as Cross Layer Design (CLD) and Swarm Intelligence (SI) are discussed for providing Quality of Service (QoS) in MANETs. Further, this chapter highlights the motivations for carrying out the research work in this thesis.

1.1 Routing in Mobile Ad Hoc Networks

MANETs belong to the family of multi-hop wireless networks in which a group of mobile nodes cooperate with each other in order to maintain the network connectivity. MANETs have no fixed network infrastructure or administrative support but have wonderful features such as *self-creation, self-organization* and *self-administration*. However, the network communications in MANETs will be decentralized due to the limited transmission range of mobile nodes and thus multiple hops are required to transmit the data between the source and destination nodes. In MANETs, a path between the source and destination nodes contains one or more intermediate or relay nodes which receive and forward the packets and thus each node also acts as *router*. MANETs extend the coverage area of the network without any additional infrastructure and thereby providing better connectivity with less energy and transmission power. MANETs are widely used in applications such as *military, public safety (disaster recovery etc.), providing access to remote regions* and *areas where deployment of cables is not possible (mountains, deserts etc.)* (Murthy and Manoj 2005).

But MANETs suffer from *routing complexity, path management overhead* and *extra delay* due to multi-hop relaying. Further, providing QoS is a critical and challenging task in MANETs due to its *dynamic network topology* (Ramanathan and Redi 2002).

Design Challenges for Routing Protocols in MANETs

The performance of MANETs depends not only on the routing technique but also on the routing metrics such as *shortest path, bandwidth, delay etc*. Since MANETs pose a significant technical challenge because of several constraints imposed by the underlying network and thus the routing is the most critical and challenging task in MANETs. Hence, the main challenges in MANETs routing are highlighted below (Imrich Chlamtac et al. 2003; Chen and Heinzelman 2007; Hanzo and Tafazolli 2007).

- *QoS aware routing* is important as MANETs are expected to provide enhanced end-to-end services such as high throughput, low end-to-end delay, less packet drops, minimal control packet overhead etc. under dynamic network topological conditions.

- *Scalability* is one of the important problems in MANETs routing and thus degrades the throughput with increased network control packet overhead under high node density scenario. Thus a MANETs routing protocol should be scalable to higher node density in such a manner that the control packet overhead is minimized.

- *Energy efficient routing* is very important in maximizing the network lifetime of MANETs since mobile nodes (mobile phones, lap tops and Personal Digital Assistant (PDAs)) have limited energy. Nodes with poor energy may breakup the link or it can cause partition in the network in the worst case scenario.

- *Route maintenance and recovery* should be one of the important characteristics of routing protocols in MANETs as hostile environment of MANETs can tamper or damage the available nodes.

- Providing *security* is one of the important challenges of MANETs routing since MANETs are vulnerable to attacks.

- Handling *hidden and exposed terminal problem* during routing is another challenging task in MANETs since it will be misinterpreted as *link breakage* or *node is not reachable*.

- Most of the MANETs routing protocols lack *Load Balancing* and hence Load Balancing across multiple paths is very important in MANETs in order to increase its efficiency.

Present MANETs provide all kinds of multimedia traffic for the users and thus designing novel QoS aware routing protocols for MANETs have opened up new challenges and a brief overview of QoS is given in the next section.

1.2 Quality of Service

The QoS is associated with the performance of computer communication networks and according to ITU-T standard E.800, QoS can be defined as *The collective effort of service performance which determines the degree of satisfaction of a user of the service*. Further, IETF contributed many protocols (Intserv, Diffserv, RSVP and MPLS) to introduce QoS in the internet; and also discussed QoS routing issue and established *IETF QoS Routing working group* (Masip-Bruin et al. 2006). Real time multimedia communication networks require QoS such as bandwidth, end-to-end delay, jitter, latency etc. and thus QoS awareness becomes a valuable parameter in designing routing protocols for MANETs. QoS can be provided over different layers in the protocol stack, from Physical layer to Application layer and each layer holds responsible for providing the QoS in different ways and the Table 1.1 gives an example of QoS services at different layers of OSI model (Mohapatra et al. 2003).

Table 1.1: QoS at Different Layers

Sl.No	Layer	QoS
1	Application Layer	Simple and Flexible user interface , Dynamic QoS Ranges
2	Transport Layer	Reliable end-to-end packet delivery
3	Network Layer	Throughput, End-to-End delay
4	MAC Layer	Variable bit error rate
5	Physical Layer	Good Transmission Quality

Providing QoS in Network layer involves developing a state dependent, QoS-aware routing protocol which search for optimal routes with sufficient network resources such as *bandwidth, end-to-end delay, jitter, latency etc.*

In MANETs, the network environment will be fluctuating and accordingly the network resources will be shared among different users and thus providing QoS is a challenging task of MANETs routing under these dynamic network conditions. Hence, the current research trends have focused on *CLD* and *SI* for designing and developing the QoS aware routing protocols for MANETs (X. Masip-Bruin et al. 2006; Gianni Di Caro et al. 2005).

1.3 Cross Layer Design

Due to the variable nature of wireless communication environment, the traditional layered architecture do not function efficiently in MANETs. However, overall network task is distributed among different layers and the service hierarchy has been defined for each layer accordingly. Further, the traditional layered architecture does not allow direct communication with nonadjacent layers; thus several efforts have been made to improve the performance of the protocol stack by using CLD and thereby providing the stack-wise network interdependence. Three main reasons for using the CLD approach in developing adaptive routing protocol for MANETs are as follows: *(1) Dynamic nature of the wireless links (2)Chance of opportunistic communication in the wireless links* and *(3) The new modalities of communication offered by the wireless medium.* In CLD, each layer is not treated as a complete independent functional entity, but the information of each layer can be shared among other layers in the protocol stack (Srivastava and Motani 2005). Further, CLD has emerged as a important design option when compared to monolithic and layered protocol stacks and there by attracting several researchers for developing adaptive QoS aware routing protocols for MANETs.

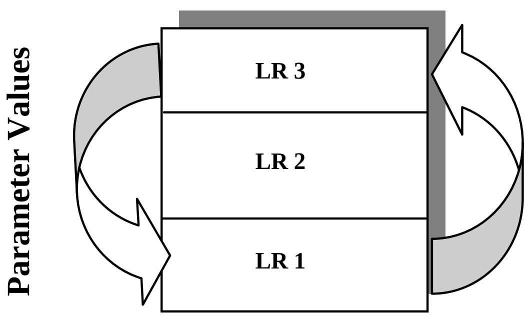


Figure 1.1: Example for CLD - A Three Layer Model

Figure 1.1 shows the working principle of CLD and let us consider a three layer model LR1, LR2 and LR3 based on ISO OSI reference model. LR1 cannot communicate with LR3 since it is not adjacent however an interface can be developed in such a manner that LR1 takes LR3 parameter values during runtime. Some of the examples of the CLD approaches are: (1) Link layer may tune transmit power of the Physical layer to control the bit-error rate (2) The Application layer can get the information about packet loss from Network layer under different situations (3) The Link layer may provide an adaptive error control mechanism based on the TCP retransmission timer information from Transport layer (Srivastava and Motani 2005).

1.4 Bio-Inspired Computing

Bio-inspired computing is a popular technique for solving the optimization problem and can be defined as *"Designing distributed problem-solving devices or algorithms inspired by the behavior of social insects and other animal societies"* (E. Bonabeau et al. 1999). Bio-Inspired computing is a technique where behavior of swarms of insects or other animals are studied and emulated to solve the complex real world problems. The bio inspired algorithms found in the literature have several applications in the areas of engineering, logistics and telecommunications in general and in particular the telecommunication domain has achieved remarkable success (Frank Neumann and Carsten Witt 2010; Gurpreet Singh et al. 2012; Vivekanand Jha et al. 2011). Bio-Inspired computing can be further classified into *Swarm Intelligence (SI)* and *Non-SI* based and the details of SI are given in the next section.

1.4.1 Swarm Intelligence Techniques

SI is inspired by the extinguishing features exhibited by the social insects (ants, termites), flocks of birds, bees, school of fish etc. and an SI system is driven by four primary principles namely, *positive feedback, negative feedback, randomness* and *multiple interactions* (Frank Neumann and Carsten Witt 2010).

- **Positive Feedback**

  It is used to emphasize the best solution in the SI system and thus it will make other insects to follow the emphasized solution.

- **Negative Feedback**

  It is used for removing old or stale solutions in SI system which consequently weakens the probability of choosing these solution. If a solution is found to be poor or non optimal then it will be negatively reinforced and thus requesting other insects not to follow it. Positive and negative feedback should be equally balanced in order to find an optimal solution.

- **Randomness**

  As many solutions will be available in the SI system, an optimal solution may become non optimal if its quality degrades or whenever a new solution is available. Thus exploring all available solutions randomly is very important in SI system and thereby selecting the best available solution.

- **Multiple Interaction**

  The SI system often contains tens to millions of individuals and the information collected by this large population is communicated through multiple interaction.

Recently another principle called stigmergy is also added to the SI system (Ruth M& Wicker S 2003a). Stigmergy is a method of indirect communication among the members of the same societies in SI system and stigmergy can increase the communication robustness and asynchronous information transfer among the individuals.

Swarms such as *ants* and *termites* are distributed in nature and these social insects are simple, independent, interdependent and co-operative in nature. While building a nest or finding the shortest path to the food source, these social insects exhibit wonderful characteristics such as adaptability, scalability and robustness. Pheromones are means of communication among these social insects and are nothing but signal carrying chemicals; these pheromones are broadly classified into two types: *releasers* and *primers*. Upon receiving releaser pheromones, insects exhibit an instantaneous behavior *(alarm, sex and recruitment or trail pheromones)* while primer pheromones cause the delayed effect on insect behavior *(sex pheromones by honeybee queen)* (Regnier F E and N. J. 1968). Recruitment or Trial pheromone is acting as navigational aid and directs other insects to a distant location (for ex. food source by ants) varying in hundred meters in bees to meters in insects. These trails are then followed by foraging insects and are either positively or negatively reinforced based on the quantity and quality of the food source and further, the trial pheromones are also used to recruit workers for colony emigration.

In social insects, alarm pheromones are the most commonly produced class of pheromones after sex pheromone and these pheromones are normally released when there is a threat to the nest or to the other insects. On reception of these releaser pheromones, insects show immediate response by dispersing from the source of alarm pheromone that detects the danger. Normally the social insects like ants, bees, wasps and termites attack the enemy which threatens the nest or exhibit a panic behavior and will escape when the defense is not realistic (Tristram D. Wyatt 2003). This will be illustrated by placing a dead ant near the nest entrance and most of the nearby ants will exhibit a panic behavior.

Thus the characteristics *(adaptability, scalability and robustness)* and intelligent behavior of these social insects (optimal path finding by ants and the hill building nature of the termites) could be exploited for design and development of routing protocols for MANETs (C. Kolias et al. 2011; Muhammad Saleem et al. 2011).

1.4.2 Non-Swarm Intelligence Techniques

Apart from salient features of Swarms *(ants, bees, flocks of birds and termites)* there exist other animal societies such as mammal *bats* and have also motivated several researchers for designing adaptive bio-inspired algorithms for solving complex real world problems. The fascinating *echolocation* characteristic of mammal bats allows them to find their prey and its status (distance and movement direction) in complete darkness. The bats produce very loud pulse and listen for the echo that bounces back after hitting the prey and the surrounding objects; depending on the species and obstacles, the pulse varies when it echoes back. According to a study, micro bats, a species of bats, builds a 3-D scenario of surrounding environment using the time delay from the emission of pulse to the detection of an echo and the loudness variations of the echoes. Thus, Bats can detect the *distance and orientation of the target, type of prey* and *its moving speed* using the Doppler Effect (Airas, M. 2003). However, Non-SI is not much explored when compared to SI in solving the real world problems, particularly in telecommunication field (Vivekanand Jha et al. 2011). The next section discusses the motivation to carry out the research work.

1.5 Motivation

The existing layered protocol stack functions inefficiently in mobile wireless environment due to highly variable and the limited nature of the mobile devices (Raisinghani et al. 2002) and thus breaking of the layered architecture can significantly improve the performance of the wireless network. Hence, current research trends have focussed on CLD in designing and developing novel routing algorithms for MANETs in order to provide QoS for the applications. Further, as centralized approach for routing in MANETs lack in scalability and fault-tolerance, SI techniques provide natural solutions through distributed approach to the adaptive routing for MANETs. Thus the characteristics and behavior of social insects could be inherited in design and development of adaptive QoS routing algorithms for MANETs (C. Kolias et al. 2011; Muhammad Saleem et al. 2011).

The mobile nodes found in MANETs are capable of monitoring the network status as well as data processing. Thus the MANETs can be made *Context Aware* with the help of mobile node‘s local monitoring capability. Traffic in MANETs is expected to carry a mix of real time multimedia and non real time file transfer etc. Providing *Quality of Service (QoS)* for these different applications is difficult in MANETs due to its distinguishing characteristics. Hence designing novel bio-inspired QoS aware routing protocols for MANETs has received ample attention as traditional routing protocols fail to meet most of the network service requirements. Motivated by these needs, this research work aims to design and develop adaptive, scalable and QoS aware bio-inspired routing protocols for MANETs by inheriting the salient features of social insect Termites, mammal Bats and CLD. In this thesis we presented an analogy of the behavior of the social insect termite and mammal bats with nodes in MANETs.

1.6 Outline of the Thesis

The remainder of this thesis is organized as follows. In Chapter 2, existing state-of-the-art bio-inspired routing protocols for MANETs are discussed and analyzed. The discussion concentrates mainly on application of SI and Non-SI in MANETs and a classification is also given based on code of practice, properties hired; further the shortcomings of these algorithms in MANETs are also noted. Then the problem statement and research objectives are derived based on the outcome of the literature survey.

Chapter 3 discusses the design of two intelligent cross layer models across Application, Network and Physical layers to get the priority information of the application and distance information of the neighbor nodes. These proposed cross layer models will be able to increase the throughput of the prioritized applications in MANETs. The validation of the proposed cross layer models is performed in terms of throughput gain of the prioritized flow when compared to un prioritized flow.

In Chapter 4, a novel bio-inspired routing protocol based on the behavior of social insect *Termites* referred to as *Context Aware or Mobility Aware-Termite (MA-Termite)* is proposed for MANETs. The primary objective of MA-Termite is to emphasize the *stable nodes* by positively reinforcing them with less pheromone decay and also to deemphasize the unstable nodes by negatively reinforcing them with high pheromone decay. The secondary objective of MA-Termite is to propose the *distance variant analytical model* for pheromone update method of *MA-Termite*. The proposed analytical model determines the average expected amount of pheromone over both single and double links by observing the asymptotic behavior of the *MA-Termite*. Further, a rigorous simulation study has been carried out to find the similarity between theoretical and simulation results and the results are compared with that of the state-of-art bio-inspired (ant based Simple Ant Routing Algorithm (SARA)(Correia F and Vazao, T 2010)) and Termite (Roth M and Wicke S 2004a) and non bio-inspired (Ad Hoc on Demand Distance Vector (AODV)(E. Perkins and Royer E.M)) routing protocols for the quantitative and qualitative analysis of MA-Termite.

In Chapter 5, a novel hybrid bio-inspired routing protocol for MANETs referred to as *Bat-Termite* is discussed. Bat-Termite combines the hill building nature of social insect Termites and the fascinating echolocation behavior of mammal Bats. The echolocation feature helps the proposed algorithm in exploring the network properly and overcomes the problem identified in MA-Termite. The simulation results of Bat-Termite are also compared with other state-of-the-art bio inspired routing algorithms (SARA and the Termite algorithm) and non bio inspired (AODV) routing algorithms for its quantitative and qualitative analysis.

In Chapter 6, a novel heuristic hybrid bio-inspired routing with load balancing approach referred to as *LB-Bat-Termite* for MANETs is discussed. The primary objective of the *LB-Bat-Termite* algorithm is to find the *stable nodes* and thereby giving preferences for these stable nodes during the path setup; thus finding the reliable route to the destination. The secondary objective of the proposed *LB-Bat-Termite* algorithm is to mitigate the *stagnation problem* identified in both MA-Termite and Bat-Termite algorithms by using *pheromone heuristic control method*. The simulation results of LB-Bat-Termite are compared with other state-of-the-art bio-inspired routing algorithms (SARA and the Termite algorithm) and non bio-inspired (AODV) routing algorithms for its quantitative and qualitative analysis.

In Chapter 7, a novel heuristic bio-inspired routing with load balancing approach referred to as *LB-Termite* for MANETs is discussed. LB-Termite algorithm overcomes the problem identified in LB-Bat-Termite algorithm by efficiently managing the primary and secondary paths in the pheromone table. LB-Termite also overcomes the control packets overhead problem of LB-Bat-Termite. The simulation results of LB-Termite is compared with other state-of-the-art bio-inspired (SARA and Termite algorithm) and non bio-inspired (AODV) algorithms for its quantitative and qualitative analysis. Further, a comparative study of all proposed algorithms(MA-Termite, Bat-Termite, LB-Bat-Termite and LB-Termite) is also done in order to give a clear picture of their merits and demerits.

Finally, Chapter 8 summarizes the contribution of the research work and highlights the possible directions for future work.

1.7 Summary

This chapter introduced MANETs and also introduced CLD, QoS aware and bio-inspired computing (SI and non-SI) techniques, which are receiving ample attention in today's research. This chapter concludes with the challenges in designing MANETs routing protocols and motivation to carry out the research work. In the next chapter literature survey of bio-inspired routing protocols for MANETs are discussed in detail.

Chapter 2

Literature Survey

In this chapter, existing state-of-the-art bio-inspired routing protocols are discussed in detail while considering several important features of social insects and animal societies. A detailed classification of *Swarm Intelligence (SI)* and *non-SI* based routing algorithms is also given based on the code of practice and properties hired and finally open issues and research challenges in designing routing algorithms in MANETs are highlighted. This literature review will give the detailed description of *Ants, Bats, Bees, Flocks of Birds* and *Termite* based bio-inspired routing protocols under *Congestion, Cross Layer Design, Load, Location, Mobility* and *Quality of Service*aware categories.

2.1 Legacy Routing Protocols for MANETs

The MANETs routing protocols have to cope with the frequent topological changes in order to provide stable path to the destination and these routing protocols can be categorized in to proactive, reactive, and Hybrid routing protocols (Murthy and Manoj 2005; Rahman M et al. 2010)). In proactive or table driven routing protocols, route updates are propagated throughout the network at regular intervals in order to maintain current routing information in each node in the network; thus the main problem with such types of protocols is *maintenance overhead*. Hence, proactive routing protocols are not suitable for highly dynamic networks and the examples for table driven protocols are Optimized Link State Routing (OLSR)(Jacquet et al. 2001), Destination Sequenced Distance Vector routing (DSDV)(He. G 2002) etc.

In reactive or source initiated routing, a route is established through a route discovery procedure only when the source needs a route to the destination. Route discovery procedure floods the route request packets to find the optimal path to the unknown destination; thus the control packets are generated only when needed but it suffers from high route discovery latency. Examples for such type of routing protocols are Dynamic Source Routing (DSR) (Johnson , D.B et al. 2001) etc. In hybrid routing protocols, the features of both on-demand and table driven protocols are combined to come with a new class of protocols. By properly handling these two techniques in a network, hybrid protocols improves the over all performance of the network. Examples for such type of protocols are Zone Routing Protocol (ZRP) (Haas, Z. J and Pearlman M. R 1997) etc. In the next section, Ad Hoc On Demand Routing protocol is explained in brief.

- **Ad-hoc On demand Distance Vector Routing (AODV)**

  AODV is one of the famous reactive routing protocol for ad hoc networks and it is continuously being updated by Internet Engineering Task Force (IETF) MANET working group. Quick adaptation to dynamic topology, low operating cost (processing and memory) and low network utilization are the highlights of AODV. Each node in AODV maintains a routing table which maintains only active routes to the destination; AODV gives a quick response to link breakage and topology changes in a timely manner and AODV uses destination sequence number to ensure loop freedom and thus shows good convergence property.

  AODV defines three types of messages for route discovery and maintenance namely, *Route Request (RREQ), Route Reply (RREP)* and *Route Error (RERR)*. When a node does not find the destination node entry in its routing table to forward the data packet, it initiates route discovery phase in which limited life time RREQ packet is broadcasted over the network to find the optimal path to the destination. On reception of RREQ packet, each node updates its routing table about the source node and it rebroadcasts RREQ packet if it do not have an entry for the intended destination in its routing table.

  Under two cases a node generates RREP packet in response to RREQ packets *(i) if it is the intended destination (ii) if it finds destination node entry in its routing table*. RREP is unicast to the RREQ source and once the source receives the RREP packet it initiates data session by transmitting the stored data packets on newly found path to the destination.

RREQ sets up the reverse path to the source on its way while RREP sets up the forward path to the destination. Duplicate RREP and RREQ packets are discarded by the nodes. If within certain time period RREP packet is not received by the source, it resends the RREQ packet with bit larger life time; the source attempts route rediscovery for limited number of times and if it does not receives RREP from the destination node then it declares destination node is unreachable.

During data transfer phase, if a node finds link break then it broadcasts a RERR packet to the neighbors in order to notify about the path loss. RERR packet contains the destination node ID which is not reachable and on reception of RERR, each node deletes the route to the particular destination node in its routing table and a fresh route to the destination is found either by the intermediate or the source node. On the other hand, AODV suffers from bandwidth overhead, high route discovery latency and poor route management thereby reducing the throughput.

2.2 Swarm Intelligence Based Routing in MANETs

The recent research trends have shown that applying Swarm Intelligence (SI) principles to find optimal paths in MANETs is giving good and encouraging results thus SI based routing has been a research topic of high interest in MANETs. In SI routing, probabilistic paths are constructed between the source and destination using simple probability update rule. Unlike traditional routing algorithms where explicit paths are builtup between source and destination, in SI routing each route will have a probability of being used. Link failures in SI methods can be immediately recovered with the next most probable route thus reduces the network control traffic. Traditional routing protocols always results in same routes in the same way whereas SI routing allows finding random solutions in a system. Traditional routing algorithms are deterministic algorithms while SI based routing algorithm are random algorithms thus do not have same properties. SI is inspired by the captivating features of ants, bees, flocks of birds and termites; multipath routing, fast route recovery, distributed fault tolerance and fast convergence are the advantages of SI based routing protocols (Sharvani G.S 2009). Ant Colony Optimization (ACO) is one of the most popular heuristic SI techniques used for solving network optimization problem in general and in particular the routing problem of wireless networks. Accordingly ACO based bio-inspired routing algorithms for MANETs are discussed first and then bees, flocks of birds and termite based bio-inspired routing protocols are explained in the following sections.

2.2.1 ACO Based Bio-Inspired Routing in MANETs

The foraging (searching for food) and nest building features of social insects (ants and termites) such as cooperative and distributed behavior could be exploited for designing network optimization algorithms (Reginer and N 1968; Wyatt 2003). The main important characteristic of ACO is its frequent acquisition of routing information through path sampling using small independent control packets referred to as ants. The task of these ants is to collect the best path information in both directions from the source to destination node and accordingly updates the routing tables. R. Schonderwoerd et al. (1996) applied ACO technique referred to as Ant Based Control (ABC) for designing the load balanced routing protocol for telecommunication networks. AntNet is the first ACO routing algorithm proposed for connectionless networks for the best effort routing by Di Caro and Dorigo (1998). Inspired by these two algorithms, several ACO based routing algorithms have been proposed for both wired and wireless communication networks. Di Caro et al. (2008) defined an Ant Colony Routing Framework (ACR) for designing ACO based routing algorithms for both wired and wireless communication networks using reinforcement learning and multi agent systems. Existing ACO based bio-inspired routing algorithms are mainly focused on challenging design issues such as Congestion Awareness, Context Awareness, Load Balancing, Quality of Service Awareness etc. and accordingly ACO based bio-inspired routing protocols for MANETs are discussed as follows.

Congestion Aware ACO Routing Algorithms

Network congestion causes packet drops and resulting in reduced throughput; hence congestion aware routing algorithms are required for real time applications in which the good throughput is maintained at reasonable level and thereby minimizing the packet drops. Several ACO based congestion aware bio-inspired routing algorithms are proposed for MANETs in the recent past and the details are given below.

- **Ad Hoc Networking with Swarm Intelligence (ANSI) (2006):**

  Rajagopalan S and Shen C (2006) have proposed a novel bio-inspired routing algorithm referred to as Ad Hoc Networking with Swarm Intelligence (ANSI). ANSI uses forward and backward ants for the route setup phase only and further uses data packets for path reinforcement. Forward ants are flooded towards the destination node whenever a new session starts or during the route repair phase. As soon as the first forward ant is received then the destination node will set up the route to the source node and this can be achieved by sending the backward ant. This route setup is evaluated based on the congestion rate of nodes over the single deterministic path through which data packets are forwarded. Further, ANSI uses periodic HELLO packets in order to refresh the active neighbor list; ANSI also uses the pheromone evaporation concept to negatively reinforce the multiple paths. Authors have demonstrated that ANSI performs well when compared to traditional non bio-inspired routing protocol (AODV).

- **Emergent Ad-hoc Routing Algorithm (EARA) (2004):**

  Zhenyu Liu et al. (2004) have proposed a novel bio-inspired multipath routing algorithm referred to as Emergent Ad-hoc Routing Algorithm (EARA) and EARA uses ants like agents to find new paths locally. During the route discovery phase these agents will take a random walk across the network in order to find the multiple routes to the destination; these optimal and sub optimal paths are reinforced further through pheromone trails. EARA uses Local Foraging Ants (LFA) in order to find the new routes locally when all the pheromone trails of the destination drops below the threshold level and periodic refreshing of the neighbor list can be carried out by HELLO packets. These agents continuously explore the new and better paths during the data session thus the best available path can be used for data transfer. EARA also uses the pheromone and a heuristic value *(congestion in the neighbor node)* in order to calculate the probability of next hop. The results demonstrated that EARA adapted very well to suit the requirements of dynamic network without any additional control packet overhead.

- **AntHocNet (2005):**

  Ducatelle F et al. (2005) and Di Caro G A et al. (2008) have proposed a novel bio-inspired routing algorithm for MANETs referred to as AntHocNet which uses reactive forward and backward ants in order to discover the new routes. These ants are sent over high priority queues and at each node these ants are either unicast or broadcast to collect the network information. Whenever forward ant reaches the destination then it will be converted into backward ant and then follows the same route in the reverse order by updating the pheromone table in each visited node. The pheromone update is proportional to the *number of hops in the path, traffic congestion* and *signal to noise ratio*. AntHoc-

Net continuously finds new paths to the destination using proactive path maintenance and path exploration mechanisms and thereby removing old (unused) paths using slow pheromone diffusion. AntHocNet performance has been extensively evaluated through simulation under different scenarios and it has been shown that it outperforms the classical AODV and OLSR routing protocols. Kalavathi and Duraiswamy (2008)have shown little improvement to AntHocNet by adding node disjoint multipath feature to it.

- **Ant routing algorithm for MANETs based on adaptive improvement (ARAAI) (2005):** Yuan-yuan Zeng and Yan-Xiang (2005) have proposed bio-inspired multipath routing algorithm referred to as ARAAI. ARAAI uses forward and backward ants for route discovery and setup phases; forward ants collect nodes local information and path information and thereby finding the routes to the destination. For each forward ant in reaching the destination one backward ant is released and thus establishes multiple paths to the destination. The next hop is chosen based on the pheromone and local heuristic *(link stability)* value of the link; then the periodic HELLO packets are used for refreshing the neighbor tables. Further, ARAAI targets stagnation problem by using adaptive parameter coordination and results have shown that ARAAI is highly efficient than AODV and DSR protocols.

- **Probabilistic Ant Routing (2009):** Prasad S et al.(2009) have proposed a bio-inspired routing algorithm referred to as PAR which uses forward ants (FANT) for exploring the routes and network information whereas backward ants (BANT) are used for updating the data structures in the nodes. FANT uses the same queue as the data packet but BANT needs high priority queues. On reaching the destination FANT transfers the collected information to BANT and then the destination node will remove the FANT from the network. BANT traces the path of the FANT in order to update the data structures and the routing table in each visited node. Ants will be either unicast or broadcast depending on the pheromone information over the neighbor links and periodic single hop HELLO packets are used by every node in order to maintain the fresh neighbor list. Simulation results have demonstrated that PAR performs better than AODV.

CLD Based ACO Routing Algorithms

Since the traditional protocol stack will not function efficiently for MANETs, several efforts have been carried out by many researchers in order to improve the performance of the protocol stack using CLD and thereby providing the stack-wise network interdependence. Thus CLD has attracted several researchers for developing adaptive QoS aware routing protocols for MANETs and the details of existing CLD based ACO routing algorithm is given below.

- **EARA-QoS (2005):**

  Zhenyu Liu et al. (2005) have developed CLD based QoS aware ACO routing algorithm referred to as EARA-QoS. EARA-QoS has been designed to improve the performance metrics such as *delay* and *congestion* and thereby providing the best quality service using CLD across MAC and Network layers. EARA-QoS uses two heuristics values (delay and congestion) for finding the next hop probability in order to forward the data packets; further it uses sequence number to avoid the loops and thereby providing multiple paths between the source and destination. Simulation results have demonstrated that EARA-QoS outperforms the traditional AODV routing protocol.

Load Aware ACO Routing Algorithms

The existing routing algorithms for MANETs used routing metrics such as *congestion, node mobility, end-to-end* delay etc. in general and in particular the *route hop count* is the widely used metric. The nodes which appear in the shortest path will get heavy load and thus exhausts the available network resources such as bandwidth, energy, memory etc. Further it creates a bottleneck in the shortest path and resulting in congestion problem. Since mobile nodes in MANETs have limited capacity there is a need for designing a load balanced routing algorithms. Hence, state-of-the-art load aware ACO based bio-inspired routing algorithms for MANETs are explained below.

- **Multi Agent ant based Routing Algorithm (MARA) (2007):**

  Sivakumar D and Bhuvaneshwaran RS (2007) have proposed a multi agent load aware ACO routing algorithm referred to as MARA and it uses *route discovery, route update, data routing, route maintenance* and *route failure handling* phases for achieving the load balance. MARA uses forward and backward ants for route discovery and setup phases and each forward ant consist of a list of visited nodes. Whenever destination node receives the forward ant then it first discards the duplicate forward ants and generates the backward ants for updating the pheromone quality along the path. Further, more numbers of paths are explored and added in the routing table using proactive path maintenance scheme; thus MARA achieves load balancing through probabilistic routing strategy.

- **AntOR (2010):**

  L.J.G. Villalba et al. (2010) have developed a load aware ACO routing algorithm referred to as AntOR which is the extended version of AntHocNet with three additional features namely *(1) disjoint link and disjoint node routes between the source and destination (2) The best paths exploration based on the distance in terms of number of hops and (3) Two types of pheromones namely regular pheromone used for denoting the path for data session and the virtual pheromone used for denoting the possibly good path for data session.* Through simulation results authors have demonstrated that AntOR outperforms AntHocNet due to its load balancing feature.

Mobility Aware ACO Routing Algorithm

The mobility of nodes makes MANETs topology dynamic in nature and resulting in difficult route setup and maintenance phases. Further, MANETs also suffer from the frequent link failures due to the mobility of nodes and thereby causing a source node to spend most of its time in route setup and maintenance phase resulting in low throughput and more control packet overhead. Thus there is a need for mobility aware routing algorithm for MANETs and the details of mobility aware ACO based bio-inspired routing algorithm is as follows.

- **imProved Ant Colony Optimization for mobile ad hoc NETworks (PACONET) (2008):**

  Osagie Eseosa et al. (2008) have proposed mobility aware ACO based bio-inspired routing algorithm referred to as PACONET which uses forward and backward ants for route discovery and setup. Forward ants are broadcasted in a controlled manner in order to discover new routes and backward ants are used for establishing the path. Forward ant takes the path of unvisited nodes thus makes sure that all possible paths to destination are discovered; further, data packets are routed stochastically on the highest pheromone link. Through simulation, PACONET is proved to be better than non bio-inspired AODV protocol.

QoS Aware ACO Routing Algorithms

Providing QoS in MANETs for different applications such as real time multimedia and non real time file transfer etc. is difficult due to its hostile network environment. Hence design and development of QoS aware routing protocols for MANETs has received great attention among several researchers. The details of existing QoS aware ACO based bio-inspired routing protocols for MANETs are given below.

- **Ant based Distributed Routing Algorithm (ADRA) (2004):**

  Zheng X et al. (2004) have developed a QoS aware ACO based bio-inspired routing algorithm for MANETs referred to as ADRA which shares the similar features of Probabilistic Emergent Routing Algorithm (PERA) with the difference lying in checking the available resource before forwarding the ants and the path will be setup only when it meets the required QoS. Apart from forward and backward ants, ADRA uses two more ants called *enforce-ant* and *anti-ant* for fast convergence. Through simulation authors have shown that ADRA outperforms non bio-inspired DSR routing algorithm.

- **Ant Routing Algorithm for Mobile Ad Hoc Networks (ARAMA) (2005):**

  Hussein O et al. (2005) have developed a novel QoS aware bio-inspired routing algorithm referred to as ARAMA which uses multiple performance metrics such as *nodes energy, processing power* and *link bandwidth*. Further, ARAMA manages the resources through fair resource distribution and continuously checks for the new and better paths using forward and backward ants. Apart from forward and backward ants, *negative backward ants* (for deemphasizing the negative paths) and *destination trail ants* (for increasing the connection setup time) are also used and the old solutions are removed through pheromone evaporation concept. Through simulation ARAMA is proved to be better than AODV and authors have claimed that ARAMA is suitable for *energy management* in MANETs.

- **Swarm based Distance Vector Routing (SDVR) (2007):**

  Asokan R et al. (2007) have developed a QoS aware bio-inspired algorithm referred to as SDVR derived from AntNet (Di caro 1998) and it targets multiple QoS metrics namely *delay, jitter and energy*. SDVR maintains pheromone table for each QoS metric in order to take routing decision and thus gives a multi constrained end to end path between the source and destination. Further, SDVR uses pheromone evaporation mechanism to remove the old paths and the simulation results show that SDVR outperforms AODV.

- **Fuzzy based ACO algorithm for MANETs (FACO) (2009):**

  M.M. Goswami et al. (2009) have developed a novel bio-inspired routing algorithm for MANETs referred to as FACO which combines fuzzy logic with ACO principles. Three different routing metrics namely *energy consumption rate, packet buffer occupancy rate* and *signal strength* are fuzzified in order to compute the pheromone over a link. FACO uses forward and backward ants for route discovery and setup phases. Each intermediate node calculates the fuzzy cost and updates in the forward ants. Whenever the destination node receives forward ant then it transfers the stored fuzzy cost to backward ant and accordingly updates the pheromone table. Through simulation results authors have shown that FACO is better than Ant-AODV (Marwaha et al. 2002) and Ant Colony Based Routing Algorithm (ARA) (Subramanian et al. 1997).

- **Ant Dynamic Source Routing (ADSR) (2008):**

  Asokan R et al. (2008) have proposed QoS aware bio-inspired routing algorithm referred to as ADSR, derived from traditional DSR routing protocol, which targets three QoS parameters namely *delay, jitter* and *energy*. ADSR uses forward and backward ants for route discovery and setup phases where the forward ant maintains a record of visited nodes and the same can be transferred to the backward ant at the destination node. Through simulation results authors have shown that ADSR outperforms DSR.

- **Multiple Disjoint QoS Enabled Ant Colony Based Multipath Routing (QAMR) (2012):**

  P. Venkata Krishna et al (2012) have proposed a QoS aware multiple disjoint bio-inspired routing algorithm referred to as QAMR, which targets the bandwidth problem of MANETs. MQAMR uses three QoS parameters namely *bandwidth, delay* and *hop count* in order to calculate the path preference probability. Forward ants use the next hop availability (a function of mobility and energy factor) during the route discovery phase and keep track of the local information of visited nodes. Whenever the destination node receives the forward ant then it calculates the path preference value based on the collected information and accordingly the backward ant will be generated for the path that satisfies the QoS requirements. Then the backward ant updates the pheromone value of each visited node proportional to the computed path preference value. Through simulation results authors have demonstrated that QAMR is better than AODV and ARMAN under high traffic load.

• **QoS Aware Ant Based Multipath Routing Algorithm (2012):**

Sungwook Kim (2012) has proposed an ACO based QoS aware routing protocol for MANETs based on the load balancing strategy for maintaining the QoS requirements of the path. In the proposed algorithm, each ant chooses the next hop based on the communication adaptability *(function of distance between the nodes and queue length)* and accordingly pheromone will be updated in each visited node. Among the available paths, the most adaptable path is selected based on the *bandwidth* and *delay* requirements and the remaining available paths are considered for load balancing. Routing adaptability is calculated based on the path's *energy level* for deciding the data distribution among paths. Through simulation the proposed algorithm is shown to be better than Ant colony Multipath Routing (AMPR) and Colony-Based Multipath-Routing (CMPR) schemes.

Location Aware ACO Routing Algorithms

The dynamic topology of MANETs causes frequent link breakups which causes source node to spend most of its time in route set up and maintenance. In position based routing algorithms, each node will have general idea about the network topology and its neighbors thus the nodes can chose the nearest neighbor towards the destination. Hence, dynamic networks like MANETs require location aware routing protocols and the state-of-the-art location aware ACO based bio-inspired routing algorithms are described below.

• **GPS and Ant-Like Algorithm (GPSAL) (2001):**

D. Camara and A.F. Loureiro (2002; 2001) have developed Global Positioning System (GPS) based ACO routing algorithm referred to as GPSAL which works on the assumption of presence of GPS based mobile devices. The forward ants are flooded towards remote nodes in order to collect and disseminate the routing information and thereby ensuring bandwidth efficiency. GPSAL is compared with non bio-inspired Location Aided Routing (LAR) and results shown that GPSAL achieves lesser routing overhead when compared to LAR.

• **Mobile Ant Based Routing (MABR) (2006):**

Heissenbuttel M et al. (2006) have designed and developed a novel location aware ACO based bio-inspired routing algorithm for large scale ad hoc networks with irregular topologies referred to as MABR derived from the Ants-based Mobile Routing Architecture (AMRA). MABR divides the network area into rectangular zones and uses the concept of logical router for long distance routing. Ants are used to update the routing tables proactively in these logical routers and position based routing protocols are used for data forwarding. Further, pheromone evaporation of biological ants is mimicked in order to remove the stale path entries. Through simulation results authors have proved that MABR achieves superior performance when compared to GFG/GPSR.

- **Ant Routing with Distributed Geographical Localization (2009):**

  Kudelski M and Pacut (2009) have proposed an extended version of AntHocNet with distributed geographical localization knowledge for MANETs. The proposed algorithm partitions the network area into cells and then the routing is considered at the cell level in order to provide the knowledge locally. Through simulation authors have demonstrated that the proposed algorithm outperforms AnthocNet.

- **HOPNET (2009):**

  Wang Jianping et al. (2009) have proposed a location aware ACO based routing algorithm referred to as HOPNET which combines the ACO principles with Zone Routing Protocol (ZRP) and DSR routing protocols and HOPNET is based on hopping of ants from one zone to the next zone. In HOPNET, the network is divided into routing zones according to the radius measured in hops which may vary in size; then the route discovery will be local proactive within the zone and reactive across the zones. HOPNET maintains two routing tables in each node namely, *IntraRT* (intrazone routing table which will be maintained proactively) and *InterRT* (inter zone routing table which will be maintained reactively). In IntraRT, HOPNET uses ACO techniques and periodically forward ants to explore the paths with in the zone and backward ants are sent with the discovered path. Ants in IntraRT are classified into 5 types namely, *internal forward ant, external forward ant, backward ant, notification ant* and *error ant*. On the other hand InterRT of HOPNET uses the DSR technique to store the route across the zone. The simulation results show that HOPNET is highly scalable for large networks when compared to AntHocNet.

- **Hybrid Routing Based on Ant Colony and ZHLS routing protocol (HRAZHLS) (2010):**

  Marjan Kuchaki Rafsanjani et al. (2010) have proposed a location based ant routing algorithm referred to as HRAZHLS which inherits the features of both ant colony and Zone Based Hierarchical Link State (ZHLS) routing protocols. In HRAZHLS, the network area is divided into non overlapping zones where each zone size is dependent on *node mobil-*

ity, network density, transmission power and *propagation characteristics*. Like HOPNET, HRAZHLS also maintains two routing tables in each node namely *IntraRT* and *InterRT*. Further, HRAZHLS uses the same 5 types of ants used in HOPNET; and it uses proactive approach for inter zone route maintenance whereas the reactive approach is used for intra zone route maintenance. Authors claimed that the proposed algorithm improves the packet delivery ratio, reduces the delay and overhead.

- **Automatic Clustering Inspired by Ant Dynamics (ACAD) (2012):**

  Chowdhury Aritra and Das Swagatam (2012) have proposed an automatic clustering based heuristic algorithm referred to as ACAD which detects the separated clusters of any shape, either convex or non-convex, using *pseudo ants*. Further, pheromone decay concept is also used to remove the old solutions and authors have demonstrated that ACAD exhibits promising results with real and synthetic data sets.

Other ACO Routing Protocols for MANETs

In this section other ACO based routing algorithms are highlighted by focusing on the performance optimization in terms of throughput and end-to-end delay and do not fall under any category as discussed above. These algorithms use ACO techniques mainly for convergence or for route dissemination etc.

- **Cooperating Mobile Agents for Dynamic Network Routing (1999):**

  Minar et al. (1999) have developed ACO based routing framework referred to as Co operating Mobile Agents for Dynamic Network Routing which generates the mobile agents independent of network and data session at the network setup. These agents *do not die* and continuously take a random walk across the network in order to keep the history of previous *N* visited nodes and updates the same whenever a new node is visited. Through simulation results authors have demonstrated that the proposed framework exhibits superior performance under highly mobile network conditions.

- **Accelerated Ants Routing (2001):**

  Matsuo H and Mori K (2001) have developed ACO based Accelerated Ants Routing algorithm for MANETs referred to as AAR which is derived from ARA (Subramanian et al. 1997). AAR mainly concentrates on convergence property and no return rule (previous hop is not chosen as the next hop to avoid the network loop); further it uses *N* Step Backward Exploration in order to explore the network effectively. For path exploration AAR uses regular ants which consist of a stack of last *N* visited nodes and accordingly updates the pheromone table of each visited node. AAR uses both pheromone and heuristic values *(queue value)* to decide the next hop in order to forward the data packet to the destination node. Through simulation results authors have shown that AAR achieves superior performance and convergence when compared to AntNet, Q-routing and PQ-Routing.

- **Ant-Colony-Based Routing Algorithm (ARA) (2002):**

  M. Gunes et al. (2002) have developed an ACO based routing algorithm referred to as ARA which inherits the features of both AODV and AntNet and uses both forward and backward ants for route discovery and setup. ARA uses data packets to update the pheromone table and thus reduces the ant population for continuous path sampling. Further, pheromone evaporation concept of real ants is also mimicked in order to remove the old data entries in the table. Simulation results have shown that ARA performs better than the traditional routing protocols such as AODV, DSDV and DSR under low and moderate dynamic topological conditions.

- **Ant-AODV (2002):**

  Marwaha S et al. (2002) have proposed an ACO based routing algorithm for MANETs referred to as Ant-AODV in which a proactive route update mechanism is exploited in addition to the salient features of traditional reactive AODV routing protocol. Ant-AODV consists of ants which are randomly traversing the network and keep the record of last *N* visited nodes and accordingly these records are updated in each visited node. Since ants are used to continuously monitor and explore the new routes and thus Ant-AODV increases the network connectivity by reducing route discovery latency. Further ants continuously explore the new and better routes thus the data packets are routed to the best available path to the destination and thereby decreasing the end-to-end delay. Data packets are routed as per the AODV norms and ants are used only for spreading the routing information. Through simulation results authors have shown that Ant-AODV achieves better network connectivity with low end-to-end delay when compared to AODV routing protocol.

- **Probabilistic Emergent Routing Algorithm (PERA) (2003):**

  Baras J and H. Mehta (2003) have developed an ACO based routing algorithm referred to as PERA which floods the forward ants only during the beginning of the communication session or when routes are stale for route discovery. For each forward ant reaching the destination, backward ants are launched thus leading to multiple paths between the source and destination. Backward ant updates two tables namely, *routing* and *statistic* tables in each visited node. Reinforcement parameter *(either delay or number of hops)* is also used during the table updating to reflect the current state of the network. Data transmission takes place in the best path available and the rest of the paths are considered for route recovery and simulation results have shown that PERA is better than AODV.

- **ABC-AdHoc (2004):**

  B. Tatomir and L. Rothkrantz (2004) have proposed the hybrid routing algorithm based on Ant Based Control and AntNet referred to as ABC-AdHoc. ABC-AdHoc uses the forward ant concept for updating the pheromone over the path and further uses AntNets framework for probabilistic routing decision. Through simulation results authors have demonstrated that ABC-AdHoc achieves superior performance when compared to AntNet.

- **W_AntNet (2007):**

  Dhillon S et al. (2007) have proposed an AntNet based routing algorithm for MANETs referred to as W_AntNet and it uses *beacon messages* for periodic neighbor discovery. W_AntNet is dependent on *buffer size* of node and exhibits satisfactory performance under static network conditions. But W_AntNet suffers from looping packets and thereby deteriorating the performance under dynamic network conditions when compared to AODV and DSR.

- **Enhanced Ant colony based algorithm (2008):**

  N. Cauvery and K. Vishwanatha (2008) have proposed enhanced ACO based routing algorithm for MAENTs which is an extended version of ARA algorithm [28] and generates all possible paths between the source and destination node. Periodic route refreshing is used to find new and better paths; further time out mechanism is also used to resend the ants to deal with ant loss during the route discovery. Memory buffer is used to hold the packets during link failure and thereby avoiding packet retransmission. The proposed algorithm is evaluated with static data and shown that its performance is better than ARA.

- **Scented Node Protocol (SNP)(2010):**

  Yalin Evren Sagduyu et al. (2010) have proposed ACO based routing protocol SNP for MANETs which uses odor localization and tracking feature of insect colonies to enhance the routing. SNP gives scent based guidance to ants for route discovery and maintenance. Whenever a source wants to find the route to the destination then it sends broadcast ant towards the destination; each destination will have a food scent that attracts these ants. If ant finds the destination scent then it will change its communication mode from broadcast to unicast mode. Further, HELLO messages are used for spreading the scent information and thus help in local route repair. Through simulation results authors have shown that SNP exhibits superior throughput when compared to that of AODV and AntHocNet.

- **Simple Ant Routing Algorithm (SARA) (2010):**

  Correia, F and Vazao, T (2008; 2010), have proposed a Simple Ant Routing Algorithm for MANETs referred to as SARA which optimizes the routing process by using *Controlled Neighbors Broadcast (CNB), control message (FANT)* and *deep search* strategies. The simulation results have shown that SARA achieves the lowest overhead while increasing the throughput.

2.2.2 Bees Based Routing Algorithms for MANETs

A honey bee colony can intelligently choose the *quality* and *quantity* of the food source through simple behavior of individual bees. A honey bee colony can contain several kinds of worker bees such as *food-storer, scout* and *forager* which are responsible for recruiting and foraging in beehive. Bees perform dance *(waggle dance and tremble dance)* in order to communicate with their fellow bees and state-of-the-art bee based bio-inspired routing algorithms for MANETs are discussed as follows.

- **BeeAdHoc Routing Protocol (2005):**

  H. F. Wedde and M. Farooq (2004) have developed an *energy efficient* bee inspired routing algorithm for MANETs referred to as BeeAdHoc by inheriting the foraging principles of honey bees. Further, BeeAdHoc uses *scouts* and *foragers* for route establishment and maintenance phase of MANETs. Through extensive simulation, authors have demonstrated that BeeAdHoc exhibits superior performance in terms of packet delivery ratio, delay and throughput with lower energy consumption.

• **BeeAIS Routing Protocol (2007):**

Mazhar N and Farooq M. (2007) have developed an extended version of BeeAdHoc for MANETs referred to as BeeAIS which uses Artificial Immune Systems (AIS) framework to detect the misbehavior in BeeAdHoc and thereby providing the security. Further, through simulation results authors have shown that BeeAIS can counter the routing attacks in BeeAdHoc efficiently.

2.2.3 Flocks of Birds Based Routing in MANETs

Birds travel in flocks of ‘V’ shape where the sphere head takes the entire burden and thereby reducing the up thrust required by the rest of the birds in the flock; thus the other birds in the flocks save its energy. When the sphere head losses its energy then it will be replaced by other bird to take the burden of the flock. Each bird in the flock will not have any idea about its position in the flock and further no specific bird directs the movement of flocks. Thus birds can travel a long distance without losing much of its energy and this feature of flocks of birds have inspired many researchers to develop energy efficient bio-inspired routing algorithms for MANETs and the details are given below.

• **Bird Flight Inspired Clustering based Routing Protocol (2010):**

Mohit Tiwari and Shirshu Varma (2010) have proposed Bird Flight Inspired clustering based scalable and energy efficient routing protocol for MANETs. The proposed algorithm actively maintains the network clusters and uses position based routing scheme in order to choose dynamic gateway nodes and thus reduces the flooding of control packets. Through simulation results authors have shown that the proposed algorithm exhibits superior performance.

• **Bird Flocking Behavior Based Routing (BFBR)(2006):**

T Srinivas et al. (2006) have developed Bird Flocking Behavior based Routing for MANETs referred to as BFBR which uses *encounter search* mechanism in order to improve the route discovery latency; further BFBR uses direction forward routing for effective route maintenance. Through simulation results authors have demonstrated that BFBR exhibits encouraging results.

2.2.4 Termite Based Routing Algorithms for MANETs

Social insect *Termites* also commonly known as *white ants* are *autonomous, cooperative* and *interdependent* social insects without any *organizational plan* or *centralized coordination* but achieve global tasks by local means through *effective mutual coordination* and can easily *recover from the setback*. The termite uses the concept of Stigmergy which can be achieved through a chemical substance referred to as the *pheromone*. The *hill building nature* of the termite shows its coordinated behaviour and termites constructs the hill in two phases: *(1) non-coordinated* and *(2) coordinated*. In the first phase, termites deposit a pheromone impregnated pellets randomly and many hills will be emerging initially due to non-coordinated behaviour of the termites as shown in Figure 2.1(a). Depending on the density of the builders one of the hills will reach the second phase due to its critical size as shown in Figure 2.1(b). Thus, it attracts more termites in turn accumulates the large pheromone gradient and spreads the strong positive feedback. Figure 2.1(c) depicts the decaying of the pheromone deposited on the other hills over a period of time and thereby spreading the negative feedback (Marco Dorigo et al. 2000).

(a) At $t = i$ (b) At $t = j$ (c) At $t = n$

Figure 2.1: Hill Building Nature of the Termite at Different Time Intervals

Inspired by the behaviour of social insect termites, an adaptive routing algorithm referred to as Termite algorithm is proposed for MANETs (Roth M and Wicker S 2003a; Roth M and Wicker S 2003b). The packet forwarding technique of MANETs is correlated to the hill building nature of termite and it achieves the strong routing robustness through multiple paths to the destination. Termite algorithm is an adaptive and per packet probabilistic routing algorithm where routing decision is taken for each packet at every node based on the pheromone deposited on the outgoing links. Through stigmergy, it reduces the amount of control traffic thus increasing the data good put. The Termite algorithm treats each mobile node as a termite hill and the packets are strongly attracted towards the hill with highest pheromone gradient. While moving towards the destination the packets lay pheromone for its source in each hill it visits. Thus it increases the likelihood of packets following the same path while traversing back from destination. Each node maintains a pheromone table which is analogous to routing tables in traditional routing protocol, for tracking the amount of pheromone on each outgoing link. To prevent the stale entries in the pheromone table, the concept of pheromone decay over time is introduced. At every node, the pheromone increase is linear and decays exponentially over time. The Termite algorithm is driven by three functions namely *pheromone update, pheromone decay* and *forwarding* functions.

Pheromone Update: Whenever a node receives packet, it increments the pheromone for the packet source by a constant γ in the pheromone table. The pheromone update equation of Termite algorithm is shown in Equation (2.1) where *s* is the source node; *i* is the previous hop and $P'_{i,s}$ is the updated pheromone value in the pheromone table

$$P'_{i,s} = P_{i,s} + \gamma \tag{2.1}$$

Pheromone Decay: In order to remove old solutions, periodically (at every 1 sec) each entry in the pheromone table is multiplied by the decay factor $e^{-\tau}$, where τ is the decay rate. The pheromone decay equation is shown in Equation (2.2)

$$P'_{n,d} = P_{n,d} \cdot e^{-\tau} \tag{2.2}$$

Forwarding Function: Based on the amount of destination pheromone deposited over the neighbor link, the probability of neighbor link usage is calculated using forwarding function shown in Equation (2.3).

$$p_{n,d} = \frac{(P_{n,d} + K)^F)}{\sum_{i=1}^{N}(P_{n,d} + K)^F} \tag{2.3}$$

where $p_{n,d}$ is the probability of using neighbour node *n* to reach destination *d* and *N* is the total neighbour nodes for node *n*. Constants *K* and *F* denotes the pheromone threshold and pheromone sensitivity which are used to tune the routing behaviour of the Termite algorithm. Simulation results of Termite algorithm are promising giving good data good put with constant control packet overhead and it has been shown to perform well over a variety of mobility conditions. Reduced control traffic overhead, quick route discovery and repair are the highlights of termite algorithm. However the Termite algorithm has many tunable parameters and the efficiency of the algorithm depends on the values chosen for these parameters. Important among them is the pheromone update and decay which should reflect the current context of the network. Ruth M. H (2005) has proposed many techniques in order to reflect the current context of the network in termite routing algorithm and the first among them is γ pheromone filter (γPF) or continuous pheromone decay. In γ pheromone filter, whenever a node receives a packet then the pheromone on its entire neighbour links will be simultaneously decayed proportionally to the *inter packet arrival time*; further pheromone is also decayed when it is checked to send the packets. Each packet contains the pheromone γ equal to the utility of the path it has traversed and the γ Pheromone filter is described by pheromone update equation as shown in Equations (2.4) and (2.5).

$$\forall_i \quad P^n_{i,s} = P^n_{i,s}.e^{-(t-t^n_{s,obs})\tau} \tag{2.4}$$

$$P^n_{r,s} = P^n_{r,s} + \gamma \tag{2.5}$$

where $P^n_{i,s}$ is the amount of pheromone from source node *s*, on the neighbour node *i*, at node *n*; *r* represents the previous hop of the packet and γ is the amount of pheromone carried by each packet; τ is the pheromone decay rate; $t^n_{s,obs}$ is the last instant of time the pheromone is observed from source node *s* at node *n* where as *t* gives the current time. Apart from γ pheromone filter, authors have also proposed several other pheromone update and decay methods such as Averaging filter, Normalized γ Pheromone Filter (NγPF), Probabilistic Bellman-Ford (pBF), Ant-Based Control + X, Box Filter and Oracle in order to keep the fresh routing information in the pheromone table. Ant-Based Control + X is a combination of ABC pheromone update methods with the γPF (ABC+γPF), NγPF (ABC+NγPF) and with pBF (ABC+pBF).

Roth M and Wicker S (2004a) have presented an analytical model in order to study the proposed pheromone update methods of the Termite algorithm. The asymptotic behaviour of the Termite algorithm is studied by using an analytical model and the time average pheromone value deposited over both single and double links is determined. The properties of the Termite algorithm is investigated for γ pheromone filter, Joint Decay Infinite Impulse Response Filter-2 (IIR2) and pDijkstra Pheromone update methods and authors have also explored the relationship between the parameters *(decay rate and packet arrival rate)* and have found the scale invariance parameter. Through theoretical results, authors have shown that the amount of pheromone on a link influences the performance of the routing algorithm in terms of both data good put and adaptability. The same authors in (2004b) have given an analytical study on undesirable property of the Termite routing algorithm in terms of *tendency to converge on only one path between source and destination*; thus the Termite algorithm will not take full advantage of multipath routes to the destination. For the study, authors have used continuous pheromone decay function which works as follows: Whenever a node receives a packet then pheromones over all its neighbor links will be decayed proportionally based on the inter packet arrival time and then the link on which packet has arrived is positively reinforced by updating the pheromone by γ.

Continuous pheromone decay function is defined by

$$P_{i,s}^{n'} = \begin{cases} P_{i,s}^{n} \cdot e^{-(t-t_s^n)\tau} + \gamma & i = k \\ P_{i,s}^{n} \cdot e^{-(t-t_s^n)\tau} & i \neq k \end{cases} \tag{2.6}$$

Where $P_{i,s}^{n'}$ is the amount of pheromone from source node *s* from neighbor node *i* at node *n*; *k* stands for the previous hop of the packet; γ is the amount of pheromone carried by the packet; τ is the decay rate and t_s^n is the previous time that a packet has arrived at node *n* from *s*where as *t* gives the current time. Authors employed normalized uncoupled pheromone update function in order to overcome the undesirable stagnation problem of Termite algorithm and thus distribute the traffic proportionally across the outgoing links. Pheromone on each link will be decayed only when a packet actually arrives on it and for normalization authors have used two filters namely: The One Tap IIR Averaging Filter and The Box Filter.

Ruth M. H (2005) has proposed ReTermite algorithm for MANETs which uses Source Pheromone Repel (SPR) which pushes the packet away from its source and at the same time the packet will be pulled towards its destination. Source and destination pheromone distributions of ReTermite are calculated using (2.7) and (2.8). The next hop is chosen randomly according to the meta distribution $\hat{p}_{i,d}^n$ as shown in (2.9).

$$p_{i,d}^n = \frac{(P_{i,d}^n + K)^F}{\sum_{j \in N^n} (P_{j,d}^n + K)^F} \tag{2.7}$$

$$p_{i,s}^n = \frac{(P_{i,s}^n + K)^F}{\sum_{j \in N^n} (P_{j,s}^n + K)^F} \tag{2.8}$$

$$\hat{p}_{i,d}^n = \frac{p_{i,d}^n (p_{i,s}^n)^{-R}}{\sum_{j \in N^n} p_{j,d}^n (p_{j,s}^n)^{-R}} \tag{2.9}$$

The salient features of the ReTermite algorithm such as true *continuous pheromone decay, piggybacked routing information* and *promiscuity* found to be very effective. The simulation results have demonstrated that ReTermite outperforms the traditional AODV routing protocol. In this thesis an effort has been made to use the distinguishing feature of social insect Termite in design and development of bio-inspired QoS aware routing algorithms for MANETs.

2.3 Non-SI Based Bio-Inspired Routing Algorithms

Apart from salient features of swarms *(ants, bees, flocks of birds and termites)* there exist other animal societies such as mammal bats and have also motivated several researchers for designing adaptive bio-inspired algorithms for solving complex real world problems. However bats based bio-inspired routing algorithms have not been found in the literature for networking and telecommunication applications; but could be found for optimizing non networking problems such as *brushless DC wheel motor problems, continuous constrained optimization problems* (Airas M 2003; Bora T.C et al. 2012; Yang X. S 2010). In this thesis an attempt has been made to exploit the salient features of mammal bats in design and development of bio-inspired routing protocols for MANETs.

2.4 Hybrid Bio-Inspired Routing Algorithms

As each of the above discussed social insects and other animal societies have wonderful characteristics and hence it could be possible to combine the unique features of both social insects and animal societies for the design and development of novel hybrid routing algorithms for MANETs. In the following the state-of-the-art hybrid bio-inspired routing protocol for MANETs is discussed.

- **Nature Inspired Scalable Routing Protocol (NISR) (2011):**

  Gudakahriz Sajjad Jahanbakhsh et al. (2011) have proposed a hybrid Nature Inspired Scalable Routing Protocol for MANETs referred to as NISR which inherits the salient features of both *ant* and *bee colonies* and NISR is derived from TORA. In NISR, *scout bees* are responsible for finding the food source and its quality; and when scout bees returns to the hive then it indicates the *distance to the food source, direction, quality* and *quantity of food source* to other bees through dance. Based on this information hive sends bees to the food source and when bees reach the food source, *an ant* and *a bee* is sent back to hive. Ant is responsible for updating the pheromone over the path while bee is responsible for informing the quality of food source to the hive. Scout bees continuously search for new paths to the food source and will periodically notify to the hive.

Tables 2.1 to 2.9 highlights the analysis of all the studied bio-inspired routing protocols for MANETs and Figure 2.2 shows the taxonomy of state-of-the-art ACO based bio-inspired routing protocols with respect to properties hired. In the Tables following acronyms are used: *F-Forward Ant, B-Backward Ant, EA-Enforce Ant, AA-Anti Ant, N-Negative Backward Ant , D-Destination Trail Ant, RA-Regular Ant, LFA-Local Foraging Ant, IFA-Internal Forward Ant, EFA-External Forward Ant, NA-Notification Ant, EA-Error Ant.*

Table 2.1: Congestion Aware ACO Based Bio-Inspired Routing Algorithms

Sl.No	Algorithm	Parameters	Agents	Purpose of Agents	Remarks
1	ANSI	Distance to the destination in hops	F,B	For Route Discovery and Setup	Targets Congestion
2	EARA	Cognestion	LFA	For Route Discovery and Setup	ARA with multipath, Targets Congestion
3	AntHocNet	No. of Hops, Traffic Congestion, SNR	F,B	For Route Discovery and Setup	Targets Congestion, Open Space Evaluation
4	ARAAI	Link Stability based on Node Mobility	F,B	For Route Discovery and Setup	Targets Stagnation
5	PAR	Queue Length	F,B	Continuous Route Exploration	Congestion Aware and it is more explorative

Table 2.2: Load and Mobility Aware ACO Based Bio-Inspired Routing Algorithms

Sl.No	Algorithm	Parameters	Agents	Purpose of Agents	Remarks
1	MARA	Constant Pheromone	F,B	For Route Discovery and Setup	Load Balancing
2	AntOR	No. of Hops	F, B	For Route Discovery and Setup	Load Balancing
3	PACONET	Active Connection Time	F,B	For Route Discovery and Setup	Targets Mobility and Route Maintenance

Table 2.3: CLD and Hybrid ACO based Bio-Inspired Routing Algorithm

Sl.No	Algorithm	Parameters	Agents	Purpose of Agents	Remarks
1	EARA-QoS	Delay, Queue Length	LFA	For Route Discovery	CLD, QoS Aware and Targets DiffServ
2	NISR	Constant Pheromone	Ants and Bees	Scout bees For Continuous Route Exploration and Setup, Ants for Reinforcement	Hybrid Routing

Table 2.4: Bees Based Bio-Inspired Routing Algorithms

Sl.No	Algorithm	Parameters	Agents	Purpose of Agents	Remarks
1	BeeAdHoc	Energy	scouts and foragers	route establishment and maintenance	CLD, QoS Aware and Targets DiffServ.
2	BeeAIS	Energy	scouts and foragers	route establishment and maintenance	Targets Security

Table 2.5: Location Aware ACO Based Bio-Inspired Routing Algorithms

Sl.No	Algorithm	Parameters	Agents	Purpose of Agents	Remarks
1	GPSAL	No pheromone Concept	F, B	For Spreading Routing Information	Bandwidth Efficient in flooding ants
2	MABR	No pheromone Concept	F	For finding new paths and for Updating Routing Info.	Part of AMRA Better than GFP/GPRS
3	HOPNET	Time taken To Reach Node K from Source S	IFA, EFA, B, NA, EA	IFA-Maintains Intra Zone Routing Table EFA-Maintains Inter Zone Routing Table B For Route Setup NA and EA Route Maintenance	Good for large networks with both low and high mobility conditions Better than GFP/GPRS
4	HRAZHLS	Time taken to reach node K from Source S	IFA, EFA, B, NA, EA	IFA-Maintains Intra Zone Routing Table EFA-Maintains Inter Zone Routing Table B For Route Setup NA and EA Route Maintenance	Targets Routing Overhead
5	ACAD	Anti pheromone	F	For Cluster Detection	Automatic and Shape Independent Clustering

Table 2.6: Flocks of Birds Based Bio-Inspired Routing Algorithms

Sl.No	Algorithm	Parameters	Remarks
1	Mohit Tiwari and Shirshu Varma	Energy	Scalable and Cluster Based
2	BFBR	Energy	Low route discovery latency and effective route maintenance

Table 2.7: QoS Aware ACO Based Bio-Inspired Routing Algorithms

Sl.No	Algorithm	Parameters	Agents	Purpose of Agents	Remarks
1	ADRA	Node velocity, queuing delay and hop count	F, B, EA, AA	For Route Discovery/setup (F,B)Congestion Handling (EA, AA)	QoS aware, Congestion and Load Balancing
2	ARAMA	Energy, Processing Power, Link Bandwidth	F, B, N, D	For Route Discovery and Setup	QoS Aware, Energy Efficient, Fair Resource Distribution
3	SDVR	Delay, Jitter and Energy	F, B	For Route Discovery and Setup	QoS Aware, Higher Overhead
4	FACO	Energy, Packet Buffered in Queue, Signal Strength	F, B	For Route Discovery and Setup	Fuzzy based, QoS Aware
5	ADSR	Delay, Jitter & Energy	F, B	For Route Discovery and Setup	QoS Aware
6	MQAMR	Bandwidth , Delay, Hop Count, Mobility and Energy	F, B	For Route Discovery and Setup	QoS aware, Fair Bandwidth Allocation, Longer Network Lifetime
7	S.Kim	Node Distance, Queue Length, Bandwidth, Delay, Energy	F, B	For Route Discovery and Setup	Load balancing, QoS Aware

Table 2.8: Termite Based Bio-Inspired Routing Algorithms

Sl.No	Algorithm	Parameters	Remarks
1	Termite Algorithm	Constant Pheromone	Robust, Reduces Control Packets
2	Termite Algorithm	Inter Packet Arrival Time	Reflects the Current Context of the Network
3	ReTermite	Inter Packet Arrival Time	True continuous pheromone decay, piggybacked routing information and promiscuity

Table 2.9: Other ACO Based Bio-Inspired Routing Algorithms

Sl.No	Algorithm	Parameters	Agents	Purpose of Agents	Remarks
1	AAR	Queue Length	F, B	Continuous Route Exploration and Path Setup	Targets convergence and average packet latency
2	ARA	Constant Pheromone	F, B	For Route Discovery and Setup	Targets Routing Overhead
3	AntAODV	No Pheromone	RA	For Route Discovery and Setup	Increased Connectivity
4	PERA	Delay or No. of Hops	F, B	For Route Discovery and Setup	Low Good Put under High Mobility Condition
5	ABC-AdHoc	Constant pheromone	F	Continuous Route Exploration and Path Setup	Combination of ABC and AntNet
6	W_AntNet	Trip time, nodes queue status	F, B	Continuous Route Exploration and Path Setup	Good for static networks
7	SNP	Hop Distance	RA	To Spread Destination Scent	Odor Localization and Tracking
8	SARA	Constant Pheromone	F, B	For Route Discovery and Setup	Targets Control Packet Overhead

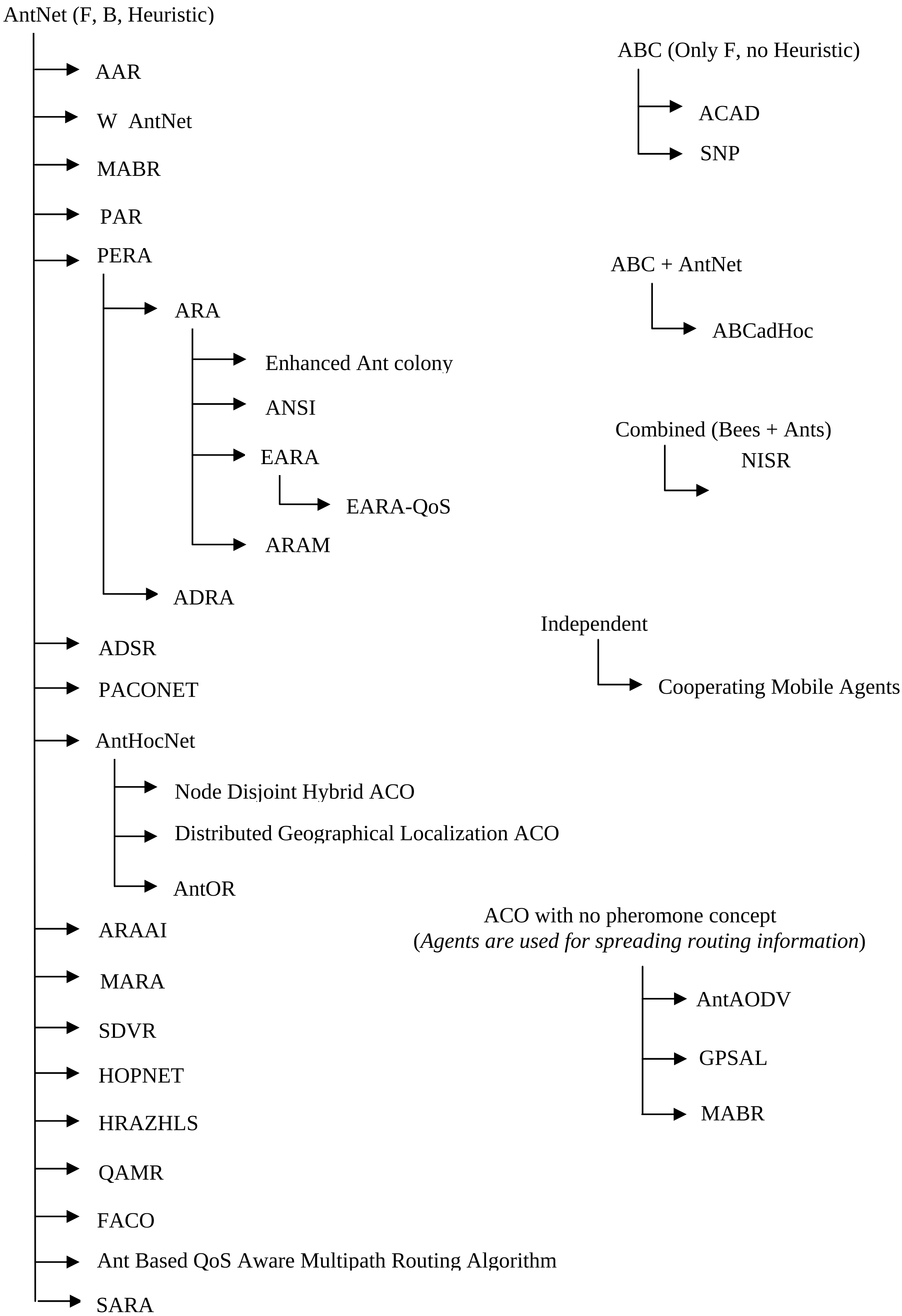


Figure 2.2: ACO Taxonomy based on Agents and Heuristics used in Routing

2.5 Outcome of The Literature Review

Based on the extensive literature review, the following issues and challenges are observed for further research:

1. *Cross Layer Design* has been receiving the tremendous attention among several researchers for increasing the efficiency of MANETs but could not be found in bio-inspired routing protocols.

2. *Swarm Intelligence* based routing protocols for MANETs by and large focused on ACO and only a handful of routing protocols which inherits the behaviour of social insect Termites could be found.

3. Existing Termite based routing protocols for MANETs concentrate on inter packet arrival time for pheromone update and decay over the link. Finding the *stable* and *reliable* nodes for the path with this approach is bit difficult as MANETs suffer from the frequent link breakups due to the mobility of nodes.

4. Very few ant colony routing algorithms for MANETs have tried to *mitigate the stagnation problem* in general and in particular the Termite based routing algorithm could be found in solving the stagnation problem (Roth M and Wicker S 2004a; 2004b). Further, *Load aware bio-inspired routing protocols* for MANETs has motivated several researchers in combating the stagnation problem efficiently.

5. A little effort could be found in exploiting the intelligent features of social insect and animal societies in *designing hybrid bio-inspired routing protocols* for MANETs (Gudakahriz Sajjad Jahanbakhsh et al. 2011).

6. Further, the requirements such as *Context Awareness, Load Balancing* and *Quality of Service Awareness* have attracted several researchers in designing a *novel and hybrid bio-inspired routing protocols* for MANETs using salient features of Cross Layer Design and social insect and animal societies.

2.6 Problem Statement

In order to solve the open issues and challenges discussed in the previous section, there is a need to develop context aware, load balanced and QoS aware adaptive routing algorithms for MANETs in order to provide the best QoS to the applications. Accordingly, the research problem is stated as follows:

"To design and develop novel bio-inspired QoS aware routing protocols for MANETs by exploiting the salient features of *Termites, Bats* and *CLD* and thereby providing the best QoS".

2.7 Research Objectives

The objectives of this research work are as follows:

1. To design *Intelligent Cross Layer Models* using software agents so that network layer in MANETs can think and react to the context of the network.
2. To design and develop an *adaptive bio-inspired Context Aware and QoS Aware routing protocol* for MANETs using the salient features of social insect *Termites*.
3. To design and develop an *hybrid bio-inspired QoS aware routing protocol* for MANETs using the salient features of both social insect *Termites* and mammal *Bats*.
4. To design and develop an *heuristic hybrid Load Balanced bio-inspired QoS aware routing protocol* for MANETs using the salient features of both social insect *Termites* and mammal *Bats*.
5. To design and develop an *heuristic Load Balanced bio-inspired QoS aware routing protocol* for MANETs using the salient features of social insect *Termites*.
6. To evaluate the QoS performance of designed protocols (objectives 2, 3, 4 and 5) with other state-of-the-art bio inspired and non-bio inspired routing protocols.

In order to accomplish the five objectives of the research work, novel bio inspired QoS aware routing protocols for MANETs are designed and developed and the brief description of the research contributions are as follows:

- **Design of Intelligent Cross Layer Models**

  Two intelligent Cross Layer Models across Application, Network and Physical layers are designed to get the priority information of the application and distance information of the neighbor node. Accordingly, the packets belonging to prioritized applications is given more chances to access the channel. Thus, these proposed Cross Layer Models will be able to increase the throughput of the prioritized applications in MANETs.

- **Bio-Inspired Context Aware and QoS Aware Routing in MANETs**

  A novel bio-inspired routing protocol based on the behavior of social insect *Termites* referred to as *Context Aware or Mobility Aware-Termite (MA-Termite)* is proposed for MANETs. The primary objective of MA-Termite is to emphasize the *stable nodes* by positively reinforcing them with less pheromone decay and also to deemphasize the unstable nodes by negatively reinforcing them with high pheromone decay. The secondary objective of MA-Termite is to propose the *distance variant analytical model* for pheromone update method of *MA-Termite*. The proposed analytical model determines the average expected amount of pheromone over both the single and double links by observing the asymptotic behavior of the *MA-Termite*. The proposed *MA-Termite* algorithm exhibits superior routing features such as *high robustness through multiple paths to the destination, high route reliability with stable nodes* and *less control packet overhead*.

- **Hybrid Bio-Inspired QoS Aware Routing in MANETs**

  A novel hybrid bio-inspired routing protocol for MANETs referred to as *Bat-Termite* is proposed based on the major behavioral advantages of both social insect *Termites* and the mammal *Bats*. The proposed Bat-Termite algorithm exhibits superior routing features such as *high robustness with efficient management of multiple routes and rapid route repair and maintenance*.

- **Hybrid Bio-Inspired QoS Aware Load Balanced Routing in MANETs**

  A novel heuristic hybrid bio-inspired routing with load balancing approach referred to as *Load Balanced-Bat-Termite (LB-Bat-Termite)* is proposed for MANETs by combining the distinguishing features of both social insect *Termites* and mammal *Bats*. The primary objective of the *LB-Bat-Termite* algorithm is to find the stable nodes and thereby giving preferences for these stable nodes during the path setup; thus finding the reliable route to the destination. The secondary objective of the proposed *LB-Bat-Termite* algorithm is to mitigate the *stagnation problem* by using pheromone heuristic control method. The tertiary objective of the proposed *LB-Bat-Termite* algorithm is to provide QoS to the applications and to manage multiple routes to the destination node efficiently. The proposed *LB-Bat-Termite* algorithm exhibits superior routing features such as *high route reliability with stable nodes, quick route discovery and repair, high robustness with efficient management of multiple routes* and *stagnation avoidance*.

- **Bio-Inspired QoS Aware Load Balanced Routing in MANETs**

  A novel heuristic bio-inspired routing with load balancing approach referred to as *Load Balanced-Termite (LB-Termite)* is proposed for MANETs by exploiting the salient features of social insect *Termites*. The primary objective of the *LB-Termite* algorithm is to find the reliable route to the destination by giving preferences for stable nodes during the path setup. The secondary objective of the proposed *LB-Termite* algorithm is to mitigate the *stagnation problem* by using pheromone heuristic control method. The tertiary objective of the proposed *LB-Termite* algorithm is to provide QoS to the applications. *LB-Termite* algorithm exhibits superior routing features such as *high route reliability with stable nodes, quick route discovery and repair, high robustness with efficient management of multiple routes* and *stagnation avoidance*.

2.8 Summary

In this chapter, existing state-of-the-art bio-inspired routing protocols are discussed in detail while considering several important features of social insects and animal societies. A detailed classification of SI and non-SI based routing algorithms is also given based on the code of practice and properties hired and finally open issues and research challenges in designing routing algorithms for MANETs are highlighted. This literature review will give the detailed description of *Ants, Bats, Bees, Flocks of Birds* and *Termite* based bio-inspired routing protocols under *Congestion, Cross Layer Design, Load, Location, Mobility* and *Quality of Service* Aware categories.

Chapter 3

Intelligent Cross Layer Models

This chapter discusses the design of two intelligent cross layer models across Application, Network and Physical layers in order to get the *priority information of the application* and *distance information of the neighbor nodes*. Accordingly, the packets belonging to prioritized applications are given more chances to access the channel. Thus, these proposed cross layer models will enhance the throughput of the prioritized applications of MANETs. The validation of the proposed cross layer models is performed in terms of throughput gain of the prioritized flow when compared to non prioritized flow. The details of cross layer models are given below.

3.1 False Route Failure in MANETs

Mobility of nodes causes frequent link failures in MANETs which in turn causes packet losses; further collision contributes for an additional packet loss resulting in misinterpretation of link failure referred to as *False Route Failure (FRF)* which triggers the route maintenance phase. Route maintenance phase is a complicated process in terms of both time and network load; thus reduces the throughput and thereby providing the QoS is difficult. Several attempts have been made in order to control the FRF while minimizing the collision (M Gunes et al. 2003; K. Nahm et al. 2005; Chen, Q and Niu, Z 2004). Finding the exact reason for the packet loss in MANETs is very difficult and challenging task; hidden node problem at the MAC layer is one of the reasons for the packet loss which in turn reduces the throughput(Xu.S and Saadawi, T 2001). The working principle of IEEE 802.11 MAC is explained in the next section.

3.2 Working Principle of IEEE 802.11 MAC

The IEEE 802.11 MAC uses *Distributed Coordination Function (DCF)* in order to share the medium among multiple contenders; further DCF uses Request To Send (RTS)-Clear To Send (CTS) mechanism to avoid packet collisions. Figure 3.1 shows the RTS-CTS scheme that consists of the special packets namely RTS and CTS which are small in size and minimize the network load when compared to the data packet.

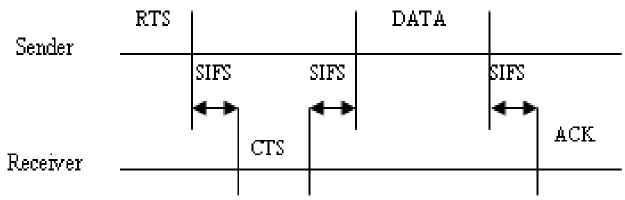


Figure 3.1: Working Principle of IEEE 802.11 DCF MAC

When a sender has data packet to be sent, then the sender transmits RTS packet to the receiver and the receiver in turn responds with the CTS packet only when it is ready to receive the packet. Upon receiving the CTS packet the sender starts transmitting the data packet. However, IEEE 802.11 DCF MAC does not work well in multi hop environment due to hidden node problem.

3.3 Hidden Node Problem at the MAC Layer

Figure 3.2 describes the hidden node problem and False Route Failure (FRF) of IEEE 802.11 DCF MAC layer of MANETs. Let us consider host node A which could not establish RTS-CTS four way handshake with target node B even though B is within As transmission range; thus A misinterprets B as unreachable node and leading to FRF. The node X which is within the interference range of node *A* will sense the transmission from *A* but *X* cannot respond to any other neighboring node *Y* if *A* is in the transmission mode. Then node *Y* is referred to as hidden node which sends RTS packets to node *X* but *Y* is ignored till the transmission channel is free from node *A* (M Gunes et al. 2003). Bigger data packets need much longer transmission time when compared to RTS/CTS packets. Thus, if node *A* is involved in transmitting a data packet, there is a high chance that subsequent retries of RTS from node *Y* may fail. However, the smaller packets with weaker burst feature are inferior when compared to bigger data packets from the channel access point of view; further, there will be higher probable channel occupancy

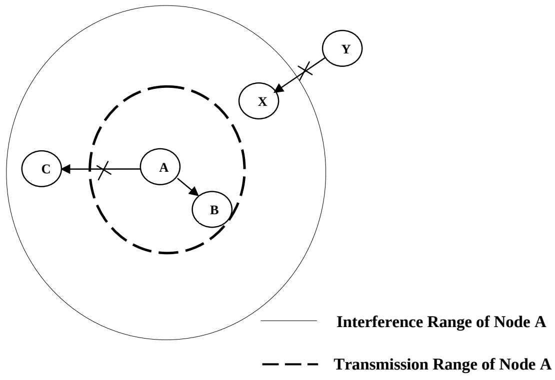


Figure 3.2: Hidden Node Problem

by these bigger data packets (Yumei Wang et al. 2004). As shown in Figure 3.2, node *Y* tries for default number of RTS retransmissions and accordingly drops the data packet assuming node *X* as non reachable and thus triggers the route maintenance phase.

In ad hoc networks collision occurs more often and whenever an RTS packet collides, the sender retransmits another RTS packet and the number of RTS retries is limited to number of SRL times *(the default value of SRL in IEEE 802.11 is 7)*. If a node does not succeed within this retry limit then it drops the corresponding data packet and thus misinterprets as link failure which triggers the route maintenance phase. It is very difficult to classify the packet loss either as the collision loss or the mobility induced loss ; in a static topology the packet loss is mainly due to the collision. If it is a collision loss then the network suffers from unnecessary overhead due to route maintenance phase and thereby reducing the throughput.

Another reason for packet loss is mobility due to which the target node may move out of the transmission range causing the transmitting node to drop the packet. As shown in Figure 3.2, node *A* is trying to establish RTS-CTS handshake with node *C*, which is not within its transmission range; node *A* tries for default number of RTS retransmissions and drops the data packet thereby reducing the throughput. Hence providing self awareness in MANETs is an important research issue. In order to find the reason for packet loss and thereby providing the QoS to the prioritized flow there is a need to design and develop cross layer models and the details are given below.

3.4 A Service Driven Cross Layer Model

Traffic in MANETs is expected to carry a mix of real time multimedia and non real time file transfer etc. but there are different services need to be prioritized and these may be either real or non real time application, for example: user may find file transfer over FTP, which is non real time, is more important than a phone call which is in real time. Since the network resources are finite and distinguished it is very difficult to give prioritized services for such kind of traffic. Hence a service driven cross layer model (Kiran M and Reddy G R M 2011b) is developed based on two important criteria: *(a) Cross Layer Design and (b) Dynamic Retry Limit* and the details are as follows.

- **Cross Layer Design**

  The proposed cross layer design allows the user to give priorities for the flows either in the beginning or in the middle of the session so that the packets are treated differently at the lower layers. These priorities are accepted by the Application layer from the user and the corresponding information has to be sent to the lower layers. It is very difficult to achieve this specific task if we use the OSI model which will not allow non adjacent layers to communicate with each other. Hence, a new cross layer design is built across Application, Network and MAC layers as shown in Figure 3.3.

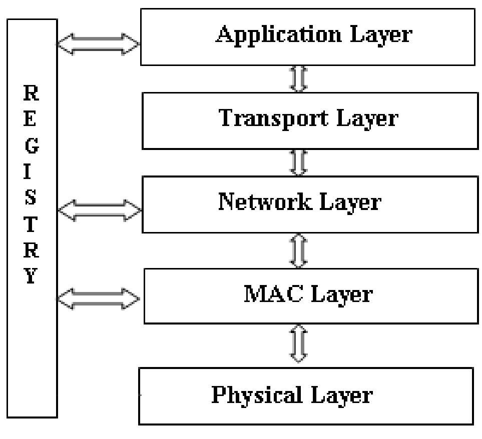


Figure 3.3: Cross Layer Model Across Application, Network and MAC Layers

The designed cross layer model consists of a shared registry which allows the non adjacent layers of OSI model in order to communicate with each other layer.

The priority information for the application is put in the shared registry by the Application layer which will be accessed by the Network layer and accordingly informs the MAC layer in order to provide the dynamic retry limit for the prioritized packets.

- **Dynamic Retry Limit**

  Whenever a network layer receives the prioritized packets, then it accordingly informs the MAC layer to adjust the Short Retry Limit (SRL). Sehoon Kim et al (2008) have shown that the SRL value of 30 yields the maximum overall throughput and for greater than 30, the throughput gets saturated. In the proposed service driven cross layer model, SRL value of 30 is considered for the higher priority packets by dynamic adjustment at the MAC layer i.e. a node will try for 30 RTS retransmissions before dropping a packet and thus triggering the route maintenance phase. For the un-prioritized packets, the default SRL value (SRL=7) is set. Here, we consider two different scenarios for adjusting the SRL and the details are given below:

  - **Scenario 1: Static Prioritized SRL**

    In this static scenario, a single or multiple flows will be given utmost priority among all other flows in the network and the priority is given in the initial stage i.e. at the beginning of the data flow and the priority will remain the same during the entire session. Further, the user has no liberty to change or swap the priority among the flows during the entire session.

  - **Scenario 2: Dynamic Prioritized SRL**

    In this dynamic scenario also a single or multiple flows will be given priority among all other flows in the network, but the priority for the flows can be assigned or changed dynamically during the entire session.

3.5 Prioritized QoS Based Cross Layer Model

The objective of the proposed Prioritized QoS Based cross layer design model is to provide the self awareness in MANETs using the neighborhood node distance concept and thereby increasing the throughput of a prioritized flow. Hence, the cross layer design based on Prioritized QoS referred to as PrQoS Cross Layer Model is accordingly designed and developed (Kiran M and

Reddy G R M 2011a). PrQoS Cross Layer Design Model consists of three parts namely *(a) Cross Layer Design (b) Neighborhood Node Distance* and *(c) Dynamic Retry Limit*. The details of PrQoS Cross Layer Model are discussed below.

- **Cross Layer Design**

  The proposed cross layer design model is developed based on two cross layer parameters namely *flow priority* and *Received Signal Strength (Pr)*. Since the objective is to give priorities to the flows, the priority information has to be accepted from the user at the Application layer and should be sent to the lower layers for further processing. For finding the distance between the nodes, *Pr* of the packet from Physical layer should be made visible to upper layers. It is difficult to achieve these two goals in the legacy protocol stack since non adjacent layers of OSI model cannot directly communicate with each other. With these two requirements, a cross layer design model is developed for PrQoS across the protocol stack as shown in the Figure 3.4.

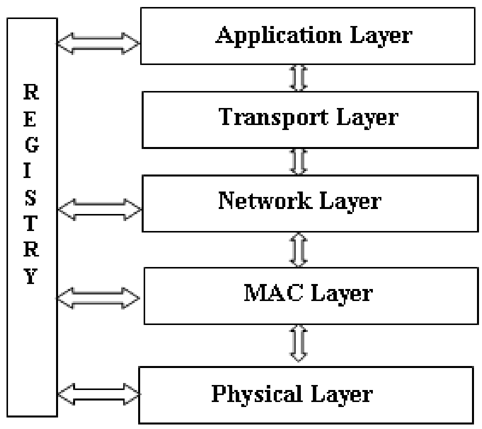


Figure 3.4: Cross Layer Model Across Application, Network, MAC and Physical Layers

  Through the shared registry, Application layer will inform the lower layers about type of service to be given on the priority and further Physical layer shares *Pr* to upper layers for finding the *distance between the nodes*.

- **Neighbourhood Node Distance and Dynamic Retry Limit**

  Since the PrQoS increases the RTS retry limit then it is important to identify whether the target node is within the vicinity of the transmitting range of the node or not. If the target node is not within the transmission range then increasing the RTS retry limit will not be useful for both *time* and *bandwidth*. This will in turn cause the performance degradation and also there will be a very late response to the actual link failure. Hence retry limit should be increased only when the node is within the transmission range. Further, *Pr* from Physical layer is used for finding whether the target node is within the transmission range or not since *Pr* is inversely proportional to the distance between the nodes (Kiran M and Reddy GRM 2011b). *Pr* is measured as soon as the packet is received from the neighbor node and the same can be used for calculating the distance *d* between the transmitting and the receiving node of the packet using the free space propagation model which is as follows:

$$d = \sqrt[4]{\frac{Pt.Gt.Gr.ht^2.hr^2}{Pr.L}} \tag{3.1}$$

where ,

Pt : Default transmission power	*Pr: Received signal power*
Gt: Antenna gains of the transmitter	*Gr: Antenna gains of the receiver*
ht and hr: Heights of the antennas	*L: System loss (1 by default)*

The distance *d* along with the time at which d is calculated is maintained in a distance table at every node and it is done only for the neighbor node which participates in the active route and not for all the neighbors and thus reduces the distance table entries (Wooi King Soo et al. 2007). Once the distance between the nodes is calculated, it is then possible to determine whether the nodes are moving apart or moving closer by calculating the difference between current and previous distances. Based on this distance information, for each node a new warning range is created as shown in the Figure 3.5.

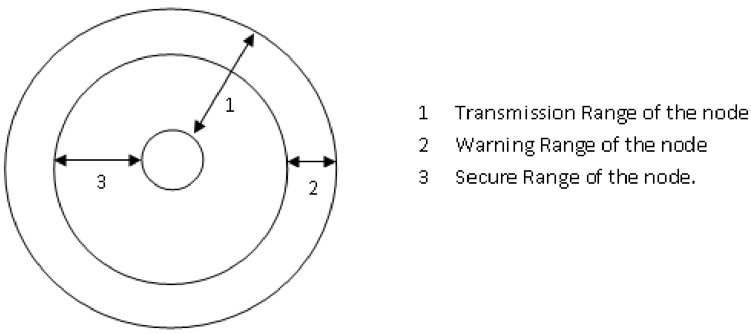


Figure 3.5: Secure and Warning Range of the Nodes

In the radio propagation model used for the study, it is assumed that each node has a transmission range of 250 meters and an interference range of 550 meters; the warning range is taken between 245 to 250 meters. Whenever a node receives a packet to be sent towards the next hop then the Network layer finds the *distance, time* and *movement information of the next hop* using the distance table. Two cases are considered for describing the working principle of PrQoS based cross layer model and the details are given below:

- **Case 1: Higher priority flow**
  - If the target node is within the secure range then the retry limit is high
  - If the target node is in the warning range AND it is coming closer then retry limit is high.
  - If the target node is in the warning range AND it is moving away then retry limit is low.
  - If the target node is in the warning range AND it is constant then the retry limit is high.
  - In any case if the information stored in the distance table is stale then the retry limit is medium.
- **Case 2: Lower priority flow**
  - Irrespective of the range where the target node lies, the retry limit is medium.

Based on these working principles of PrQoS, Network layer informs the MAC layer in order to adjust the retry limit accordingly. A high retry limit is set as twice the default RTS retry limit (SRL=14) whereas the low retry limit is set to half the default RTS retry limit (SRL=4) and the medium retry limit is set to the default RTS retry limit(SRL=7). If the target node is within the secure range then the RTS retry limit is set high as it is obvious that the RTS time out is mainly due to interference but not due to the link failure. If the target node is within the warning range and it is moving towards the transmitting node, RTS retry limit is set high as chances of link failure due to target node moving out of the transmission range is almost nil. If the target node is within the warning range and it is moving away from the transmitting node, the retry limit is set to low as the node may go out of the transmission range soon. If the target node is within the warning range and it is constant, the retry limit is set high since the chances of node moving out of the transmission range is less. In any case if the information stored in the distance table is old, then the default behavior of the protocols will be considered.

3.6 Results and Discussion

Simulation has been carried out to evaluate the two intelligent cross layer models in NS-2 and in the simulation AODV is used as Network layer protocol and IEEE 802.11 DCF is used as the MAC layer protocol; the data rate of wireless channel is fixed at 2Mbps. The simulation parameters are listed in Table 3.1.

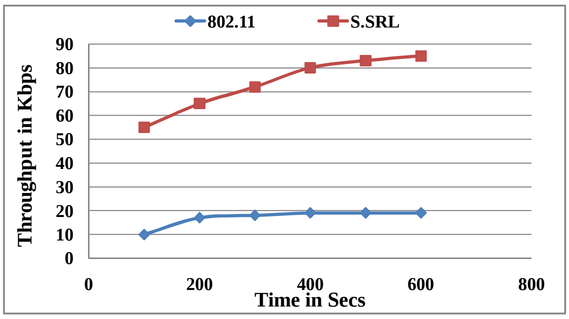
Table 3.1: Simulation Parameters For Evaluation of Intelligent Cross Layer Models

Sl.No	**Parameter**	**Value**
1	Simulation Time	600 Sec
2	Transmission Range	250 m
3	Interference Range	550 m
4	Traffic Type	FTP
5	Payload	1500 bytes
6	Node Density	50 nodes
7	Pause Time	0
8	Simulation Area	1000m x 1000m
9	Topology	Nodes are uniformly distributed in the network

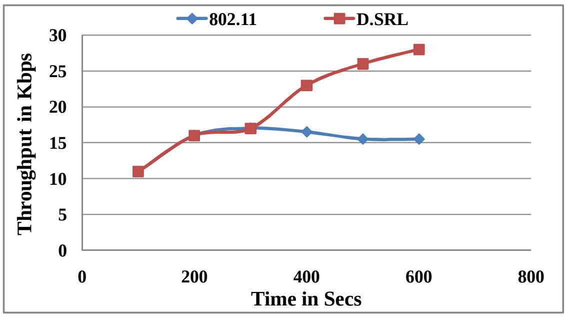
3.6.1 Simulation Results of Service Driven Cross Layer Model

Two experiments were conducted using single flow F1 and for Static Prioritized SRL (S.SRL), priority is assigned to F1 at the beginning of the data flow and the same priority is kept till the end of the simulation. For Dynamic Prioritized SRL (D.SRL), F1 will start with normal priority and the priority is dynamically given to F1 after 200 seconds of simulation time. Further, AODV and DSR protocols are considered for performance evaluation of service driven cross layer design model and the experimental results are shown in Figures 3.6 and 3.7.

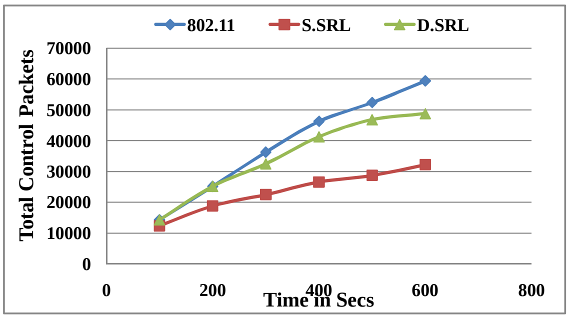
• **Simulation Using AODV Protocol**



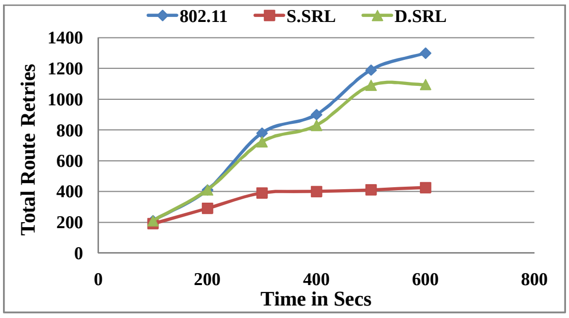
(a) Throughput-S.SRL



(b) Throughput-D.SRL



(c) Total Control Packets



(d) Total Route Retries

Figure 3.6: Simulation of Service Driven Cross Layer Model Using AODV Routing Protocol

Throughput:

The throughput comparison of IEEE 802.11 DCF MAC and S. SRL is shown in Figure 3.6 (a) and it is clearly observed that the throughput of S.SRL dominates over IEEE 802.11 DCF MAC right from the beginning of the simulation. This is mainly due to increased value of SRL (30) in case S.SRL and thus there is higher probability of channel occupancy by S.SRL for most of the simulation time. Figure 3.6(b) shows the behavior of D. SRL and it is clearly observed that the throughput of both IEEE 802.11 DCF MAC and D. SRL is almost similar up to 200 seconds. After 200 seconds, the priority is dynamically given to D.SRL; thus D.SRL yields higher throughput when compared to IEEE 802.11 DCF MAC after 200 seconds of simulation time.

Control Packet Overhead:

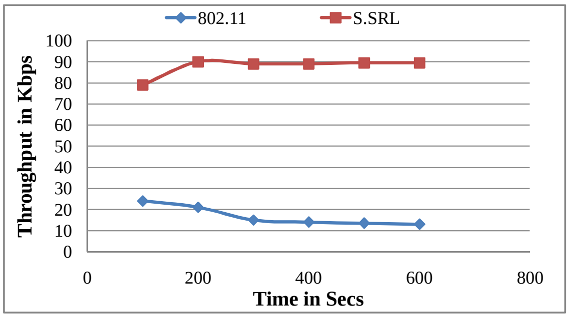
Figure 3.6(c) shows the control packets overhead which increases the network load. As S. SRL makes less route re-discovery attempts, its control packet overhead should be less compared to other two methods and it is also one of the reasons for high overall throughput of S.SRL. On the other hand, D.SRL produces equal amount of control packets till 200 seconds of simulation time and once the priority is given at 200 seconds, it also produces less control packet overhead when compared to IEEE 802.11 DCF MAC. Where as IEEE 802.11 DCF MAC contributes more control packets to the network thereby increasing the network load while decreasing the throughput; this is mainly due to more route retries of IEEE 802.11 DCF MAC.

Total Route Re-Discovery Attempts:

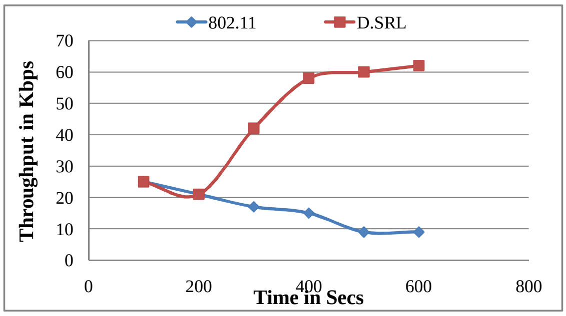
Figure 3.6(d) shows the route re-discovery attempts made by all three competing methods namely IEEE 802.11, S.SRL and D.SRL. It is clearly observed that S. SRL makes very less route re-discovery attempts compared to other two methods IEEE 802.11 DCF MAC and D.SRL. This is mainly due to the fact that, chances of FRF is very less in S.SRL and thus gives more overall throughput; whereas IEEE 802.11 DCF MAC makes more route re-discovery attempts due to high chances of FRF. On the other hand D. SRL makes almost equal attempts similar to IEEE 802.11 MAC up to 200 seconds and D.SRL observes less FRF and thereby reducing the route-rediscovery attempts after 200 seconds.

- **Simulation Using DSR Protocol**

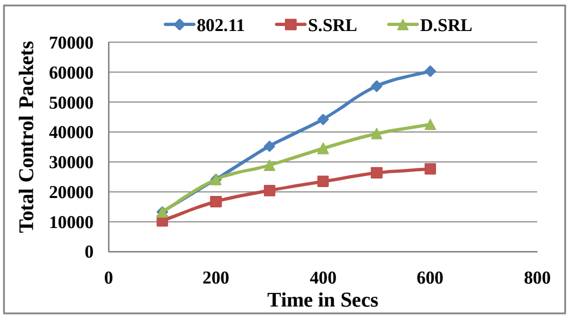
  Further simulation has been carried out to evaluate the performance of service driven cross layer design model using DSR routing protocol and the results are shown in Figure 3.7. The experimental results show that in case of DSR also proposed service driven cross layer model exhibits superior performance both in case of S.SRL and D.SRL. The experimental results depict that service driven cross layer model is adaptable to both AODV and DSR routing protocols.



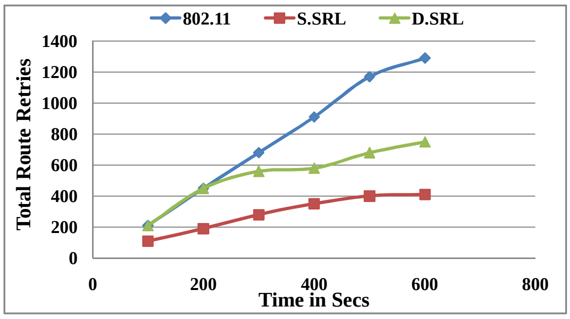
(a) Throughput-S.SRL



(b) Throughput-D.SRL



(c) Total Control Packets



(d) Total Route Retries

Figure 3.7: Simulation of Service Driven Cross Layer Model Using DSR Routing Protocol

3.6.2 Simulation Results of Prioritized QoS Cross Layer Model

The experiment is conducted with a single flow *F1* and in case of PrQoS the priority is assigned to F1 at the beginning of the simulation and the same priority is kept till the end of the simulation; further for Traditional Method (TM) *F1* is treated as un prioritized flow. PrQoS is evaluated under different node velocity namely 5, 10, 15, 20 and 25 m/s and the results are given below.

Throughput Analysis:

Figure 3.8 shows the throughput comparison of PrQoS and TM and the graph clearly depicts that PrQoS dominates over TM throughout the simulation. The main reason for this behavior of PrQoS is due to the increased RTS retry limit; further the *context awareness* of PrQoS also increases its throughput. The mobility information of each neighbor node will be reflected in PrQoS and hence PrQoS can predict the chances of neighbor node stay in the transmission range. The graph also depicts that PrQoS shows convincing results under high node velocity conditions. On the other hand, as TM is context unaware and also its priority is less its per-

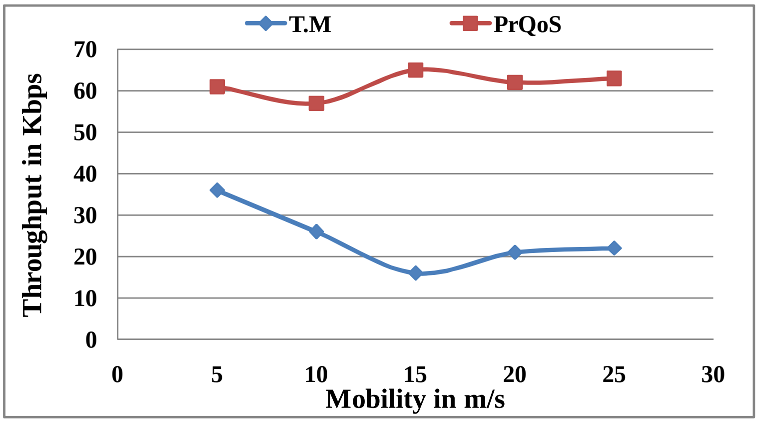


Figure 3.8: Throughput comparison of PrQoS and Traditional Method

formance deteriorates when compared to PrQoS. Further, TM finds it difficult to handle the mobility as it is *context unaware* and thus gives low throughput under high node velocity conditions.

Control Packet Overhead Analysis:

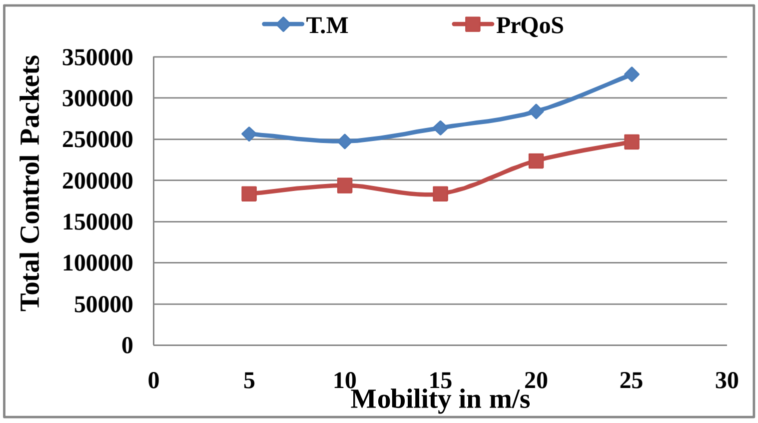


Figure 3.9: Control Packet Overhead Comparison of PrQoS and Traditional Method

Figure 3.9 shows control packet overhead comparison of both the methods and it clearly depicts that the curve of TM raises rapidly; the nodes in TM will misinterpret the packet drop under high mobility conditions as link breakage and thus triggers route rediscovery phase. Whereas the PrQoS curve rises slowly since it is aware of the reason for the packet drop and thus reduces the FRF.

Total Route Re-Discovery Attempts:

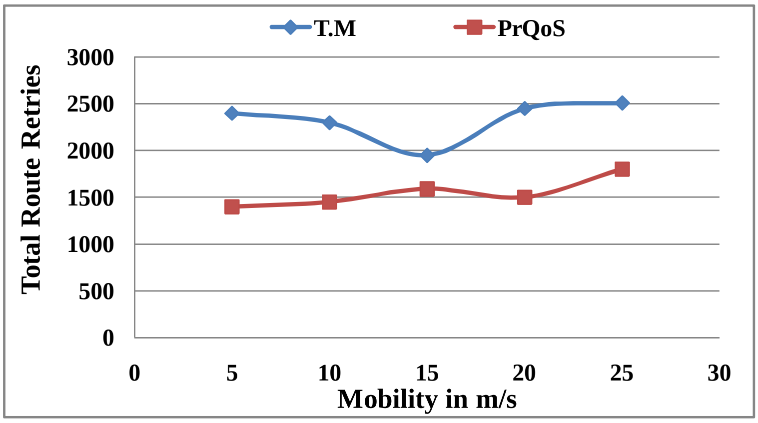


Figure 3.10: Route Re-Discovery Attempts Comparison of PrQoS and Traditional Method

Figure 3.10 compares the total route rediscovery attempts made by the PrQoS and the TM and it is clearly observed that the PrQoS makes less route re-discovery attempts as chances of FRF are very less; whereas traditional method makes more attempts due to FRF.

3.7 Summary

In this chapter we proposed a two intelligent cross layer models namely *service driven cross layer model* and *Prioritized QoS (PrQoS)* for providing QoS to different flows in the network. The cross layer model is designed across Application, Network, Physical and MAC layers in order to share the *priority of the application, distance between the nodes* and *RTS retry limit* information across non adjacent layers. The simulation results clearly depicts that the proposed cross layer models increases the throughput of prioritized flow when compared to unprioritized flow. Further, the proposed cross layer models also reduces the FRF there by reducing the total route rediscovery attempts; hence reduces the total control packets in the network.

On the other hand, increased retry limit causes the packets to spend more time in the MAC layer queue which increases the end to end delay. Thus the proposed cross layer models are suitable to the less delay sensitive or non real time applications and require more throughput (ex. file transfer over FTP). However, the context awareness feature of the proposed PrQoS is found to be convincing and encouraging when compared to traditional context unaware protocols. Thus motivated us in design and development of context aware bio-inspired routing protocol for MANETs by using the feature of SI techniques and the details are given in the next chapter.

Chapter 4

Mobility Aware Termite: A Novel Bio-Inspired Routing in MANETs

The frequent link failures due to mobility of nodes make source nodes to spend most of its time in route setup and maintenance than sending the messages, thus MANETs suffer from *low throughput* and *control packet overhead*. Hence, MANETs require the context or mobility aware environment for taking decisions adaptively under dynamic network conditions and thereby providing the QoS to applications. This chapter deals with proposed novel bio inspired routing protocol for MANETs referred to as Mobility Aware Termite (MA-Termite) which is based on the behaviour of *social insect Termites*, which shares the similar features of ants. The primary objective of MA-Termite is to emphasize the *stable nodes* by positively reinforcing them with less pheromone decay and also to deemphasize the unstable nodes by negatively reinforcing them with high pheromone decay. The secondary objective of MA-Termite is to propose the *distance variant analytical model* for pheromone update method of *MA-Termite*. The details of the MA-Termite are explained in the following sections.

4.1 Motivation for Development of MA-Termite Algorithm

Since MANETs suffer from low throughput and control packet overhead due to frequent link breakups, finding the stable nodes for routing is a challenging and critical task (Kim Ki & Kim S H 2006; Q.Han et al. 2011; WU Da-Peng et al. 2009). The mobile nodes found in existing MANETs are capable of monitoring the network status as well as data processing. Thus, context aware routing in MANETs can be achieved by local monitoring capability of nodes. In recent years, a lot of research has been carried out in building SI based adaptive bio-inspire routing protocols for MANETs. As centralized approach for routing in MANETs lack in scalability and fault-tolerance, SI techniques give natural solutions through distributed approach to the adaptive routing for MANETs. In this section, a novel bio inspired routing protocol based on the behaviour of social insect *Termites*, which shares the similar features of ants, referred to as Mobility Aware Termite (MA-Termite) is proposed for MANETs.

In MA-Termite, a new distance based pheromone update and decay functions which reflect the current context of the network in terms of node mobility are defined. The pheromone update and decay over the link will be directly proportional to the distance between nodes on successive packet arrivals; thus causing high pheromone decay for the high mobility nodes and low pheromone decay for the low mobility nodes. During data transfer, the nodes with low mobility will be given the preference over high mobility nodes. The working principle of MA-Termite is discussed in the following section.

4.2 Working Principle of MA-Termite

In MA-Termite, the coordinated behaviour exhibited by the social insect Termites during the hill building phase, is used to guide each packets in reaching the destination. Further, each node is treated as a termite hill which contains the pheromone table for storing the pheromone gradients of all reachable destinations. Packets leaving the source node are forwarded towards the node with highest destination pheromone gradient and the selection of next hop will be decided probabilistically at every node. At a given node, the packet arrival updates the source pheromone in its pheromone table proportional to the utility of the path it has traversed and MA-Termite removes the old routing solutions by mimicking the pheromone decay concept of social insect Termites. Hence, MA-Termite is mainly driven by two important functions namely, pheromone update-decay and forwarding functions.

When a node has the data packet to be sent and if the destination pheromone entry is not present in its pheromone table, then the corresponding node triggers a route setup phase in which a new pheromone trail to the destination can be found. Once the new route is established then MA-Termite will start the data session phase between the source and destination nodes. The proposed MA-Termite exhibits salient features such as high robustness through multiple paths to the destination, high route reliability with context awareness and less control packet with stable nodes in the path. Apart from these salient features, each node can also have a general idea about the position of neighboring nodes and the corresponding movements in order to find the stable nodes in reaching the destination.

The pheromone update-decay functions will play an important role in any ant or termite based routing protocols. A slow decay rate may cause stale entries in the pheromone table and a high decay rate may quickly remove the entry from the pheromone table thus closing the path in reaching the destination. In MA-Termite, the inter packet distance between the nodes is considered for pheromone update and decay (Kiran et al. 2012; Kiran and Reddy 2013) in order to reflect the mobility of each nodes as shown in Figure 4.1.

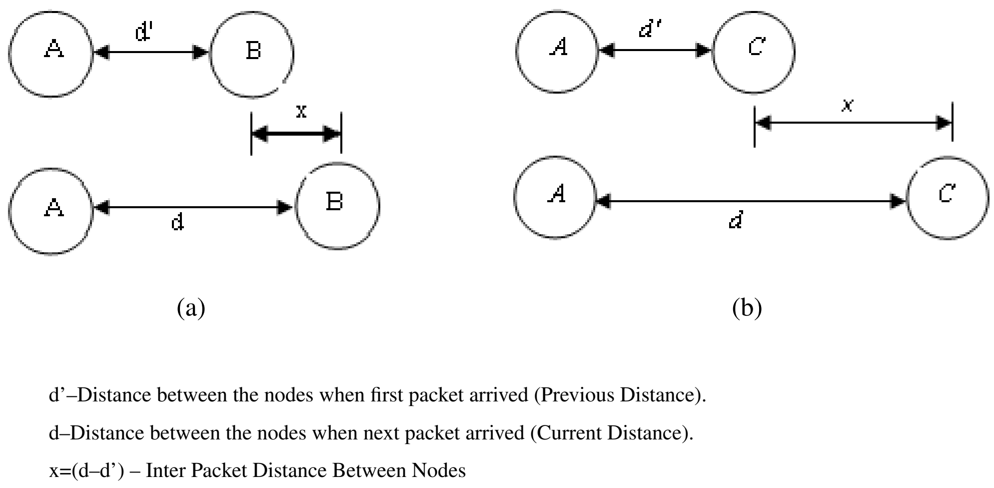


Figure 4.1: Principle of Computing Inter Packet Distance Between the Nodes

If the inter packet distance between the nodes is small (Figure 4.1a) then the decay will also be small. On the other hand, the decay will be large when the inter packet distance between the nodes is large (Figure 4.1b) thus reflecting the current context of the network. Accordingly new *pheromone update-decay function* is designed in such a manner that it meets the requirements of MA-Termite and thereby finding the distance between the nodes using the cross layer design model as explained below.

4.2.1 Cross Layer Design Model

Figure 4.2 shows the Cross Layer Design Model across the Physical and Network layers which can be used for finding the distance between the nodes.

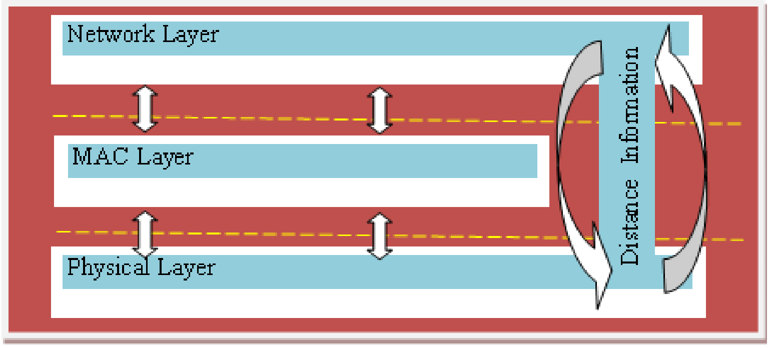


Figure 4.2: Cross Layer Design Model for Finding the Distance Between the Mobile Nodes

From the Physical layer, variable *Received Signal Strength Pr* of each packet is made visible at the Network layer since *Pr* is inversely proportional to the *distance between nodes* (Kiran M et al. 2012; A.G Zakeerhusen 2012). Whenever any mobile node receives the packet from the neighbor node, *Pr* will be measured and then the distance *d* between nodes can be computed using the free space propagation model as shown in Eqn. 3.1 The distance *d* and the time interval at which *d* is calculated will be stored in a table of the Network layer. The time interval gives the freshness of table entries which are refreshed at regular intervals in order to remove the stale entries and the table details are discussed in Subsection 4.2.4 The computed distance *d* will be used for updating-decaying the trail pheromone in mobile nodes and the details of pheromone update-decay functions are given in the following subsection.

4.2.2 Trail Pheromone Update-Decay Function

The continuous pheromone update-decay function of basic Termite algorithm (Roth M and Wicker S 2004a;2004b) has been modified in order to suit the requirements of the proposed MA-Termite algorithm and is as follows.

$$\boldsymbol{P_R^n} = \begin{cases} \boldsymbol{P_{k,s}^n \cdot e^{-(d_c - d_p)\tau} + \alpha} & \boldsymbol{k = l} \\ \boldsymbol{P_{k,s}^n \cdot e^{-(d_c - d_p)\tau}} & \boldsymbol{k \neq l} \end{cases} \tag{4.1}$$

where P_R^n is the amount of pheromone from source node *s* deposited on neighbor link *k* at node *n*; *l* represents the previous hop of the packet; α is the amount of pheromone carried by the packet; d_c represents the current distance of the neighbor node *k* and d_p is the distance of the same neighbor node *k* when the last packet arrived; τ is the decay rate. When a packet arrives at a mobile node, the pheromone on its entire neighbor node links will be decayed simultaneously based on the distance travelled by the neighbor nodes. Then, on only one link, on which the data packet has arrived, will be positively reinforced with the pheromone update equal to the amount of pheromone carried by the data packet. The computed trail pheromone of each neighbor node will be used for finding the probability of neighbour node and thereby forwarding the data packets towards the destination node and the details of forwarding function are given in the following subsection.

4.2.3 Forwarding Function of MA-Termite Algorithm

The forwarding function (Roth M and Wicker S 2004a;2004b) handles each data packet independently and it is given as

$$p^n_{i,d} = \frac{(P^n_{i,d} + K)^F}{\sum_{j=1}^{N}(P^n_{j,d} + K)^F} \tag{4.2}$$

where $p^n_{i,d}$ is the probability of using neighbor node *i* to reach destination *d* at node *n* and *N* is the total number of neighbor nodes of *n*; constants *K* and *F* indicates the *pheromone threshold* and *pheromone sensitivity* respectively used for tuning the routing behavior of the MA-Termite.

4.2.4 Node Structure

Figure 4.3 illustrates the node structure of MA-Termite algorithm. Each node consists of two tables; (1) the forwarding function uses the *Trail Pheromone Table* for deciding the next node along the path (2) pheromone update function uses the *Local Status Table* in order to update the pheromone on the links and the node structure is shown in Figure 4.3.

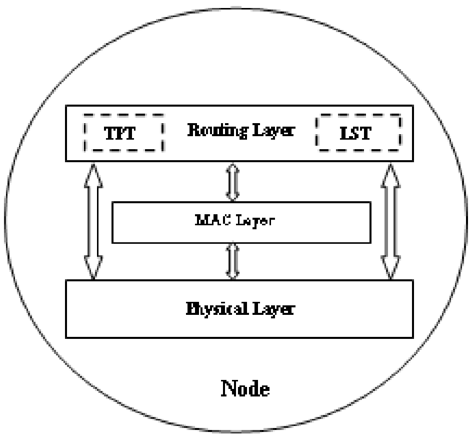


Figure 4.3: Node Structure of MA-Termite

4.2.5 Pheromone Bounds

MA-Termite uses three bounds for the pheromones namely, *PH_BEGIN, PH_MIN* and *PH_MAX* in order to limit the pheromone value in the pheromone table. Whenever a node receives a packet from an unknown source then a new entry will be created for that node in the *Trail Pheromone Table* with an initial value of *PH_BEGIN*. Further, pheromones in the pheromone table will neither be allowed to fall below the minimum value *PH_MIN* nor allowed to exceed the maximum value *PH_MAX* and these pheromone bounds will be chosen according to the context of the network. The details of *Trail Pheromone Table* are given below.

4.2.6 Trail Pheromone Table

Trail Pheromone Table (TPT) records the amount of destination pheromone on each neighbor link; *TPT* helps in establishing the best path from the source to the destination and more or less holds the same elements as routing table in terms of *neighbor ID, destination ID* and *routing metric (trial pheromone)*. In *TPT*, rows represent neighbor nodes whereas columns are corresponding to the destination nodes. When a node detects a new neighbor node then it adds an additional row to the *TPT*; further when a new destination is discovered then a new column will be added to the *TPT* and in both cases the trail pheromone will be initialized to *PH_MIN*. Further, *TPT* will be updated based on the *Local Status Table (LST)* of the node and the details of *LST* are given below.

4.2.7 Local Status Table

LST gives the current context of the network in terms of the neighbor nodes distance information which will be maintained by the Network Layer along with the *TPT*. Further, *LST* consists of neighbor node's *current and previous distances between the packet arrivals* and *the time at which the distance is calculated*. This time entry will give the freshness of *LST* information and *LST* will be refreshed at regular intervals of *LST_TIME_OUT*. Whenever a node receives any control *(RREQ, RREP and RERR)* or data packet then the node will calculate the distance of the transmitting node using Eqn. 3.1 and the same will be stored in *LST*. Thus every node will be aware of its neighboring nodes movement and also aware of chances of neighbor nodes staying within the transmission range. Based on the current and previous distances of the neighbor nodes information from *LST*, the trail pheromone will be updated in *TPT* in order to reflect the neighbor nodes movements. Thus, the stable node has higher pheromone entry in *TPT* whereas the unstable neighbor node has lower pheromone entry in *TPT*. Further, both tables *LST* and *TPT* are updated and maintained during the route discovery and maintenance phases of MA-Termite and the details are given in the following subsection.

4.2.8 Route Setup and Maintenance

When a node does not find the destination pheromone entry in its *TPT* in forwarding the data packet towards the destination, then it sends a *Route Request (RREQ)* packet towards its neighbour nodes and thus finds a new pheromone trail for the intended destination by taking a random walk across the network. During the route setup phase, the *RREQ* packets will not be attracted by the highest pheromone link and the *RREQ* packets will be forwarded to a random next hop except the link on which it has arrived; further if a node cannot forward the *RREQ* packet then it will automatically drop the *RREQ* packet. Whenever a node receives the *RREQ* packet then it makes the necessary entries in its pheromone table in order to setup the reverse path and it replies to *RREQ* source with *Route Reply (RREP)* packet only under two different scenarios *(i) if the node is the intended destination (ii) if the node has the destination trail pheromone;* hence *RREP* packets will set up the forward path in each visited node. When the source receives the *RREP* packet, it will forward the stored data packets according to the forwarding function and these paths will be further strengthened by the data packets traversing in that path. If a node is left with empty neighbour list to forward the data packets then it will store the data packets and sends the *RREQ* packet to find new pheromone trail to the intended destination.

This concludes the description of entire MA-Termite and the Algorithm 4.1 illustrates the route setup phase of *MA-Termite* and further Algorithm 4.2 describes the data session phase of *MA-Termite*.

4.3 Asymptotic Behaviour of MA-Termite

In this section, an analytical model is designed for finding the average expected amount of pheromone on both single and double links based on the asymptotic behaviour of MA-Termite. For this study, the MANET model is designed with the source and destination nodes *A* and *B* respectively with two independent paths *(Path 0 and Path 1)* as shown in Figure 4.4 (Subramanian et al. 1997; Roth M and Wicker S 2004a; 2004b). It is clearly observed from Figure 4.4 that

Algorithm 4.1: Route Setup Phase of MA-Termite

Input: TPT and LST

Output: Updated TPT and LST and an optimal path between the source and the destination node

1. ***while*** node is intended destination or it consists of pheromone trail to the intended destination
2. At each intermediate node
3. Update the TPT and LST
4. ***if*** no neighbour exists
5. Discard the RREQ
6. ***else***
7. Randomly choose the next hop neighbour and forward the RREQ packet.
8. ***end if***
9. ***end while***
10. Unicast the RREP packet to the source
11. Discard the RREQ

each path *k* has a utility which will be characterized by a non negative random process $\Gamma_x(d)$ with mean $\mu_x(d)$. Each packet carries pheromone α which is a finite independent sample of above said random process; further at each node pheromone is decayed at rate τ (constant) and stationary Γ is considered in this study. The nodes *A* and *B* are located at two different positions with distance *d* and the mobile node separation distance distribution is explained below.

4.3.1 Node Separation Distance Distribution

Consider the nodes are spread out uniformly over the network area then the nodes separation distance is approximately defined by Rayleigh distribution. Further, if the transmission range of a node is much smaller than the network area then the node separation distance can be approximately distributed in the range of *0* to *r* and the density function is given as (Sanlin Xu et al. 2007; Ross, S S 2002),

$$\boldsymbol{f}(d) = \begin{cases} \boldsymbol{f}(d) = \frac{\boldsymbol{2d}}{\boldsymbol{r^2}} & \boldsymbol{0 \le d \le r} \\ \boldsymbol{0} & \boldsymbol{d > r} \end{cases} \tag{4.3}$$

The node separation distance distribution *f(d)* can be used to calculate the expected pheromone decay *E(*$e^{(-d\tau)}$*)* between packet arrivals corresponding to node separation distance *d* as ex-

Algorithm 4.2 Data Session Phase of MA-Termite

Input: TPT and LST

Output: Updated TPS and LST and reinforcement of optimal path between the source and the destination.

1. ***while*** node = destination node
2. At each intermediate node
3. ***for*** all the neighbours
4. Decay the pheromone
5. ***end for***
6. for the neighbour from which the data packet received
7. Update the pheromone on the link with α
8. Calculate the next hop probability for all the neighbours which lead to the destination
9. Update α value in the packet based on the path utility.
10. Choose the highest probability neighbour and forward the data packet
11. ***if*** no neighbour exists which lead to the destination
12. Store the data packet
13. Trigger the Route Setup phase
14. ***end if***
15. ***end while***

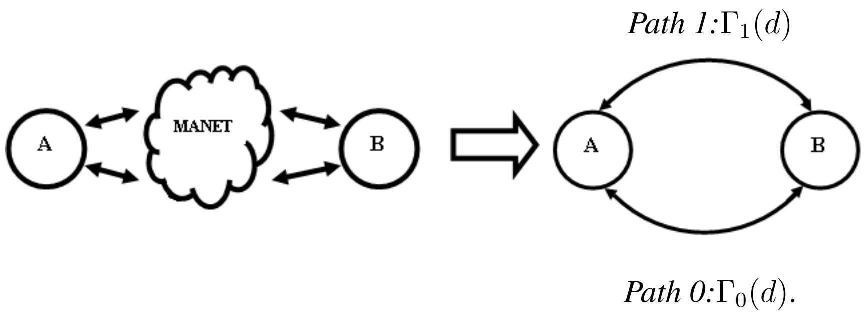


Figure 4.4: MANET Model for Studying Asymptotic Behavior of MA-Termite

plained in next subsection.

4.3.2 Expected Pheromone Decay

The random variable y is defined such that $y = e^{(-d\tau)}$ which describes the fraction of pheromone decay between packet arrivals corresponding to node separation distance *d*. Then the expected pheromone decay can be calculated using *f(d)* for $0 \le d \le r$ and the details are as follows:

Since $f(d) = \frac{2d}{r^2}$ for $0 \le d \le r$ and $y = e^{(-d\tau)}$ then,

$$\begin{aligned}
d &= \frac{-logy}{\tau} \\
\frac{dd}{dy} &= \frac{-1}{y\tau} \\
f(y) &= f(d)|\frac{dd}{dy}| \\
&= \frac{2}{r^2}\left(\frac{-logy}{\tau}\right)|\frac{-1}{y\tau}| \\
f(y) &= \frac{2}{r^2\tau^2}\frac{1}{y}(-logy) \qquad e^{-r\tau} \le y \le 1
\end{aligned} \tag{4.4}$$

Hence, Expected Pheromone Decay will be computed as:

$$\begin{aligned}
E(e^{-d\tau}) = E(y) &= \int_{e^{-r\tau}}^{1} \frac{2}{r^2\tau^2}\frac{1}{y}y\,(-log\,y)\,dy \\
&= \frac{2}{r^2\tau^2}\int_{e^{-r\tau}}^{1} -log\,y\,dy \\
&= \frac{2}{r^2\tau^2}\big[-(ylogy - y)\big]_{e^{-r\tau}}^{1} \\
&= \frac{2}{r^2\tau^2}\big[-1log1 + 1 + e^{-r\tau}loge^{-r\tau} - e^{-r\tau}\big] \\
&= \frac{2}{r^2\tau^2}\big[e^{-r\tau}\big(-r\tau - 1\big) + 1\big] \\
E(e^{-d\tau}) &= \frac{2}{r^2\tau^2}\big[1 - e^{-r\tau}\big(1 + r\tau\big)\big]
\end{aligned} \tag{4.5}$$

This expected decay rate corresponding to the node separation distance *d* is then used for calculating the average expected amount of pheromone on both single and double links using the pheromone update function of the proposed MA-Termite and the justification for the pheromone update-decay function is given in Subsection 4.3.3.

4.3.3 Analytical Justification for Pheromone Update-Decay Function

Let p be the population at distance *m* and p_0 be the initial population then ,

$$
\begin{aligned}
\frac{dp}{dm} &= -\tau p \\
\frac{dp}{p} &= -\tau dm \\
logp &= -\tau m + c' \qquad (4.6)
\end{aligned}
$$

Hence pheromone update function is derived such that, $p = ce^{-\tau m} = ce^{-m\tau}$

The computed pheromone update-decay function of MA-Termite algorithm is then used for computing the average expected amount of pheromone over both single and double links as per the following procedure.

4.3.4 Average Expected Amount of Pheromone Over Single Link

Based on the assumptions *(i) number of packets received within a given distance d is Poisson distributed with mean λ packets/meter (ii) the average value of the received pheromone as μ (iii) P_0 is the initial amount of pheromone on the link*, an analytical formula is derived for finding the average expected amount of pheromone over single link. Then, pheromone update equation of MA-Termite is applied for *n* consecutive times indicating *n* arrived packets on a single link as given below:

Pheromone over a single link for *n* arrived packets is defined as

$$
\begin{aligned}
P(n) &= \left(\left(\left(P_0.e^{-d_1\tau} + \mu\right).e^{-d_2\tau} + \mu\right)...\right).e^{-d_n\tau} + \mu \\
&= P_0 e^{-(d_1+d_2+...+d_n)\tau} + ... + \mu e^{-(d_n)\tau} + \mu \\
&= P_0 e^{-\left(\sum_{i=1}^{n} d_i\right)\tau} + \mu \sum_{j=2}^{n} e^{-\left(\sum_{i=j}^{n} d_i\right)\tau} \qquad (4.7)
\end{aligned}
$$

Then the Expected amount of Pheromone EP(n) for *n* arrived packets can be computed as:

$$
\begin{aligned}
EP(n) &= P_0\big(E(e^{-d\tau})\big)^n + \mu\left(\frac{\big(E(e^{-d\tau})\big)^n}{\big(E(e^{-d\tau})\big)}\right) \\
EP(n) &= P_0\left[\frac{2}{r^2\tau^2}\big[1-e^{-r\tau}\big(1+r\tau\big)\big]\right]^n + \mu\left(\frac{1-\left[\frac{2}{r^2\tau^2}\big[1-e^{-r\tau}\big(1+r\tau\big)\big]\right]^n}{1-\left[\frac{2}{r^2\tau^2}\big[1-e^{-r\tau}\big(1+r\tau\big)\big]\right]}\right) \\
Let \beta &= \left[\frac{2}{r^2\tau^2}\big[1-e^{-r\tau}\big(1+r\tau\big)\big]\right] \\
then, EP(n) &= P_0\beta^n + \mu\Big(\frac{1-\beta^n}{1-\beta}\Big) \qquad (4.8)
\end{aligned}
$$

Hence, the Expected amount of pheromone on a link over node separation distance *d* for *n* arrived packets EP(d) is defined as function of both node separation distance *d* and number of received packets *n* with Poisson distribution rate of λ and the details are as follows:

$$
\begin{aligned}
EP(d) &= \sum_{n=0}^{\infty}\big[Poisson(\lambda d, n)\big]\big[EP(n)\big] \\
&= \sum_{n=0}^{\infty}\left[e^{-\lambda d}\frac{(\lambda d)^n}{n!}\right]\left[P_0\beta^n + \mu\Big(\frac{1-\beta^n}{1-\beta}\Big)\right] \\
&= \frac{\mu}{1-\beta} + e^{-\frac{\lambda\tau d}{\lambda+\tau}}\Big(P_0 - \frac{\mu}{1-\beta}\Big) \qquad (4.9)
\end{aligned}
$$

The mean pheromone behaviour on long term basis is defined as,

$$
\begin{aligned}
\lim_{x\to\infty} EP(d) &= \frac{\mu}{1-\beta} \\
&= \frac{\mu(\lambda+\tau)}{\tau} \\
&\triangleq EP \qquad (4.10)
\end{aligned}
$$

The *EP(d)* will give the average expected amount of pheromone over a single link, where packets do not have any other links to reach the destination. Thus the pheromone behaviour over double links will be studied where packets will have two options in reaching the destination and the details are given in the following subsection.

4.3.5 Average Expected Amount of Pheromone Over Double Links

For the study, a double link system is considered as shown in Figure 4.5 and a set of equations for both the nodes *A* and *B* are derived using pheromone update and forwarding functions of MA-Termite, namely $P^A_{0,B}$ (between *A* to *B* on link 0), $P^B_{0,A}$ (between *B* to *A* on link 0) , $P^A_{1,B}$ (between *A* to *B* on link 1) and $P^B_{1,A}$ (between *B* to *A* on link 1). These derived equations will recursively compute the mean pheromone at each node on each link of MANET as shown in Figure 4.5.

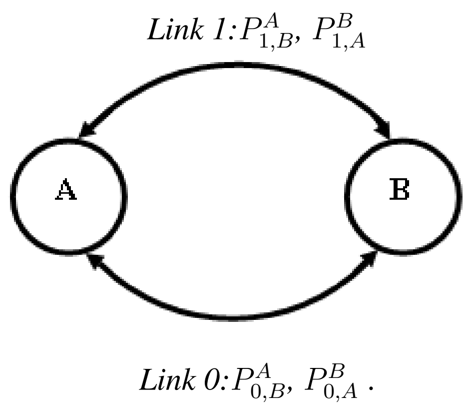


Figure 4.5: Double Link System for studying Asymptotic Behaviour of MA-Termite

The average expected amount of pheromone over a link for double link system is found by considering the following assumptions: Whenever a node receives a packet, then the pheromone will be decayed by an expected amount $(E(e^{(-d\tau)}))$ on all the neighbour links, however the pheromone will be incremented by both average amount (μ_i) and the opposite node link probability. Let P_0 and P_1 be the initial amount of pheromones on the link 0 and link 1 respectively; μ_0 and μ_1 be the average values of the received pheromones on link 0 and link 1 respectively. Hence, the average expected amount of pheromone over all links: $P^A_{1,B}$, $P^B_{1,A}$ $P^A_{0,B}$, $P^B_{0,A}$ will be computed as follows:

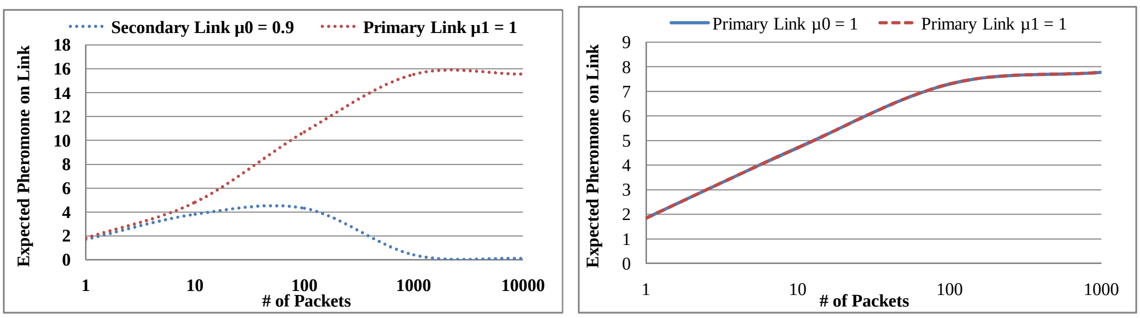
$$P^A_{0,B} = P^A_{0,B}\left[\frac{2}{r^2\tau^2}\left[1-e^{-r\tau}\left(1+r\tau\right)\right]\right] + \left[\frac{(P^B_{0,A}+K)^F}{(P^B_{0,A}+K)^F+(P^B_{1,A}+K)^F)}\right]\mu_0 \tag{4.11}$$

$$P^A_{1,B} = P^A_{1,B}\left[\frac{2}{r^2\tau^2}\left[1-e^{-r\tau}\left(1+r\tau\right)\right]\right] + \left[\frac{(P^B_{1,A}+K)^F}{(P^B_{0,A}+K)^F+(P^B_{1,A}+K)^F)}\right]\mu_1 \tag{4.12}$$

$$P^B_{0,A} = P^B_{0,A}\left[\frac{2}{r^2\tau^2}\left[1-e^{-r\tau}\left(1+r\tau\right)\right]\right] + \left[\frac{(P^A_{0,B}+K)^F}{(P^A_{0,B}+K)^F+(P^A_{1,B}+K)^F)}\right]\mu_0 \tag{4.13}$$

$$P^B_{1,A} = P^B_{1,A}\left[\frac{2}{r^2\tau^2}\left[1-e^{-r\tau}\left(1+r\tau\right)\right]\right] + \left[\frac{(P^A_{1,B}+K)^F}{(P^A_{0,B}+K)^F+(P^A_{1,B}+K)^F)}\right]\mu_1 \tag{4.14}$$

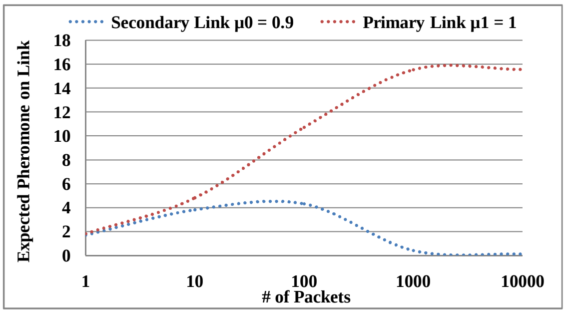
The above equations are applied for *n* consecutive times indicating *n* arrived packets; *Pheromone sensitivity (F)* and *Pheromone Threshold (K)* in the above equations are assigned with values *1* and *0* respectively. Thus, the expected pheromone behaviour over all four links with respect to *r (distance between the nodes)* is depicted in Figures 4.6 with different values of *F* and τ. Further, the behaviour of only $P^A_{0,B}$ and $P^A_{1,B}$ will be considered for the asymptotic analysis since the other two links $P^B_{0,A}$ and $P^B_{1,A}$ will exhibit the similar behaviour based on μ.



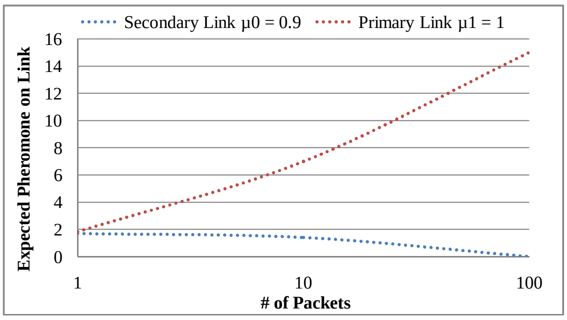
(a) (b)

Figure 4.6: Expected Link Pheromone Vs Distance β = 0.9357, τ=0.01, r = 10, F=1, K = 0

Figure 4.6(a) shows the asymptotic analytical model behaviour when one of the links of MANET model (Figure 4.5) will be considered as dominant or primary link (Link 1). Whenever a packet arrives at a node, then the pheromone on its entire neighbour links (both Link 0 and Link 1) will be decayed simultaneously by expected decay $E(e^{(-d\tau)})$ and then, only one link, on which the data packet has arrived (Link 1), will be positively reinforced with the pheromone update equal to the average value of received pheromone μ_i. Since the Primary Link (Link 1) is positively reinforced when compared to the Secondary Link (Link 0), then the Link 1 will produce a strong positive feedback with more pheromone and thereby attracting more number of packets. Hence, Link 0 fails to attract the packets and thus it produces a strong negative feedback resulting in exponential decay of pheromones as shown in Figure 4.6(a). Further, it is also clearly observed from Figure 4.6(a) that the MA-Termite always uses the highest pheromone link exclusively for the data transfer. Figure 4.6(b) shows the model behaviour when both the links, Link 1 and Link 0 have equal preference ($\mu_0 = \mu_1$).



(a) F=1

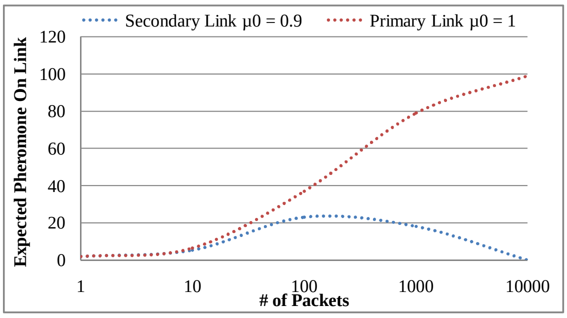


(b) F=5

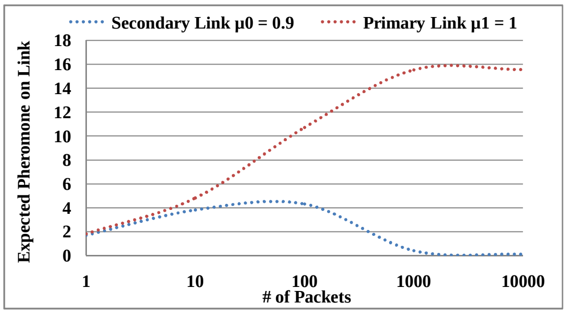
Figure 4.7: Expected Link Pheromone for different F Values, β = 0.9357, r = 10, K=0

Figure 4.7 shows the behavior of the model for different values of sensitivity parameter *F*. Sensitivity parameter controls how best the good links are effectively used and also ensures that the probability estimation is dependent on *F*. If *F* is high then the model can ensure that the good links will be preferred without any ambiguity. It is clearly observed from Figure 4.7(a) that the Primary Link will start dominating the Secondary Link after receiving ten packets when *F*=1, where as in Figure 4.7(b) the Primary Link dominates the Secondary Link right from the reception of the first packet when *F* = 5.

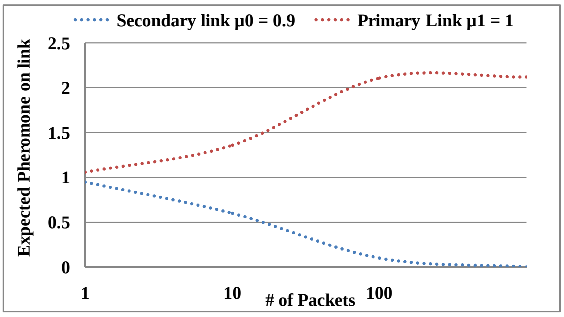
It is well known that pheromone decay rate τ plays a vital role in the behavior of ACO and Termite based routing algorithms in general and MA-Termite algorithm in particular. Hence, the optimum value of this parameter τ should be carefully chosen in such a manner that the decay rate τ should not be kept too low in order to avoid wrong routing decision because of stale entries in the pheromone table. On the other hand, the decay rate τ should not be kept too high in order to avoid the quick saturation of the pheromone. Hence, in order to choose the optimum value for the τ, the model behavior for different values of τ (0.001, 0.01, 0.1 and 1) is studied and the results are shown in Figure 4.8. Model takes longer time to converge (on reception of approximately 10000 and 1000 packets) when τ = 0.001 (Figure 4.8(a)) and τ = 0.01 (Figure 4.8(b)). On the other hand, model will converge to final value soon (on reception of



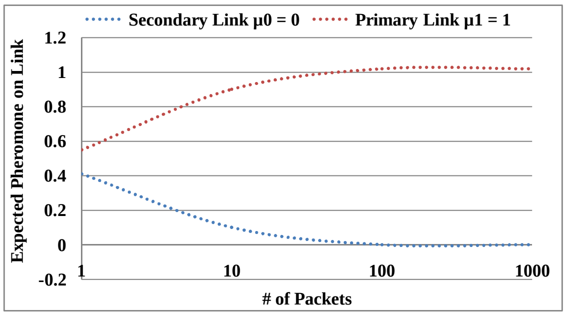
(a) τ=0.001



(b) τ=0.01



(c) τ=0.1



(d) τ=1

Figure 4.8: Expected Link Pheromone for different decay rate F=1, K = 0, r = 10

approximately 80 packets) when τ = 1 (Figure 4.8(d)). However, model converges to final value moderately (on reception of approximately 120 packets) (Figure 4.8(c)) when τ = 0.1. Thus, based on these theoretical results, the optimum value of τ is now chosen as 0.1 for evaluating the performance of MA-Termite algorithm.

4.4 Performance Evaluation of MA-Termite

The proposed MA-Termite algorithm (here after called as MA-T for plotting the graphs) is simulated in NS-2 and the results are compared with both bio-inspired (SARA (Correia, F and Vazao, T 2010) and basic Termite Algorithm (BT) (Roth M and Wicker S 2004a; 2004b)) and non bio-inspired (Ad hoc On Demand Distance Vector (AODV) (C.M, Perkins and E.M, Royer 1999)) routing protocols for its performance evaluation. The protocols are compared with respect to QoS metrics such as *throughput, total packet drops, end-to-end delay* and *control packet overhead* and the simulation environment is explained below.

Simulation Environment:

Each simulation run was carried out for 100 sec and Random Waypoint Mobility model is considered for the simulation. At the MAC layer, IEEE 802.11 DCF is used and File Transfer Protocol (FTP) is used as a traffic type with the packet payload of 1500 bytes. Ten traffic source and destination nodes are considered for all the simulation runs and each Transmission Control Protocol (TCP) source destination pairs are selected randomly for each simulation run. Table 4.1 shows the different parameter values used for evaluating the performance of MA-Termite algorithm. Further, the default values of SARA protocol are considered for performance evaluation throughout the experiment.

Table 4.1: Parameter Values used for MA-Termite

Sl.No	Parameter	Value
1	*PH_INITIAL*	1
2	*PH_CEILING*	50
3	*PH_FLOOR*	0
4	τ	0.105
5	F	1

4.4.1 Experimental Setup

A rigorous simulation study was carried out under two different scenarios to see how the proposed algorithms behaves under *scalability* and the *node velocity* factor.

- **Scenario A**

  In this scenario *scalability* of the proposed algorithms is evaluated by considering two test areas analogous to academic campus of size 500m x 500m and 1000m x 1000m. Under each simulation area, the algorithms are tested for different node density of 100, 200, 400 and 600 nodes. The node velocity is kept constant at 2m/s when a normal walking scenario of a person is considered and no pause time is given to any mobile node.

- **Scenario B**

  In this scenario behavior of the proposed algorithms under different set of *node velocity* is studied for performance evaluation. Mobile nodes with varying node velocity (2, 4, 6, 8 and 10 m/s) and with different node density (100, 200, 400 and 600 nodes) are considered for each simulation run. Test area analogous to academic campus of size 1000m x1000m is considered and no pause time is given to any mobile node.

4.4.2 Scenario A

Throughput Analysis:

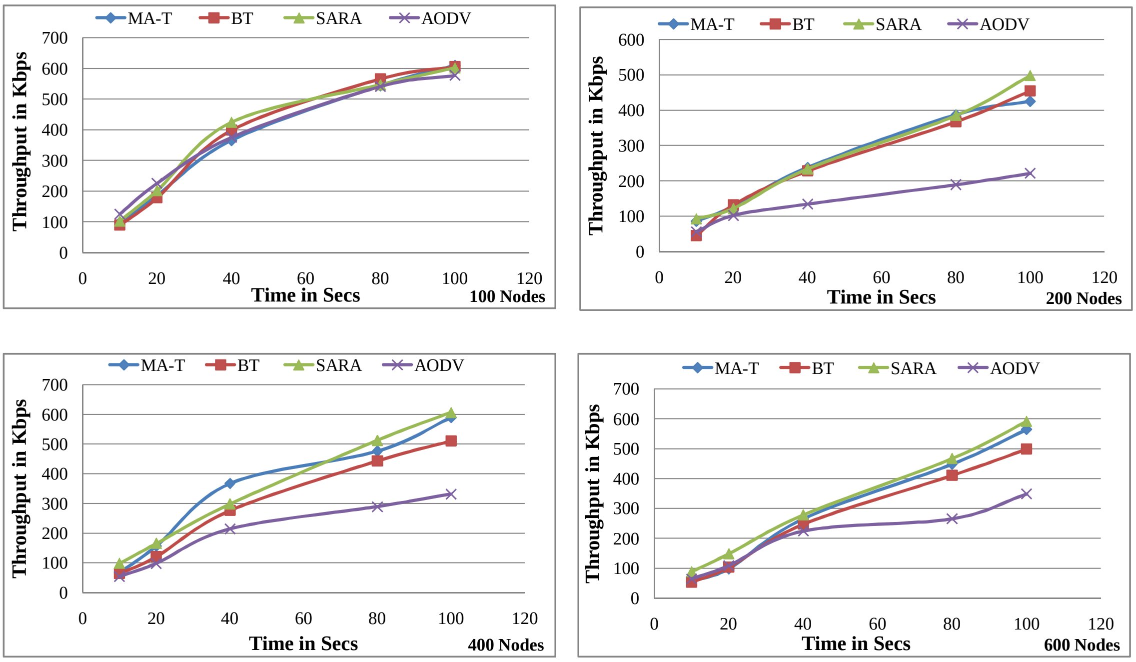


Figure 4.9: Throughput Vs Simulation Time in Sec (500m x 500m Sim. Area)

Figures 4.9 and 4.10 shows the throughput comparison of all competing protocols and it is clearly observed that as node density increases from 100 nodes to 600 nodes, bio-inspired protocols (MA-T, SARA and BT) exhibit superior performance when compared to non bio-inspired protocol (AODV). This is due to the multipath routing feature of bio-inspired protocols and thereby providing better connectivity. But AODV suffers from the poor path management, bad route recovery procedure due to its single path feature. On the other hand BT provides good throughput when compared to AODV due to its quick route discovery and robustness features; but its performance is inferior when compared to MA-T and SARA. MA-T offers higher throughput when compared to AODV and BT due to its reliable paths in reaching the destination node, context awareness of the network and steady state data transmission rate. It is also observed that SARA and MA-T algorithms exhibit good scalability when compared to BT and AODV. On the other hand SARA exhibits superior performance when compared to AODV, BT and MA-T. Thus, SARA exhibits good scalability in terms of better throughput even when the node density is increased to 600 nodes and thereby reducing the complexity of the protocol due to its simple routing phases.

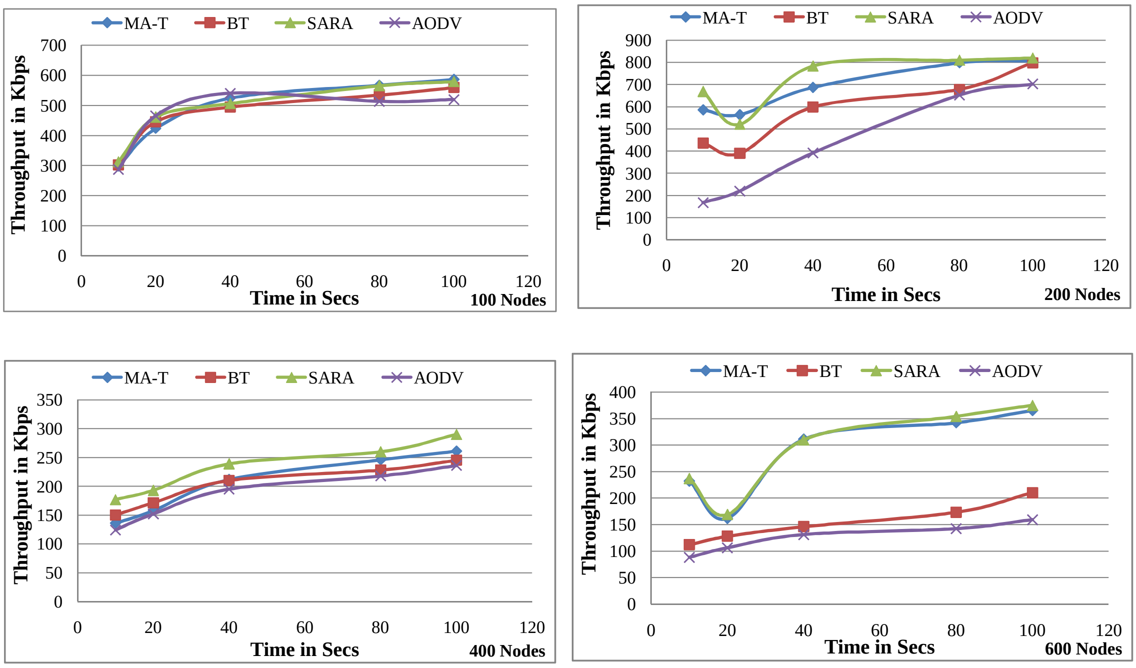


Figure 4.10: Throughput Vs Simulation Time in Sec (1000m x 1000m Sim. Area)

Packet Drop Analysis

Figures 4.11 and 4.12 show the total packet drop of all competing protocols and it is clearly observed that the total packet drop is directly proportional to the node density. SARA dominates all the other competing protocols (MA-T, BT and AODV) due to its optimized routing process. MA-Termite also reduces the total packet drops with its reliable path between the source and destination nodes when compared to BT and AODV. Context awareness causes MA-Termite to choose stable nodes for the path in turn reducing the total packet drops. On the other hand, BT and AODV produces more packet drops when compared to SARA and MA-T due to their context unawareness.

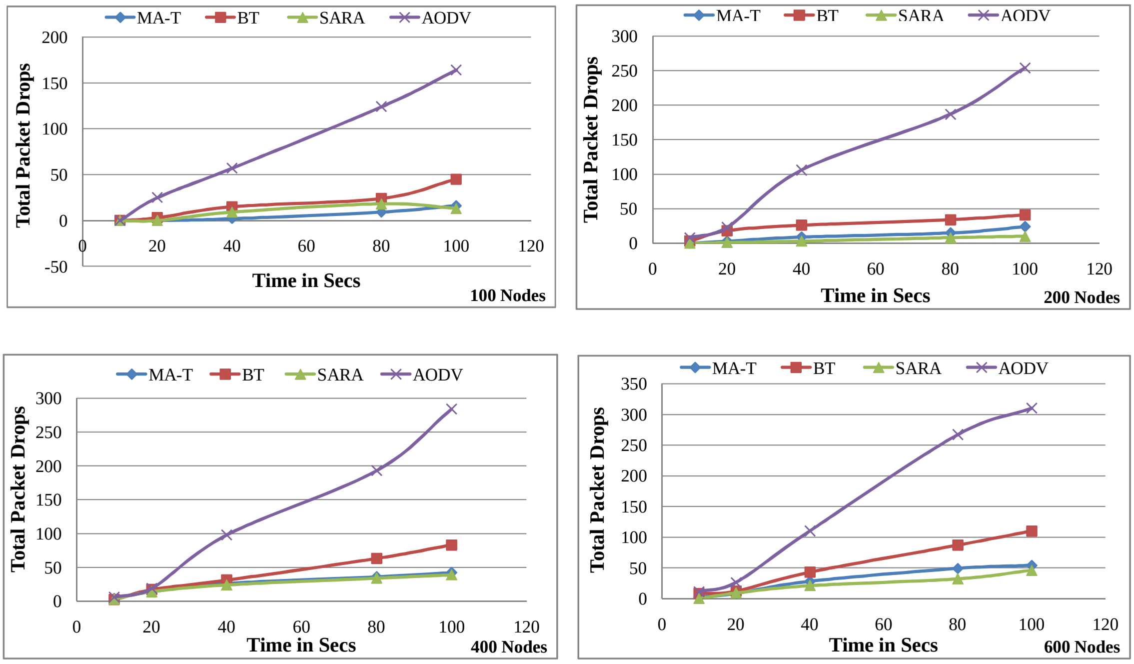


Figure 4.11: Packet Drop Vs Simulation Time in Sec (500m x 500m Sim. Area)

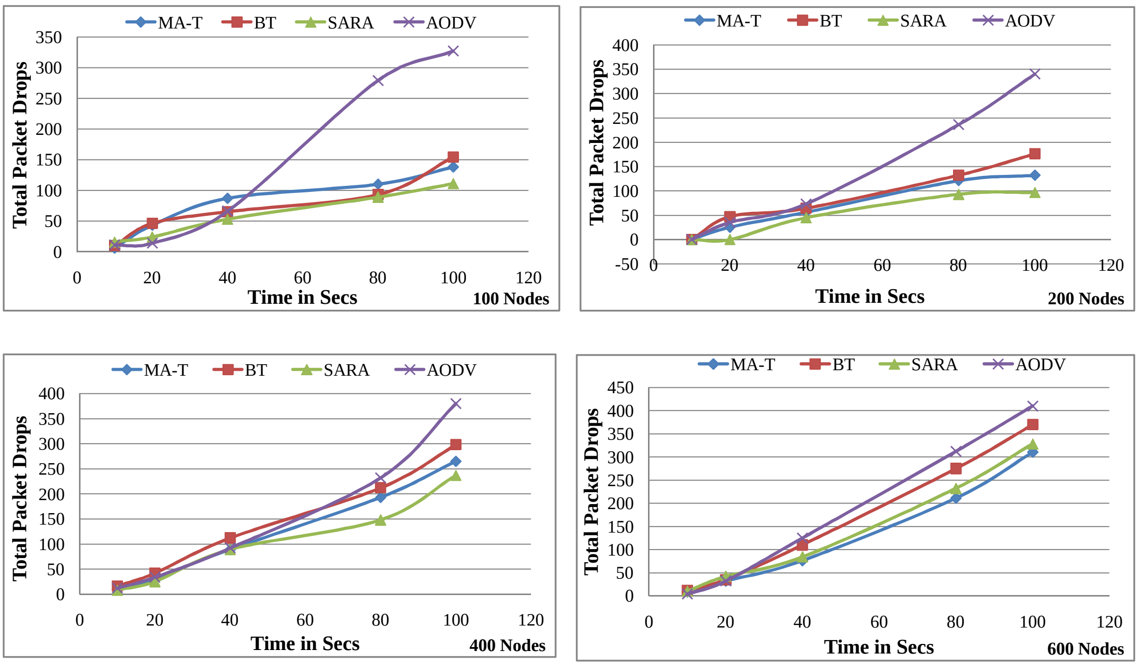


Figure 4.12: Packet Drop Vs Simulation Time in Sec (1000m x 1000m Sim. Area)

End-to-End Delay Analysis

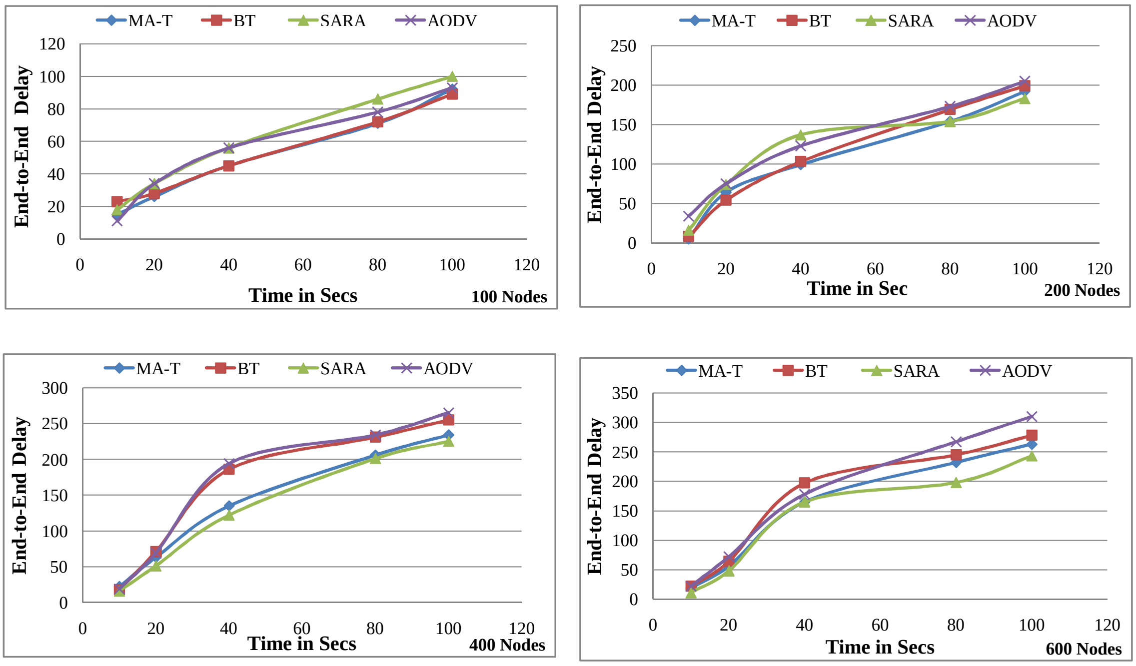


Figure 4.13: End-to-End Delay Vs Simulation Time in Sec (500m x 500m Sim. Area)

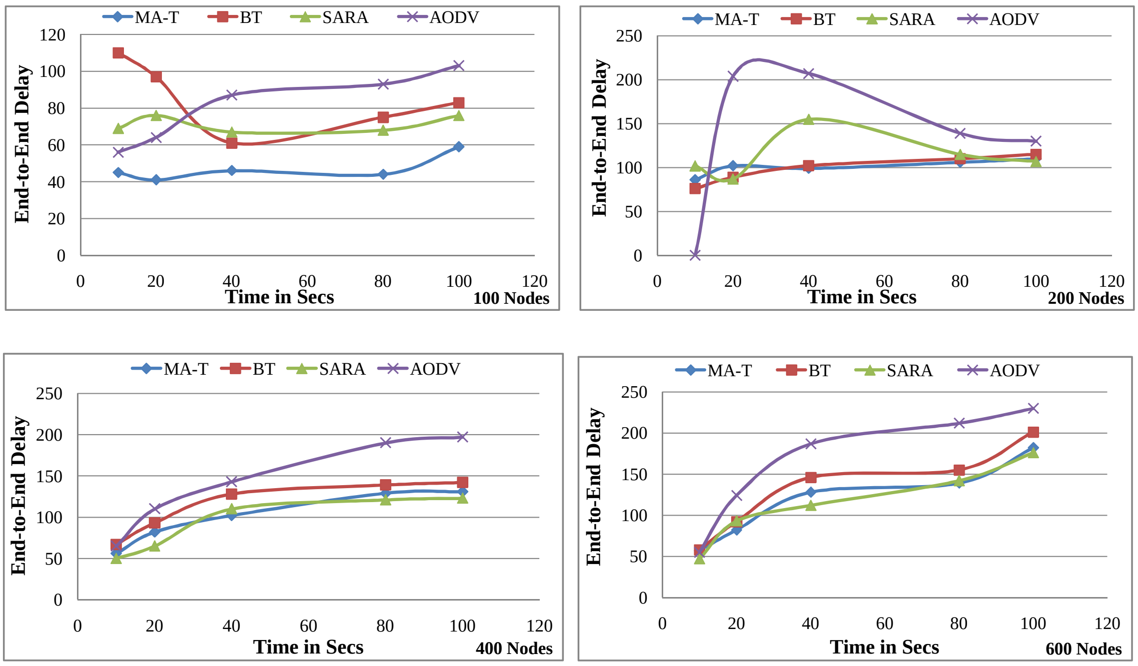


Figure 4.14: End-to-End Delay Vs Simulation Time in Sec (1000m x 1000m Sim. Area)

It is clearly observed from Figures 4.13 and 4.14 that, the end-to-end delay is directly proportional to the node density. When the node density is low, then the path consists of less numbers of relay nodes and thus exhibits low end-to-end delay. On the other hand, the path consists of more number of relay nodes when the node density is high and resulting in high end-to-end delay. Based on this simulation results, one can easily observe that SARA gives better performance when compared to MA-T, BT and AODV algorithms due to its efficient route management techniques. On the other hand, as node density increases BT and AODV finds it difficult to manage the root and thus gives high end-to-end delay. However, MA-T with its reliable path maintains the steady state data transmission and thus reduces the end-to-end delay when compared to BT and AODV.

Control Packet Overhead Analysis

Figures 4.15 and 4.16 show the control packet overhead for all competing protocols and there

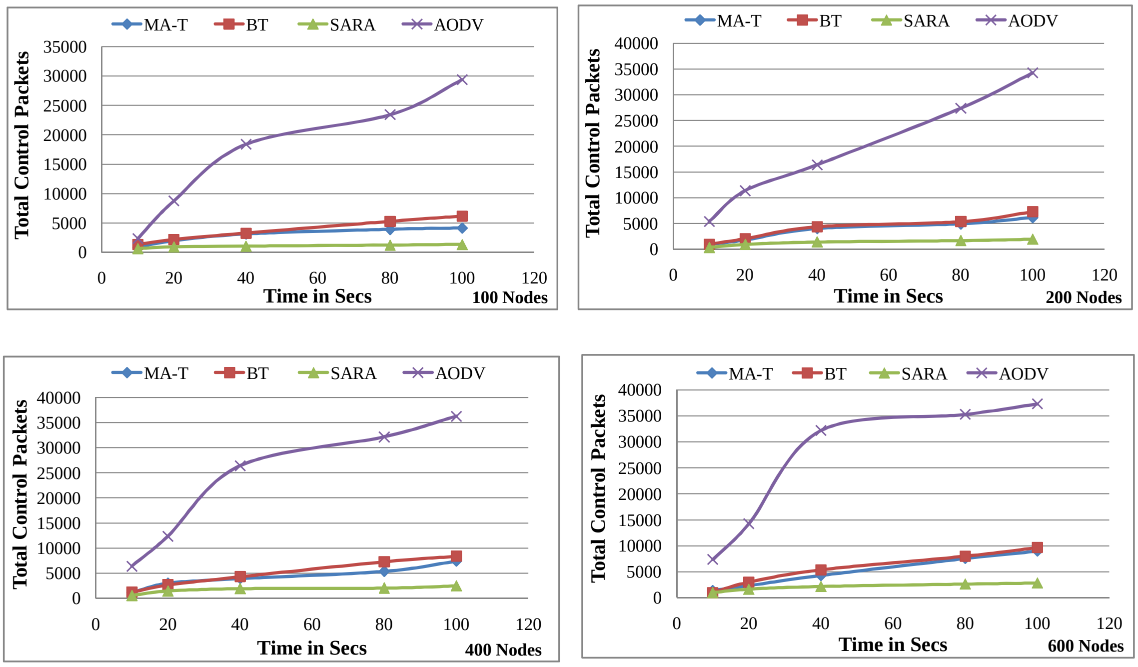


Figure 4.15: Control Packet Overhead Vs Simulation Time in Sec (500m x 500m Sim. Area)

by determining the complexity of the protocols. MA-T produces less control packets due to stable nodes in the path and thereby reducing path break ups. Whereas BT and AODV protocols exhibits frequent link breakups due to poor path management when the node density is high. On the other hand SARA exhibits superior performance when compared to MA-T, BT and AODV due to its efficient route management techniques such as Controlled Neighbor Broadcast (CNB), refreshing the paths with data packets and deep search procedure. Further, it is also observed from the graphs that MA-T and SARA produces less control packet even under high node density (600 nodes) and thus exhibit good scalability feature when compared to BT and AODV.

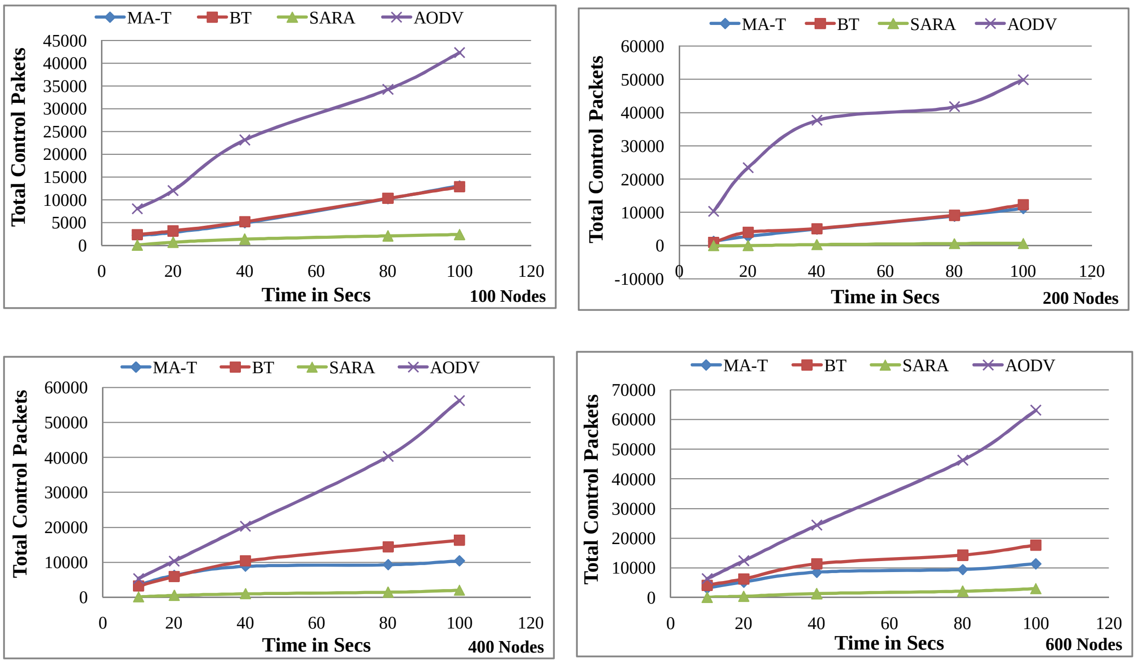


Figure 4.16: Control Packet Overhead Vs Simulation Time in Sec (1000m x 1000m Sim. Area)

4.4.3 Scenario B

Throughput Analysis: It is clearly observed from Figure 4.17 that AODV offers low throughput when compared to other protocols (MA-T, BT and SARA). This is mainly because of mobility of nodes that causes frequent link breakups and thus node spends most of its time in route discovery and maintenance than handling the data messages. On the other hand MA-T, SARA and BT algorithms outperform AODV due to their multi path routing, fast route recovery and fault tolerance features. But SARA suffers from low throughput due to high route setup latency when compare to MA-T; further SARA fails to handling high node velocity in finding reliable path between the source and destination node. However, MA-T shows considerable improvement when compared to BT and SARA due to its context awareness of the neighboring mobile nodes. Hence this context awareness of MA-T produces less packet drops when compared SARA, BT and AODV algorithms and the results shown in Figure 4.18.

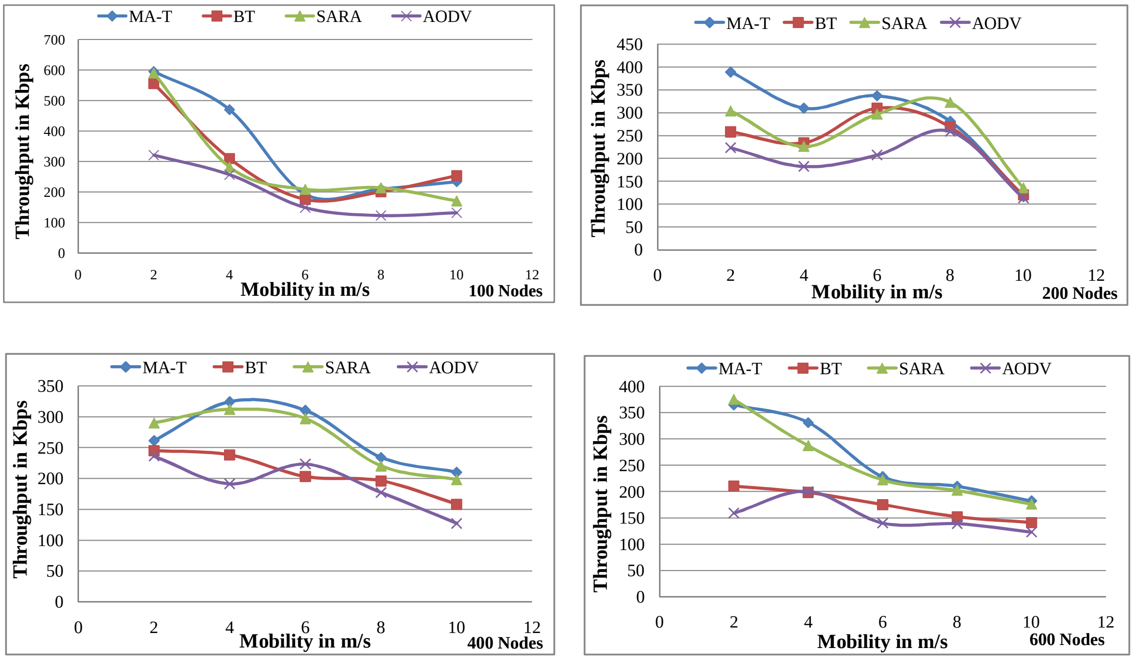


Figure 4.17: Throughput Vs Node Velocity

Packet Drop Analysis:

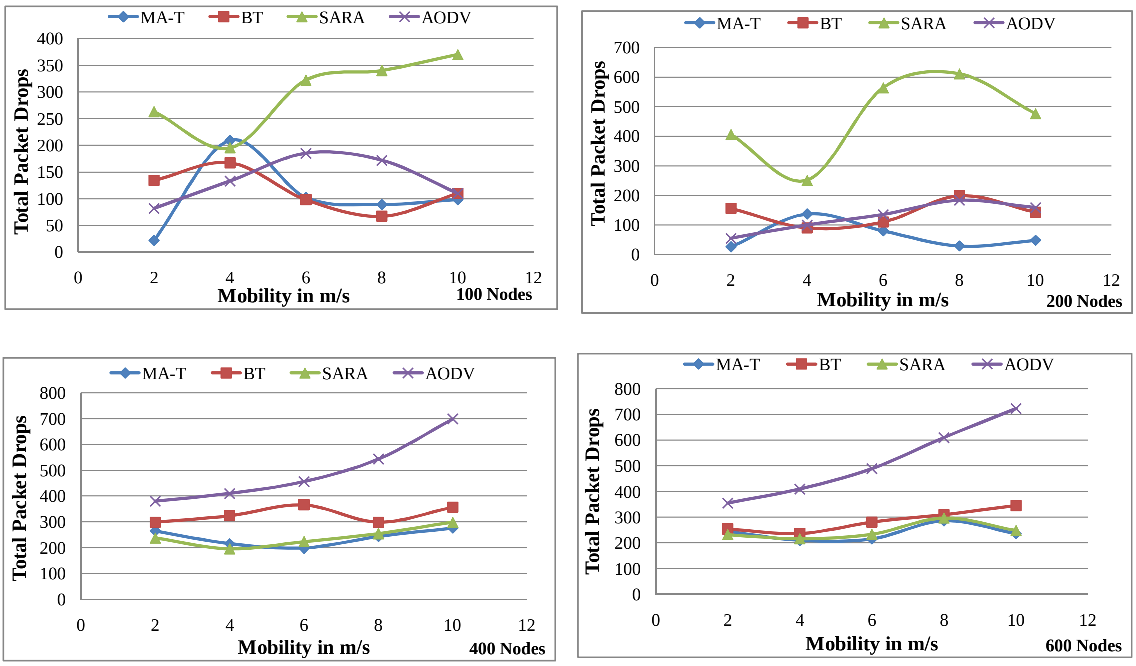


Figure 4.18: Packet Drop Vs Node Velocity

End-to-End Delay Analysis:

Figure 4.19 shows the end-to-end delay of all competing protocols and MA-T offers low end-to-end delay when compared to SARA, BT and AODV. Also it could be observed that node density and end-to-end delay are inversely proportional in case of MA-T and SARA algorithms; this is due to more alternative paths to reach the destination provided by more number of nodes. Further, MA-T and SARA exhibits superior route maintenance when compared to BT and AODV due to efficient use of alternative paths and thereby giving low end-to-end delay when the node velocity is high.

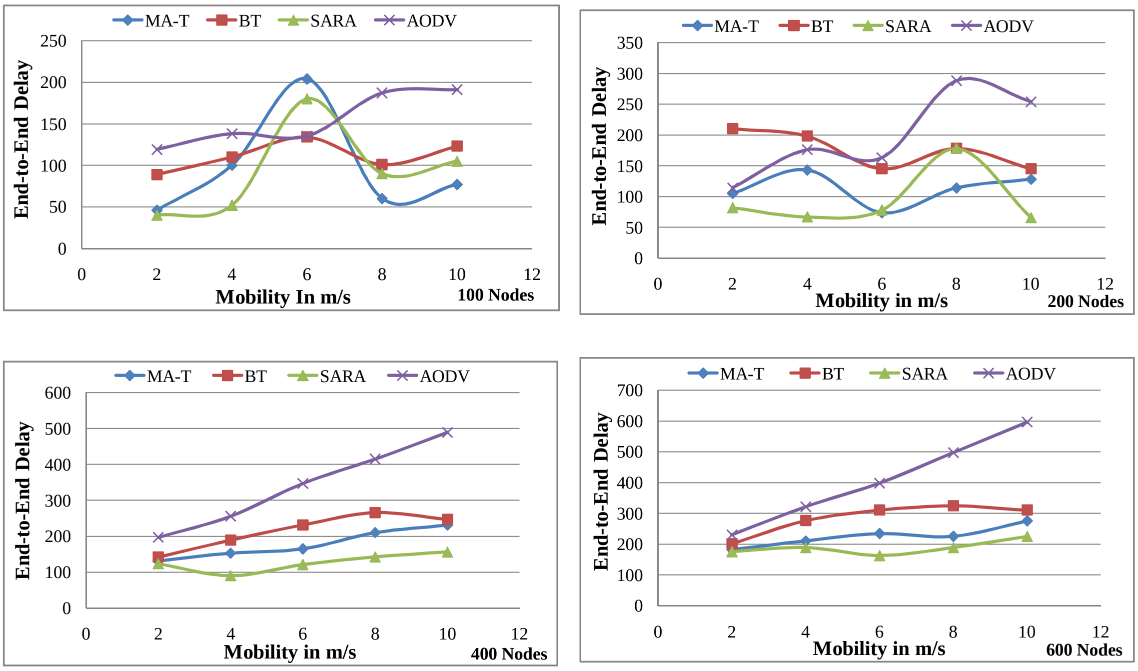


Figure 4.19: End-to-End Delay Vs Node Velocity

Control Packet Overhead Analysis:

Figure 4.20 shows the control packet overhead of all the competing protocols and it is clearly observed that, total control packet and node velocity are directly proportional in case of AODV. Further, it is also observed that SARA outperforms the MA-T, BT and AODV protocols. Although MA-T produces less control packets when compared to AODV and BT algorithms but it is inferior to SARA. In general, bio-inspired protocols do not show much variation under all mobility conditions whereas non bio-inspired protocol (AODV) shows some variations with respect to mobility. Further, AODV suffers from frequent link breakups under high mobility conditions and thereby causing more control packets in the network.

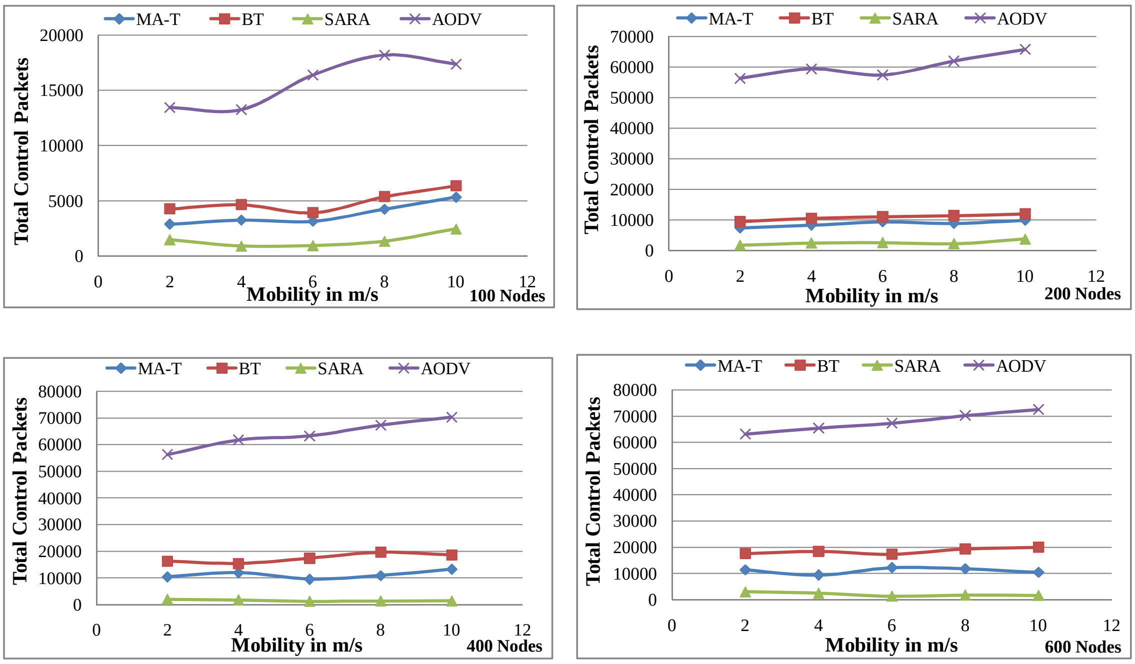


Figure 4.20: Control Packet Overhead Vs Node Velocity

4.5 Summary

In this chapter, a QoS aware bio inspired routing algorithm referred to as MA-Termite is presented for MANETs with analytical justification. A proof for pheromone update-decay function of MA-Termite algorithm is given and a analytical model is used for studying the asymptotic behavior of MA-Termite over both single and double links system. The asymptotic behavior is studied by using two different parameters *decay rate* and *pheromone sensitivity* and the model further reveals the selection of the optimum values of decay rate and pheromone sensitivity. A rigorous simulation study is carried out with different test scenarios in order to find the similarities between theoretical and simulation results under scalability and node velocity conditions. Further, the simulation results are also compared against bio-inspired (SARA and Termite algorithm) and non bio-inspired (AODV) routing algorithms for its performance evaluation. It is observed that MA-Termite achieves 2-3% improvement when compared to SARA with respect to QoS metrics such as *throughput, end-to-end delay* and *total packet drops*. It is also observed that MA-Termite achieves 5-8% improvement when compared to basic Termite and AODV algorithms in terms of QoS metrics such as *throughput, end-to-end delay, total packet drops* and *control packet overhead*. Simulation results also demonstrate that MA-Termite exhibit superior performance when compared to SARA, basic Termite and AODV *under dynamic network conditions*.

The asymptotic pheromone behaviour of MA-Termite studied in this chapter reveals that MA-Termite always chooses the highest pheromone link at any given time thus it suffers from *stagnation problem*. Further, MA-Termite also suffers from *poor network exploration* and *improper backup path maintenance* which causes the early removal of secondary paths from Trail Pheromone Table. Thus, in the next chapter a hybrid bio inspired routing protocol based on the behaviour of social insect *Termites* and mammal *Bats* will be discussed in order to overcome the poor network exploration and improper backup path maintenance problems of MA-Termite.

Chapter 5

Bat-Termite: A Novel Hybrid Bio-Inspired Routing

This chapter discusses a novel hybrid bio-inspired routing protocol for MANETs referred to as *Bat-Termite* which combines the unique features of both social insect *Termites* and the mammal *Bats*. The Termite is well known for its *hill building nature* and the Bat, a fascinating mammal is known for its advanced and distinguishing *echo-location feature*; hence in Bat-Termite, major behavioral advantages of both the *Termites* and the *Bats* are combined for building a new hybrid bio-inspired routing in MANETs. The echolocation feature of Bat-Termite will be studied in order to solve the problem of *poor network exploration* and *improper back up route maintenance* of MA-Termite in the following sections.

5.1 Motivation for Development of Bat-Termite Algorithm

As discussed in the previous chapter, the MA-Termite algorithm (Kiran M and Reddy G R M 2013) has two important drawbacks namely, *(i) poor network exploration and (ii) improper back up route maintenance*. To illustrate this, let us consider a two path system with the source node *A* and destination node *B* as shown in the Figure 5.1(a). Node *A* has two independent paths in reaching node *B* through neighboring nodes *X* and *Y*. Further, asymptotic behavior of MA-Termite will be experimentally verified based on the pheromone update-decay function with only node *A* neighbor links as shown in Figure 5.1(b). Assume that link L2 is the dominant link when compared to L1 and thus, the packets will choose L2 in reaching the destination node *B*. When the link L2 is positively reinforced then L2 will attract the entire traffic and thereby

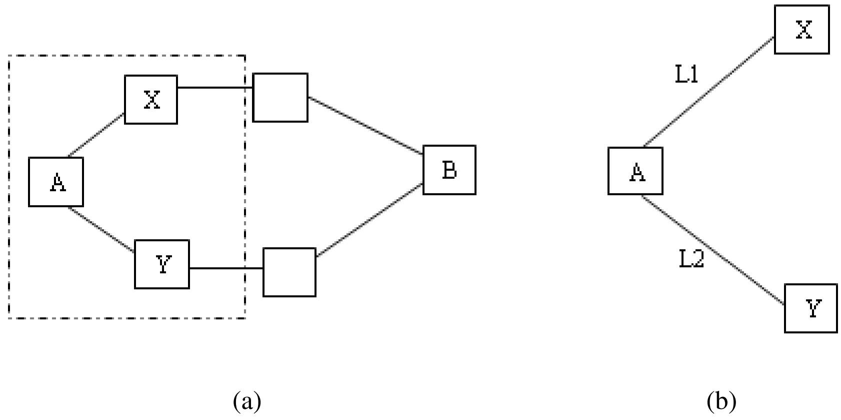


Figure 5.1: Asymptotic Behavior of MA-Termite on a Two Link System

accumulating more pheromone concentration. On the other hand L1 will be neglected because of less pheromone concentration which will be gradually decayed over certain time period; under these circumstances a single dominant link system will be emerging out of double link system.

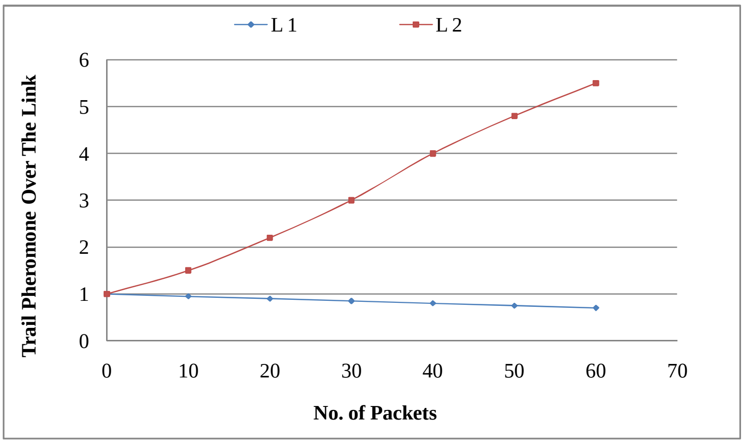


Figure 5.2: Experimental Results of Asymptotic Behavior of MA-Termite on a Two Link System

The experimental results of this situation are shown in Figure 5.2 and it is clearly observed from the graph that the MA-Termite algorithm has exclusively chosen the dominant link L2 and the less dominant link L1 will be gradually losing its priority over L2. Further, the pheromone on L1 reaches the minimum value over a period of time and the corresponding L1 entry will be removed from the *Trail Pheromone Table (TPT)* of node *A*; thus another option of reaching the destination node using L1 will be closed.

A situation will arise under such circumstances that the neighbor node is *reachable, active* and *could be used in reaching the destination* but neighbor node's pheromone entry will not be present in the TPT. If the Link L2 fails due to some reasons, then the node *A* will be left with empty neighbors in its pheromone table even though a neighbor node is active and could be used for reaching the destination. Thus, data packets from node *A* have to find an alternate pheromone trail in reaching the destination and in turn create a pause in the data transmission. If a node has an option for checking the existence of neighbors current status before removing its entry from the TPT then an alternate route could be maintained in reaching the destination under worst case scenario. In the above example, if the node A had such option then, it could have continued its data transmission using L1 without data transmission hick ups. Thus, the proposed Bat-Termite algorithm utilizes the fascinating echolocation feature of mammal Bats in order to address the poor network exploration and improper back up route maintenance of the MA-Termite. The echolocation feature of mammal Bats is explained in the next section.

5.2 Echolocation in Bats

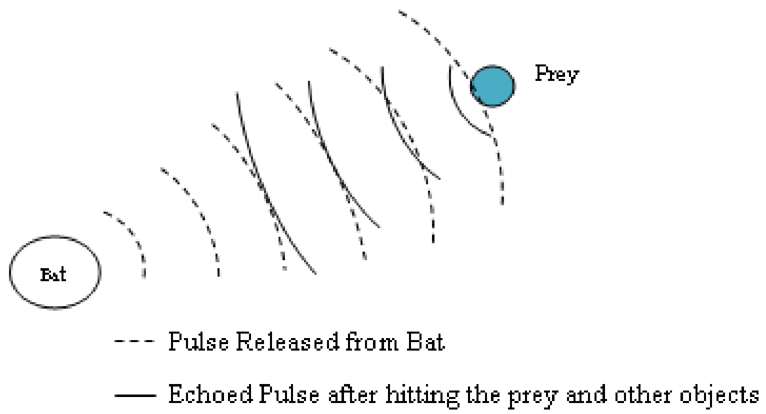


Figure 5.3: Echolocation Features of Bats

Two distinct characteristics of bats make them stand apart from other mammals: First, being the *only mammal to have wings* and the second, having advanced capability of *echolocation*. The fascinating *echolocation* feature of bats will allow them to find their *prey* and its status in terms of *distance* and *movement direction* in complete darkness. As shown in Figure 5.3, the bats produce a very loud pulse and listen for the echo that bounces back after hitting the prey and surrounding objects; depending on the species and obstacles, the pulse varies when it echoes back. According to a study, micro bats, a species of bats, builds a 3-D scenario of surrounding using the time delay between the emission of pulse and reception of an echo and the loudness variations of the echoes. Thus, Bats can detect the *type of prey, its distance, its orientation* and *its moving speed* using the Doppler Effect. This echolocation feature of bats has inspired many researchers in design and development of novel algorithms in non-telecommunication domain applications (Airas. M 2003; Yang. X.-S 2010; Bora T. C et. al. 2012). Thus, the echolocation feature of bats will be exploited in design and development of a novel hybrid routing algorithm referred to as *Bat-Termite* for telecommunication domain applications in general and in particular for MANETs and the echolocation feature of *Bat-Termite* is explained below.

5.3 Echolocation in Bat-Termite

In MA-Termite, once the pheromone of the neighbor node, which is still within the communication range, reaches its minimum value *(PH_FLOOR)* in the TPT then its facility cannot be used as an alternative route in reaching the destination; thus it will lead to poor network exploration and improper back up route maintenance. But the Bat-Termite sends a special *echo_location* packet towards the neighbor node (Figure 5.4(a)) in unicast mode and thereby observing its current status. Upon receiving the *echo_location* packet, the neighbour node then broadcasts the same *echo_location* packet (Figure 5.4(b)) by piggy backing its current status *(available bandwidth, queue status, distance etc.)*. Just like bats, which collects information about its prey *(its distance, movement direction etc.)* from the echoed signals, the current status of the neighbor node *(available bandwidth, queue status, distance etc.)* will be discovered using the broadcasted *echo_location* packet. Based on whether the neighbor node replies for

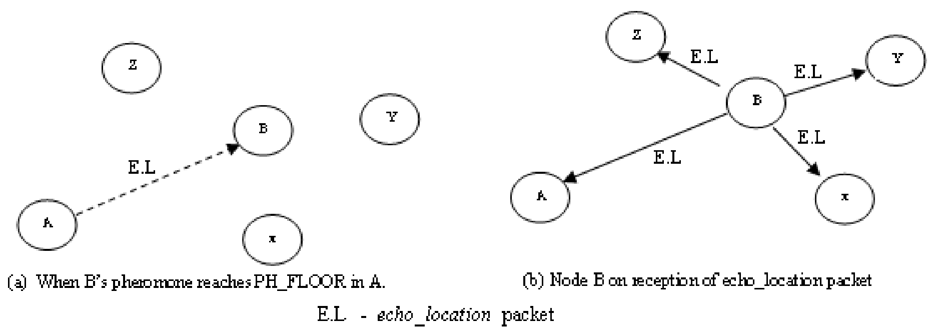


Figure 5.4: Echolocation Feature of Bat-Termite

echo_location packet or not and if it replies, based on its current status which are piggy backed with *echo_location* packet then the neighbor node will be reconsidered for an entry in the TPT. Thus, the echo location feature of Bat-Termite helps in keeping the fresh entry in the TPT with proper network exploration and thus reflects the current status of the network.

Mammal Bats, while using echolocation, eavesdrop the information to other specimens as well and this leakage of information may be intentional or unintentional (Airas. M 2003). Study shows that the echoed signals will also be used by other bats and animals for identifying the surrounding objects. This feature of bats will also be exploited in designing Bat-Termite algorithm by means of *broadcasting echo_location* packet by the neighbor node as shown in Figure 5.4(b). Upon receiving the *broadcasted echo_location* packet, the other nodes will also get the neighbors status and thereby reconsidering the broadcasting nodes entry in the TPT. The echolocation characteristics are explained in the following section.

Echolocation Characteristics of Bat-Termite

When a mobile node receives a broadcasted *echo_location* packet then it extracts the following:

1. The *distance* of the neighbor node can be found with the help of *Cross Layer Model* (Chapter 4, Subsection 4.2.1).

2. The neighbor nodes *moving direction* can be found whether the neighbor node is *moving towards* (current distance (d_c) $<$ previous distance (d_p)) or *moving away* from the node ($d_c > d_p$) or the neighbor node is *stable* ($d_c = d_p$).

3. The *mobility information* (speed at which the neighbor node is moving) can be determined by finding the *difference* between current and previous distances (d_c-d_p). Thus a node can decide whether the neighbor node is *reliable or not*.

4. The characteristics of the neighbor node can also be extracted with the piggy backed information such as *available bandwidth, queue status, energy etc*.

5.4 Route Maintenance Phase of Bat-Termite

Unlike basic Termite (Roth M and Wicker S 2004a; 2004b) and MA-Termite (Kiran M and Reddy G R M 2013) routing algorithms, the proposed hybrid Bat-Termite algorithm has a new route maintenance mechanism for issuing a *one hop route error packet (RERR)* and thereby notifying the path loss to the immediate neighbour. If a node is left with an empty neighbour list while forwarding the data packet towards the destination then the node will send a one hop *RERR* packet to its immediate neighbour node from where the data packet has arrived. This one hop *RERR* packet causes the recipient node to remove *RERR* source node entry from its TPT and thus drops the *RERR* packet. The recipient node then finds an alternative route from its remaining neighbour list and then forwards the stored data packet towards the destination. If the recipient node is the source node, then it will re initiate the route setup phase by issuing the *RREQ* packets. This concludes the working principle of Bat-Termite algorithm and the route maintenance and network exploration features of Bat-Termite are explained in Algorithm 5.1. The structure of the *RERR* and *echo_location* packets will be discussed in the next section.

5.5 Packet Structure of Bat-Termite

Apart from the regular packets of MA-Termite algorithm *(RREQ, RREP and DATA packets)*, the proposed Bat-Termite algorithm utilizes two additional control packets namely *RERR* and *echo_location* packets. Except *RERR*, the *echo_location* packet will be similar to *HELLO* packets of traditional AODV (C.M, Perkins and E.M, Royer 1999) protocol and further *echo_location* packets are event based and used for finding the current status of the neighbor nodes.
RERR packets: The *RERR* packets employed in Bat-Termite mirrors the *RERR* packet of traditional AODV protocol; but the only change is, *RERR* in Bat-Termite has only one hop life.

echo_location packets: The purpose of the *echo_location* packets is to broadcast the current status *(available bandwidth, energy, queue status etc.)* of a node towards its neighbor nodes. For the current study, *echo_location* packet does not include any status information as its main objective is to inform its neighbors about its existence. Thus *echo_location* packet looks like *HELLO* packets of AODV protocol.

Algorithm 5.1 : Route Maintenance and Network Exploration of Bat-Termite

Input : TPT and LST

Output: Updated TPT and LST for the each neighbors

1. ***When*** neighbour node $A's$ pheromone value reaches *PH_FLOOR* in the pheromone table
2. Unicast *echo_location* packet to the neighbour node *A*
3. ***if*** broadcasted *echo_location* packet is received as reply from neighbour node *A*
4. ***Calculate*** the current distance of the neighbour node *A*
5. ***Find*** the moving direction of neighbour node *A*
6. ***Find*** the mobility of neighbour node *A*
7. ***Find*** the current status *(Bandwidth, Queue status, Energy etc.)* of the neighbour node *A*
8. Based on the extracted information, reconsider the neighbour node $A's$ entry in the TPT.
9. ***else***
10. Remove the neighbour node $A's$ entry from the pheromone table
11. ***end if***

1. ***When*** a node receives unicasted *echo_location* packet
2. Piggyback its current status *(Bandwidth, Queue status, Energy etc.)*
3. Broadcast the *echo_location* packet.

1. ***When*** a broadcasted *echo_location* packet from node *A* is received by the other nodes
2. ***Calculate*** the current distance of the neighbour node *A*
3. ***Find*** the moving direction of neighbour node *A*
4. ***Find*** the mobility of neighbour node *A*
5. ***Find*** the current status of the neighbour node *A (Bandwidth, Queue status, Energy etc.)*
6. Based on the extracted information, update node $A's$ pheromone entry in the TPT

5.6 Performance Evaluation of Bat-Termite

The proposed Bat-Termite algorithm (here after called as Bat-T for plotting the graphs) is simulated in NS-2 and the results are compared with both bio-inspired (SARA (Correia, F and Vazao, T 2010) and basic Termite Algorithm (BT) (Roth M and Wicker S 2004a; 2004b)) and non bio-inspired (Ad hoc On Demand Distance Vector (AODV) (C.M, Perkins and E.M, Royer 1999)) routing protocols for its performance evaluation. The protocols are compared with respect to QoS metrics such as *throughput, total packet drops, end-to-end delay* and *control packet overhead*.

Simulation Environment:

Each simulation run was carried out for 100 secs and Random Waypoint Mobility model is considered for the simulation. At the MAC layer, IEEE 802.11 DCF is used and File Transfer Protocol (FTP) is used as a traffic type with the packet payload of 1500 bytes. Ten traffic source and destination nodes are considered for all the simulation runs and each Transmission Control Protocol (TCP) source destination pairs are selected randomly for each simulation run. Table 5.1 shows the different parameter values used for evaluating the performance of Bat-Termite algorithm. Further, the default values of SARA protocol are considered for performance evaluation throughout the experiment.

Table 5.1: Parameter Values used for Bat-Termite

Sl.No	Parameter	Value
1	*PH_INITIAL*	1
2	*PH_CEILING*	50
3	*PH_FLOOR*	0
4	τ	0.105
5	F	1

5.6.1 Experimental Setup

A rigorous simulation study was carried out under two different scenarios to see how the proposed algorithm behaves under *scalability* and the *node velocity* factor.

- **Scenario A**

  In this scenario *scalability* of the proposed algorithms is evaluated by considering two test areas analogous to academic campus of size 500m x 500m and 1000m x 1000m. Under each simulation area, the algorithms are tested for different node density of 100, 200, 400 and 600 nodes. The node velocity is kept constant at 2m/s when a normal walking scenario of a person is considered and no pause time is given to any mobile node.

- **Scenario B**

  In this scenario behavior of the proposed algorithms under different set of *node velocity* is studied for performance evaluation. Mobile nodes with varying node velocity (2, 4, 6, 8 and 10 m/s) and with different node density (100, 200, 400 and 600 nodes) are considered for each simulation run. Test area analogous to academic campus of size 1000m x1000m is considered and no pause time is given to any mobile node.

5.6.2 Scenario A

Throughput Analysis

From the Figures 5.5 and 5.6 it is clearly observed that Bat-Termite shows considerable through-

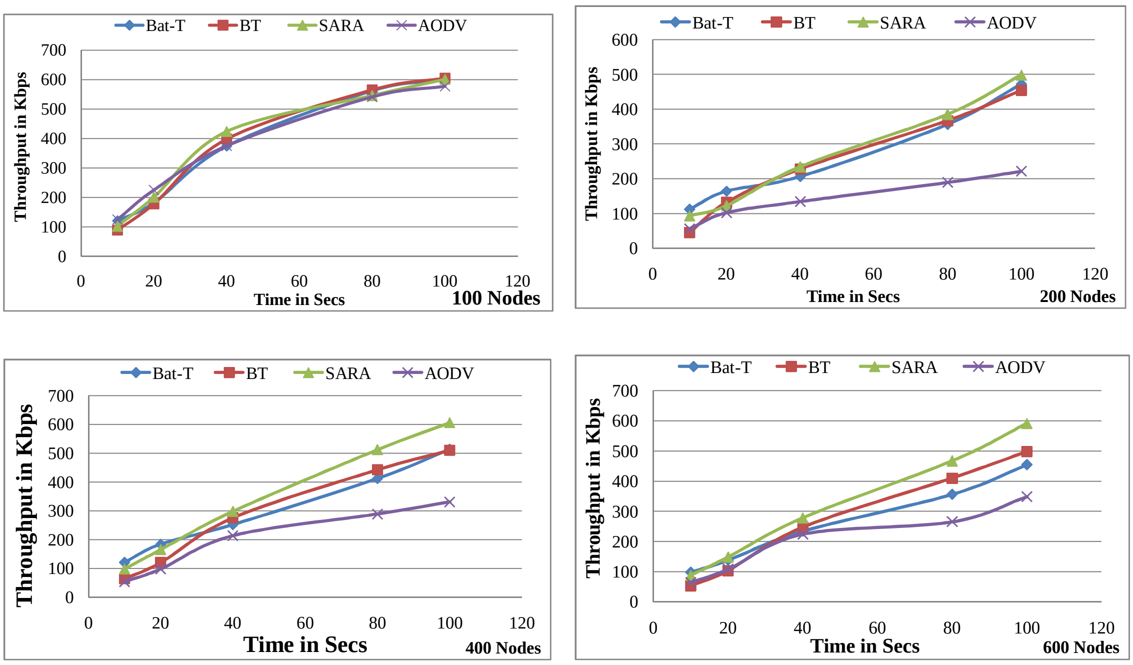


Figure 5.5: Throughput Vs Simulation Time in Sec (500m x 500m Sim. Area)

put enhancement under low node density (100 and 200 nodes) when compared to other routing protocols SARA, BT and AODV. There are mainly two reasons for the increased throughput of Bat-Termite, (1) Bat-Termite finds the reliable path to the destination with stable nodes; hence Bat-Termite maintains a steady state data transmission by choosing the stable nodes to the destination (2) Bat-Termite will manage the secondary paths efficiently and thus the chances of having the alternate paths to the destination node are high when compared to other protocols; whenever route breaks up, immediately Bat-Termite will find the alternative route to the destination without any pause in the data transmission. On the other hand, under high node density (400 and 600 nodes), Bat-Termite fails to achieve expected throughput because Bat-Termite finds it difficult to manage too many routes in the TPT. Since Bat-Termite will maintain all possible routes to the destination, nodes will be engaged in managing the routes instead of sending the data packets; thus gives low throughput when compared to SARA and BT. Also, increased number of *echo_location* packets under high node density cause congestion in the network in turn reduces the throughput of Bat-Termite.

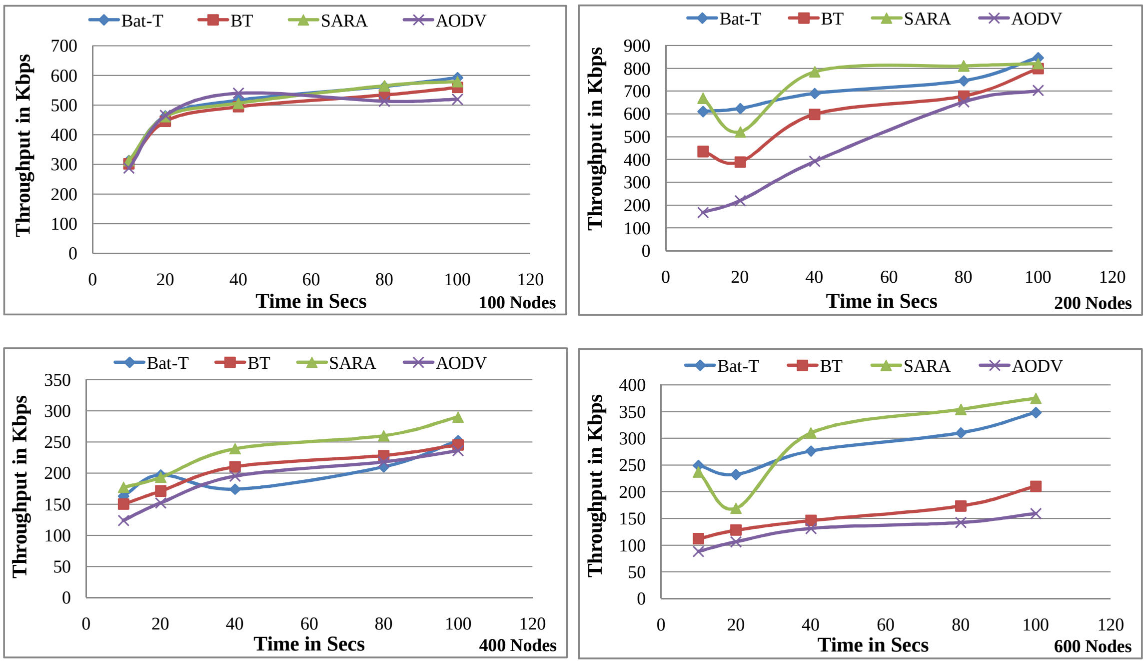


Figure 5.6: Throughput Vs Simulation Time in Sec (1000m x 1000m Sim. Area)

Packet Drop Analysis

The Figures 5.7 and 5.8 shows the total packet drops of the Bat-Termite, SARA , BT and AODV algorithms and it is clearly observed that Bat-Termite drops less packets at low node density (100 and 200 nodes) when compared to SARA, BT and AODV since reliable paths causes less number of route break ups in Bat-Termite. Further, even if route breaks up, Bat-Termite finds the new route immediately with its effective one hop RERR mechanism and thus will not allow further data packet drops. Whereas at high node density(400 and 600 nodes), Bat-Termite will produce more packet drops due to the increased number of *echo_location* packets which cause congestion in the network.

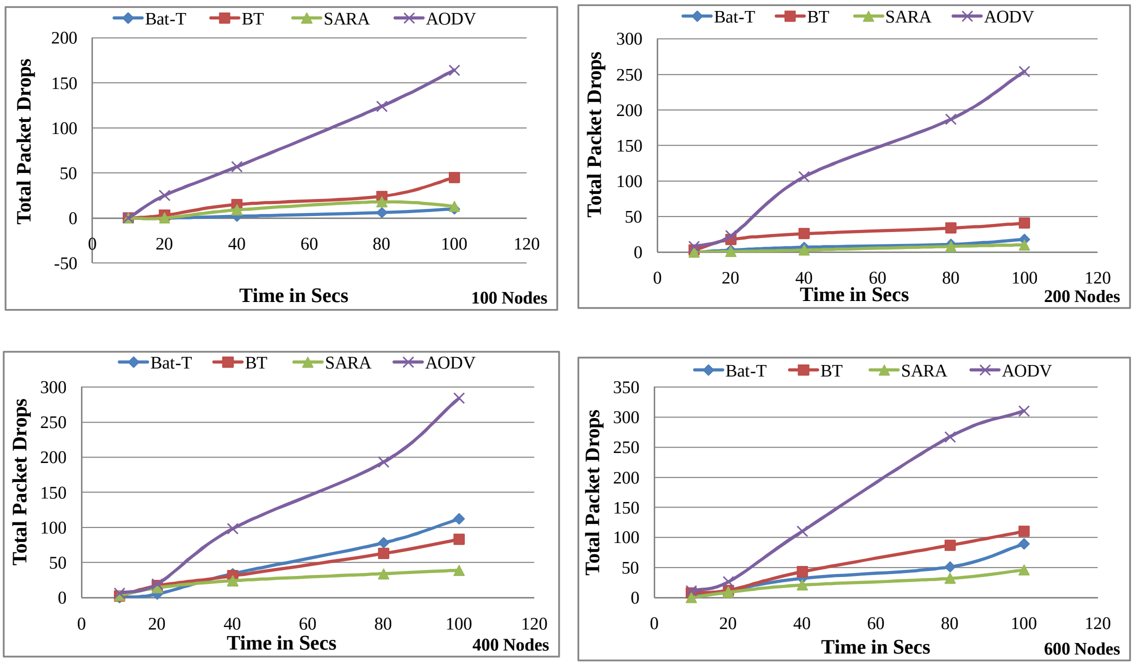


Figure 5.7: Packet Drop Vs Simulation Time in Sec (500m x 500m Sim. Area)

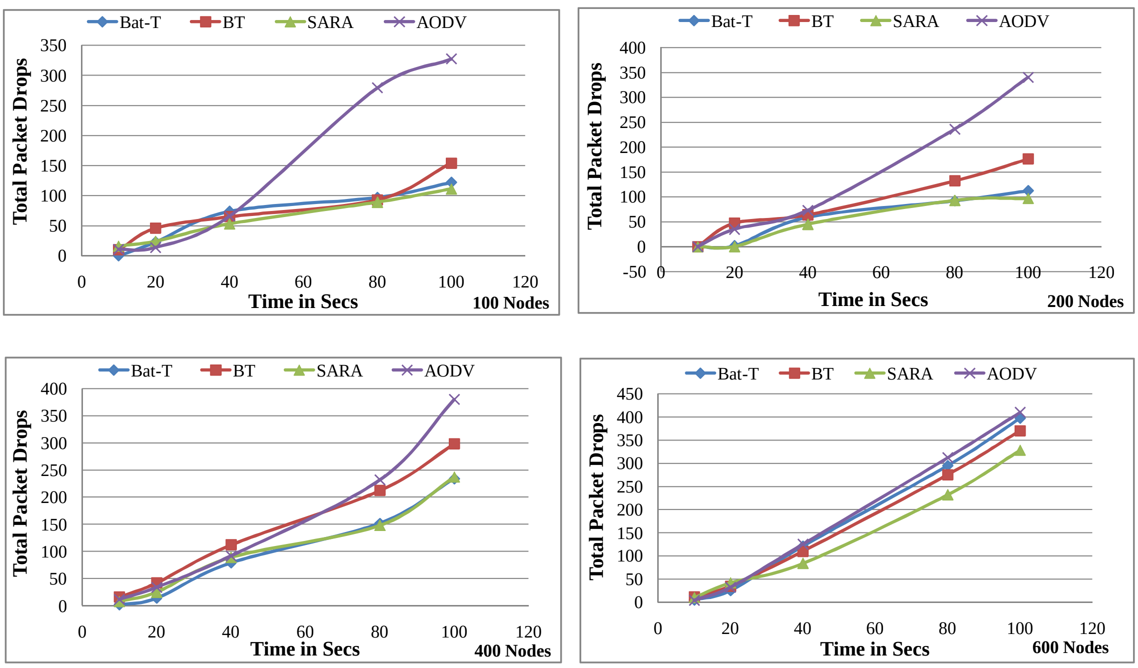


Figure 5.8: Packet Drop Vs Simulation Time in Sec (1000m x 1000m Sim. Area)

End-to-End Delay Analysis

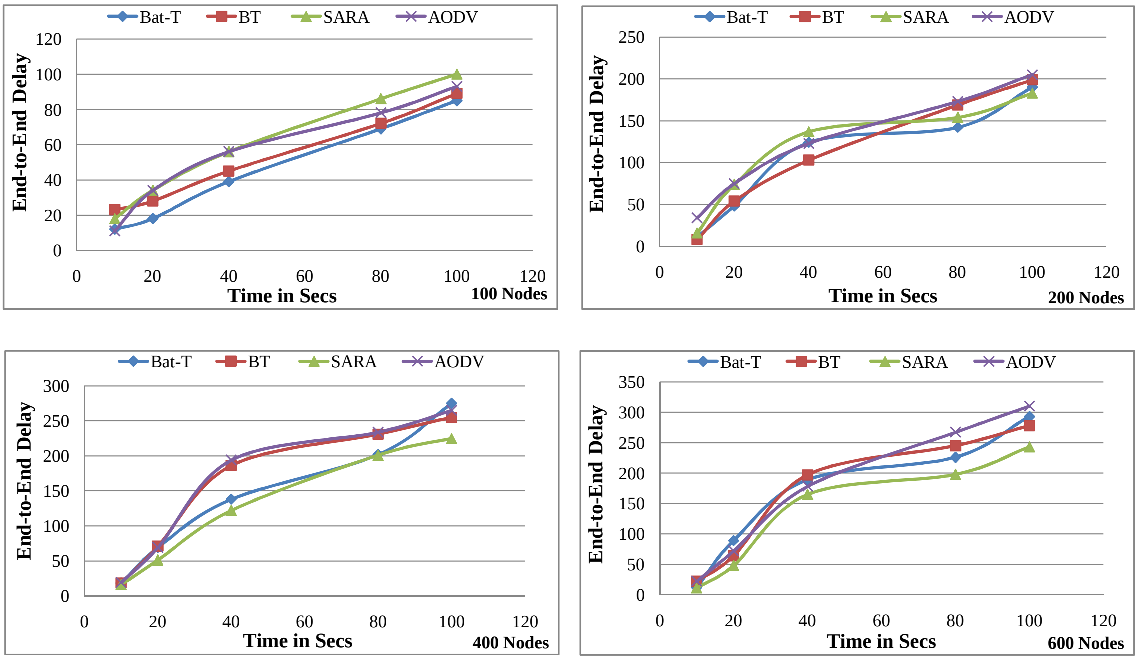


Figure 5.9: End-to-End Delay Vs Simulation Time in Sec (500m x 500m Sim. Area)

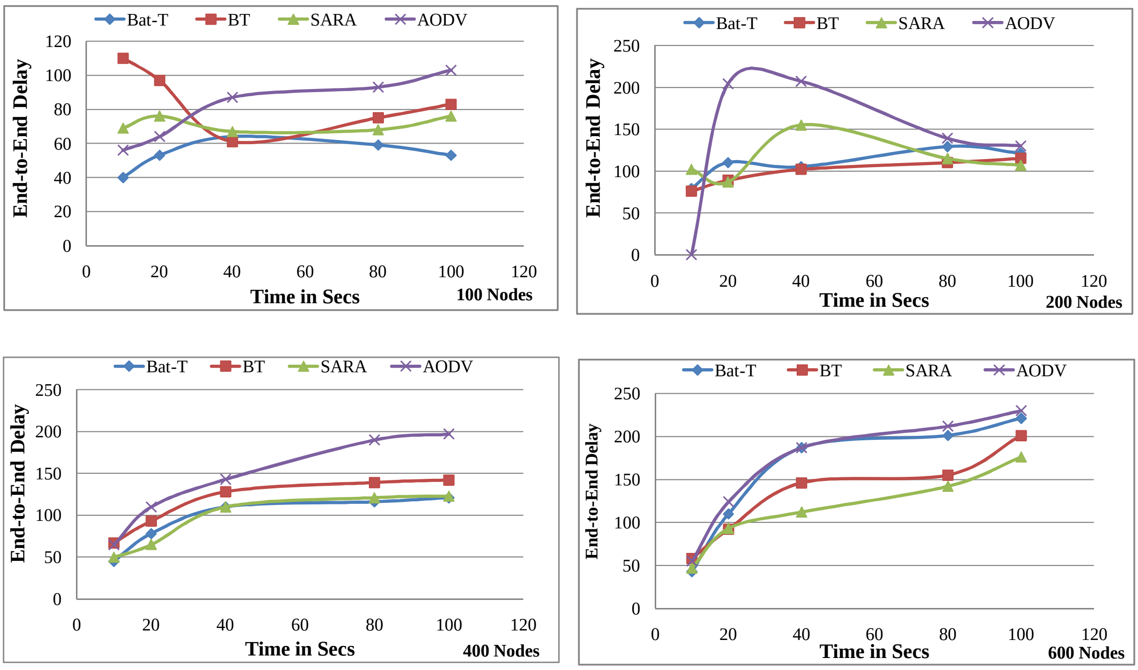


Figure 5.10: End-to-End Delay Vs Simulation Time in Sec (1000m x 1000m Sim. Area)

Figures 5.9 and 5.10 shows the End-to-End delay analysis of all competing protocols and it is observed that under low node density (100 and 200 nodes) Bat-Termite produces less end-to-end delay when compared to SARA, BT and AODV with its stable and reliable paths to the destination. But under high node density (400 and 600), Bat-Termite suffers from too many *echo_location* packets which blocks the node queue and causes the other packets to wait in the queue; thus packets takes time to reach destination and hence Bat-Termite produces high end-to-end delay.

Control Packet Overhead Analysis

Figures 5.11 and 5.12 shows the control packet overhead analysis of Bat-Termite, SARA, BT and AODV protocols. SARA with its simple and optimized routing technique produces very less control packets thus dominates Bat-Termite, BT and AODV. On the other hand, though Bat-Termite produces less control packets at low node density (100 and 200 nodes), but as node density increases Bat-Termite produces more control packets. These extra control packets are from echolocation feature of Bat-Termite which produces echolocation packets in order to maintain the alternative paths to the destination. It is also observed that the number of *echo_location* packet is directly proportional to the node density in case of Bat-Termite.

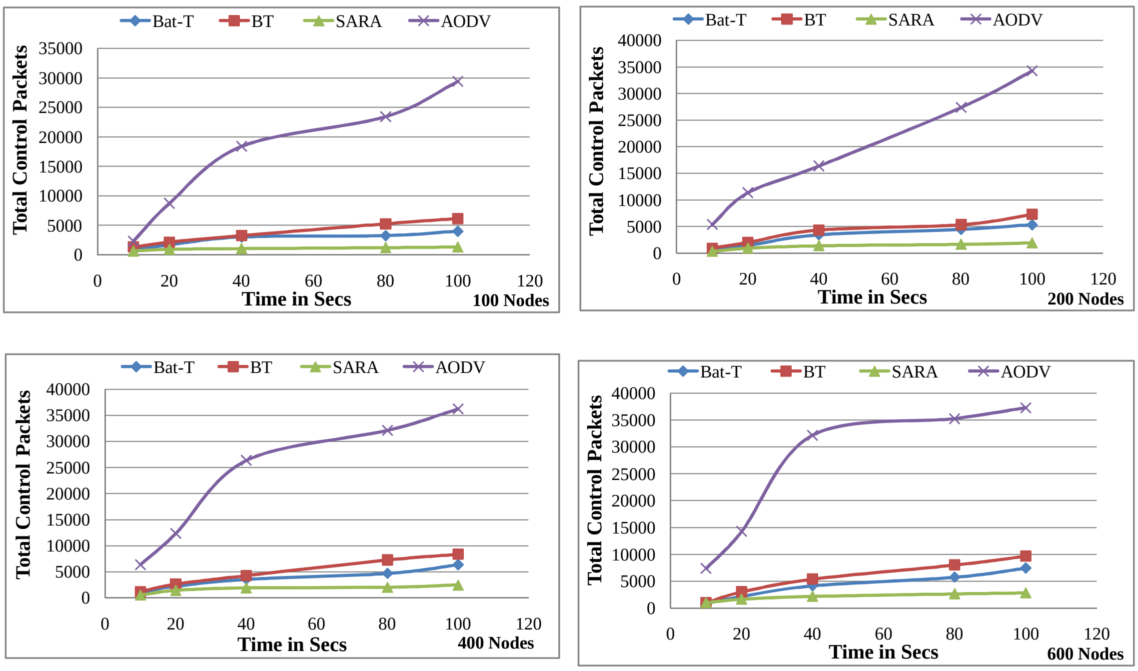


Figure 5.11: Control Packet Overhead Vs Simulation Time in Sec (500m x 500m Sim. Area)

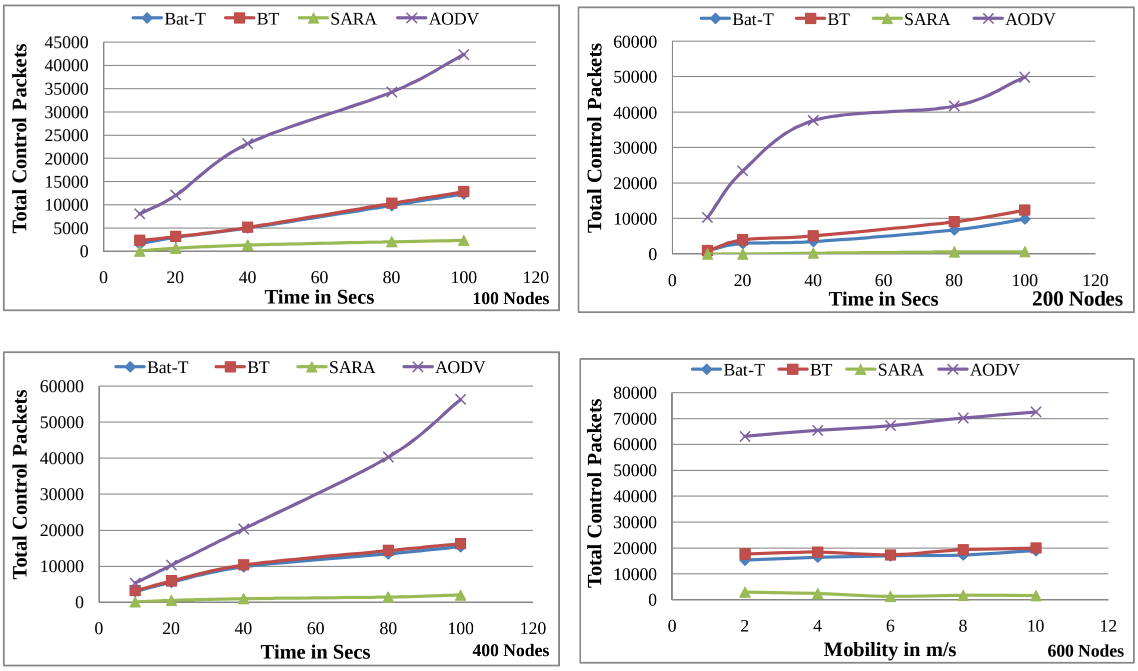


Figure 5.12: Control Packet Overhead Vs Simulation Time in Sec (1000m x 1000m Sim. Area)

5.6.3 Scenario B

Throughput Analysis

Figure 5.13 shows the Throughput comparison of competing protocols under different node velocity. As with the case of Scenario A, Bat-Termite produces good throughput under low node density (100 and 200 nodes) and under all node velocity (2, 4, 6, 8 and 10 m/s) conditions with its context awareness feature when compared to SARA, BT and AODV. As the node mobility will be reflected in the TPT, each node will have information about its neighbor node movements and their chances of staying in the transmission range; thus a node can select the stable and reliable node for the paths in order to reach the destination node. Further, the echolocation feature of Bat-Termite helps each node to maintain the alternative routes in the worst case scenario thereby causing a continuous data transmission without any hiccups. But Bat-Termite suffers from too many echolocation packets under high node density which causes congestion in the network and thus reduces the throughput. These *echo_location* packets even though serve their intended purpose of back up route maintenance but will create much load in the network.

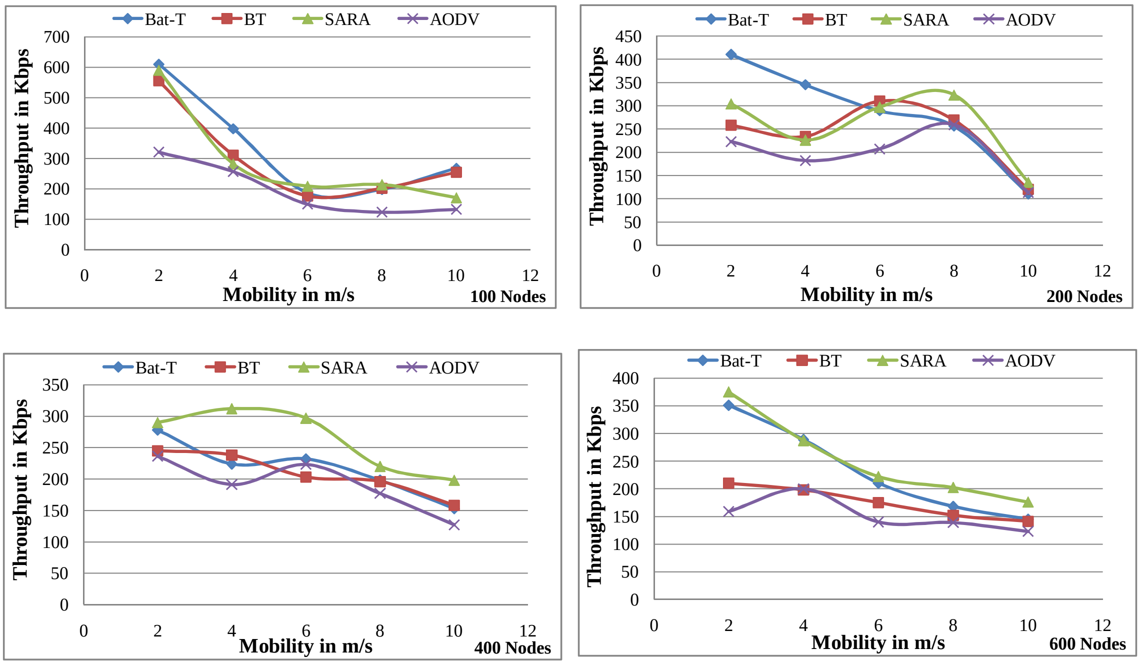


Figure 5.13: Throughput Vs Node Velocity

Packet Drop Analysis

It is clearly observed from Figure 5.14 that Bat-Termite with its context awareness feature pro-

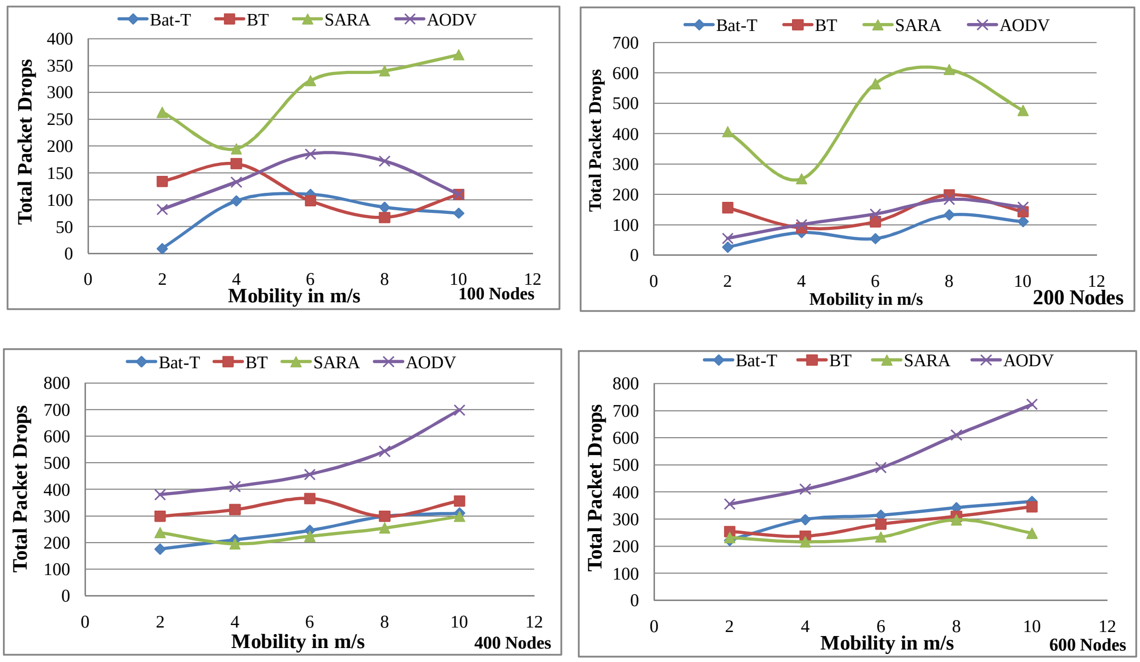


Figure 5.14: Packet Drop Vs Node Velocity

duces less number of packet drops under low node density (100 and 200 nodes) and under all node velocity (2, 4, 6, 8 and 10 m/s) conditions. But Bat-Termite produces too many packets drops under high node density conditions (400 and 600 nodes) since *echo_location* packets are directly proportional to the node density which creates congestion in the network. It is also observed from the graphs that the total packet drop of all the protocols is directly proportional to the node density.

End-to-End Delay Analysis

End-to-End delay analysis of all the competing protocols is shown in the Figure 5.15. Under low node density conditions Bat-Termite produces less end-to-end delay because it chooses stable nodes in the path; under high node density conditions even though the path will be stable with low mobility nodes, *echo_location* packets blocks the other packets in the queue in turn delays the other packets in reaching destination node; hence Bat-Termite produces high end-to-end delay under high node density irrespective of the node velocity.

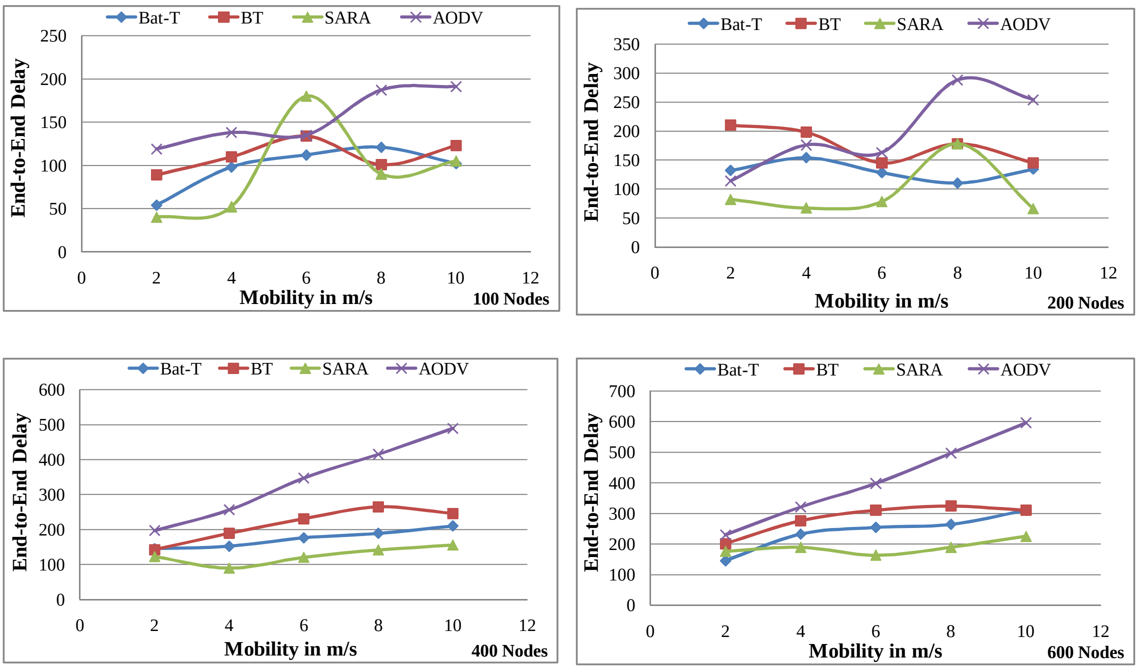


Figure 5.15: End-to-End Delay Vs Node Velocity

Control Packet Overhead Analysis

Figure 5.16 shows the control packet overhead and it show that, SARA with its simple routing techniques dominates all the protocols under all node density and node velocity conditions. On the other hand, even though Bat-Termite handles node velocity efficiently, produces more control packets under high node velocity because of its *echo_location* feature.

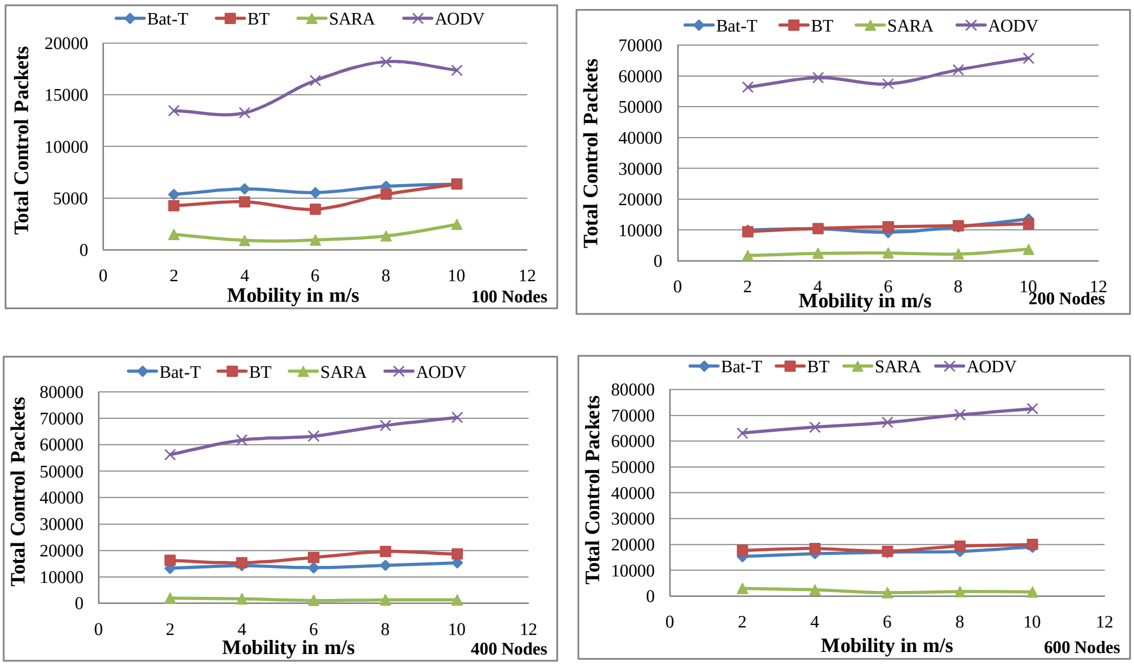


Figure 5.16: Control Packet Overhead Vs Node Velocity

5.7 Summary

The proposed hybrid bio-inspired routing protocol for MANETs referred to as *Bat-Termite*, combines the major attractive features of Termites *(hill building nature)* and Bats *(echolocation)*. The drawback of MA-Termite algorithm is it will not use the active neighbor as a next hop to forward the data packets if its entry is not there in the TPT; whereas Bat-Termite algorithm using *echolocation* feature keeps all the active neighbor nodes entries in the TPT which lead to destination. *Quick route discovery, high robustness with efficient management of multiple routes* and *rapid route repair* are the highlights of Bat-Termite algorithm. With its reliable path and effective backup route maintenance scheme, the proposed Bat-Termite algorithm maintains a study data transmission between the source and the destination there by giving less end-to-end delay. With its robustness and with less number of link breakups due to stable mobile nodes in the path, Bat-Termite produces less control packet overhead. The simulation results clearly depicts that, even though Bat-Termite solves the *poor network exploration and improper route maintenance* problems of MA-Termite, but still *stagnation problem* is not addressed by Bat-Termite algorithm. It is also observed that Bat-Termite suffers from *additional echolocation control packets under high node density conditions* and thereby deteriorating the performance of Bat-Termite algorithm.

Since hybrid bio-inspired Bat-Termite algorithm did not achieve the intended objectives at this juncture, hence a novel *load balanced hybrid bio-inspired routing protocol* based on the salient features of social insect *Termites* and mammal *Bats* will be discussed in the next chapter. This novel hybrid *Load Balanced Bat Termite (LB-Bat-Termite)* algorithm will try to mitigate the stagnation problem of Bat-Termite algorithm.

Chapter 6

Load Balanced-Bat-Termite: A Novel Hybrid Bio-Inspired Routing

This chapter discusses the design and development of a novel heuristic hybrid bio-inspired routing protocol with load balancing approach referred to as *Load Balanced-Bat-Termite (LB-Bat-Termite)* for MANETs. Apart from the salient features of MA-Termite and Bat-Termite, LB-Bat-Termite will also consider the *nodal load* in order to achieve the *load balancing among the available multiple paths* and thereby *mitigating the stagnation problem* of both MA-Termite and Bat-Termite algorithms. Further, LB-Bat-Termite will also *overcome the poor backup maintenance and improper route maintenance* problems of MA-Termite by efficiently managing the multiple paths in Trial Pheromone Table (TPT). The details of LB-Bat-Termite are discussed in the following sections.

6.1 Motivation for Development of LB-Bat-Termite

The existing routing protocols for MANETs use many routing metrics for finding the optimal path to the destination in general and in particular the route *hop count* which is the most popular metric for finding the shortest path (Douglas S.J. De Couto et al. 2003). The nodes which appear in the shortest path will get the maximum load when compared to other mobile nodes and thus consume most of their available resources such as *bandwidth, energy, memory etc*. This heavy load creates a bottleneck in the shortest path and resulting in *congestion problem*; congestion causes packet loss and will result in reduced throughput and in the worst case scenario the connection may be lost due to heavy packet loss. The major problem in many ACO and termite based routing algorithms is the *stagnation problem* which occurs when a network reaches its *convergence* or *equilibrium state*. Hence, both MA-Termite and Bat-Termite algorithms will also suffer from the stagnation problem. To illustrate the asymptotic behavior of MA-Termite and Bat-Termite, let us consider a two path system as shown in Figure 6.1(a) where source node *X* wants to communicate with destination node *Y*. To reach the destination node *Y*, the source

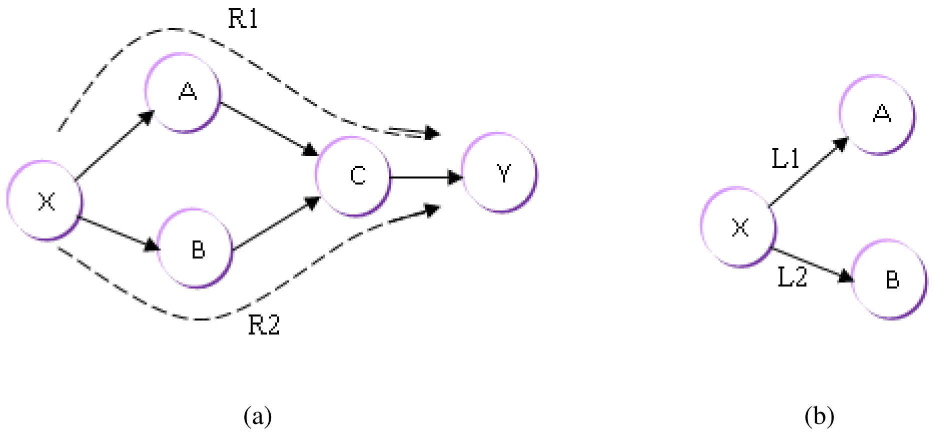


Figure 6.1: Two Path System for Studying the Asymptotic Behavior of MA and Bat-Termite

node *X* should have two independent paths *R1* and *R2* and each path consists of a series of relay nodes with different characteristics such as *mobility, bandwidth* etc. Figure 6.1(b) shows the subset of Figure 6.1(a), where node *X* has a choice to choose either link *L1* or *L2* in reaching the node *Y*. During the route setup phase, suppose node *X* chooses Link *L2* for reaching the destination node *Y* then the trial pheromone on link *L2* increases since *L2* is positively reinforced and thereby attracting the entire traffic; on the other hand *L1* is negatively reinforced and thus the pheromone on link *L1* decays gradually over certain period of time. Under such circumstances the network appears like a single dominant Link (L2 in the above example) between the source and destination nodes. The experimental results of this situation are depicted in Figure 6.2

It is clearly observed from Figure 6.2 that the MA-Termite and Bat-Termite algorithms *exclusively choose the dominant link always* and once any link dominates other links, then the entire traffic is diverted towards the dominant link only and thus creates a *bottleneck in the dominant link*. On the other hand, the less dominant link *L1* is gradually losing its priority over the dominant link. Hence, over a period of time there are chances that *L2* may get congested by the packets and may become non optimal. As packets are attracted towards the highest

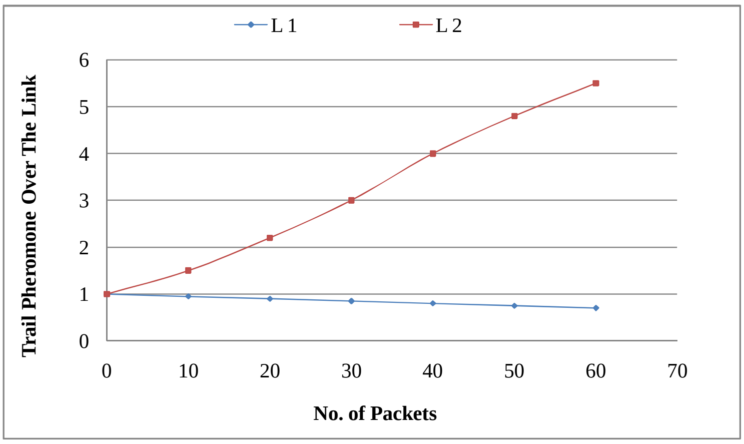


Figure 6.2: Experimental Results of Asymptotic Behavior of MA and Bat-Termite on a Two Link System

pheromone link, even though *L2* is congested since the trial pheromone on *L2* is very high, the packets still choose *L2* to reach the destination. This scenario further worsens the condition of *L2* resulting in heavy packet loss and low throughput. Such worst case scenario is referred to as *stagnation* which will lead to:

- Decrease in probability of selecting less dominant links (L1 in the above example)
- Congestion in optimal link (L2 in the above example)
- Other non optimal link may become optimal link (L1 in the above example)

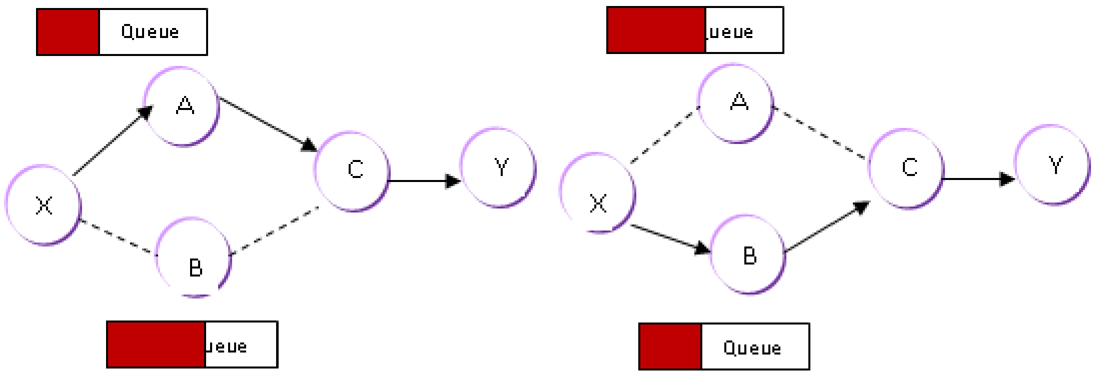
Since mobile nodes of MANETs have limited resources, there is a need for good load balancing strategy to make use of the available resources efficiently and effectively. The term *load* is broadly classified into *channel load* (loads on the channel), *nodal load* (nodes activity) and *neighboring load*(load generated by the neighbor nodes). The term *nodal load* implies how busy a node is in terms of its activity and it can be further described by *memory, power, processing* and *bandwidth* loads. Most of the existing routing protocols are mainly addressing the load balancing issue *during the route setup phase only* and these protocols are broadly classified into *delay, traffic* or *hybrid* based load balancing protocols. Further, these protocols will target either single or multiple load metrics such as *active paths, traffic size, packets in the interface queue, channel access probability* and *node delay* (Chai Keong Toh et al. 2009). But these algorithms will not adequately deal with issues such as load balancing during the route maintenance phase and dynamic topology. Further, these context unaware protocols are less stable and also suffer from the *scalability*. Hence, the current trends have focused on Swarm Intelligence (SI) based load balancing routing protocols. On the other hand, the undesirable behavior of these bio-inspired routing protocols is the *stagnation problem* caused by always choosing the highest pheromone link at any given time. This stagnation is unacceptable as the link selected may get congested over a period of time and thus reduces the throughput (Sim K.M and Sun W.H 2003).

Stagnation problem can be avoided using different techniques namely *evaporation, aging, pheromone smoothing and limiting, privileged pheromone laying* and *pheromone heuristic control* (Sim K.M and Sun W.H 2003). These approaches reduce the influence of past experience and allow new and better links for effective data transfer. Hence, the proposed LB-Bat-Termite will address the stagnation problem by balancing the load among the available multiple paths using *pheromone heuristic control* method and the details of load balancing approach are discussed below.

6.2 Load Balancing Approach of LB-Bat-Termite

The proposed LB-Bat-Termite introduces another kind of pheromone called *alarm pheromone* which is used by the social insect termites and thereby indicating the *danger*. Alarm pheromone in LB-Termite indicates the load on each neighbor node and it is defined as a function of *total number of packets waiting in the queue* (Hoolimath, P et al. 2012). In LB-Termite, a new forwarding function based on heuristics will be defined for finding the probability of the neighbor node and thereby forwarding the packet towards the destination node. Apart from the trail pheromone update/decay function of MA-Termite, a new heuristic function is also proposed for updating the alarm pheromone on each neighbor link to reflect the current load on the neighbor node.

The proposed LB-Bat-Termite algorithm achieves the load balancing by avoiding the links with higher alarm pheromone concentration. During the route discovery phase, LB-Bat-Termite considers only the trail pheromone for finding the multiple paths towards the destination while combination of alarm and trail pheromones will be used for forwarding the data packets. Thus the proposed LB-Bat-Termite algorithm achieves load balancing during both route discovery and route maintenance phases and thereby providing high robustness. Further, LB-Bat-Termite does not need additional control packets for finding the neighbor nodes load as queue status is piggybacked in each outgoing packets. Figure 6.3 depicts the load balancing approach of the proposed LB-Bat-Termite algorithm.



(a) At time t (b) At time t+n

Figure 6.3: Illustration of Nodal Load Balancing of LB-Bat-Termite Algorithm

As shown in Figure 6.3(a), let us consider that a node *X* initially chooses a neighbor node *A* with less load for reaching the destination. When node *A* gets higher load than node *B*, then the node *X* switches the traffic to node *B* as shown in Figure 6.3(b), thus LB-Bat-Termite achieves the *dynamic load balancing*. The alarm pheromone update function of LB-Bat-Termite is explained in the following Section.

6.3 Alarm Pheromone Update Function of LB-Bat-Termite

Alarm pheromone can be updated by any kind of packets either by control or data packets. Whenever the mobile node receives a packet from the neighbor node, then it first extracts the piggybacked queue status from the packet and then calculates the alarm pheromone using 6.1.

$$\phi_i^n = 1 - \frac{TotalNo.ofPacketsintheQueue}{Q\_MAX} \tag{6.1}$$

where ϕ_i^n is the amount of alarm pheromone of neighbor node i at node n and *Q_MAX* is the maximum number of packets that the queue can accommodate. Alarm pheromone and the time interval at which alarm pheromone is calculated will be stored in a table at routing layer for further processing. The time interval gives the freshness of the table entry and the table details are discussed in Section 6.6. The differences between the alarm pheromone and trail pheromone is given in the next section.

6.4 Trial Pheromone Vs Alarm Pheromone

Comparative features of Trail and Alarm pheromone are given below:

- Like trial pheromone, there will be no decaying concept for alarm pheromone.
- Trail pheromone will be used for finding all available multiple paths to the destination node while a combination of trail and alarm pheromones will be used for finding the reliable and resource rich path among the available multiple paths to the destination node.
- Trail pheromones can be updated by the data packets where as alarm pheromones can be updated by both data and control packets
- The trial pheromone can be defined as a function of total number of packets passing through the node, on the other hand an alarm pheromone can be defined as a function of total number of packets in the queue which completely depends on the entire node traffic

The computed alarm pheromone will be further used by the forwarding function in order to estimate the next hop probability as discussed below.

6.5 Forwarding Function of LB-Bat-Termite

In the proposed heuristic model of LB-Bat-Termite, the next hop probability can be estimated based on the trail pheromone $P^n_{i,d}$ and a heuristic function of alarm pheromone η^n_i. At each mobile node, the packet can be forwarded as a functional composition of $P^n_{i,d}$ and η^n_i . The heuristic function is defined by

$$\eta^n_i = \frac{\phi^n_i}{\sum_{k\in N}\phi^n_k} \tag{6.2}$$

where ϕ^n_i is the alarm pheromone for the neighbor node *i* at node *n*, *N* is the total number of neighbor nodes of node *n*. The forwarding function is defined by

$$p^n_{i,d} = \frac{P^n_{i,d}(\eta^n_i)^F}{\sum_{k\in N} P^n_{k,d}(\eta^n_k)^F} \tag{6.3}$$

where $P^n_{i,d}$ is the trail pheromone of node *i* to reach the destination *d* at node *n*; *N* is the total number of neighbor nodes of node *n*; η^n_i is the heuristic value calculated based on an alarm pheromone for neighbor node i at node n. The constant F is defined as the inertia factor and it can be used for controlling the routing behavior of LB-Bat-Termite. The preferences for the next hop selection can be varied by choosing the proper value of F; a lower value of F can be generally considered for selecting the highest trail pheromone link for forwarding the data packets. On the other hand, a higher value of F orients the data packets towards more optimistic heuristic value; but the optimal value of F is dependent on the pheromone ceiling *PH_CEILING*. The node structure of LB-Bat-Termite is explained below.

6.6 Node Structure of LB-Bat-Termite

In the proposed LB-Bat-Termite algorithm, each node maintains two tables at network layer namely *Local Status Table (LST)* and *Trail Pheromone Table (TPT)* like MA-Termite (Chapter 4, Subsection 4.2.4).

6.6.1 Local Status Table

Each node will refer to LST while forwarding a data packet towards the destination node and LST consists of the *current status of neighbor nodes in terms of alarm pheromone, current and previous distances*; further LST will also maintain the *time interval* at which these parameters are calculated. This time entry gives the freshness of the information available in the table and at every *LST_TIME_OUT*, the LST can be refreshed in order to remove the stale entries. The current and previous distances of neighbor nodes will give the mobility information and these distances can be computed as follows.

Neighborhood Node Distances: whenever a node receives a packet then it calculates the distance of the transmitting node using the free space propagation model (chapter 4, Subsection 4.2.1) and stores it in Local Status Table (LST) along with the time interval at which the distance is calculated. Thus, every node will be aware of its neighbor node movements, its mobility and its chances of staying within the transmission range. Further, alarm pheromone can be calculated in order to reflect the load on each neighbor node and the details of computing alarm pheromone are given below.

Alarm Pheromone: Whenever a node receives a data or control packet then it calculates the alarm pheromone based on the total number of packets in the sending nodes queue using Equation 6.1 and this alarm pheromone along with the time interval at which it is calculated are stored in LST. The stored information of LST will be used by the forwarding function in order to estimate the next hop probability. The Algorithm 6.1 gives the Alarm Pheromone Update function of LB-Termite; further based on the LST entries, the trail pheromone in the TPT will be updated and the TPT structure is given in Subsection 6.6.2.

Algorithm 6.1 : Alarm Pheromone Update Function of LB-Bat-Termite

---

Input: LST

Output: Neighbor nodes Alarm Pheromone and its Distance will be updated in LST

For every packet received from neighbor node

1. Extract the piggybacked queue information from the packet
2. Calculate the alarm pheromone for the neighbor node
3. Calculate the current distance of the neighbor node
4. Update the LST with the calculated information for neighbor node
5. Piggyback the queue status of the current node in the packet.

6.6.2 Trail Pheromone Table

The Trial Pheromone Table (TPT) can be used for maintaining the trail pheromone for each reachable destination node. Each node records the amount of destination trail pheromone on each neighbor link in TPT and it is analogous to routing table of traditional routing protocol such as AODV. TPT more or less holds the same elements as traditional routing table i.e, *neighbor node ID, destination node ID* and *routing metric (pheromone in LB-Termite and number of hops in AODV)*. In TPT, the rows represent neighbor nodes where as the columns are corresponding to the destination nodes.

When a node detects a new neighbor then it adds an additional row in the TPT; whenever a new destination is discovered then a new column will be added in the TPT and in both the cases the trial pheromone is initialized to *PH_INITIAL*. While updating the trail pheromone of neighbor nodes in TPT, each node has to refer to the LST for finding the *alarm pheromones, current and previous distances* of the neighbor nodes in order to reflect the mobility and load information. Since each node movement is reflected in TPT then the neighbor node with low mobility will have higher trail pheromone entry; whereas the neighbor node with high mobility will have less trail pheromone entry. Further, load on each neighbor will also be reflected in TPT based on the alarm pheromones of each neighbor node and the Algorithm 6.2 gives the

Trail Pheromone Update function of LB-Bat-Termite. Both TPT and LST of each node can be updated during the route discovery and maintenance phases of LB-Bat-Termite as explained in the following subsections 6.6.3 and 6.6.4 respectively.

Algorithm 6.2 Trail Pheromone Update Function of LB-Bat-Termite

---

Input: TPT

Output: Trail pheromone for the source and the destination node will be updated in the TPT

For every data and RREQ (Route Request) packet received

1. ***If*** data packet
2. ***for*** all the neighbors
3. Decay the pheromone
4. ***end for***
5. ***End if***
6. Calculate the trail pheromone for the neighbor from which the data packet received.
7. Update the calculated Trail Pheromone in TPT.
8. Piggyback the queue status of the current node in the packet.

6.6.3 Route Discovery Phase of LB-Bat-Termite

When a mobile node does not find the destination trail pheromone entry in its TPT, then it initiates the route discovery phase by issuing Route Request (RREQ) packets. Mobile nodes upon receiving RREQ packets, first updates the TPT as well as LST tables then piggy backs its current queue status in the RREQ packets. A mobile node replies to RREQ packet with Route Reply (RREP) packet only if it is the intended destination node or if it finds the trail pheromone entry for the requested destination node in its TPT and then discards the RREQ packet. Otherwise the RREQ packet can be forwarded towards the randomly chosen neighbor node except the node from which it has arrived on. Further, if a node cannot forward the RREQ packet then it will drop the corresponding packet. On receiving the RREP packet, the source node forwards the cached data packets according to the forwarding function and the Algorithm 6.3 describes the Route Setup Phase of LB-Termite.

This selected path will be further reinforced by the data packets over a period of time during the route maintenance phase and the details of route maintenance phase of LB-Termite are given in the next section.

Algorithm 6.3 : Route Setup Phase of LB-Bat-Termite

Input: TPT and LST

Output: Trail pheromone and local status will be updated in the TPT and LST for each neighbors

1. ***While*** node consists of trail pheromone to the intended destination
2. At each intermediate node on reception of RREQ packet
3. Update the Alarm and Trail Pheromone in LST and TPT
4. ***If*** no neighbor exists
5. Discard the RREQ
6. ***else***
7. Piggyback the queue status of the current node in the packet.
8. Randomly choose the next hop neighbor and forward the RREQ packet
9. ***End If***
10. ***End While***
11. Unicast the RREP packet to the source
12. Discard the RREQ packet

6.6.4 Route Maintenance Phase of LB-Bat-Termite

LB-Bat-Termite algorithm has a new route maintenance mechanism which issues a single hop Route Error packet (RERR) in order to notify the path loss. When a node is left with empty neighbor node list then it sends a single hop RERR packet to the immediate neighbor node from which data packets have arrived. This single hop RERR packet causes the recipient node in removing RERR source from its TPT for the particular destination node and thus drops the RERR packet. The recipient node then finds an alternative route to the intended destination node from its remaining neighbor node list. This concludes the working principles of LB-Bat-Termite and Algorithm 6.4 discusses the Route Maintenance Phase and Algorithm 6.5 gives the Data Session Phase of LB-Bat-Termite algorithm.

Algorithm 6.4 : Route Repair Phase of LB-Bat-Termite

Input: Local Status Table (LST) and Trail Pheromone Table (TPT)

Output: 1. Neighbor Nodes Alarm Pheromone and its Distance will be updated in LST

2. Trail Pheromone Entry for the Neighbor Node Will be Removed from TPT

1. ***For*** every RERR packet received from neighbor node
2. Update the Alarm Pheromone in LST
3. ***If*** source node
4. Remove the Trail Pheromone entry of the neighbor node from TPT
5. Store the data packet
6. Send RREQ to find the new Trail pheromone source to the intended destination
7. Trigger the Route Setup phase .
8. Drop the RERR packet
9. ***Else***
10. Remove the Trail Pheromone entry of the neighbor node from TPT
11. Update α value in the data packet based on the path utility
12. Calculate the next hop probability for all the neighbors which lead to the destination
13. Choose the highest probability neighbor and forward the data packet
14. Drop the RERR Packet
15. ***End if***

Algorithm 6.5 : Data Session Phase of LB-Bat-Termite

Input: Trail Pheromone Table(TPT) and Local Status Table (LST)

Output: Reinforcement of the selected link in TPT and updated LST table

1. ***While*** node = destination node
2. Update the Alarm and Trail Pheromone in LST and TPT
3. Update α value in the data packet based on the path utility
4. Calculate the next hop probability for all the neighbors which lead to the destination
5. Choose the highest probability neighbor and forward the data packet
6. ***if*** no neighbor exists that lead to the destination
7. Send RERR to immediate neighbor node from which data packet is received
8. ***End if***
9. ***End While***

6.7 Performance Evaluation of LB-Bat-Termite

The proposed LB-Bat-Termite algorithm (here after called as LB-Bat-T for plotting the graphs) is simulated in NS-2 and the results are compared with both bio-inspired (SARA (Correia, F and Vazao, T 2010) and basic Termite Algorithm (BT) (Roth M and Wicker S 2004a; 2004b)) and non bio-inspired (Ad hoc On Demand Distance Vector (AODV) (C.M, Perkins and E.M, Royer 1999)) routing protocols for its performance evaluation. The protocols are compared with respect to QoS metrics such as *throughput, total packet drops, end-to-end delay* and *control packet overhead*.

Simulation Environment:

Each simulation run was carried out for 100 sec and Random Waypoint Mobility model is considered for the simulation. At the MAC layer, IEEE 802.11 DCF is used and File Transfer Protocol (FTP) is used as a traffic type with the packet payload of 1500 bytes. Ten traffic source and destination nodes are considered for all the simulation runs and each Transmission Control Protocol (TCP) source destination pairs are selected randomly for each simulation run. Table 6.1 shows the different parameter values used for evaluating the performance of LB-Bat-Termite algorithm. For SARA protocol, the default values are considered throughout the experiment.

Table 6.1: Parameter Values used for LB-Bat-Termite

Sl.No	Parameter	Value
1	*PH_INITIAL*	1
2	*PH_CEILING*	50
3	*PH_FLOOR*	0
4	τ	0.105
5	F	3

6.7.1 Experimental Setup

A rigorous simulation study was carried out under two different scenarios to see how the proposed algorithm behaves under *scalability* and the *node velocity* factor.

- **Scenario A**

  In this scenario *scalability* of the proposed algorithms is evaluated by considering two test areas analogous to academic campus of size 500m x 500m and 1000m x 1000m. Under each simulation area, the algorithms are tested for different node density of 100, 200, 400 and 600 nodes. The node velocity is kept constant at 2m/s when a normal walking scenario of a person is considered and no pause time is given to any mobile node.

- **Scenario B**

  In this scenario behavior of the proposed algorithms under different set of *node velocity* is studied for performance evaluation. Mobile nodes with varying node velocity (2, 4, 6, 8 and 10 m/s) and with different node density (100, 200, 400 and 600 nodes) are considered for each simulation run. Test area analogous to academic campus of size 1000m x1000m is considered and no pause time is given to any mobile node.

6.7.2 Scenario A

Throughput Analysis

Figures 6.4 and 6.5 show the throughput comparison of LB-Bat-Termite, Termite, SARA and

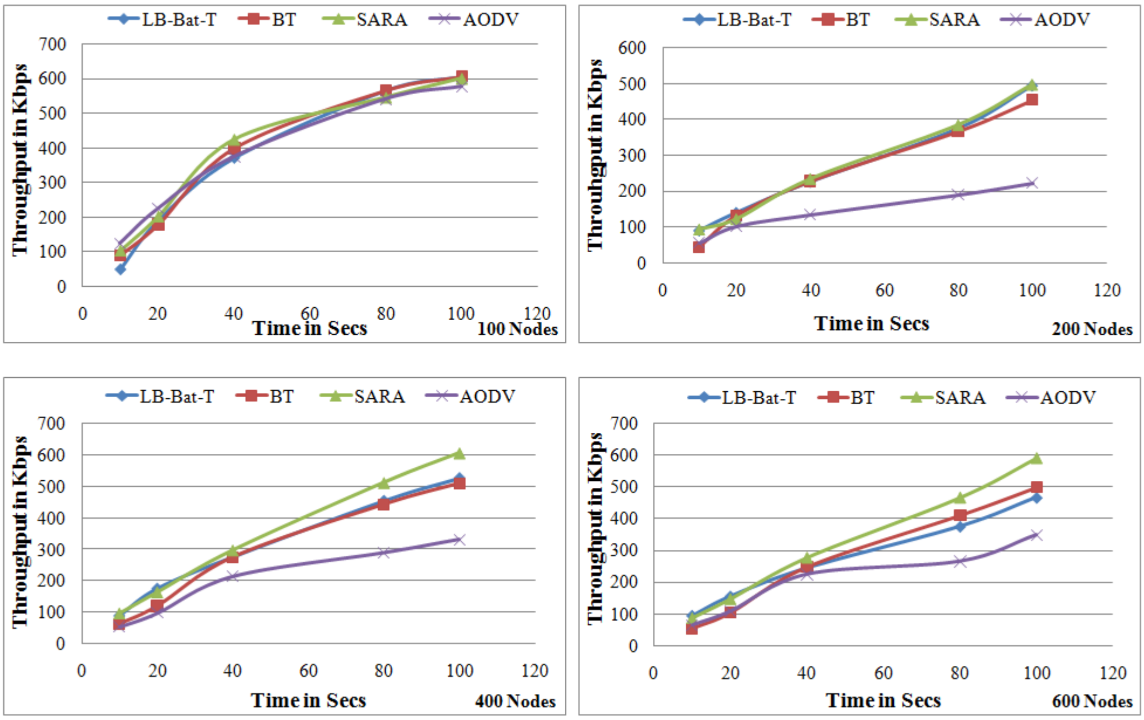


Figure 6.4: Throughput Vs Simulation Time in Sec (500m x 500m Sim. Area)

AODV protocols. LB-Bat-Termite shows considerable performance gain over SARA, Termite and AODV algorithms under low node density conditions (up to 200 nodes) as bottleneck links are avoided in destination paths. LB-Bat-Termite algorithm makes sure that no links are dominant and gives an equal preference to all the destination routes. LB-Bat-Termites performance degrades as node density increases when compared to SARA, Termite and AODV algorithms. The primary reason for such bahavior of LB-Bat-Termite is, LB-Bat-Termite suffers from additional control packets in terms of echo_location packets in order to manage all possible paths to the destination. Thus creates extra burden on the mobile nodes to process echo_location packets; hence mobile nodes spends most of its time in managing echo_location packets than data transfer. SARA on the other hand, with its simple routing techniques dominates LB-Bat-

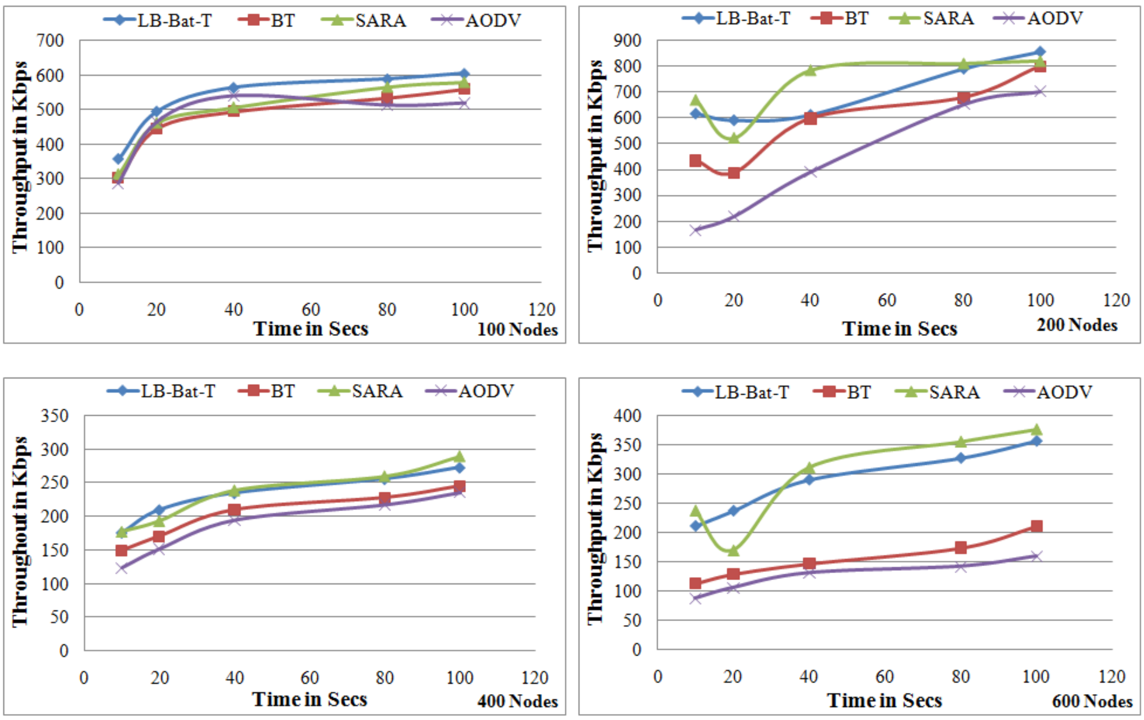


Figure 6.5: Throughput Vs Simulation Time in Sec (1000m x 1000m Sim. Area)

Termite, Termite and AODV protocols. As Termite algorithm exclusively chooses the single dominant link, the dominant link will get congested gradually resulting in low throughput. Also as the pheromone in the less dominant link decays soon, its entry will be removed from the TPT closing the alternate routes to the destination. Thus, nodes will be left with single dominant link sooner than expected. All these reasons causes low throughput of Termite algorithm. AODV is poor in load balancing as it chooses a single path to the destination and depends on this single path throughout the data transmission. AODV selects the path based on the minimum hop and the path chosen may be a shorter path but it may not be a resource rich path. The other nodes also chose the same path since it is shorter, thus the shortest path gets congested over a period of time resulting in low throughput.

Total Packet Drops Analysis

Figures 6.6 and 6.7 show the total packet drops of all routing protocols considered for the performance evaluation. When node density is 100 nodes, all the competing protocols shows very good performance as packet collision will be less. But as node density increases, the packet drops will also increase as chances of collision is more. In both the test cases (500m x 500m and 1000m x 1000m), AODV shows poor performance due to congestion and improper route management. Even though the Termite Algorithm is better than AODV but it is not as good as SARA and LB-Bat-Termite. LB-Bat-Termite with its efficient route maintenance technique show less packet drops than other competing protocols when node density is low. As node density increases, the increased echo_location packets in the network causes collision with the data packets thus increases the total packet drops when compared to SARA, BT and AODV. SARA shows very good performance with its new route repair technique, deep search procedure.

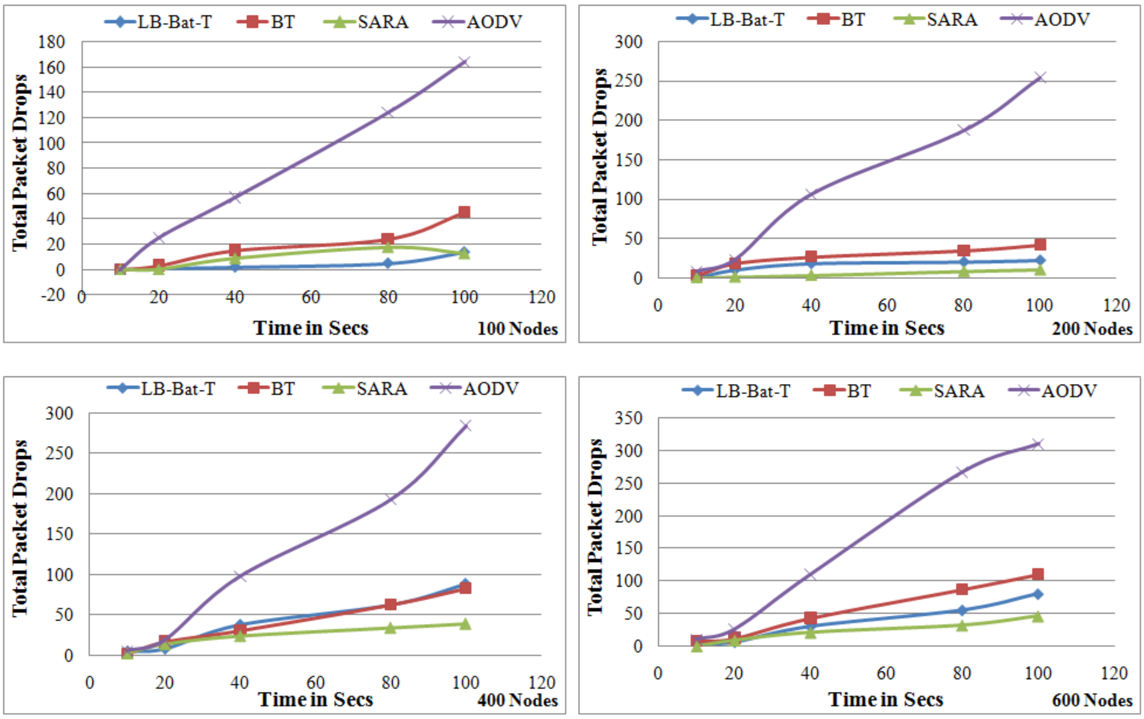


Figure 6.6: Total Packet Drops Vs Simulation Time in Sec (500m x 500m Sim. Area)

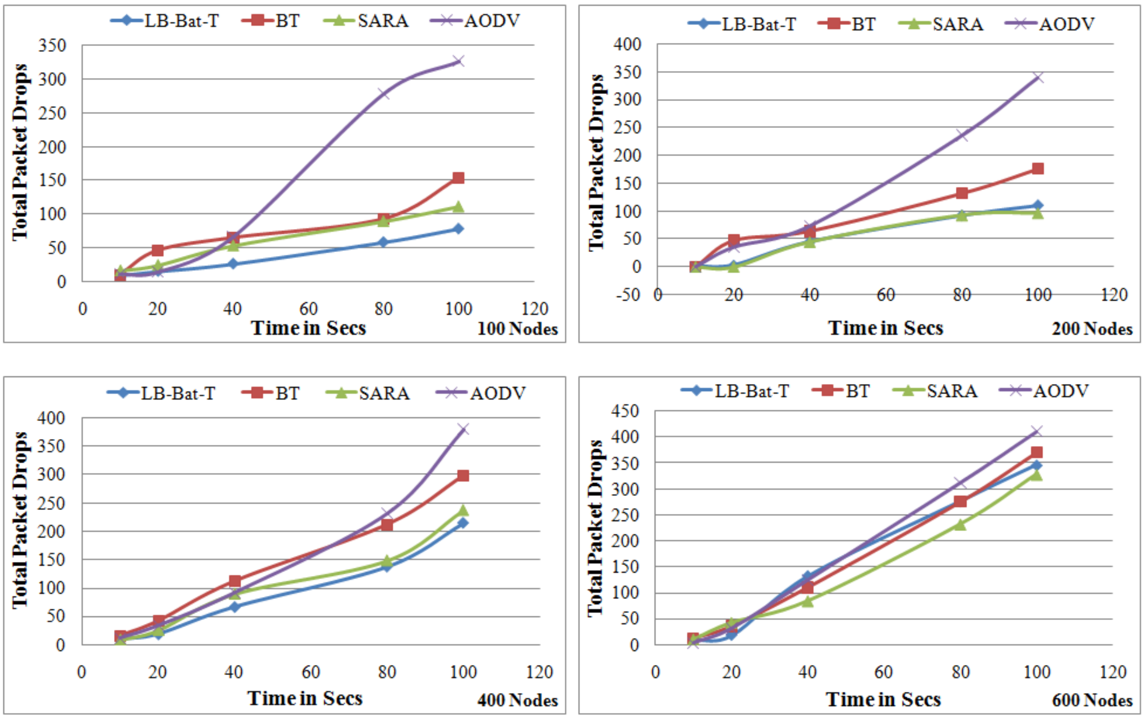


Figure 6.7: Total Packet Drops Vs Simulation Time in Sec (1000m x 1000m Sim. Area)

End-to-End Delay Analysis

Figures 6.8 and 6.9 show the end-to-end delay analysis of all the four protocols considered for the performance evaluation. It is observed that as node density increases AODV shows high end-to-end delay because it maintains a single path to the destination and also due to more number of packet drops. Termite shows better results than AODV as it is multipath routing protocol. Packets find different options to reach destination there by reducing and end-to-end delay. As mobile nodes in LB-Bat-Termite spends most of its time in responding to echo_location packets, data packets takes longer time to reach the destination when compared SARA but on the other hand it shows considerable improvements when compared to BT and AODV. SARA with its optimized routing process achieves less end-to-end delay than LB-Bat-Termite, Termite as well as AODV protocols.

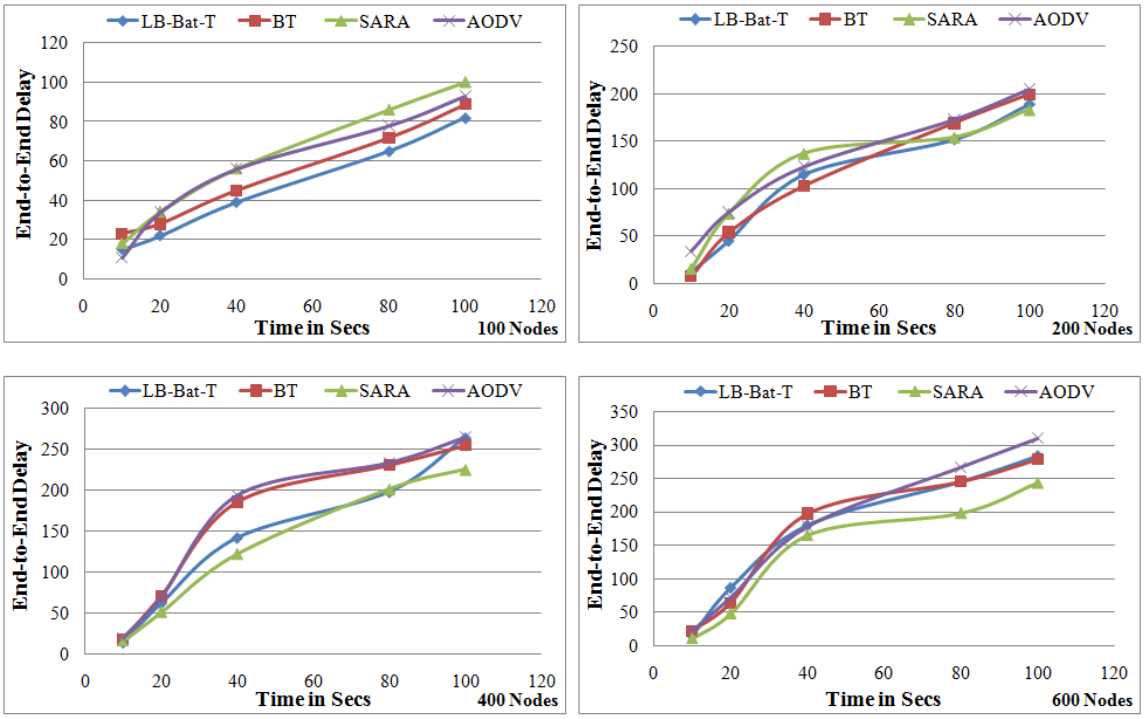


Figure 6.8: End-to-End Delay Vs Simulation Time in Sec (500m x 500m Sim. Area)

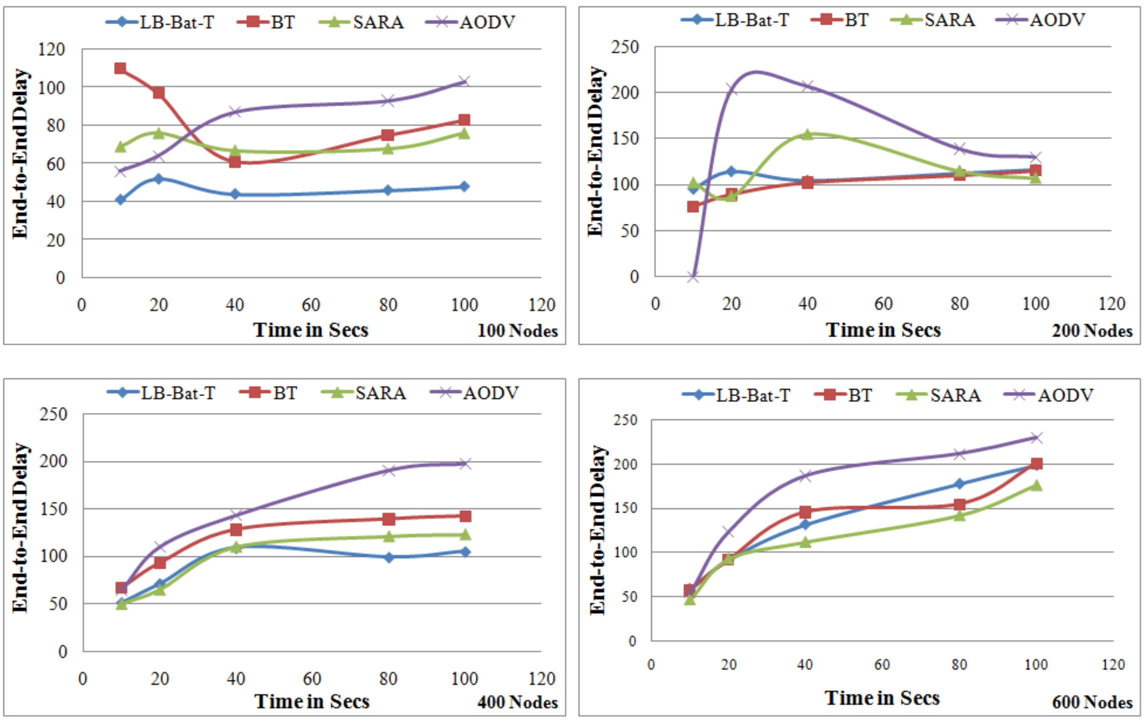


Figure 6.9: End-to-End Delay Vs Simulation Time in Sec (1000m x 1000m Sim. Area)

Control Packet Overhead Analysis

Figures 6.10 and 6.11 show the control packet overhead of all the four competing protocols

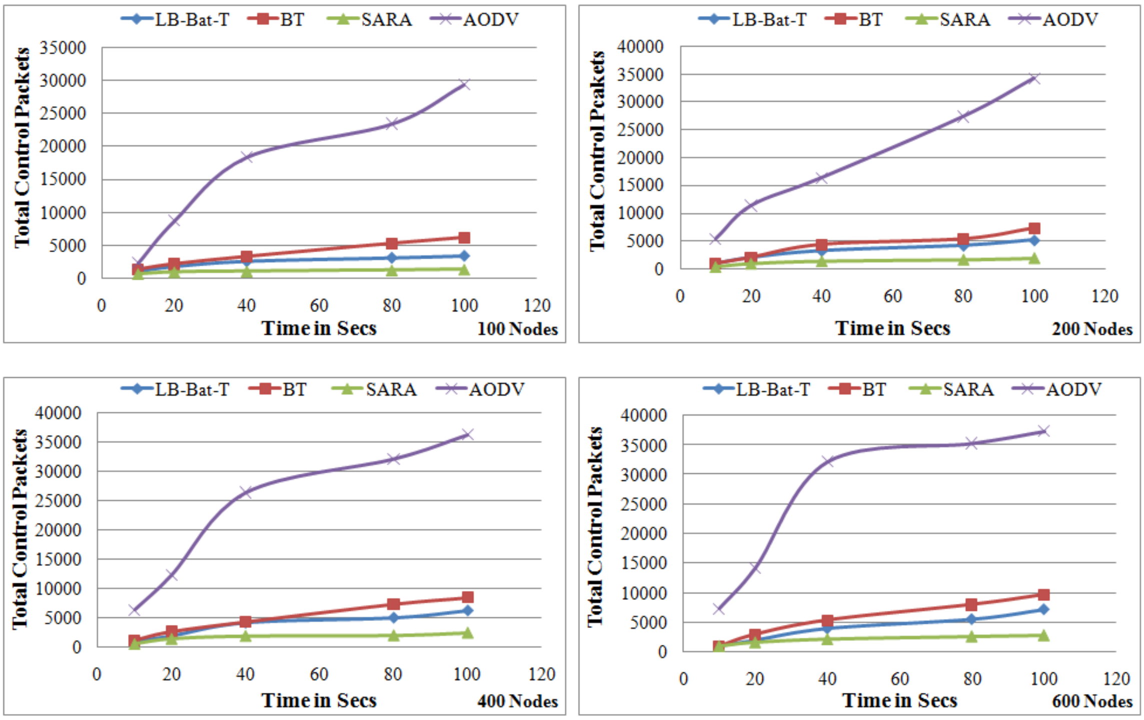


Figure 6.10: Total Control Packets Vs Simulation Time in Sec (500m x 500m Sim. Area)

used in this performance evaluation. AODV produces large amount of control packets due to its poor route setup and maintenance strategy. Also as chances of link breakup are high in AODV, route maintenance is triggered every now and then producing more control packets. Termite algorithm, due to its multipath environment and quick route discovery and maintenance strategy, produces less control packet overhead than AODV. LB-Bat-Termite algorithm with its efficient route maintenance and load balancing techniques maintains less control packet overhead than AODV algorithms but LB-Bat-Termite in an effort to maintain all possible paths to the destination produces almost same amount of control packets when compared to BT. On the other hand, SARA with its efficient route discovery and route maintenance procedure dominates all the protocols in both the test cases (500m x 500m and 1000m x 1000m).

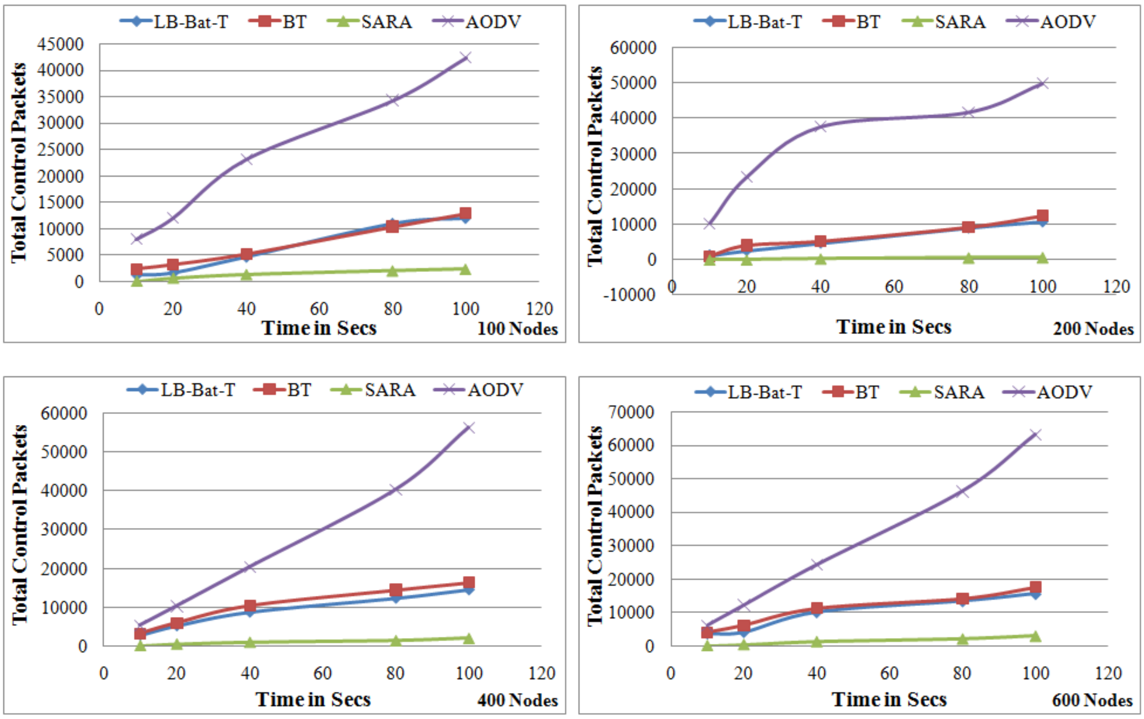


Figure 6.11: Total Control Packets Vs Simulation Time in Sec (1000m x 1000m Sim. Area)

6.7.3 Scenario B

Throughput Analysis

Figure 6.12 show the throughput of all the four protocols under performance analysis and it is clearly observed that LB-Bat-Termite algorithm shows good performance under low mobility and low node density conditions but as node mobility increases LB-Bat-Termite finds it difficult to manage all possible routes to the destination; hence finds low throughput when compared to SARA, BT and AODV. SARA performs better than Termite and AODV protocols with its efficient route management technique but Termite and AODV protocols fail to handle the mobility and thus both Termite and AODV protocols offer low throughput.

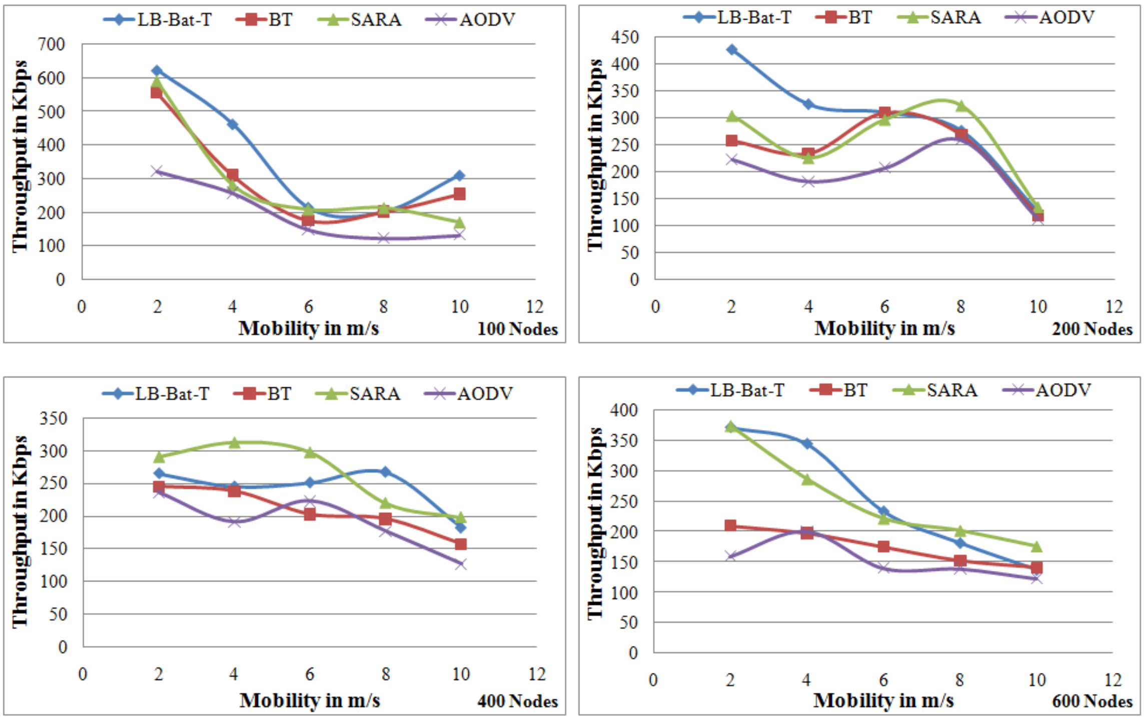


Figure 6.12: Throughput Vs Node Velocity

Total Packet Drops Analysis

Figures 6.13 show the total packet drops of the four competing protocols considered for performance evaluation and the graph shows that LB-Bat-Termite provides better results when compared to SARA, Termite and AODV protocols under low mobility conditions. As node mobility increases LB-Bat-Termite fails to handle multiple routes and thus finds more packet drops when compared to SARA. And for the same reason, LB-Bat-Termite algorithm offers lower end-to-end delay than SARA, Termite and AODV under low mobility and low node density conditions and offers high end-to-end delay under high mobility and high node density conditions as shown in Figure 6.14.

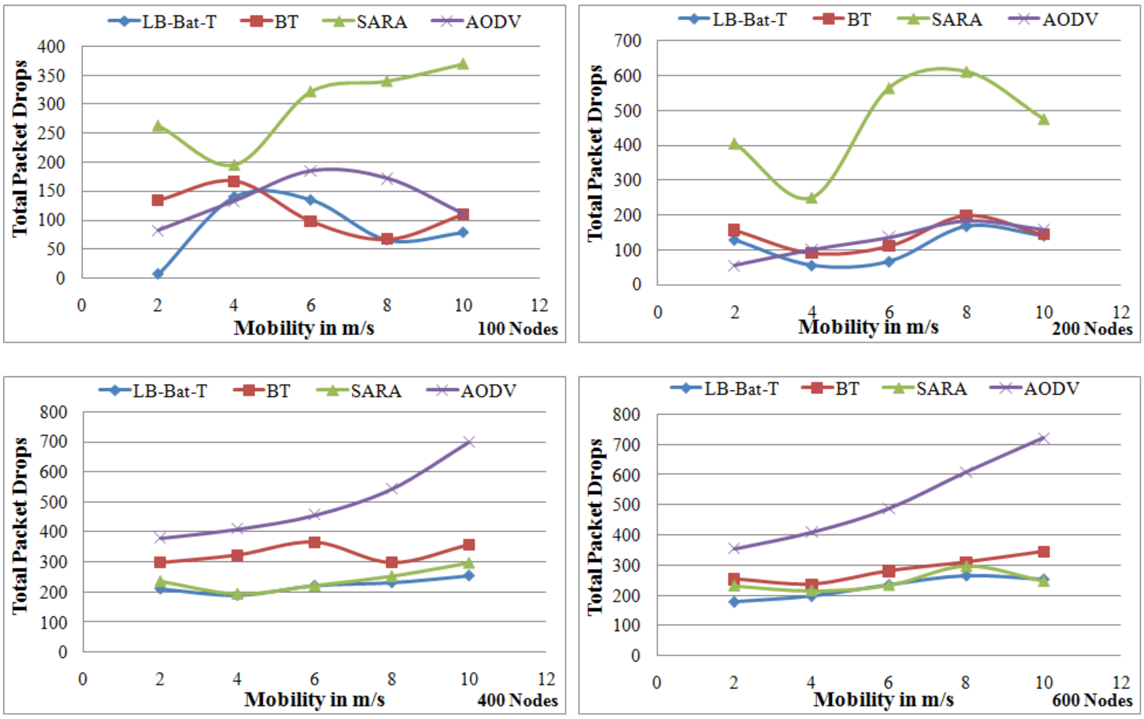


Figure 6.13: Total Packet Drops Vs Node Velocity

End-to-End Delay Analysis

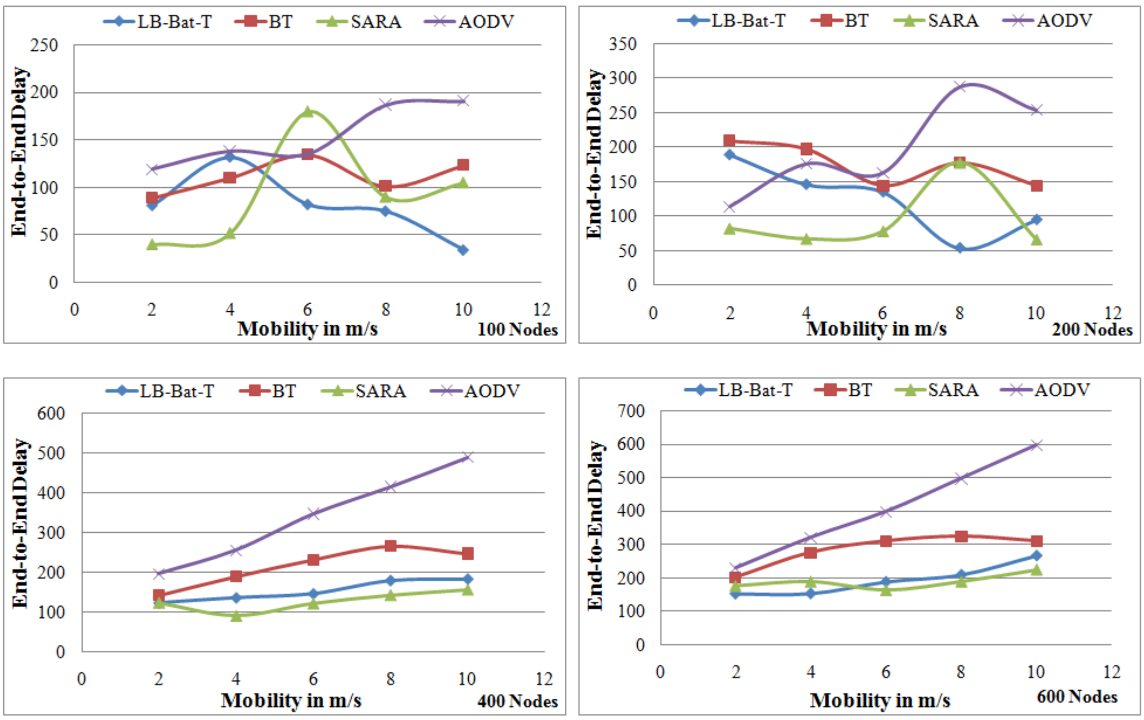


Figure 6.14: End-to-End Delay Vs Node Velocity

Control Packet Overhead Analysis

Figure 6.15 shows the control packet overhead of all the four protocols considered for the per-

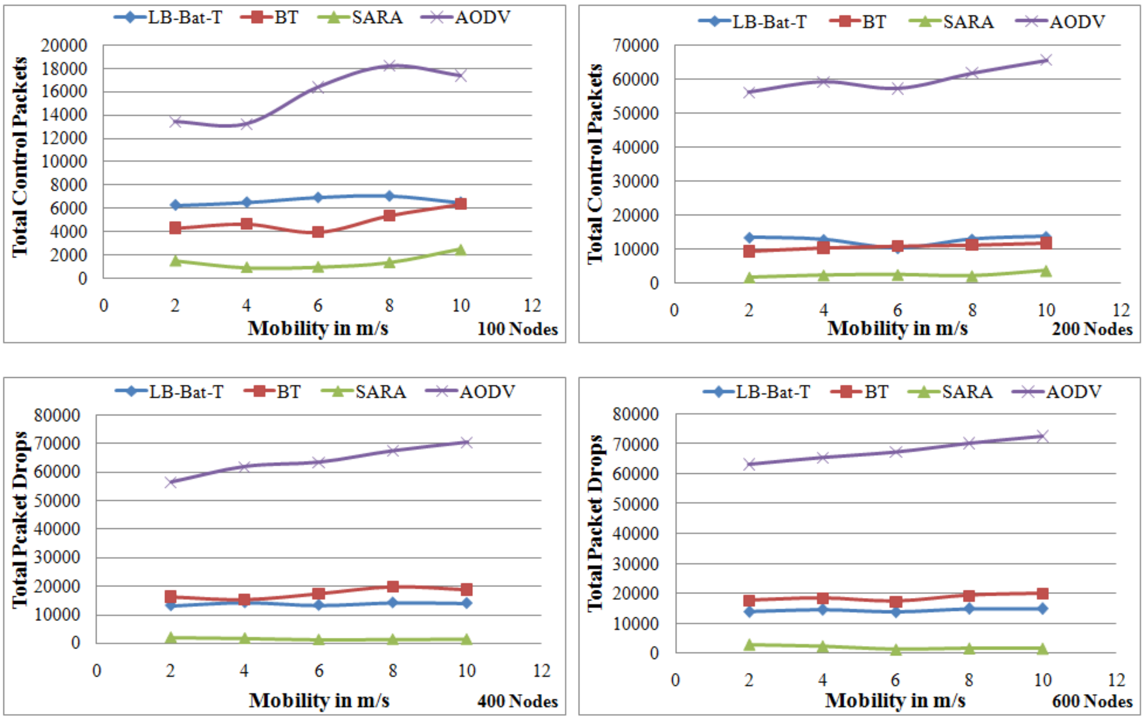


Figure 6.15: Total Control Packets Vs Node Velocity

formance evaluation. LB-Bat-Termite produces slightly higher control packets when compared to SARA mainly due to its additional echo_location packets but produces considerably less control packets when compared to BT and AODV. SARA outperforms LB-Bat-Termite, Termite and AODV protocols by producing very less control packets than other competing protocols even under high mobility. The reason lies in the fact that, SARA uses optimized routing process with three different optimizing techniques namely, Controlled Neighbor Broadcast (CNB), active sessions refresh by data packets and deep search procedure. It is observed that, as node mobility increases, the control packet also increases in case of AODV, but the other bio inspired protocols, LB-Bat-Termite, Termite and SARA algorithms, shows similar behavior even under high mobility mainly due to their efficient route maintenance and fast route recovery techniques.

6.8 Summary

The proposed hybrid bio-inspired routing algorithm with load balancing approach based on the social insect *Termites* and mammal *Bats* referred to as *Load Balanced-Bat-Termite (LB-Termite)* finds the *stable* and *reliable nodes* for the path and also it tries to mitigate the stagnation by introducing a new kind of pheromone called as alarm pheromone. *High route reliability, quick route discovery and repair & maintenance, high robustness with efficient management of multiple routes* and *stagnation avoidance* are some of the highlights of LB-Bat-Termite algorithm. LB-Bat-termite algorithm is implemented in NS-2 and the results are compared with other state-of-the-art bio inspired (SARA and Termite algorithms) and non bio-inspired (AODV) routing algorithms for its performance evaluation. It is observed that LB-Bat-Termite achieves 2% to 4% of increase in the evaluation metrics throughput, end-to-end delay and total packet drops when compared to SARA and achieves more than 4% increase in throughput, end-to-end delay, total packet drops and control packet overhead when compared to Termite and AODV algorithms when node density is low (upto 200).

But on the other hand, as node density increases (above 200), LB-Bat-Termite algorithm suffers from additional control packet overhead thereby deteriorating the performance. Further, the resultant graphs clearly depict that LB-Bat-Termite under high node density show poor performance when compared to SARA and Termite algorithms. Thus, even though LB-Bat-Termite algorithm solves stagnation problem of MA-Termite and Bat-Termite and solves the poor netwrok exploration and backup route maintenance problem of MA-Termite, it suffers from additional control packets overhead inorder to maintain the active paths towards the destination nodes. Hence, inorder to overcome the control packet overhead problem of LB-Bat-Termite, a novel heuristic load balanced bio-inspired routing algorithm referred to as *Load Balanced-Termite* will be discussed in the next chapter.

Chapter 7

Load Balanced-Termite:

A Novel Load Aware Bio-Inspired Routing

This chapter discusses the design and development of a novel heuristic bio-inspired routing protocol with load balancing approach referred to as *Load Balanced-Termite (LB-Termite)* for MANETs. Apart from the salient features of only MA-Termite algorithm, LB-Termite will also consider the *nodal load* in order to achieve the *load balancing among the available multiple paths* and thereby *mitigating the stagnation problem* of both MA-Termite and Bat-Termite algorithms. The details of LB-Termite are discussed in the following sections.

7.1 Motivation for Development of LB-Termite Algorithm

LB-Bat-Termite algorithm as discussed in the previous chapter suffers from additional control packet overhead which is caused by its echo_location packets. LB-Bat-Termite algorithm uses echolocation packets in order to over come the poor network exploration and improper back up maintenance problem of MA-Termite algorithm. Using echolocation packets LB-Bat-Termite will maintain all possible paths to the destination node in its TPT. Due to load balancing across multiple paths, LB-Bat-Termite algorithm will not allow trail pheromone of primary and secondary paths to fall below PH_FLOOR in TPT. Thus, at any point of time only two paths (Primary and Secondary) among all possible paths to the destination will be used for data transfer. To illustrate this, let us consider a n path system as shown in Figure 7.1(a) where source node x wants to communicate with destination node y.

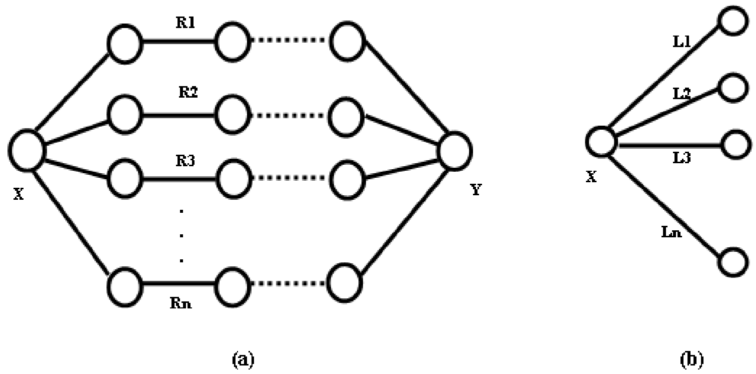


Figure 7.1: n path system for studying the asymptotic behavior of LB-Bat-Termite

To reach the destination node *Y*, the source node should have *n* independent paths *R1*, *R2*,...,*Rn* and each path consists of a series of relay nodes connecting source node *X* and the destination node *Y* with different characteristics such as *bandwidth, moblity, etc.* Figure 7.1(b) shows the subset of Figure 7.1(a) where the source node *X* has many links such as *L1, L2, L3, ... , Ln* to reach the destination node *Y*. Suppose node *X* chooses link *L2*(primary link) to reach the destination during the route setup phase and let link *L3* be the secondary link among the available links to the destination node *Y*. LB-Bat-Termite with its load balancing approach efficiently manges the load across primary and secondary links and thus the pheromone on these links will never fall below *PH_FLOOR* in node *X's* TPT; on the other hand pheromone on the other links (L1, L4,...,Ln) decays gradually over a period of time and reaches PH_FLOOR. Under such circumstances the souce node *X* generates echo_location packets on links (L1, L4,...,Ln) in order find the current status of the its neighbor nodes and thereby maintaining the back up route to the destination node *Y*. Eventhough these back up links are maintained and refreshed at regular time intervals in the TPT, chances of using these links for data transfer is very less as primary and seconadary links will dominate all other links. Further, refreshing of these backup links at regular intervals generates additional echo_location packets thereby increasing the network load.

The nodes in LB-Bat-Termite tries to maintain all possible routes to the destination and thus spends most of its time in refreshing TPT than data transfer. Further, these additional echo_location packets increases the load on the network and inturn decreases the network performance. Hence, the proposed LB-Termite will address this additional control packet overhead issue and the details of LB-Termite are discussed in the following sections.

7.2 Working Principle of LB-Termite Algorithm

In order to reduce the additional control packet overhead problem of LB-Bat-Termite algorithm, the proposed LB-Termite algorithm drops the echolocation feature of LB-Bat-Termite and retains rest of LB-Bat-Termite's features. LB-Termite uses eqn. 4.1 for trail pheromone update-decay and uses eqn. 6.1 in order to update the alarm pheromone over the link; Further, LB-Termite uses eqns. 6.2 and 6.3 as forwarding functions. LB-Termite uses Algorithm 6.1 and 6.2 for updating Alarm and Trail pheromone in the TPT and uses Algorithms 6.3, 6.4 and 6.5 for route discovery, route maintenance and data session phase.

7.3 Performance Evaluation of LB-Termite

The proposed LB-Termite algorithm (here after called as LB-T for plotting the graphs) is simulated in NS-2 and the results are compared with both bio-inspired (SARA (Correia, F and Vazao, T 2010) and basic Termite Algorithm (BT) (Roth M and Wicker S 2004a; 2004b)) and non bio-inspired (Ad hoc On Demand Distance Vector (AODV) (C.M, Perkins and E.M, Royer 1999)) routing protocols for its performance evaluation. The protocols are compared with respect to QoS metrics such as *throughput, total packet drops, end-to-end delay* and *control packet overhead*.

Simulation Environment:

Each simulation run was carried out for 100 sec and Random Waypoint Mobility model is considered for the simulation. At the MAC layer, IEEE 802.11 DCF is used and File Transfer Protocol (FTP) is used as a traffic type with the packet payload of 1500 bytes. Ten traffic source and destination nodes are considered for all the simulation runs and each Transmission Control Protocol (TCP) source destination pairs are selected randomly for each simulation run. Table 7.1 shows the different parameter values used for evaluating the performance of LB-Termite algorithm. For SARA protocol, the default values are considered throughout the experiment.

7.3.1 Experimental Setup

A rigorous simulation study was carried out under two different scenarios to see how the proposed algorithm behaves under *scalability* and the *node velocity* factor.

Table 7.1: Parameter Values used for LB-Termite

Sl.No	Parameter	Value
1	*PH_INITIAL*	1
2	*PH_CEILING*	50
3	*PH_FLOOR*	0
4	τ	0.105
5	*F*	3

- **Scenario A**

  In this scenario *scalability* of the proposed algorithms is evaluated by considering two test areas analogous to academic campus of size 500m x 500m and 1000m x 1000m. Under each simulation area, the algorithms are tested for different node density of 100, 200, 400 and 600 nodes. The node velocity is kept constant at 2m/s when a normal walking scenario of a person is considered and no pause time is given to any mobile node.

- **Scenario B**

  In this scenario behavior of the proposed algorithms under different set of *node velocity* is studied for performance evaluation. Mobile nodes with varying node velocity (2, 4, 6, 8 and 10 m/s) and with different node density (100, 200, 400 and 600 nodes) are considered for each simulation run. Test area analogous to academic campus of size 1000m x1000m is considered and no pause time is given to any mobile node.

7.3.2 Scenario A

Throughput Analysis

Figures 7.2 and 7.3 show the throughput comparison of all four protocols (LB-Termite, Termite, SARA and AODV). LB-Termite shows considerable performance gain over SARA, Termite and AODV algorithms as bottleneck links are avoided in destination paths. LB-Termite algorithm makes sure that no links are dominant and gives an equal preference to all the destination routes. As node density increases, LB-Termite shows better performance than SARA, Termite and AODV algorithms. The primary reason for this is, LB-Termite finds stable nodes in the path in terms of the node mobility, thus makes the path reliable. The secondary reason is, LB-Termite manages all the available links by switching the traffic between links as and when the load on the link increases thus the pheromone over the links will not reach *PH_FLOOR* at all and thereby has multiple options to reach the destination. Whereas in SARA and Termite algorithms, the improper route management causes the pheromone over inactive links to reach *PH_FLOOR* and thus the pheromone entries will be removed from the pheromone table resulting in fewer options to reach the destination. SARA on the other hand, with its simple routing

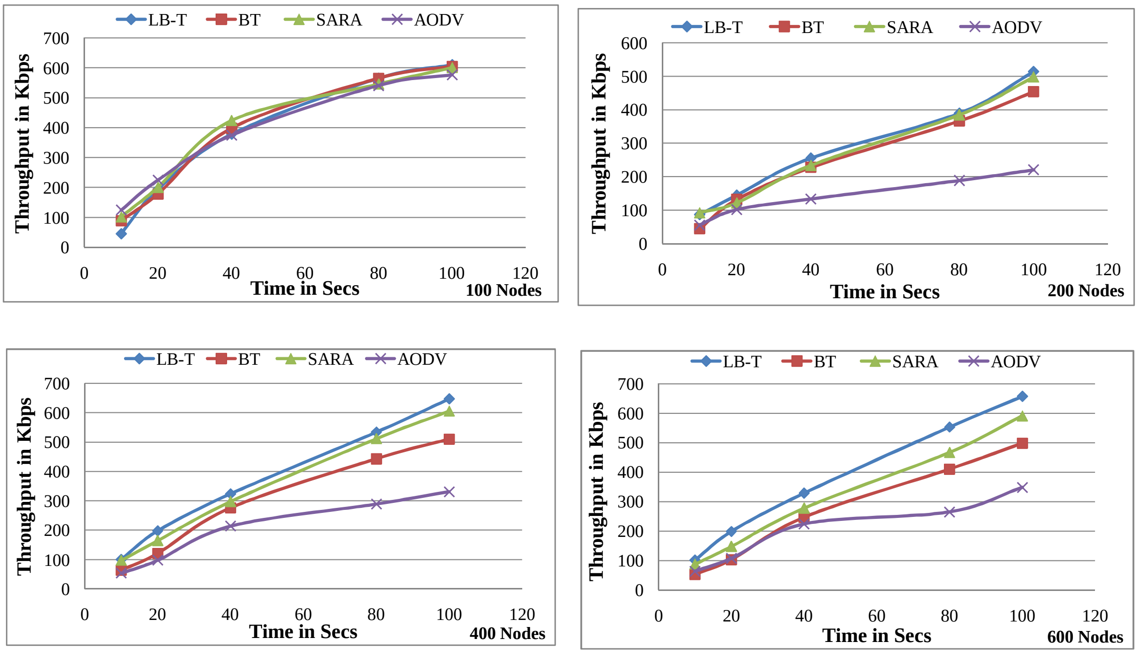


Figure 7.2: Throughput Vs Simulation Time in Sec (500m x 500m Sim. Area)

techniques dominates Termite and AODV protocols but as it takes more time to discover routes and lags in load balancing, shows low performance when compared to LB-Termite. As Termite algorithm exclusively chooses the single dominant link, the dominant link will get congested gradually resulting in low throughput. Also as the pheromone in the less dominant link decays soon, its entry will be removed from the TPT closing the alternate routes to the destination. Thus, nodes will be left with single dominant link sooner than expected. All these reasons causes low throughput of Termite algorithm. AODV is poor in load balancing as it chooses a single path to the destination and depends on this single path throughout the data transmission. AODV selects the path based on the minimum hop and the path chosen may be a shorter path but it may not be a resource rich path. The other nodes also chose the same path since it is shorter, thus the shortest path gets congested over a period of time resulting in low throughput.

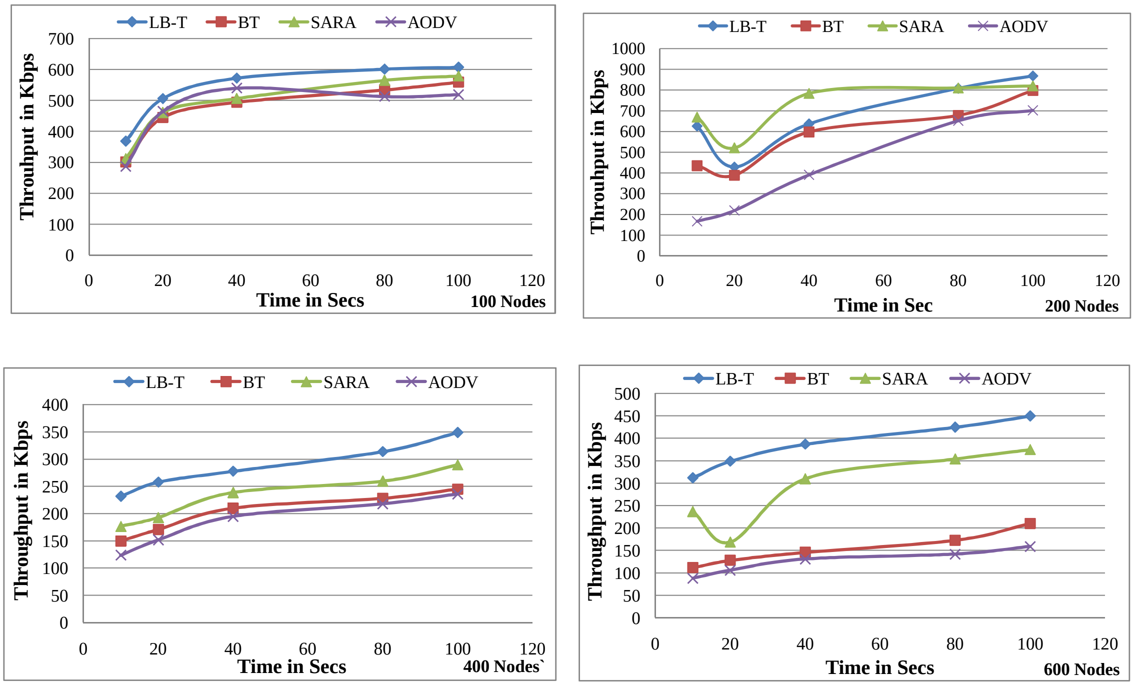


Figure 7.3: Throughput Vs Simulation Time in Sec (1000m x 1000m Sim. Area)

Total Packet Drops Analysis

Figures 7.4 and 7.5 show the total packet drops of all routing protocols considered for the per-

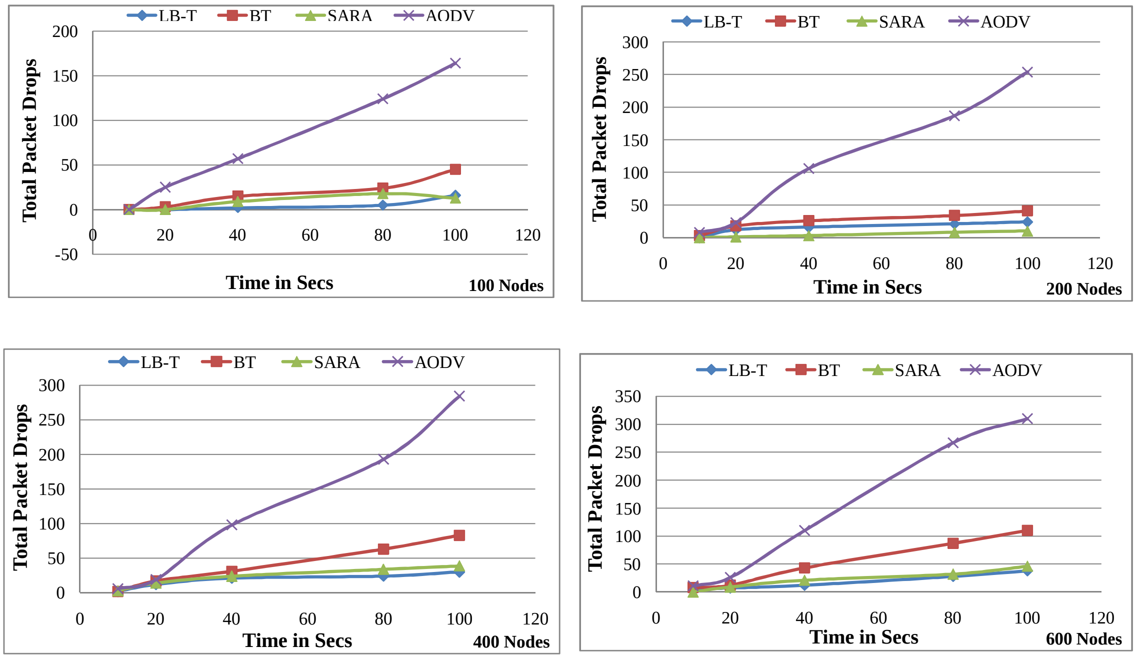


Figure 7.4: Total Packet Drops Vs Simulation Time in Sec (500m x 500m Sim. Area)

formance evaluation. When node density is 100 nodes, all the competing protocols shows very good performance as packet collision will be less. But as node density increases, the packet drops will also increase as chances of collision is more. In both the test cases (500m x 500m and 1000m x 1000m), AODV shows poor performance due to congestion and improper route management. Even though the Termite Algorithm is better than AODV but it is not as good as SARA and LB-Termite algorithms. SARA shows very good performance when scenario is 500m x 500m with its new route repair technique, deep search procedure. But as the network size increases it gives less performance when compared to LB-Termite algorithm. LB-Termite with its efficient route maintenance technique show less packet drops than other competing protocols in the second scenarios (1000m x 1000x simulation area). The reason lies in the fact that, as nodes are spread out in a large area, finding the distance between the nodes and guessing their movements becomes easy for LB-Termite whereas in case of 500m x 500m simulation area where nodes are dumped in a small simulation area, LB-Termite finds it difficult to manage the neighbor movements. Further, LB-Termite deals with the stagnation problem effectively and it will not allow the links to get congested thereby increasing the throughput while reducing the number of packet drops.

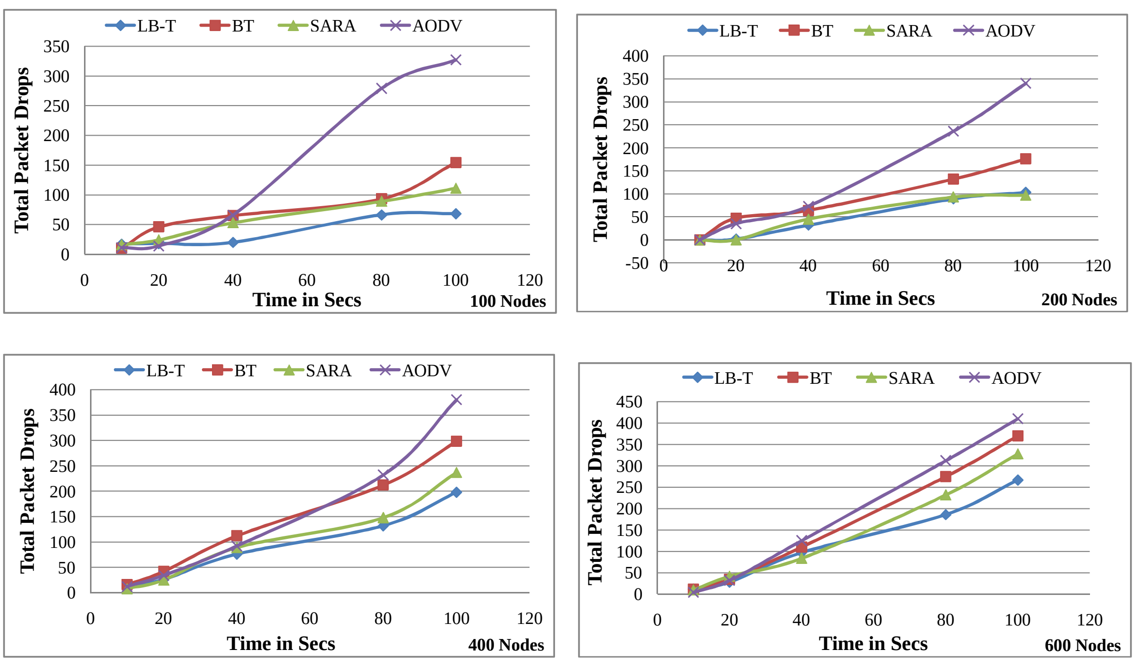


Figure 7.5: Total Packet Drops Vs Simulation Time in Sec (1000m x 1000m Sim. Area)

End-to-End Delay Analysis

Figures 7.6 and 7.7 show the end-to-end delay analysis of all the four protocols considered for

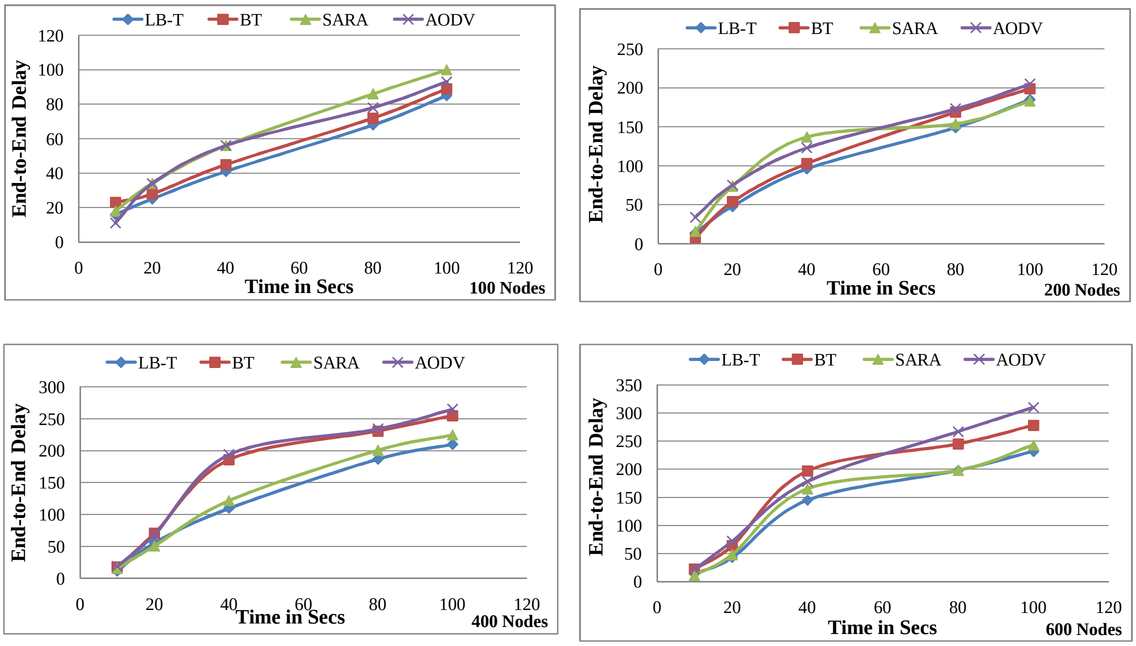


Figure 7.6: End-to-End Delay Vs Simulation Time in Sec (500m x 500m Sim. Area)

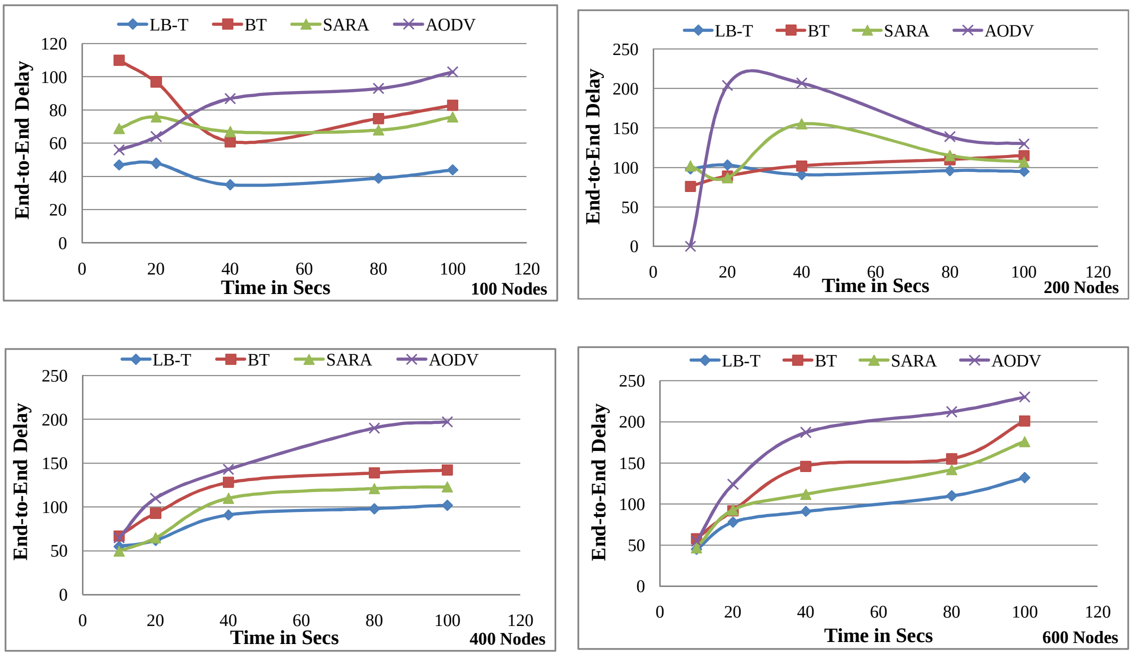


Figure 7.7: End-to-End Delay Vs Simulation Time in Sec (1000m x 1000m Sim. Area)

the performance evaluation. It is observed that as node density increases AODV shows high end-to-end delay because it maintains a single path to the destination and also due to more number of packet drops. Termite shows better results than AODV as it is multipath routing protocol. Packets find different options to reach destination there by reducing and end-to-end delay. SARA with its optimized routing process achieves less end-to-end delay than Termite as well as AODV protocols. When a route breaks up, instead of searching a new path from source and destination, SARA finds a new path between the two end nodes of the broken path, thus reduces route rediscovery time thereby reducing the end-to-end delay. As LB-Termite finds stable nodes in the path, path will be more reliable and chances of path breaks up will be less. Also LB-Termite will not allow any link to get congested as it deals with stagnation problem with the help of an alarm pheromone concept. Thus, LB-Termite maintains a steady data transmission between the source and the destination and thereby achieving lower value of end-to-end delay.

Control Packet Overhead Analysis

Figures 7.8 and 7.9 show the control packet overhead of all the four competing protocols used

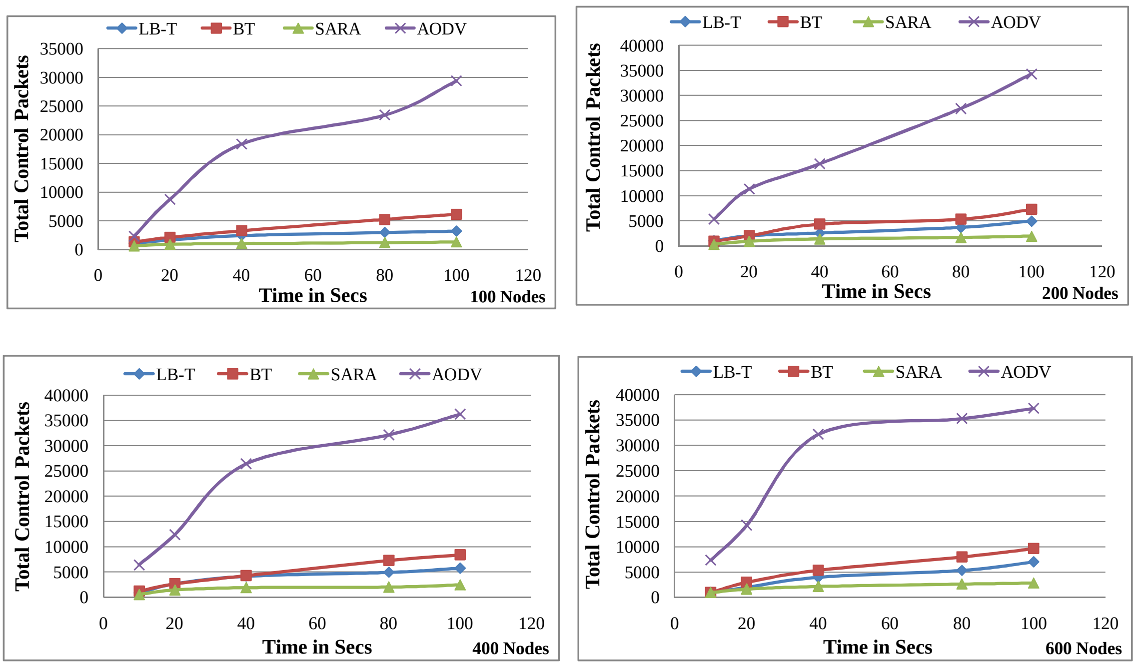


Figure 7.8: Control Packet Overhead Vs Simulation Time in Sec (500m x 500m Sim. Area)

in this performance evaluation. AODV produces large amount of control packets due to its poor route setup and maintenance strategy. Also as chances of link breakup are high in AODV, route maintenance is triggered every now and then producing more control packets. Termite algorithm, due to its multipath environment and quick route discovery and maintenance strat-

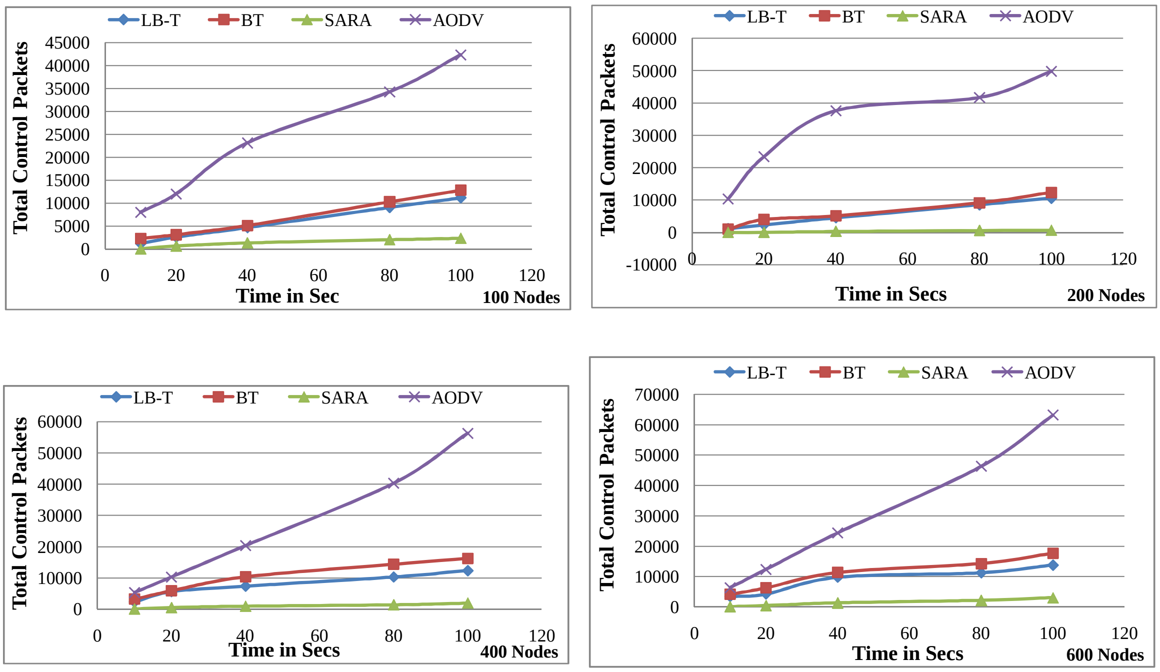


Figure 7.9: Control Packet Overhead Vs Simulation Time in Sec (1000m x 1000m Sim. Area)

egy, produces less control packet overhead than AODV. LB-Termite algorithm with its efficient route maintenance and load balancing techniques maintains less control packet overhead than Termite and AODV algorithms. On the other hand, SARA with its efficient route discovery and route maintenance procedure dominates all the protocols in both the test cases (500m x 500m and 1000m x 1000m).

7.3.3 Scenario B

Throughput Analysis

Figure 7.10 show the throughput of all the four protocols under performance analysis and it is clearly observed that LB-Termite algorithm dominates the other protocols. The graph shows that LB-Termite exhibit good performance at low mobility as the nodes will have multiple links to reach destination and it uses these available links efficiently by switching among them. But as mobility increases LB-Termites performance degrades as the node finds it difficult to maintain the multiple paths as mobility causes frequent link breakups. Thus nodes will not have many alternatives to switch among them resulting in low throughput. Under such conditions the LB-Termite behaves much like SARA algorithm. Reliable and load balanced paths, effective route maintenance and stagnation avoidance are the reasons for the high throughput of LB-Termite algorithm under different node velocity. Further, SARA performs better than Termite and AODV protocols with its efficient route management technique but not better than LB-Termite as it takes more time to discover routes. But Termite and AODV protocols fail to handle the mobility and thus both Termite and AODV protocols offer low throughput.

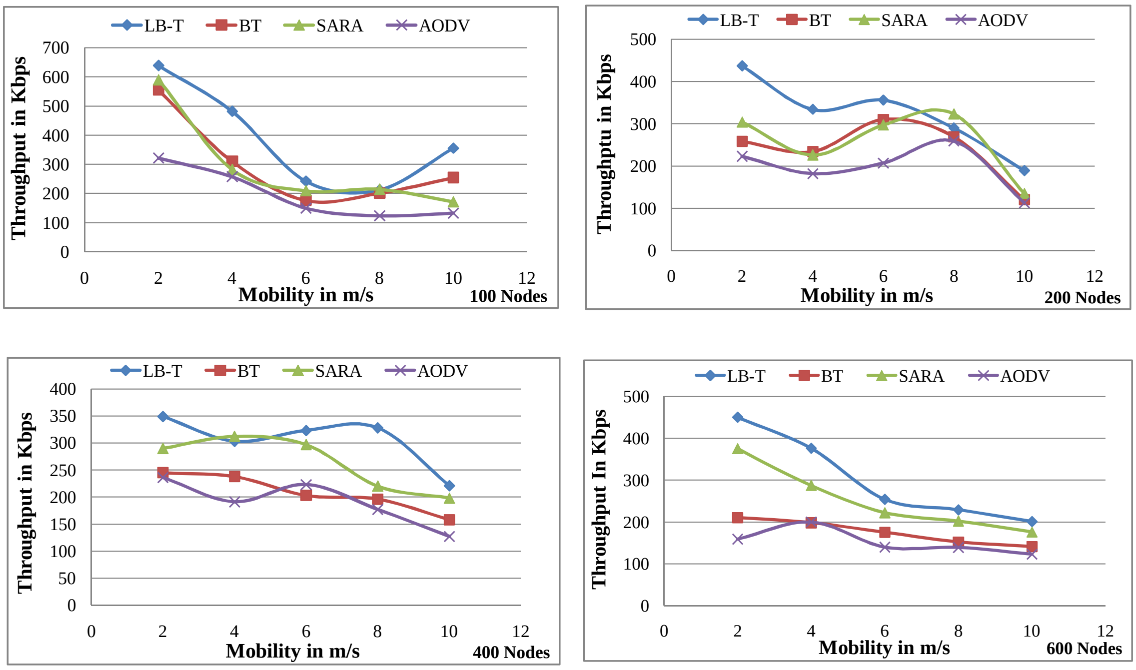


Figure 7.10: Throughput Vs Node Velocity

Total Packet Drops and End-to-End Delay Analysis

Figures 7.11 show the total packet drops of the four competing protocols considered for performance evaluation and the graph shows that LB-Termite provides better results when compared to SARA, Termite and AODV protocols since LB-Termite selects the reliable path with stable nodes between the source and destination nodes. The graphs shows clearly that LB-Termite handles the node mobility efficiently as LB-Termite reflects the current context of the network in the routing tables whereas SARA, Termite and AODV fails to handle the mobility of the nodes and drops more packets than LB-Termite. And for the same reason, LB-Termite algorithm offers lower end-to-end delay than SARA, Termite and AODV as shown in Figure 7.12.

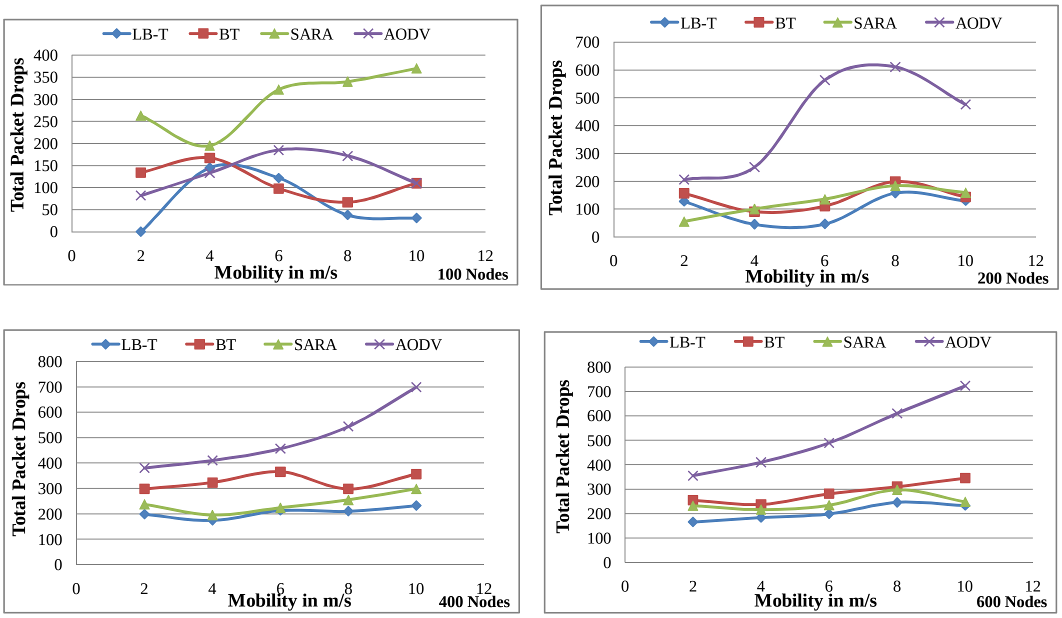


Figure 7.11: Total Packet Drops Vs Node Velocity

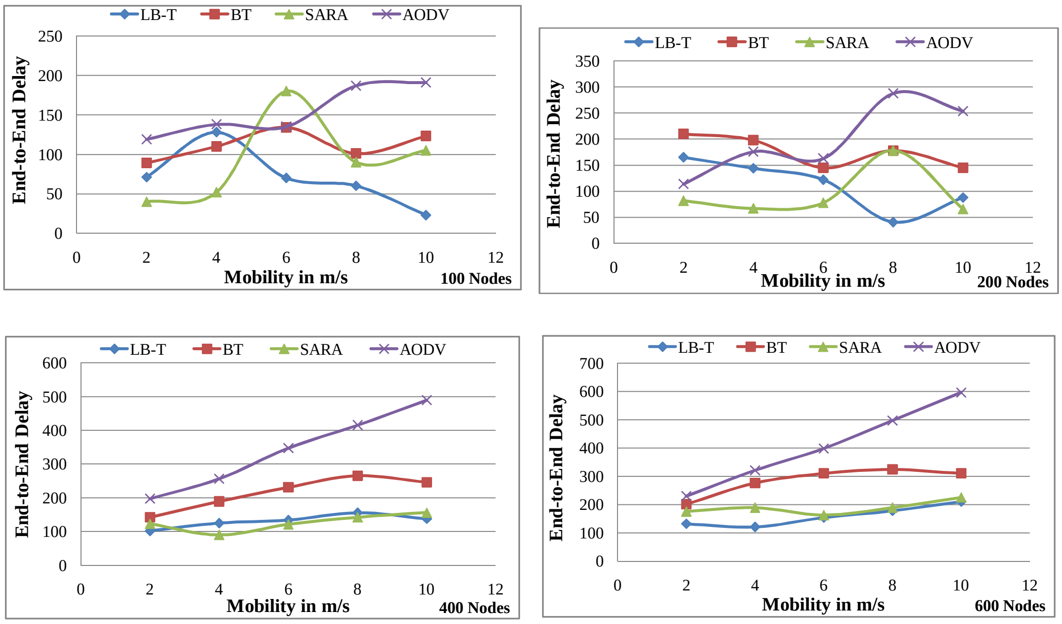


Figure 7.12: End-to-End Delay Vs Node Velocity

Control Packet Overhead Analysis

Figure 7.13 shows the control packet overhead of all the four protocols considered for the performance evaluation. LB-Termite produces slightly higher control packets than Termite algorithm mainly because of its new route repair technique which is triggered when path to the destination node is lost. SARA outperforms LB-Termite, Termite and AODV protocols by producing very less control packets than other competing protocols even under high mobility. The reason lies in the fact that, SARA uses optimized routing process with three different optimizing techniques namely, Controlled Neighbor Broadcast (CNB), active sessions refresh by data packets and deep search procedure. It is observed that, as node mobility increases, the control packet also increases in case of AODV, but the other bio inspired protocols, LB-Termite, Termite and SARA algorithms, shows similar behavior even under high mobility mainly due to their efficient route maintenance and fast route recovery techniques.

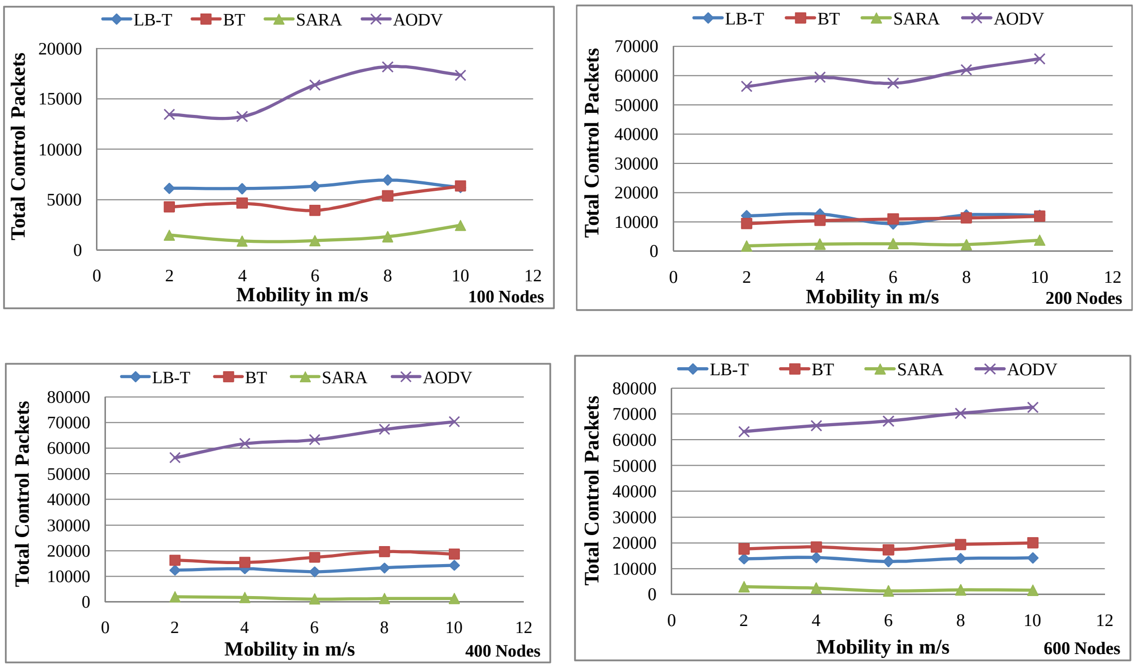


Figure 7.13: Control Packet Overhead Vs Node Velocity

7.4 Comparative Study of MA-Termite, Bat-Termite, LB-Bat-Termite and LB-Termite Algorithms

For the comparative study of MA-Termite, Bat-Termite, LB-Bat-Temite and LB-Termite the same experimental setup of LB-Termite is used and the results are discussed below.

Throughput Analysis

Figures 7.14, 7.15 and 7.16 show the throughput comparison of proposed MA-Termite, Bat-Termite, LB-Bat-Temite and LB-Termite algorithms under both Scenarios A and Scenario B and it is clearly observed that LB-Termite dominates the other two protocols MA-Termite and Bat-Termite in both the scenarios. With the stable nodes and proper load balancing among available multiple paths LB-Termite gives high throughput under all mobility and node density conditions. Both Bat-Termite and LB-Bat-Temite algorithms with their efficient backup

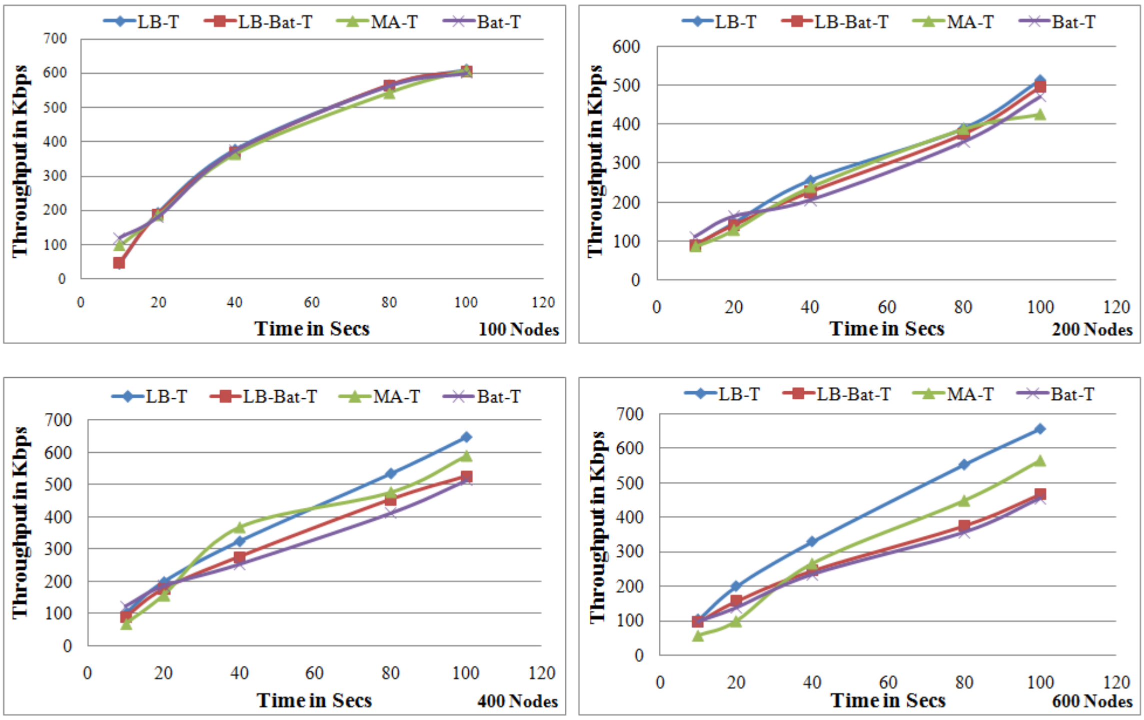


Figure 7.14: Throughput Vs Simulation Time in Sec (500m x 500m Sim. Area)

route maintenance technique and with proper network exploration for new and better routes gives high throughput only under low node density conditions when compared to MA-Termite. Bat-Termite and LB-Bat-Termite algorithms will maintain all available multiple paths to the destination and thus gives less route rediscovery latency and hence maintains a steady state data

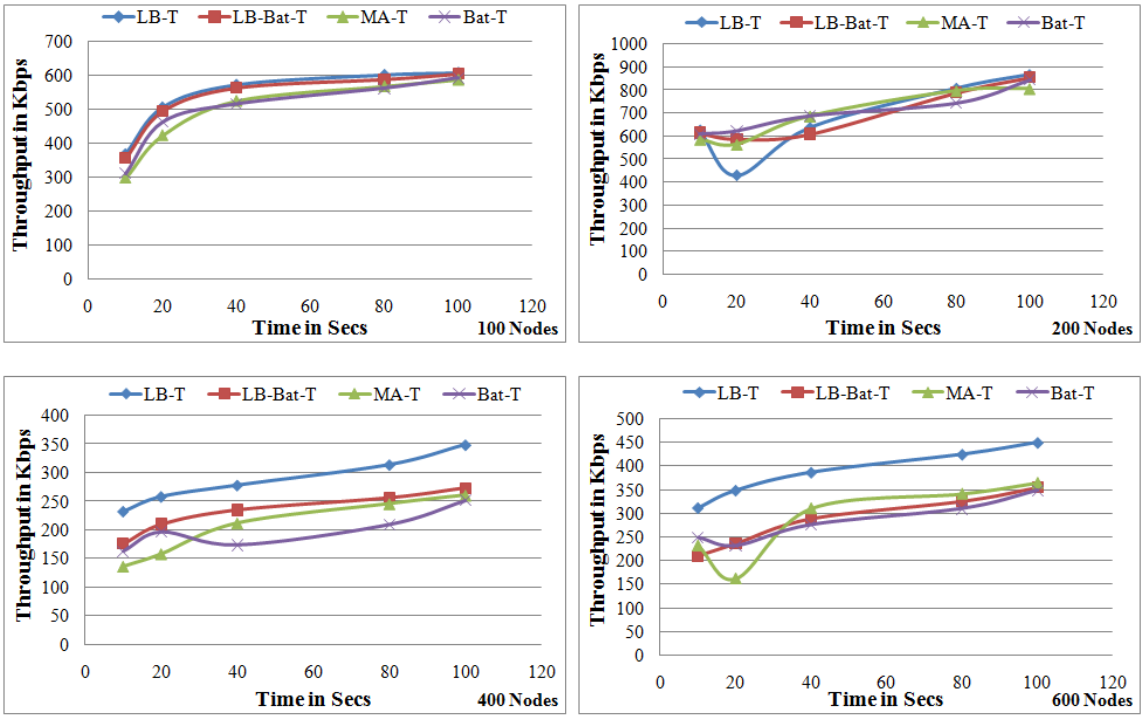


Figure 7.15: Throughput Vs Simulation Time in Sec (1000m x 1000m Sim. Area)

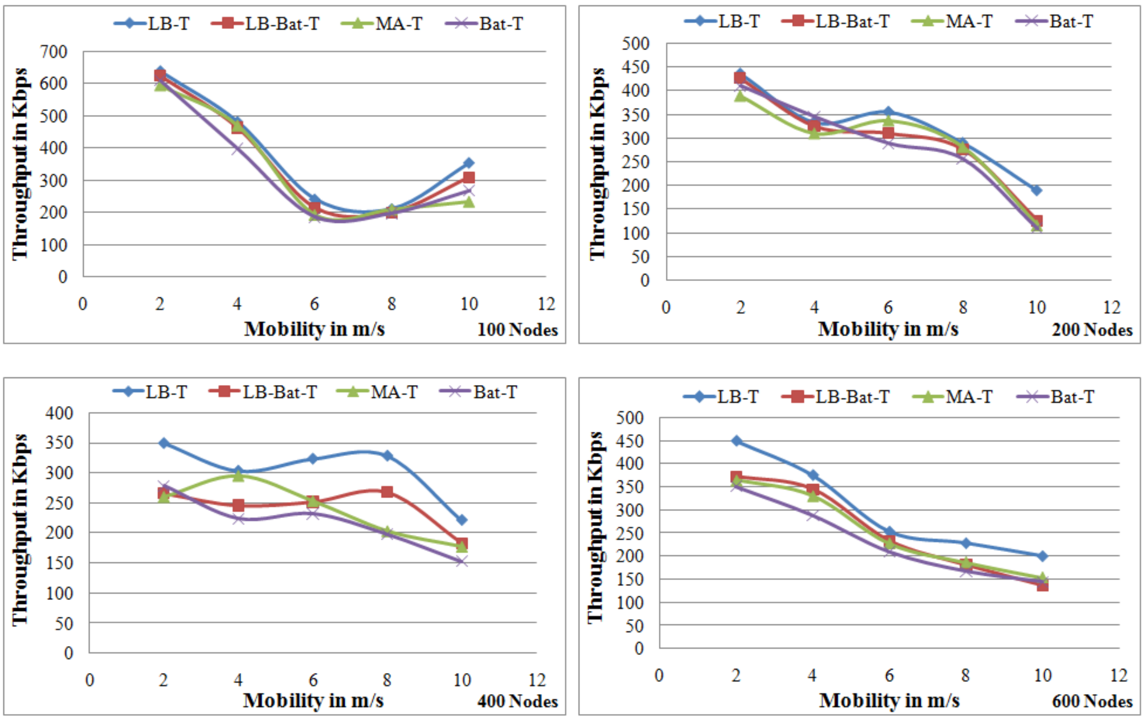


Figure 7.16: Throughput Vs Node Velocity

transmission. But Bat-Termite and LB-Bat-Termite algorithms suffers from additional control packets in terms of *echo_location* packets under high node density conditions and thus gives low throughput when node density is high. On the other hand, MA-Termite will suffer from the stagnation problem and thus exhibits low throughput under both the scenarios.

Packet Drop Analysis

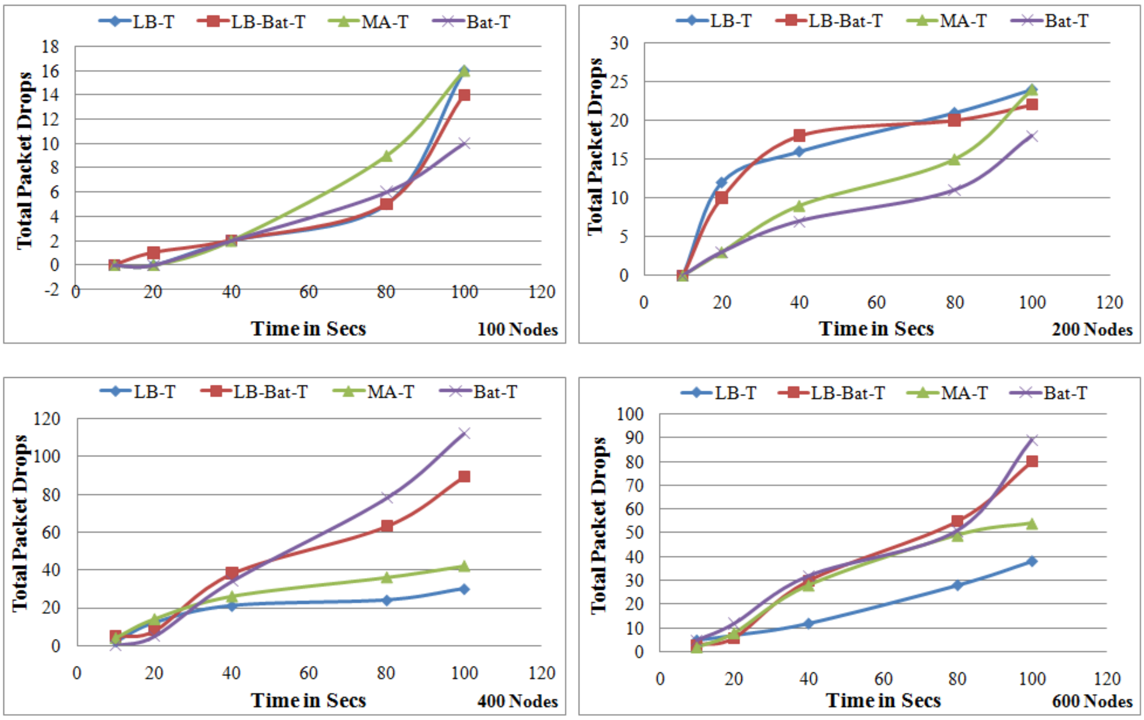


Figure 7.17: Packet Drop Vs Simulation Time in Sec (500m x 500m Sim. Area)

Figure 7.17, 7.18 and 7.19 show the total packet drops of all competing protocols and it shows that LB-Termite with its efficient route management and route rediscovery techniques drops fewer packets when compared to LB-Bat-Termite, Bat-Termite and MA-Termite under all mobility and node density conditions. Bat-Termite and LB-Bat-Termie shows fewer packets drops under low node density but under high node density conditions drops too many packets and hence its performance deteriorates when compared to MA-Termite. MA-Termite shows poor performance as its backup route maintenance is poor.

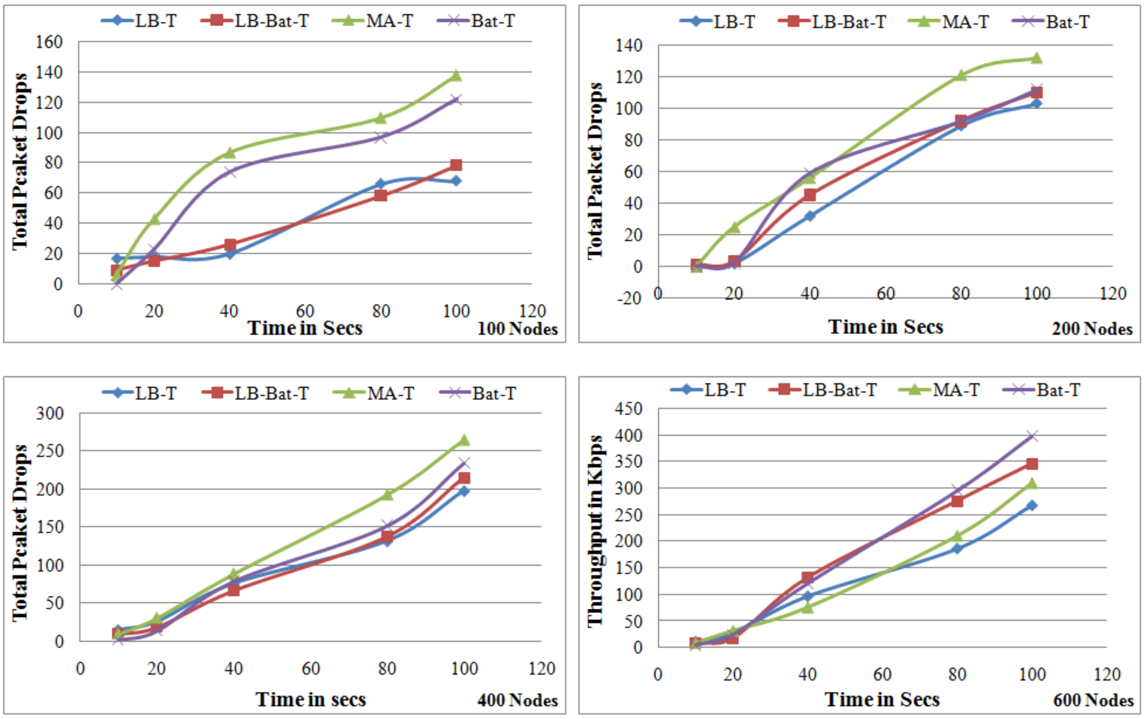


Figure 7.18: Packet Drop Vs Simulation Time in Sec (1000m x 1000m Sim. Area)

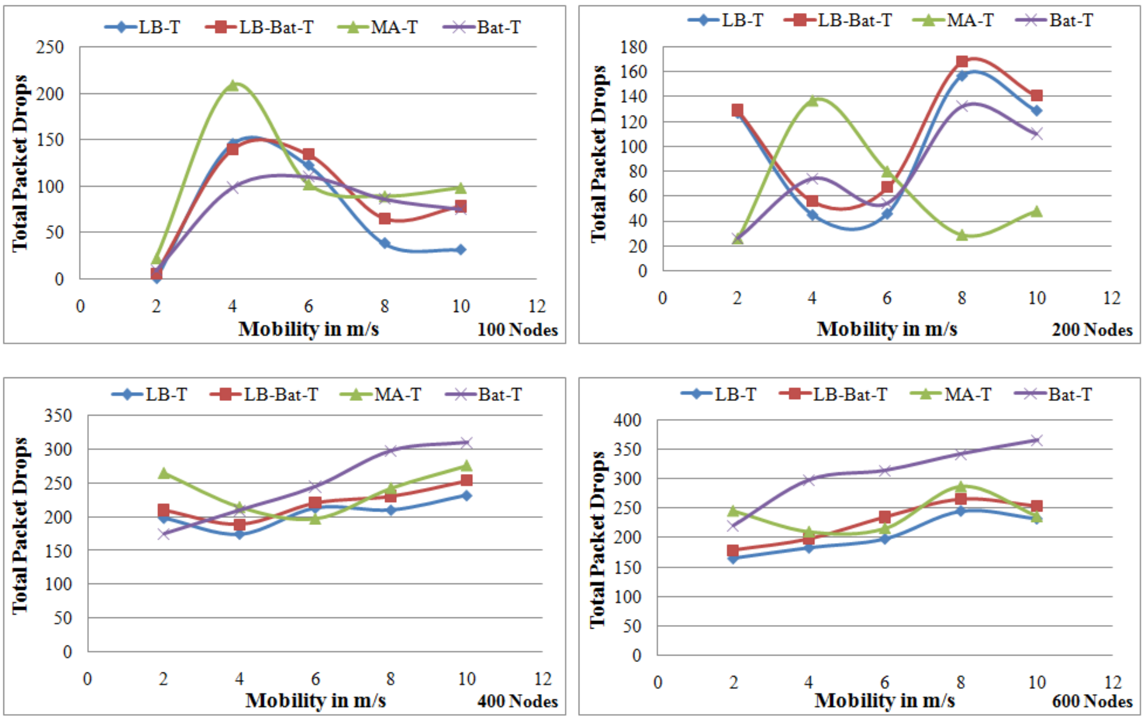


Figure 7.19: Packet Drop Vs Node Velocity

End-to-End Delay Analysis

End-to-end delay analysis is shown in Figures 7.20, 7.21 and 7.22. LB-Termite shows less end-

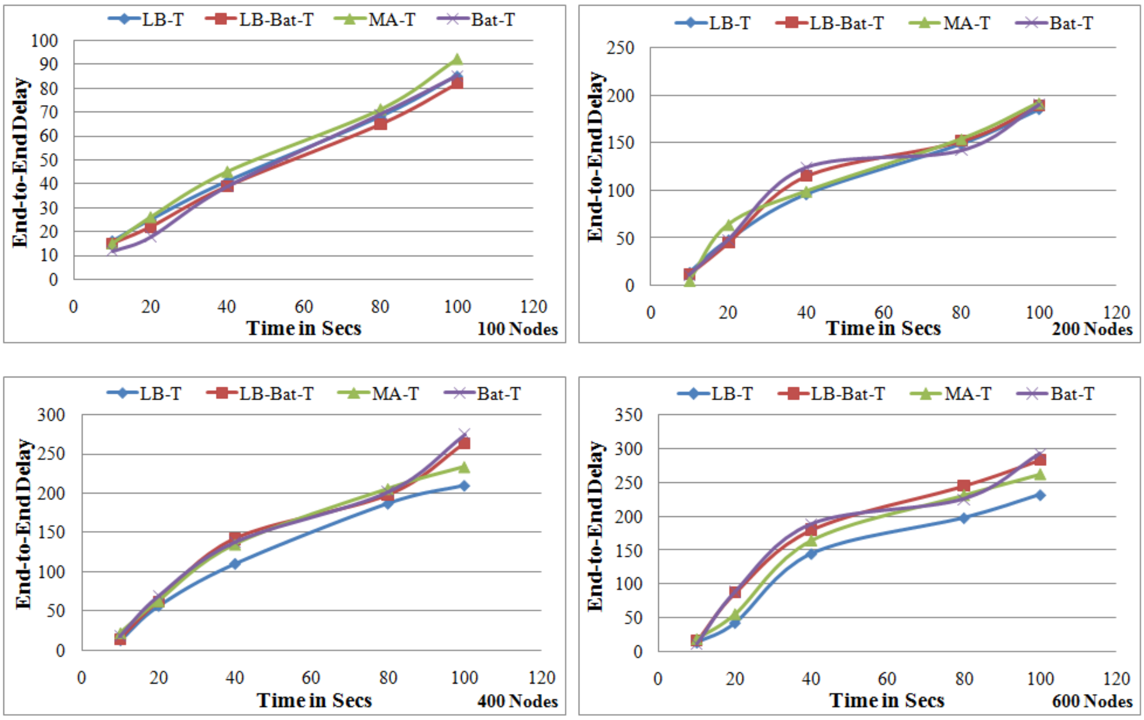


Figure 7.20: End-to-End Delay Vs Simulation Time in Sec (500m x 500m Sim. Area)

to-end delay under all conditions because the stable nodes in the path cause packets to reach the destination node early; proper route back up maintenance reduces the route rediscovery latency; the one hop RERR route rediscovery scheme further reduces the end-to-end delay. On the other hand, Bat-Termite and LB-Bat-Termite with its efficient backup route maintenance and network exploration schemes shows good performance under low node density and conditions; but when node density increases, it finds it difficult to manage routes and thus increases the end-to-end delay. Due to stagnation problem and improper backup route maintenance, packets in MA-Termite will take reasonable time to reach destination and thus shows high end-to-end delay under both the Scenarios.

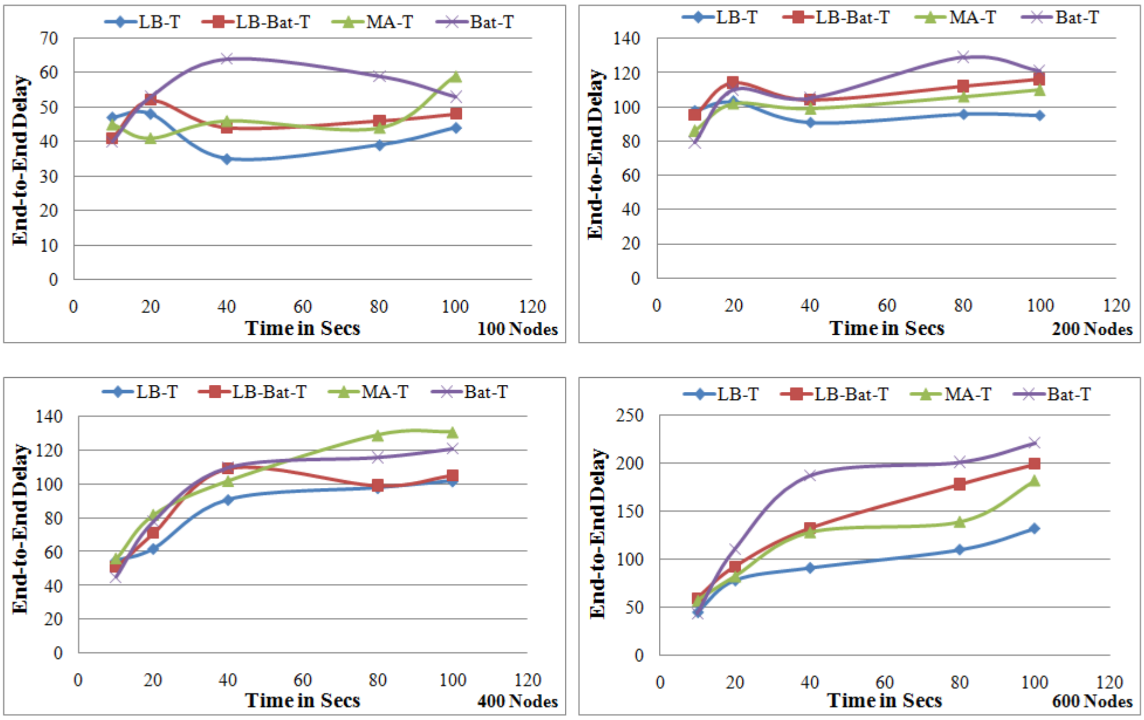


Figure 7.21: End-to-End Delay Vs Simulation Time in Sec (1000m x 1000m Sim. Area)

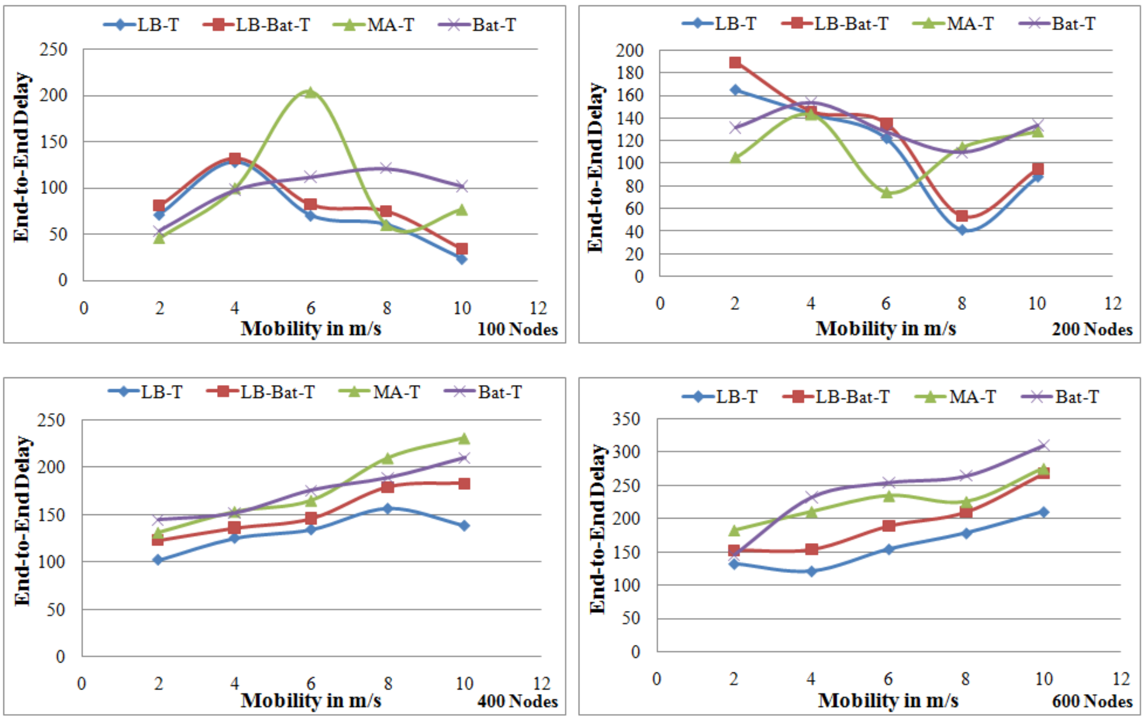


Figure 7.22: End-to-End Delay Vs Node Velocity

Control Packet Overhead Analysis

Control packet overhead of all three protocols are shown in Figures 7.23, 7.24 and 7.25. Since LB-Termite uses one hop RERR packets for route rediscovery, the control packets produced by LB-Termite is more when compared to MA-Termite. On the other hand, Bat-Termite and LB-Bat-Termite suffers from additional echolocation packets. MA-Termite produces considerable low control packet when compared to LB-Termite and Bat-Termite. Tables 7.2 and 7.3 summarizes the protocols behaviour under scalability and node velocity factor for different QoS parameters.

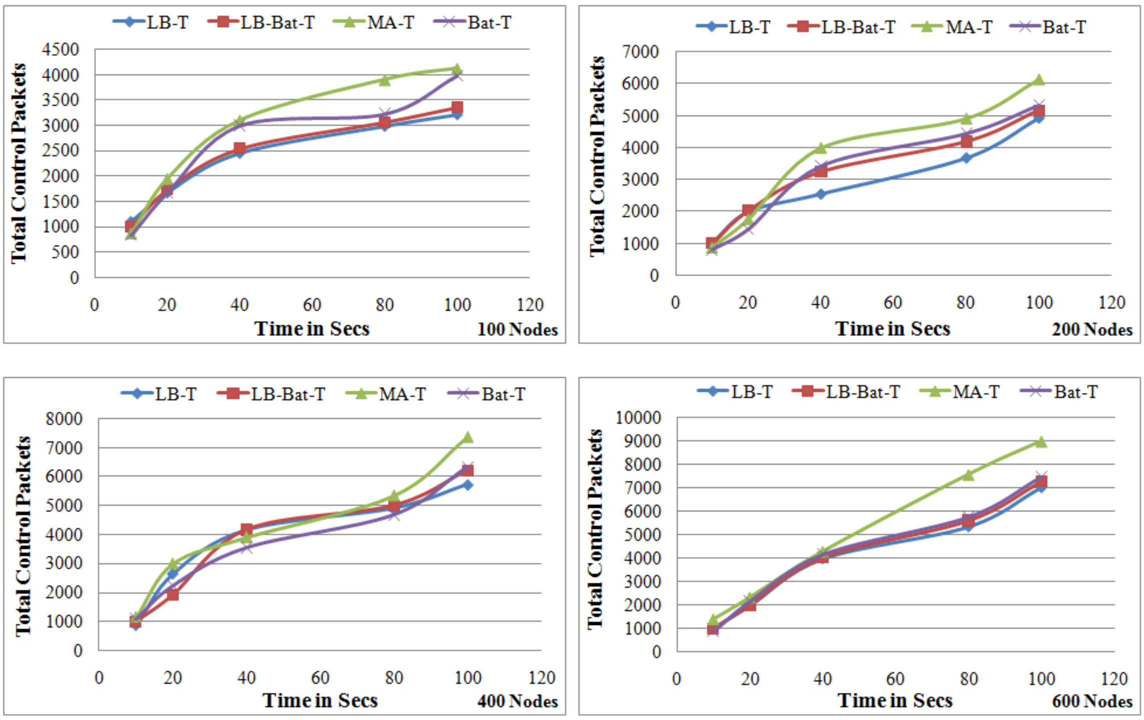


Figure 7.23: Control Packet Overhead Vs Simulation Time in Sec (500m x 500m Sim. Area)

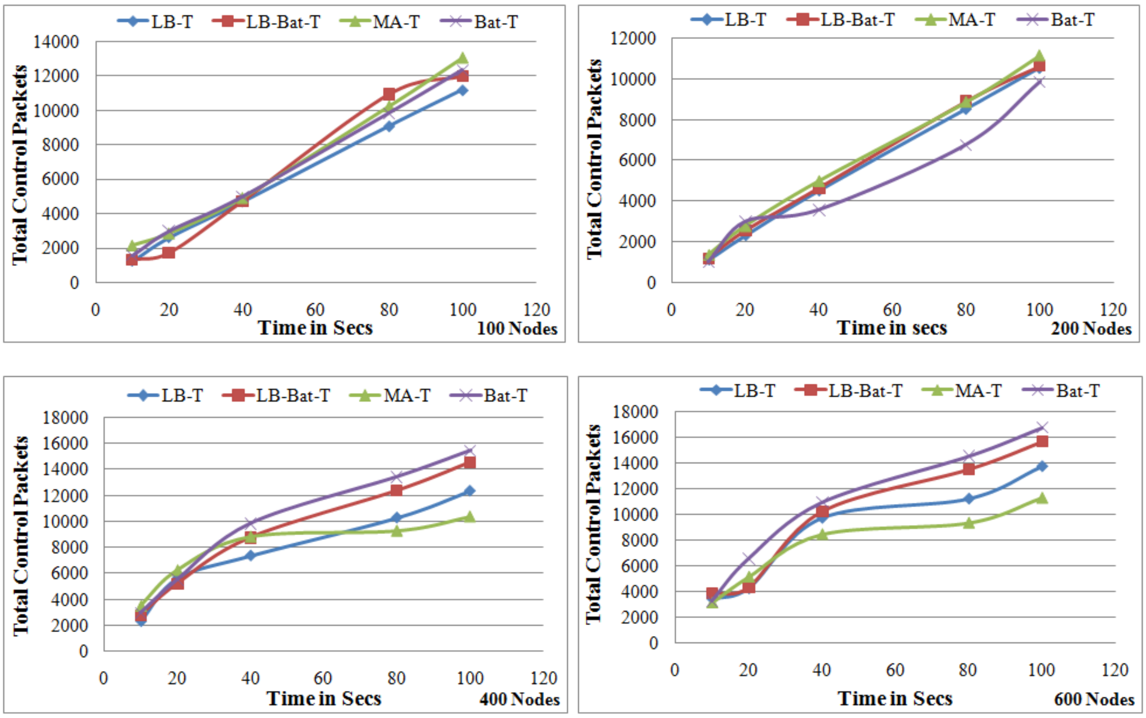


Figure 7.24: Control Packet Overhead Vs Simulation Time in Sec (1000m x 1000m Sim. Area)

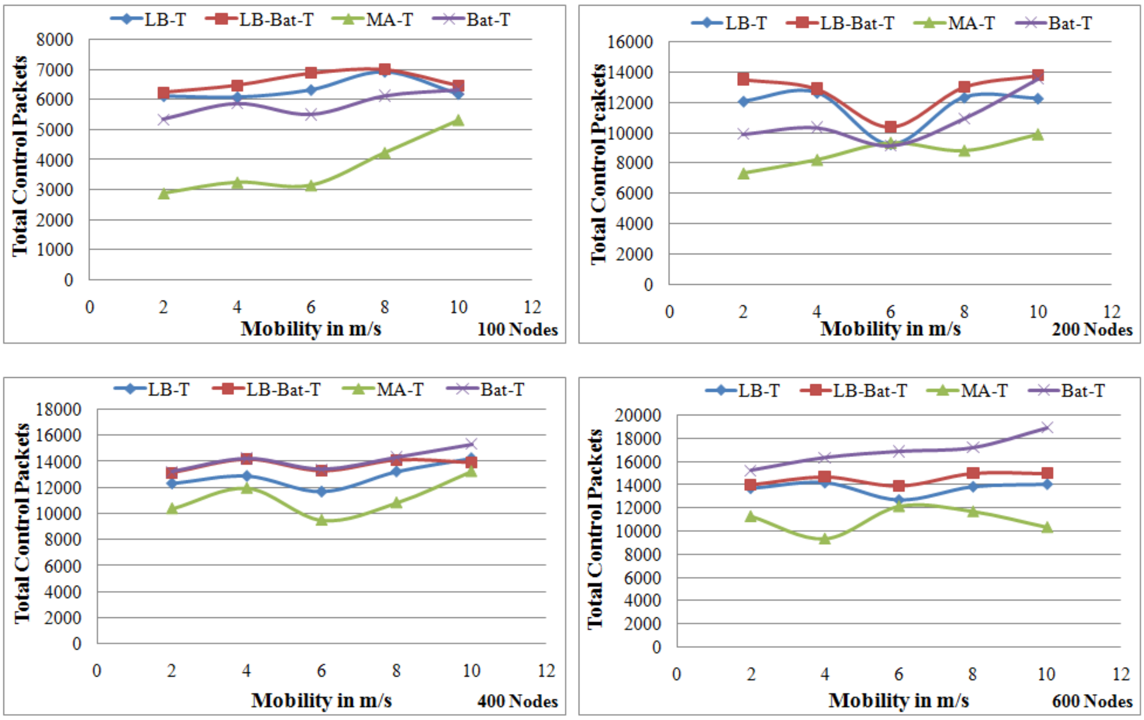


Figure 7.25: Control Packet Overhead Vs Node Velocity

Table 7.2: Protocol Behavior Under Increasing Node Density (100 to 600 Nodes)

Sl.No	Algorithm	Throughput	Drops	End-to-End Delay	Control Packets
1	AODV	Low	High	High	High
2	Termite	Medium	High	High	High
3	SARA	High	Low	Low	Low
4	MA-T	High	Medium	Medium	Medium
5	Bat-T	Low	High	High	High
6	LB-Bat-T	Low	High	High	High
7	LB-T	High	Low	Low	Low

Table 7.3: Protocol Behavior Under Increasing Node Velocity(2 - 10 m/s)

Sl.No	Algorithm	Throughput	Drops	End-to-End Delay	Control Packets
1	AODV	Low	High	High	High
2	Termite	Medium	High	High	High
3	SARA	Medium	Medium	Low	Low
4	MA-T	High	Medium	Medium	Medium
5	Bat-T	Low	High	High	High
6	LB-Bat-T	Low	High	High	High
7	LB-T	High	Low	Low	Low

7.5 Summary

Bio-inspired routing, an attractive and promising technique, is outperforming the traditional routing protocols by maintaining high throughput with less control packet overhead. But the bio-inspired algorithms such as ACO and Termite based routing suffer from the stagnation which leads to the congestion in the optimal path and thereby reducing the throughput. The proposed bio-inspired routing algorithm based on the social insect Termite with load balancing approach referred to as *Load Balanced-Termite (LB-Termite)* finds the *stable* and *reliable nodes* for the path and also it tries to mitigate the stagnation by introducing a new kind of pheromone called as *alarm pheromone*. *High route reliability, quick route discovery and repair & maintenance, high robustness with efficient management of multiple routes* and *stagnation avoidance* are some of the highlights of LB-Termite algorithm. LB-termite algorithm is implemented in NS-2 and the results are compared with other state-of-the-art bio inspired (SARA and Termite algorithms) and non bio-inspired (AODV) routing algorithms for its performance evaluation. It is observed that LB-Termite achieves 3% to 5% of increase in the evaluation metrics throughput, end-to-end delay and total packet drops when compared to SARA and achieves more than 5% increase in throughput, end-to-end delay, total packet drops and control packet overhead when compared to Termite and AODV algorithms.

The proposed LB-Termite algorithm can be applied for a typical Virtual Class Room (VCR) scenario. A VCR is a teaching-learning process where group of students and teacher are interconnected through web. VCR can be established immediately and students may join or leave the session in an ad-hoc manner. A teacher can establish a VCR session from his chamber or from his residence; students can join this VCR session being in their room or from any other place in the campus. During the session the students can send instant messages to the group members or they can request for study material or carrying out file transfer among the student members. Under such scenario, LB-Termite with its efficient route maintenance and multipath features can ensure a constant data rate among the group members. With its reduced control packet overhead, LB-Termite not only improves the network lifetime but also avoids the session hiccups due to less packet drops. Further, less end-to-end delay feature of LB-Termite makes the VCR more interactive and lively session of teaching-learning process. Similarly, LB-Termite algorithm can be used for other application scenarios such as university or campus based management information systems, live data sharing during real time class room lecturing or video conference based meetings.

Chapter 8

Conclusions and Future Work

MANETs are rapidly gaining popularity because of its distinguishing characteristics such as no fixed infrastructure, easy to deploy, no central administration, increased connectivity etc. However, MANETs suffer from *routing complexity, path management* and *extra delay* due to multi-hop relaying; thus providing the best QoS in MANETs is challenging. In the recent past many researchers have developed SI based adaptive algorithms for solving real world problems in general and in particular routing problems of telecommunication domain. Further, the CLD has not been explored much in developing SI based routing protocols for increasing the efficiency of mobile wireless networks. Hence, the research work in this thesis directed towards designing and developing a novel bio-inspired QoS aware routing protocols for MANETs by exploiting the salient features of social insect Termites, mammal Bats and Cross Layer Design. Accordingly, three bio-inspired QoS aware routing protocols namely, MA-Termite, Bat-Termite and LB-Termite are proposed for MANETs and the merits and demerits of these proposed bio-inspired QoS aware routing algorithms are summarized as follows.

In Chapter 3, two intelligent cross layer models namely *Service Driven Cross Layer Model* and *Prioritized QoS* are designed and developed in order to provide the QoS for different applications of MANETs. *Service Driven Cross Layer Model* targets mainly *False Route Failure* and uses priority of the flow as the cross layer parameter across Application and Network Layers. Accordingly the limit of RTS retransmission at the MAC layer is dynamically adjusted for the prioritized flows. The second cross layer model; *Prioritized QoS* uses priority of the flow from the Application Layer and *received signal strength (Pr)* from the Physical Layer as the cross layer parameters across Application, Physical and Network Layers. *Prioritized QoS* uses distance between the nodes and accordingly adjusts the RTS retransmission limit at MAC layer for the prioritized flow. The proposed cross layer models are evaluated through simulation in ns-2 and the results are analyzed. The results clearly demonstrates that the proposed intelligent cross layer models have successfully enhanced the throughput of the prioritized flow while reducing the overall control packet overhead during the route rediscovery phase; on the other hand, the high RTS retry limit increases the end-to-end delay. However, the distance parameter *(distance between the nodes)* is found to be convincing and encouraging when compared to *shortest hop path* parameter used in traditional AODV and DSR protocols and hence motivated in design and development of bio-inspired routing algorithms for MANETs by exploiting the features of *Cross Layer Design, social insect Termites and mammal Bats*.

In order to make MANETs Context or Mobility aware under dynamic network conditions, a novel bio-inspired QoS aware routing protocol referred to as Mobility Aware-Termite (MA-Termite) is designed and developed for MANETs in Chapter 4. MA-Termite uses the captivating hill building nature of social insect Termites in order to guide each packet to reach the destination node. Further, MA-Termite defines a novel distance based pheromone update-decay function in order to reflect the current context of the network in terms of node mobility; the pheromone update/decay function decay the pheromone proportionally to the inter packet distance between the nodes. Thus MA-Termite uses not only the cross layer model architecture but also exploits the salient features of social insect Termites in order to provide best QoS for the MANETs. In MA-Termite, the mobility of each neighbour node will be exactly reflected in the Trial Pheromone Table; thus higher the node mobility then less will be the selection preference for data transfer. Hence, MA-Termite shows considerable performance improvement under different node velocity conditions.

Further, an analytical model for pheromone update-decay function in describing the asymptotic behaviour of the MA-Termite using both single and double links is developed by considering two different parameters namely *decay rate* and *pheromone smoothing*. A rigorous simulation study is carried out with different test scenarios in order to find the similarities between theoretical and simulation results under node density (100, 200, 400 and 600 nodes) and node velocity (2,4,6,8 and 10 m/s) conditions. The simulation results demonstrates that MA-Termite achieves an average of 2-3% improvement when compared to SARA with respect to QoS metrics such as throughput, end-to-end delay and total packet drops. It is also observed that MA-Termite achieves 5-8% improvement when compared to basic Termite and AODV algorithms in terms of QoS metrics such as throughput, end-to-end delay, total packet drops and control packet overhead and thus MA-Termite exhibits superior performance when compared to SARA, basic Termite and AODV routing algorithms for MANETs under dynamic network conditions only. But the performance of MA-Termite under stable network conditions is inferior when compared to both Bat-Termite and LB-Termite. Though MA-Termite is context aware, its performance degrades when the node density increased to 400 and 600 nodes. The main cause for such inferior performance of MA-Termite is it always chooses the highest pheromone link at any given time thus it suffers from *stagnation problem*. Further, MA-Termite also suffers from *poor network exploration* and *improper backup path maintenance* which causes the early removal of secondary paths from Trail Pheromone Table.

To solve the *poor network exploration* and *improper backup path maintenance* problem of MA-Termite, a novel hybrid bio-inspired routing protocol for MANETs referred to as Bat-Termite is proposed for MANETs in Chapter 5 based on the salient features of social insect *Termites (hill building nature)* and mammal *Bats (echolocation feature)*. Bat-Termite with its efficient backup route maintenance and with good network exploration techniques shows the superior performance when compared to MA-Termite under low node density conditions (100 and 200 nodes). Since the asymptotic behavior of Bat-Termite is same as MA-Termite, hence Bat-Termite also exhibits inferior performance when the node density is high (400 and 600 nodes). On the other hand, in an effort to maintain the available paths efficiently and to explore the network for new and better paths, Bat-Termite produces more control packets under high node density conditions. The echolocation packets produced by the Bat-Termite is directly proportional to the node density in the network. Hence, under high node density conditions Bat-Termite performance deteriorates. Simulation results demonstrate that, Bat-Termite lags in scalability feature and further adds extra burden for the nodes in terms of *echo_location* packets thereby causing maintenance problem in maintaining all the alternative paths to the destination. Thus, each node spends more time in maintaining the routes and exploring the network for better routes than sending the data packets. Hence, Bat-Termite is suitable for low density network conditions and since Bat-Termite uses the framework of MA-Termite, the stagnation problem still unresolved by Bat-Termite algorithm.

In Chapter 6, a novel heuristic Load Balanced hybrid based QoS aware routing protocol referred to as Load Balanced-Bat-Termite (LB-Bat-Termite) is proposed for MANETs in order to solve the *stagnation problem* of both MA-Termite and hybrid Bat-Termite. Firstly, LB-Bat-Termite is context aware: the mobility of the neighbour nodes will be clearly reflected in the

Trail Pheromone Table of each node. Thus each node will have general idea about the current status of the network and node can choose the stable and reliable node for the path in order to forward the data packets towards the destination. Secondly, LB-Bat-Termite is load aware: the concept of alarm pheromone is used by the LB-Bat-Termite in order to find and avoid the heavily loaded nodes in the paths. Thus LB-Bat-Termite not only solves the stagnation problem of Bat-Termite but also solves the stagnation problem and improper route maintenance/poor back up route maintenance problem of MA-Termite. Thirdly, LB-Bat-Termite is QoS aware: LB-Bat-Termite properly manages the load among all the available multiple paths to the destination. Further, by constantly switching the traffic among all the available multiple paths to the destination, LB-Bat-Termite keeps all the routes active; thus chances of availability of multiple routes in worst case scenario is very high. On the other hand, eventhough multiple routes are available to reach the destination, LB-Bat-Termite emphasises only on primary and secondary paths for data transfer. Thus, mobile nodes in LB-Bat-Termite wastes most of its time in maintaining all possible paths to the destination eventhough only praimary and secondary paths are used for data transfer. On the other hand, since LB-Bat-Termite uses the architecture of Bat-Termite, LB-Bat-Termite also suffers from additional control packets in terms of *echo_location* packets thus creates an extra burden on the mobile nodes in terms of handling *echo_location* packets. Hence LB-Bat-Termite is not scalable and is suitable only for low density network conditions.

Finally, in Chapter 7, a novel Load Balanced Termite based QoS aware routing protocol referred to as Load Balanced-Termite (LB-Termite) is proposed for MANETs in order to overcome the scalability problem of LB-Bat-Termite algorithm. LB-Termite solves the stagnation problem of both MA-Termite and Bat-Termite algortihm and solves the poor network exploration and backup root maintenance problem of MA-Termite algorithm. Furhter, LB-Termite also solves the scalability problem of LB-Bat-Termite algorithm. LB-Termite inherits the features such as context awareness, load awareness and QoS awareness from LB-Bat-Termite algorithm. In order to improve the scalability LB-Termite drops the echolocation feature of LB-Bat-Termite algorithm. Experimental results demonstrates that LB-Termite achieves 3% to 5% of increase in the evaluation metrics throughput, end-to-end delay and total packet drops when compared to SARA and achieves more than 5% increase in throughput, end-to-end delay, total packet drops and control packet overhead when compared to Termite and AODV algorithms. Further, a comparative study of proposed MA-Termite, Bat-Termite, LB-Bat-Termite and LB-

Termite is also done to list out the merits and demerits of each of the proposed protocols. The study reveals that, LB-Termite outperforms MA-Termite, Bat-Termite and LB-Bat-Termite under both Scenario A and Scenario B.

In future, the proposed QoS aware bio inspired routing algorithms can be enriched further in the following aspects:

- Scalability is the important factor in MANETs that too in large scale networks (5000 to 6000 nodes) hence an effort will also be made in immediate future in order to study the performance of the proposed LB-Termite algorithm under large network conditions.

- To evaluate the performance of the proposed LB-Termite algorithm using testbeds.

- To consider other QoS parameters namely, *bandwidth, latency* and *jitter* for performance evaluation of MA-Termite, Bat-Termite, LB-Bat-Termite and LB-Termite algorithms.

Bibliography

A.G., Z. et al. (2012). ”cognitive network layer- mobility aware routing protocol”. In Sixth International Conference on Information Processing ICIP-2012, volume 292, pages 84–92.

Airas, M. (2003). ”Echolocation in Bats”. In Proceedings of Spatial sound perception and reproduction.

Aritra, C. and Swagatam, D. (2012). Automatic shape independent clustering inspired by ant dynamics. International Journal of Swarm and Evolutionary computation, pages 33–35.

Asokan R, N. A. and A, N. (2007). A swarm based distance vector routing to support multiple quality of service metrics in mobile ad hoc networks. Journal of computer science, 3(9):700– 707.

Asokan R, N. A. and C, V. (2008). Ant based dynamic source routing protocol to support multiple quality of service (qos) metrics in mobile ad hoc networks. International Journal of computer Science and security, 2(3).

B, K. and K, D. (2008). Ant colony based node disjoint hybrid multi-path routing for mobile ad hoc networks. Journal of Computer Science, 4(2):80–86.

B, T. and L, R. (2004). Dynamic routing in mobile wireless networks using abc-adhoc. In The Fourth International Workshop on ant colony optimization and swarm intelligence(ANTS), LNCS , Springer, volume 3172.

Baras, J. and Mehta, H. (2003). ”A Probabilistic Emergent Routing Algorithm (PERA) for Mobile Ad Hoc Networks”. In Proceedings of Modeling and optimization in mobile, ad hoc and wireless networks (WiOpt 03), pages 20–24.

C. Kolias, G. K. and Maragoudakis, M. (2011). ”Swarm Intelligence in Intrusion Detection : A Survey ”. Computers and Security, 30(8):625–642.

Camara, D. and Loureiro, A. (2000). A novel routing algorithm for ad hoc networks. In HICSS, IEEE Press.

Camara, D. and Loureiro, A. (2001). Gps/ant-like routing in ad hoc networks. Telecommunication Systems, 18(1–3):85–100.

Caro, G. A. D., Ducatelle, F., and Gambardella, L. M. (2008). Theory and practice of Ant Colony Optimization for routing in dynamic telecommunications networks. Idea Group, Hershey, Pa, USA.

Caro, G. D. and Dorigo, M. (2011). Antnet: Distributed stigmergetic control for communications networks. CoRR, abs/1105.5449.

C.E., P. and E.M, R. (1999). ”ad hoc on demand distance vector (aodv) routing”. In Second IEEE Mobile Computing Systems and Applications, pages 90–100.

Chai Keong Toh, A.-N. and cho, Y.-Z. (2009). ”Load Balanced Routing Protocols for Ad Hoc Mobile Wireless Networks”. Topics in Ad Hoc and Sensor Networks, IEEE Communications Magazine, 47(8):78–84.

Chen, L. and Heinzelman, W. B. (2007). ” A Survey of Routing Protocols that Support QoS in Mobile Ad Hoc Networks ”. IEEE Networks, 21(6):30–38.

Chen, Q. and Niu, Z. (2004). A delayed adaptive retransmission scheme for false route failure in manet. In The 2004 Joint Conference of the 10th Asia-Pacific Conference on Communications, 2004 and the 5th International Symposium on Multi-Dimensional Mobile Communications, Volume 2, pages 858–862.

Correia, F. and Vazao, T. (2008). Simple ant routing algorithm. In International Conference on Information Networking (ICOIN 2008), IEEE, pages 1–8.

Correia, F. and Vazao, T. (2010). Simple ant routing algorithm strategies for a (multipurpose) manet model. Ad Hoc Networks, Elsevier, 8:810–823.

D, S. and RS, B. (2007). Proposal on multi agent ants based routing algorithm for mobile ad hoc networks. International Journal of Computer Science and Network Security (IJCSNS), 17(6).

Dhillon S., A. X. and P, V. M. (2007). Ant routing in mobile ad hoc networks. In Proceedings of third international conference on networking and services (INCS) , IEEE computer Society, pages 67–74.

Di Caro, G. and Dorigo, M. (1998). Mobile agents for adaptive routing. In System Sciences, 1998, Proceedings of the Thirty-First Hawaii International Conference on, volume 7, pages 74–83.

Douglas S.J. Dec Couto, Daniel Aguayo, J. B. and Morris, R. (2003). A high-throughput path metric for multi-hop wireless routing. In Proceedings of the 9th ACM International Conference on Mobile Computing and Networking(MobiCom03).

Ducatelle F, Di Caro G A, G. L. M. (2005). ”Using ant agents to combine reactive and proactivestrategies for routing in mobile ad hoc networks”. International Journal of Computational intelligence and Applications, Special issue on nature inspired approaches to networks and telecommunications, 5(2):169–184.

E. Bonabeau, M. and G.Theraulaz (1999). ”Swarm Intelligence: From Natural to Artificial Systems”. Oxford University Press, New York, USA.

Farooq, M. and Caro, G. A. D. (2008). ”Routing Protocols for Next Generation Networks Inspired by Collective Behaviors of Insect Societies: An Overview ”. Swarm Intelligence, Natural Computing Series, Springer, pages 101–160.

Frederick Ducatelle, G. A. D. and Gambardella, L. M. (2010). ”Principles and Applications of Swarm Intelligence for Adaptive Routing in Telecommunications Networks”. Swarm Intelligence, Springer, 4(3):173–198.

Gianni Di Caro, F. D. and Gambardella, L. M. (2005). Swarm Intelligence for Routing in Mobile Ad Hoc Networks. In IEEE Swarm Intelligence Symposium, pages 76–83.

Gudakahriz Sajjad Jahanbakhsh, J. S. and Esmaeel, Z. (2011). NISR: A nature inspired scalable routing protocol for mobile ad hoc networks. International Journal of computer Science engineering and Technology, 1(4):947–950.

H, M. and K, M. (2001). Accelerated ants routing in dynamic netwokrs. In 2nd ACIS International Conference on Software engineering, Artificial intelligence, Networking and parallel/ distributed computing, pages 333–339.

Haas, Z. J. and Pearlman, M. R. (1997). The Zone Routing Protocol (ZRP) for Ad Hoc Networks. Technical report, Cornell University.

Hanzo, L. and Tafazolli (2007). ” A Survey of QoS Aware Routing Solutions for Mobile Ad Hoc Networks ”. Communications Surveys Tutorials, IEEE, 9(2):55–70.

He, G. (2002). Destination-Sequenced Distance Vector (DSDV) Protocol. Technical report, Helsinki University of Technology.

Heissenbuttel M, Braun T, J. D. and T, H. (2006). A framework for routing in larger ad hoc networks with irregular topologies. In Challenges in Ad Hoc Networking, IFIP International Federation for Information Processing, volume 197, pages 119–128.

Hoolimath, P. et al. (2012). ”optimized termite: A bio inspired routing algorithm for manet”. In IEEE Int. Conf. on Signal Processing and Communications (SPCOM-2012), pages 1–5.

Hussein O, S. T. and M, J. L. (2005). Probability routing algorithm for mobile ad hoc networks resources management. IEEE Journal of Selected Areas in Communications, 23(12):2248– 2259.

Imrich Chlamtac, M. C. and J.N.Liu, J. (2003). ” Mobile ad hoc networking: imperatives and challenges ”. Ad Hoc Networks, Elsevier, 1:13–64.

Jacquet, P., Muhlethaler, P., Clausen, T., Laouiti, A., Qayyum, A., and Viennot, L. (2001). Optimized link state routing protocol for ad hoc networks. In Multi Topic Conference, 2001. IEEE INMIC 2001. Technology for the 21st Century. Proceedings. IEEE International, pages 62–68.

Johnson, D. B., Maltz, D. A., and Broch, J. (2001). Dsr: The dynamic source routing protocol for multi-hop wireless ad hoc networks. In Ad Hoc Networking, edited by Charles E. Perkins, Chapter 5, pages 139–172. Addison-Wesley.

Kim, K.-I. and Kim, S.-H. (2006). ”Effectiveness of Reliable Routing Protocols in Mobile Ad Hoc Networks”. Wireless Personal Communications, Springer, 38:377–390.

Kim, S. (2012). An antbased multipath routing algorithm for qos aware mobile ad hoc networks. Wireless Personal Communications, Springer, 66:739–749.

Kiran, M. et al. (2012). ”distance based termite algorithm for mobile ad-hoc networks”. In International Conference on Information Processing (ICIP-2012), pages 35–45.

Kiran, M. and Reddy, G. R. M. (2011a). ”cross layer service driven adaptive retry limit for ieee 802.11 mobile ad hoc networks”. In IEEE IFIP Wireless Days 2011, pages 1–3.

Kiran, M. and Reddy, G. R. M. (2011b). ”throughput enhancement of the prioritized flow in self aware manet based on neighborhood node distances”. In IEEE International Conference on Computer Applications and Industrial Electronics, pages 497–502.

Kiran, M. and Reddy, G. R. M. (2013). ”Mobility Aware Termite: A Novel Bio Inspired Routing Protocol for Mobile Ad Hoc Networks”. IET Journal of Networks, doi: 10.1049/ietnet. 2012.0203.

K.Nahm, A. H. and Kuo, C.-C. (2005). Improving stability and performance of multihop 802.11 networks. In ACM International symposium on Mobile AdHoc Networking and Computing (MOBIHOC05).

L.J.G. Villalba, D. C. and Orozco, A. (2010). Bio-inpired routing protocol for mobile ad hoc networks. IET communications, 4(8):2187–2195.

M, K. and A, P. (2009). Ant routing with distributed geographical localization of knowledge in ad-hoc networks. In EvoCOMNET- 2009 , LNCS , Berlin, Springer, pages 111–116.

M. Gunes, M. H. and Bouazizi, I. (2003). Influence of adaptive rts/cts retransmission on tcp in wireless and ad-hoc networks. In 8th IEEE international symposium on computers and communications (ISCC03), pages 855–860.

M. Gunes, U. s. and Bouazizi, I. (2002). Ara the ant-colony based routing algorithm for manets. In International Workshop on Ad Hoc Networks (IWAHN 2002), pages 79–85.

Marco Dorigo, E. B. and Theraulaz, G. (2000). Ant algorithm and stigmergy. Future Generation Computer Systems, 16(8):851–71.

Marjan Kuchaki Rafsanjani, S. A. and Pakzad, F. (2010). A hybrid routing algorithm based on ant colony and zhls routing protocol for manet. Communications in Computer and Information Science , Springer, 120:112–122.

Marwaha S, T. C. and D, S. (2002). ”Mobile Agents based routing protocol for mobile ad hoc networks”. In Procedding of the IEEE global telecommunications conference (Globecom), IEEE Computer Society, Washington, pages 163–167.

Masip-Bruin, X. and Others (”2006”). Research Challenges in Qos Routing. ”Computer Communications, Elsevier”, ”29”(”5”):”563–581”.

Mazhar, N. and Farooq., M. (2007). Beeais: Artificial immune system security for nature inspired, manet routing protocol, beeadhoc. In Proceedings of the 6th International Conference on Artificial Immune Systems, LNCS, Springer, volume 4628.

Minar N, K. K. and P, M. (1999). Cooperating mobile agents for dynamic network routing. Software agents for future Communication systems, Chapter 12, A. Hayzelden and J. Bigham (Eds.), Springer,.

M.M. Goswami, R. D. and Thakare, V. (2009). ”Fuzzy ant Colony Based Routing Protocol for Mobile Ad Hoc Networks”. In Proceedings of International Conference on Computer Engineering and Technology (ICCET 09), IEEE computer Society, volume 2, pages 438–444.

Mohapatra P., J. L. and Gui, C. (2003). ”QoS in mobile ad hoc networks”. Wireless Communications, IEEE, 10(3):44–52.

MohitTiwari and ShirshuVarma (2010). Bird flight inspired clustering based routing protocol for mobile ad hoc networks. IJCSNS International Journal of Computer Science and Network Security, 10(3).

Muhammad Saleem, G. A. D. C. and Farooq, M. (2011). ”Swarm Intelligence Based Routing Protocol for Wireless Senson Networks: Survey and Future Directions ”. Information Sciences, 181(20):4597–4624.

Murthy, C. S. R. and Manoj, B. S. (2005). ” Ad hoc Wireless Networks”. 2nd Edition, Pearson Education.

N, C. and K, V. (2008). Enhanced ant colony based algorithm for routing in mobile ad hoc network. In Proceedings ofWorld Academy of Science, Engineering and Technology, volume 36, pages 30–35.

Neumann, F. and Witt, C. (2010). ”Bioinspired Computation in Combinatorial Optimization Algorithms and Their Computational Complexity”. Natural Computing Series, Springer.

Osagie Eseosa, T. P. and K, T. R. (2008). ”PACONET : ImProved ant colony optimization routing algorithm for mobile ad hoc networks”. In Proceedings of 22nd international conference on advanced information networking and applications, pages 204–211.

P. Venkata Krishna, V. S., Bhiwal, G. V. A., and Chawla, A. (2012). Quality-of-service enabled ant colony-based multipath routing for mobile ad hoc networks. IET communications, 6(1):76–83.

Prasad S, S. Y. and CS, R. (2009). Swarm based intelligent routing for manets. International Journal of Recent Trends in Engineering.

Q Han, Y. b., L.Gong, and wu, W. (2011). ”Link Availability prediction-based reliable routing for mobile ad hoc networks”. IET Communications, 5(16):2291–2300.

R. Schoonderwoerd, O. Holland, J. B. and Rothkrantz, L. (1996). Ant-Based load Balancing in Telecommunication Networks. Technical report, HP Labs Technical Reports.

Rahman, M., Anwar, F., Naeem, J., and Abedin, M. (2010). A simulation based performance comparison of routing protocol on mobile ad-hoc network (proactive, reactive and hybrid). In Computer and Communication Engineering (ICCCE), 2010 International Conference on, pages 1–5.

Raisinghani, V., Singh, A., and Iyer, S. (2002). Improving tcp performance over mobile wireless environments using cross layer feedback. In Personal Wireless Communications, 2002 IEEE International Conference on, pages 81–85.

Ramanathan, R. and Redi, J. (2002). ”A Brief Overview of Ad Hoc networks: Challenges and directions ”. IEEE Communication Magazine, 40(5):20–22.

Regnier, F. E. and N, J. (1968). ”Insect Pheromones”. Journal of Lipid Research, 14.

Ross, S. S. (2002). ”Jointly Distributed Random Variables in A First Course in Probability”. Pearson Education.

Roth, M. and Wicker, S. (2003a). ”termite: Ad-hoc networking with stigmergy”. In IEEE Global Communication Conference, volume 5, pages 2937–2941.

Roth, M. and Wicker, S. (2003b). ”termite: Emergent ad-hoc networking”. In the 2nd Mediterranean workshop on Ad-Hoc Networking (MedHoc 2003).

Roth, M. andWicker, S. (2004a). ” asymptotic pheromone behavior in swarm intelligent manets : an analytical analysis of routing behavior”. In Sixth IFIP IEEE International Conference on Mobile and Wireless Communication Networks MWCN 2004.

Roth, M. andWicker, S. (2004b). Network routing with filters: Link utility estimation in swarm intelligent manets. Technical report, This Research is Sponsored by the Defence Advance Research Project Agency (DARPA) and administrated by the Army Research Office under Emergent Surveilliance Plexus MURI Award No. DAAD 19-01-1-0504.

Ruth, M. H. (2005). Termite: A Swarm Intelligent Routing Algorithm for Mobile Ad Hoc Networks. PhD thesis, Cornell University.

S, R. and C, S. (2006). ”ANSI: A unicast routing protocol for mobile ad hoc networks using swarm intelligence”. In proceedings of the international conference on artificial intelligence (ICAI), pages 104–110.

Sanlin Xu, K. L. and M.Jones, H. (2007). ”An Analysis Framework for Mobility Metrics in Mobile Ad Hoc Networks ”. EURASIP Journal on Wireless Communication and Networking, 2007.

Scheduling, B. E.-A. and Framework, R. (2004). H. F. Wedde and M. Farooq. In LSIII, School of Computer Science, University of Dortmund.

Sehoon Kim, J. L. and Yeom, I. (2008). Neighbour-aware adaptive retry limit for ieee 802.11 based mobile ad hoc network. In Wireless Communication and Networking Conference (WCNC-2008) , IEEE, pages 1402–1407.

Sharvani G.S, N. C. . T. R. (2009). Different types of swarm intelligence algorithm for routing. In International Conference on Advances in Recent Technologies in Communication and Computing , IEEE computer Society, pages 1–5.

Sim, K. M. and Sun,W. H. (2003). ”Ant Colony Optimization for Routing and Load Balancing: Survey and New Directions”. IEEE Transactions on Systems, MAN and Cybernetics Part A : Systems and Humans, 33(5):560–572.

Singh, G. et al. (2012). ”Ant Colony Algorithms in MANETs: A Review ”. Journal of Network and Computer Applications, 35(6):1964–1972.

Srivastava, V. and Motani, M. (2005). ”Cross Layer Design : A survey and Road Ahead”. IEEE communication Magazine, 43(12):112–119.

Subramanian D, D. P. and J, C. (1997). ”Ants and reinforcement Learning ; A Case Study in Routing in dynamic Networks”. In Procedding of International Joint Conference on Artificial Intelligence (IJCAI), pages 832–838.

T. Srinivasan, T. N., Mahadevan, V., M.Nivedita, G.Sahitya, A.Meyyappan, and J.V.Sahana (2006). Bfbr a novel bird flocking behavior based routing for highly mobile ad hoc networks. In Proceedings of International Conference on Computational Intelligence for Modelling Control and Automation,and International Conference on Intelligent Agents,Web Technologies and Internet Commerce (CIMCA-IAWTIC’06)2006, IEEE.

Teodoro C. Bora, L. d. S. C. and Lebensztajn, L. (2012). Bat-inspired optimization approach for the brushless dc wheel motor problem. IEEE Transactions on Magnetics, 48(2):947–950.

Vivekanand Jha, Khetarpal, K. and Sharma, M. (2011). ”a survey of nature inspired routing algorithms for manets ”. In 3rd IEEE International Conference on Electronics Computer Technology (ICET-2011), volume 6, pages 16–24.

Wang Jianping, O. E., Parimala, T., and ruppa K, T. (2009). Hopnet: A hybrid ant colony optimization routing algorithm for mobile ad hoc networks. Ad Hoc Networks, Elsevier, 7:690–705.

Wooi King Soo, K. K. P., Ling, T. C., and Ang, T. G. (2007). ”Intelligent IEEE 802.11B wireless Networks MAC Layer Diagnostic Controller in Mobile Ad Hoc Networks”. Malaysian Journal of Comupter Science, 20(2).

WU Da Peng, W. M.-q. and yan, Z. (2009). ”Reliable routing mechanism based on neighbor stability for MANET”. The Journal of China Universities of Posts and Telecommunications, 16(3):33–39.

Wyatt, T. D. (2003). ”Pheromones and animal behaviour: Communication by smell and taste”. Cambridge University Press.

Xu, S. and Saadawi, T. (2001). ”Does IEEE 802.11 MAC protocol work well in Multihop wireless Adhoc networks?”. IEEE Communications Magazine, 39(6):130–137.

Yalin Evren Sagduyu, S. L. and Li, J. H. (2010). On the overhead and throughput performance of scent-based manet routing. In The Military Communications Conference-Unclassified Program-Networking Protocols and Performance Track, IEEE.

Yang, X.-S. (2010). ”a new metaheurastic bat-inspired algorithm”. In Nature Inspired Cooperative Strategies for Optimization (NICSO 2010), Studies in Computational Intelligence, Springer.

yuan Zeng, Y. and Yan-Xiang (2005). ”Ant Routing Algorithm for Mobile Ad Hoc Networks based on Adaptive Improvement”. In Proceedings of international conference on wireless communications, networking and mobile computing, volume 2, pages 678–681.

Yumei Wang, K. Y., Liu, Y., and Zhang, H. (2004). Effects of mac retransmission on tcp performance in ieee 802.11 based ad-hoc networks. In Vehicular Technology Conference 2004 , IEEE.

Zheng X, G. W. and R, L. (2004). ”An Ant-Based distributed Routing Algorithm for Ad- Hoc Networks”. In Proceedings of the international conference on communications, Circuits and systems (ICCCAS), IEEE computer Society, pages 412–417.

Zhenyu Liu, M. Z. K. and Constantinou, C. (2004). ”A Swarm Intelligence Routing Algorithm for MANETs”. In Proceddings of the IASTED international conference on communications internet and information technology.

Zhenyu Liu, M. Z. K. and Constantinou, C. (2005). ”A Biologically Inspired QoS routing Algorithm for Mobile Ad Hoc Networks”. In Proceedings of International conference on Advanced information networking and applications , IEEE computer Society, Washington, pages 426–431.

Publications

List of Publications Based on the Research Work:

International Journals

1. Kiran M and G Ram Mohana Reddy,A Study on Termite Based Routing Protocols for Mobile Ad Hoc Networks, International Journal of Information Processing (IJIP), 6(4), 48-54, 2012 (Published by Society of Information Processing, Bangalore, India).

2. Kiran M and G Ram Mohana Reddy, Mobility Aware Termite: A Novel Bio Inspired Routing Protocol for Mobile Ad Hoc Networks, IET Journal of Networks, 2013, DOI: 10.1049/iet-net.2012.0203

3. KiranMand G Ram Mohana Reddy,Design and Evaluation of Load Balanced-Termite: A Novel Load Aware Bio Inspired Routing Protocol for Mobile Ad Hoc Networks, Springer Journal of Wireless Personal Communications, 2013, DOI 10.1007/s11277-013-1453-9.

4. Kiran M and G Ram Mohana Reddy, Bat-Termite: A Novel Hybrid Bio Inspired Routing Protocol for Mobile Ad Hoc Networks, International Journal of Wireless and Mobile Computing, Inderscience (Accepted for Publication).

5. Kiran M and G Ram Mohana Reddy, A Taxonomy of Bio-Inspired Routing Protocols for MANETs, IET Journal of Networks, January 2013 (Communicated).

6. Kiran M and G Ram Mohana Reddy, Load Balanced-Bat-Termite: A Novel Hybrid Load Aware Bio Inspired Routing Protocol for Mobile Ad Hoc Networks, Ad-Hoc Networks, Elsevier, January 2014 (Communicated).

7. KiranM and G Ram Mohana Reddy, Performance Evaluation of Bio-Inspired QoS Aware Routing Protocols of MANETs, Networks, IEEE, January 2014 (Communicated).

Conference Publications

1. Kiran M, G Ram Mohana Reddy, Cross Layer Service Driven Adaptive Retry Limit for IEEE 802.11 Mobile Ad Hoc Networks, IEEE IFIP Wireless Days 2011, pp. 1-3, October 10-12, Niagara Falls, Ontario, Canada.DOI: 10.1109/WD.2011.6098180

2. Kiran M , G Ram Mohana Reddy, Throughput Enhancement of the Prioritized Flow in Self Aware MANET Based on Neighborhood Node Distances, IEEE Int. Conf. on Computer Applications and Industrial Electronics, pp. 497-502, Dec 4-7, 2011, Penang, Malaysia. DOI:10.1109/ICCAIE.2011.6162186

3. Kiran M, Praveenkumar Hoolimath, G Ram Mohana Reddy, Distance based Termite algorithm for Mobile Adhoc Networks, Sixth International Conference on Information Processing ICIP-2012, August 10-12, 2012 , Springer Verlag, Bangalore, India. DOI:10.1007/978- 3-642-31686-9_5

4. Praveenkumar Hoolimath, Kiran M, G Ram Mohana Reddy, Optimized TERMITE: A Bio Inspired Routing Algorithm for MANET, IEEE Int. Conf. on Signal Processing and Communications (SPCOM-2012), pp. 1-5, July 22-25, 2012, IISc, Bangalore, India. DOI:10.1109/SPCOM.2012.6290230

5. Zakeerhusen A.G. , Kiran M, G Ram Mohana Reddy, Cognitive Network Layer-Mobility Aware Routing Protocol , Sixth International Conference on Information Processing ICIP- 2012, Volume 292, pp. 84-92 August 10-12, 2012 , Springer Verlag, Bangalore , India, DOI:10.1007/978-3-642-31686-9 10

Other Publications

1. Yeshavanta Kubusada, Giridhar Mohan, Kiran Manjappa, G. Ram Mohana Reddy A Gene Expression Based Quality of Service Aware Routing Protocol for Mobile Ad Hoc Networks, Computer Networks & Communications (NetCom), Lecture Notes in Electrical Engineering Volume 131, 2013, pp 283-290 DOI:10.1007/978-1-4614-6154-8 28

2. Likewin Thoman, Kiran Manjappa, B.Annappa, G Ram Mohana Reddy Parallelized KMeans clustering algorithm for self aware mobile ad-hoc networks, In Proceedings of the 2011 International Conference on Communication, Computing and Security, Pages 152-155, 2011, DOI: 10.1145/1947940.1947973

Brief Bio-Data

Kiran M

Research Scholar,

Department of Information Technology,

National Institute of Technology Karnataka, Surathkal,

P.O. Srinivasnagar,

Mangalore-575025

Permanent Address

Kiran M

C-Block, C-101,

Mythri Mansion,

Sominakoppa Road, Near Kashipura Railway Gate,

Vinobanagar, Shimoga - 577201

Karnataka.

Phone : 9886539400 , 9449816147

Email : kiranmanjappa@gmail.com