

The copyright © of this thesis belongs to its rightful author and/or other copyright owner. Copies can be accessed and downloaded for non-commercial or learning purposes without any charge and permission. The thesis cannot be reproduced or quoted as a whole without the permission from its rightful owner. No alteration or changes in format is allowed without permission from its rightful owner.



**ALGORITHMS BASED ON SPIDER DADDY LONG LEGS FOR  
FINDING THE OPTIMAL ROUTE IN SECURING MOBILE AD  
HOC NETWORKS**



**KHALIL IBRAHIM GHATHWAN**

**UUM**  

---

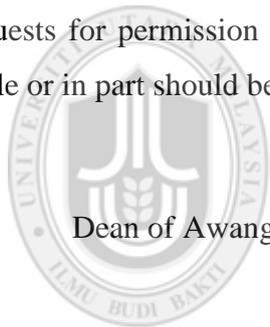
**Universiti Utara Malaysia**

**DOCTOR OF PHILOSOPHY  
UNIVERSITI UTARA MALAYSIA  
2016**

## **Permission to Use**

In presenting this thesis in fulfilment of the requirements for a postgraduate degree from Universiti Utara Malaysia, I agree that the Universiti Library may make it freely available for inspection. I further agree that permission for the copying of this thesis in any manner, in whole or in part, for scholarly purpose may be granted by my supervisor(s) or, in their absence, by the Dean of Awang Had Salleh Graduate School of Arts and Sciences. It is understood that any copying or publication or use of this thesis or parts thereof for financial gain shall not be allowed without my written permission. It is also understood that due recognition shall be given to me and to Universiti Utara Malaysia for any scholarly use which may be made of any material from my thesis.

Requests for permission to copy or to make other use of materials in this thesis, in whole or in part should be addressed to:



Dean of Awang Had Salleh Graduate School of Arts and Sciences

UUM College of Arts and Sciences Malaysia

Universiti Utara Malaysia

06010 UUM Sintok

## Abstrak

Rangkaian Ad hoc Bergerak (MANETs) adalah rangkaian wayarles yang tertakluk kepada serangan yang teruk, seperti serangan lohong hitam. Salah satu matlamat dalam penyelidikan ini adalah untuk mencari kaedah mencegah serangan lohong hitam tanpa mengurangkan daya pemprosesan rangkaian atau meningkatkan penghalauan overhead. Mekanisme penghalauan dalam menggunakan permintaan laluan (RREQs; untuk menerokai laluan) dan dihantar laluan (RREPs; untuk menerima laluan). Walau bagaimanapun, mekanisme ini adalah terdedah kepada serangan hasad nod lohong hitam. Mekanisme ini dibangunkan untuk mencari jalan selamat terdekat dan mengurangkan overhead penggunaan maklumat yang terdapat dalam jadual penghalauan sebagai input untuk mencadangkan algoritma sifat-inspirasi yang lebih kompleks. Kaedah baru dipanggil protokol PGO-DLLA, yang mengubah AODV standard dan mengoptimumkan proses penghalauan. Protokol ini mengelakkan bergantung semata-mata pada kiraan hop dan nombor urutan destinasi (DSNs) yang dieksploitasi oleh nod berhasad jahat dalam protokol piawai AODV. Percubaan oleh metrik prestasi nisbah kelewatan dan paket penghantaran End-to-End dibandingkan untuk menentukan trafik usaha terbaik. Keputusan menunjukkan peningkatan PGO-DLLA penghalauan yang dekat singkat dan selamat daripada serangan lohong hitam dalam MANET. Di samping itu, keputusan menunjukkan prestasi yang lebih baik daripada kerja-kerja yang algoritma yang berkaitan berkenaan dengan semua metrik tidak termasuk pemprosesan yang InterNet adalah yang terbaik dalam routing apabila masa jeda menjadi lebih daripada 40 saat. PGO-DLLA mampu meningkatkan penemuan laluan terhadap serangan lubang hitam dalam AODV. Eksperimen dalam tesis ini telah menunjukkan bahawa PGO-DLL dapat mengurangkan kelewatan beban routing normal, hujung-ke-akhir, dan kehilangan paket dan mempunyai daya pemprosesan dan paket nisbah penghantaran yang baik jika dibandingkan dengan protokol AODV standard, protokol BAODV, dan protokol berkaitan semasa yang meningkatkan keselamatan routing protokol AODV.

**Kata Kunci:** Rangkaian Ad hoc Bergerak, Keselamatan Penghalauan, Serangan Lohong Hitam, Algoritma Berilhamkan Alam

## Abstract

Mobile ad hoc networks (MANETs) are wireless networks that are subject to severe attacks, such as the black hole attack. One of the goals in the research is to find a method to prevent black hole attacks without decreasing network throughput or increasing routing overhead. The routing mechanism in define uses route requests (RREQs; for discovering routes) and route replies (RREPs; for receiving paths). However, this mechanism is vulnerable to attacks by malicious black hole nodes. The mechanism is developed to find the shortest secure path and to reduce overhead using the information that is available in the routing tables as an input to propose a more complex nature-inspired algorithm. The new method is called the Daddy Long-Legs Algorithm (PGO-DLLA), which modifies the standard AODV and optimizes the routing process. This method avoids dependency exclusively on the hop counts and destination sequence numbers (DSNs) that are exploited by malicious nodes in the standard AODV protocol. The experiment by performance metrics End-to-End delay and packet delivery ratio are compared in order to determine the best effort traffic. The results showed the PGO-DLLA improvement of the shortest and secure routing from black hole attack in MANET. In addition, the results indicate better performance than the related works algorithm with respect to all metrics excluding throughput which AntNet is best in routing when the pause time be more than 40 seconds. PGO-DLLA is able to improve the route discovery against the black hole attacks in AODV. Experiments in this thesis have shown that PGO-DLLA is able to reduce the normalized routing load, end-to-end delay, and packet loss and has a good throughput and packet delivery ratio when compared with the standard AODV protocol, BAODV protocol, and the current related protocols that enhance the routing security of the AODV protocols.

**Keywords:** Mobile Ad hoc Networks, Routing Security, Black Hole Attack, Nature-inspired Algorithms

## **Acknowledgement**

All praises to ALLAH for helping me to accomplish this PhD study. Also, my thanks to ALLAH who has seen me through to this level in my academic achievement; I would like to seize this opportunity to extend my gratitude to my supervisor, Professor Dr. Abdul Razak Yaakub, for kindly supervising this study. His priceless instruction and guidance had a greater role in the accomplishment of this thesis.

I would like to thank my wife and my family for everything they have done and the love they have showered up on me. Without their dedication and sacrifices, I would not have come up to this level in life.

I would also like to thank everyone who has assisted me in the School of Computing (SOC), InterNetWorks Research Laboratory, and the School of Quantity (SQS), in Universiti Utara Malaysia (UUM) for their support.

And, finally, I would like to particularly thank my friends who have supported me during my research.

Generally, thanks to everyone who have assisted me in completing my thesis.

## Table of Contents

Permission to Use .....	i
Abstrak.....	ii
Abstract .....	iii
Acknowledgement .....	iv
Table of Contents .....	v
List of Tables .....	x
List of Figures .....	xii
List of Appendices .....	xviii
List of Abbreviations .....	xix
<b>CHAPTER ONE INTRODUCTION .....</b>	<b>1</b>
1.1 Background .....	1
1.1.1 Ad Hoc On-Demand Distance Vector Routing Protocol (AODV).....	5
1.1.2 The Black Hole Problem in Mobile Ad Hoc Networks.....	7
1.2 Problem Statement.....	9
1.3 Research Questions.....	12
1.4 Research Objectives.....	13
1.5 Scope of the Research.....	13
1.6 The Significance of the Research .....	14
1.7 Structure of the Research .....	15
<b>CHAPTER TWO LITERATURE REVIEW .....</b>	<b>18</b>
2.1 Introduction.....	18
2.2 Vulnerability of MANETs .....	18
2.2.1 MANET Security.....	21
2.2.2 Routing Attacks in MANET .....	22
2.2.3 The Black Hole Attack .....	23
2.2.3.1 Black hole Attack Caused by RREQ.....	23
2.2.3.2 Black hole Attack Caused by RREP .....	24
2.2.4 The Roles of DSN and Hop Count .....	24

2.2.4.1 Destination Sequence Number (DSN) .....	25
2.2.4.2 Hop Count .....	25
2.3 Prevention Techniques Against Black Hole Attack.....	26
2.3.1 The Computation Restricted Type .....	27
2.3.1.1 Trusted Neighboring Nodes .....	28
i. Feedback .....	28
ii. Acknowledge Based .....	30
iii. Reputation.....	31
2.3.1.2 Cross Layer Cooperation.....	33
2.3.1.3 Route Redundant and Message Parameter .....	34
2.3.1.4 Other Computation Restricted Types .....	35
2.3.2 The Computation Unrestricted Type .....	36
2.3.2.1 Genetic Algorithm.....	37
2.3.2.2 Fuzzy Logic.....	37
2.3.2.3 Clustering Algorithm.....	38
2.3.2.4 Mobile Agents .....	38
2.3.2.5 Others Computation Unrestricted Types.....	38
2.4 Population Meta-heuristic Algorithms.....	42
2.4.1 Optimization Algorithms .....	42
2.4.1.1 Particle Swarm Optimization (PSO) .....	43
2.4.1.2 Differential Evolution (DE).....	44
2.4.1.3 Bat-Inspired Algorithm (BA) .....	45
2.5 Nature-Inspired Algorithms for MANET .....	46
2.6 Limitation of Current Work .....	49
<b>CHAPTER THREE Research Methodology.....</b>	<b>51</b>
3.1 Introduction.....	51
3.2 Research Clarification.....	53
3.3 Descriptive Study I .....	54
3.3.1 Enhanced Ad hoc On-demand Distance Vector (EAODV).....	56
3.3.2 Shortest Secure Path for Ad hoc On-demand Distance Vector (SSP-AODV)	
.....	58

3.3.3 Parallel Grid Optimization by Daddy Long-Legs Algorithm (PGO-DLLA)	59
3.4 Prescriptive Study	60
3.5 Descriptive Study II	61
3.6 Evaluation of Network Performance	63
3.6.1 Performance Metrics	63
3.6.1.1 The Packet Delivery Ratio (PDR)	64
3.6.1.2 Packet Loss (PL)	64
3.6.1.3 The End-to-End Delay (EtoE)	65
3.6.1.4 Throughput (TH)	65
3.6.1.5 Normalized Routing Load (NRL)	65
3.7 Summary	66
<b>CHAPTER FOUR Design and implementation of the proposed protocol</b>	<b>67</b>
4.1 Introduction	67
4.2 The Design of EAODV Routing Protocol	68
4.2.1 EAODV Messages Format	70
4.2.2 Data Structure of EAODV	70
4.2.3 Route Discovery of EAODV Protocol	70
4.2.3.1 Example I: Process of A* in EAODV	72
4.3 A Dynamic Heuristic Search Algorithm	77
4.3.1 The Proposed SSP-AODV	78
4.3.1.1 Phase One: Shortest Path	81
4.3.1.2 Example II: Process of A* in SSP-AODV	82
4.3.1.3 Phase Two: Prevent Black Hole Attack	86
4.4 PGO-DLLA to Prevent Black Hole Attack	87
4.4.1 Parallel Grid Optimization by Daddy Long-Legs Algorithm (PGO-DLLA)	87
4.4.1.1 Daddy Long-Legs Spider	88
4.4.1.2 The Spider's Leg Behaviors	89
4.4.2 Virtual Daddy Long-Legs Algorithm (VDLLA)	90
4.4.2.1 The Motivations of the Design of VDLLA	91

4.4.2.2 The Process of VDLLA.....	92
4.4.2.3 Implementation of VDLLA.....	96
4.4.3 Problem Formulation and Solution Representation.....	98
4.4.3.1 The Proposed PGO-DLLA Algorithm .....	99
4.4.3.2 Solution Representation .....	102
4.5 Implementation of EAODV, SSP-AODV, PGO-DLLA and BAODV .....	104
4.5.1 Implementing Black Hole Behavior Protocol in NS-2 .....	104
4.6 Summary .....	105
<b>CHAPTER FIVE the results and performance analysis.....</b>	<b>107</b>
5.1 Introduction.....	107
5.2 Empirical Experiments for AODV .....	107
5.2.1 Empirical Experiment on Regular Environment .....	108
5.2.2 Empirical Experiment on Hostile Environment.....	112
5.3 Experiments Setup, Results and Analysis of EAODV Protocol.....	118
5.3.1 Packet Loss: Results and Discussion.....	119
5.3.2 Average End-to-End Delay: Results and Discussion .....	120
5.3.3 Packet Delivery Ratio (PDR) Results and Discussion .....	121
5.3.4 The Normalized Routing Load (NRL) Results and Discussion .....	121
5.3.5 The Throughput Ratio (TH): Results and Discussion .....	122
5.4 Experiments Setup, Results and Analysis of SSP-AODV Protocol .....	123
5.4.1 Packet Loss (PL) Results and Discussion.....	124
5.4.2 Average End-to-End Delay: Results and Discussion .....	125
5.4.3 Packet Delivery Ratio (PDR) Results and Discussion .....	126
5.4.4 The Normalized Routing Load (NRL): Results and Discussion .....	127
5.4.5 The Throughput Ratio (TH): Results and Discussion .....	128
5.5 Evaluation of Virtual Daddy Long-Legs Algorithm.....	129
5.6 Experimental Results of PGO-DLLA Protocol .....	136
5.6.1 Results and Discussion of Comparison between PGO-DLLA and AntNet .....	136
5.6.2 Comparison of PGO-DLLA with AODV and BAODV .....	140
5.7 Performance Comparison of Proposed Protocols with Current Work.....	145

5.7.1 True-link Cross-checking Enhanced AODV Protocol (EDRIAODV) ....	145
5.7.1.1 Comparison of Proposed Protocols with EDRIAODV Protocol.	146
5.7.2 Secure Ad hoc On-demand Distance Vector Protocol (SAODV) .....	152
5.7.2.1 Comparison of Proposed Protocols with SAODV Protocol.....	153
5.7.3 Impact of Black Hole and Gray Hole Attacks in AODV Protocol (IAODV)	
.....	158
5.7.3.1 Comparison of Proposed Protocols with IAODV Protocol.....	159
5.7.4 Fuzzy-Based Intrusion Detection Protocol (Fuzzy-IDs) .....	168
5.7.4.1 Comparison of Proposed Protocols with Fuzzy-IDs Protocol....	169
5.7.5 Swarm Based Intrusion Detection Protocol (SBDT).....	175
5.7.5.1 Comparison of Proposed Protocols with SBDT Protocols.....	175
5.8 Summary .....	181
<b>CHAPTER SIX THE CONCLUSION AND FUTURE WORK .....</b>	<b>182</b>
6.1 Introduction.....	182
6.2 Research Contribution .....	182
6.3 Objectives of the Research.....	184
6.4 Future Work.....	186
<b>REFERENCES .....</b>	<b>188</b>
Integration of BAODV Protocols to NS-2 Environment.....	205
Integration of EAODV Protocol to NS-2 Environment.....	206
Configuration and Installation of EAODV Protocol to NS-2.....	207
Integration of SSP-AODV Protocol to NS-2 Environment.....	208
Integration of PGO-DLLA Protocol to NS-2 Environment.....	210

## List of Tables

Table 2.1	The reasons which make the vulnerabilities in MANETs. ....	19
Table 2.2	Security attributes in MANET. ....	21
Table 2.3	The Contents of the Routing Table in Standard AODV Routing Protocol . .....	26
Table 2.4	Summary of the Computation Restricted Methods Based on The Trusted Neighboring Nodes: Feedback, Acknowledge Based, and Reputation. ....	32
Table 2.5	The Computation Restricted Methods Based on Mechanisms: Cross Layer Cooperation, Route Redundancy, and Message Parameters. ....	36
Table 2.6	The Computation Unrestricted Methods Based on Mechanisms: Genetic Algorithm, Fuzzy Logic, Clustering Algorithm and Mobile Agents. ....	41
Table 2.7	The Nature-Inspired Algorithms for MANET: Ant and Bee Colonies. ....	48
Table 3.1	Simulation Parameters for Regular Environment. ....	61
Table 4.1	Example of Estimated Distances. ....	73
Table 4.2	The positions of each agent (spider). ....	93
Table 5.1	The comparison of performance metrics result for AODV. ....	111
Table 5.2	The Performance Metrics for AODV and BAODV. ....	117
Table 5.3	Simulation Parameters for EAODV, BAODV, and Standard AODV. ..	118
Table 5.4	Simulation Parameters for three Scenarios with SSP-AODV, BAODV and Standard AODV Routing protocol. ....	124
Table 5.5	Performance Results of proposed VDLLA vs. DE vs. PSO vs. BA. ....	132
Table 5.6	P-value generated by T-Test for independent samples comparing VDLLA vs. BA, VDLLA vs. PSO, and VDLLA vs. DE. ....	133
Table 5.7	The PGO-DLLA, SSP-AODV, and EAODV Scenarios identical to the Scenarios in the Simulation of DRIAODV and EDRIAODV Protocols . .....	147
Table 5.8	The PGO-DLLA, SSP-AODV, and EAODV Scenarios Identical to Scenario in the Simulation of SAODV Protocol . ....	154
Table 5.9	The PGO-DLLA, SSP-AODV, and EAODV Scenarios identical to the Scenarios in the Simulation of IAODV Protocol. ....	160

Table 5.10	Comparison Result of EAODV,SSP-AODV, and PGO-DLLA Protocols with IAODV and Standard AODV Protocol.....	164
Table 5.11	The Fuzzy Rule Base.....	169
Table 5.12	The PGO-DLLA, SSP-AODV, and EAODV Scenarios identical to the Scenarios in the Simulation of Fuzzy-IDs Protocol. ....	170
Table 5.13	Comparison Result of EAODV,SSP-AODV, and PGO-DLLA Protocols with FUZZY-IDs and Standard AODV Protocol.....	174
Table 5.14	The PGO-DLLA, SSP-AODV, and EAODV Scenarios identical to the Scenarios in the Simulation of SBDT . ....	176
Table 5.15	Comparison Result of EAODV,SSP-AODV, and PGO-DLLA Protocols with SBDS and Standard AODV Protocol .....	181



## List of Figures

Figure 1.1. Classification of routing protocols in MANETs .....	2
Figure 1.2. The route request message (RREQ) on AODV protocol .....	6
Figure 1.3. The route reply message (RREP) in the AODV protocol .....	6
Figure 1.4. The source node forwards packets to the destination node .....	7
Figure 1.5. Black hole attack: (a) single black hole attack and (b) cooperative black hole attack with two malicious nodes.....	9
Figure 2.1. Passive and Active attacks in MANET .....	22
Figure 2.2. Black hole attack in the AODV routing protocol.....	24
Figure 2.3. The prevention techniques against black hole attack in MANET.....	27
Figure 2.4. The pseudo code of PSO .....	44
Figure 2.5. The pseudo code of DE .....	45
Figure 2.6. The pseudo code of bat algorithm .....	46
Figure 3.1. Research methodology stages.....	52
Figure 3.2. RREQ packet format in AODV routing protocol.....	53
Figure 3.3. RREP packet format in AODV routing protocol .....	54
Figure 3.4. The proposed new protocols.....	55
Figure 3.5. Pseudo code for A* function in the EAODV routing protocol .....	57
Figure 3.6. Pseudo code for Floyd-Warshall function in SSP-AODV routing protocol .....	58
Figure 3.7. Empirical experiments for AODV .....	60
Figure 3.8. Flow diagram for NS-2 scenario implementation .....	62
Figure 3.9. The methods that are used to evaluate performance in networks.....	63
Figure 4.1. The taxonomy of the proposed algorithms design .....	68
Figure 4.2. The elements of EAODV routing protocol .....	69
Figure 4.3. The flowchart of proposed EAODV.....	71
Figure 4.4. Example of six nodes.....	73
Figure 4.5. An example of best route between node (a) and node (b).....	74
Figure 4.6. An example of best route between node (b) and node (d).....	75
Figure 4.7. An example of best route between node (d) and node (f) .....	76

Figure 4.8. An example of the best route {a→b→d→f} .....	77
Figure 4.9. Pseudo code of Dijkstra's algorithm [116].....	78
Figure 4.10. The routing table: SSP-AODV, RREQ-AODV and RREP-AODV.....	79
Figure 4.11. The flowchart of the proposed SSP-AODV .....	80
Figure 4.12. An example of six nodes topology with estimated distance.....	82
Figure 4.13. An example of the first phases of Floyd-Warshall's algorithm .....	83
Figure 4.14. The second and final phases of Floyd-Warshall's algorithm .....	84
Figure 4.15. The first and second steps of Floyd-Warshall's with A* algorithm.....	85
Figure 4.16. Third step of Floyd-Warshall's with A* algorithm.....	86
Figure 4.17. The 8-legs in VDLLA .....	89
Figure 4.18. The Rosenborck's function.....	94
Figure 4.19. The Michalewicz's function .....	94
Figure 4.20. The EggCrate's function.....	95
Figure 4.21. The Beal's function .....	96
Figure 4.18. Pseudo code of VDLLA .....	97
Figure 4.19. PGO-DLLA routing tables .....	100
Figure 4.20. The pseudo code of parallel grid optimization based on virtual daddy long-legs algorithm (PGO-DLLA).....	103
Figure 5.1. Empirical experiment of AODV .....	108
Figure 5.2. The performance metrics of standard AODV with various number of nodes.....	110
Figure 5.3. Comparison result of the performance metrics TH, PL, and NRL.....	112
Figure 5.4. The performance metrics of standard AODV with various pause time ..	114
Figure 5.5. The performance metrics of BAODV with various pause time .....	115
Figure 5.6. Packet loss percentage for AODV, black hole AODV and EAODV .....	119
Figure 5.7. The average end-to-end delay for AODV, black hole AODV and EAODV .....	120
Figure 5.8. The packet delivery ratio for AODV, black hole AODV and EAODV ..	121
Figure 5.9. The normalized routing load for AODV, black hole AODV and EAODV .....	122
Figure 5.10. The throughput ratio for AODV, black hole AODV and EAODV .....	123

Figure 5.11. The packet loss percentage for AODV, BAODV and SSP-AODV .....	125
Figure 5.12. The average end-to-end delay for AODV, BAODV and SSP-AODV ..	126
Figure 5.13. The packet delivery ratio for AODV, BAODV and SSP-AODV .....	127
Figure 5.14 . The normalized routing load for AODV, BAODV and SSP-AODV...	128
Figure 5.15. The throughput ratio for AODV, BAODV and SSP-AODV .....	129
Figure 5.16. PDR results: PGO-DLLA vs. AntNet .....	137
Figure 5.16. End-to- End results PGO-DLLA vs. AntNet.....	138
Figure 5.17. Throughput results PGO-DLLA vs. AntNet .....	139
Figure 5.18. Packet Loss results PGO-DLLA vs. AntNet .....	140
Figure 5.19. PDR results: PGO-DLLA vs. AODV vs. BAODV.....	141
Figure 5.20. End-to-End results PGO-DLLA vs. AODV vs. BAODV .....	142
Figure 5.21. Throughput results PGO-DLLA vs. AODV vs. BAODV .....	143
Figure 5.22. Packet Loss results PGO-DLLA vs. AODV vs. BAODV.....	144
Figure 5.23. The system architecture of the DRIAODV and EDRIAODV [83].....	146
Figure 5.24. The throughput in various mobility models of proposed protocols EAODV, SSP-AODV and PGO-DLLA that are compared with DRIAODV, EDRIAODV and standard AODV protocol .....	148
Figure 5.25. The packet delivery ratio in various mobility models of proposed protocols EAODV, SSP-AODV and PGO-DLLA compared with DRIAODV, EDRIAODV and standard AODV protocol .....	149
Figure 5.26. The normalized routing overhead in various mobility models of proposed protocols EAODV, SSP-AODV and PGO-DLLA compared with DRIAODV, EDRIAODV and the standard AODV protocol.....	150
Figure 5.27. The average delay in various mobility models of proposed protocols EAODV, SSP-AODV and PGO-DLLA Compared with DRIAODV, EDRIAODV and standard AODV protocol .....	152
Figure 5.28. The packet delivery ratio in various number of nodes in proposed protocols EAODV, SSP-AODV and PGO-DLLA to compare with SAODV, BAODV and standard AODV protocol.....	155

Figure 5.29. Throughput in various number of nodes of proposed protocols EAODV, SSP-AODV and PGO-DLLA compared with SAODV, BAODV and standard AODV protocol.....	156
Figure 5.30. The packet delivery percentage in various nodes speed of proposed protocols EAODV, SSP-AODV and PGO-DLLA compared with SAODV, BAODV and standard AODV protocol.....	157
Figure 5.31. The Throughput in various nodes speed of proposed protocols EAODV, SSP-AODV and PGO-DLLA compared with SAODV, BAODV and standard AODV protocol.....	158
Figure 5.32. The packet delivery ratio in various nodes speed of proposed protocols EAODV, SSP-AODV and PGO-DLLA compared with IAODV and BAODV protocol .....	161
Figure 5.33. Normalized routing load in various speed of proposed protocols EAODV, SSP-AODV and PGO-DLLA compared with IAODV and BAODV protocol .....	162
Figure 5.34. The throughput in various speed of proposed protocols EAODV, SSP-AODV and PGO-DLLA compared with IAODV and BAODV protocol .....	163
Figure 5.35. Normalized routing load with various pause times of EAODV, SSP-AODV and PGO-DLLA Compared with IAODV and BAODV Protocol .....	165
Figure 5.36. Normalized routing load with various number of nodes of EAODV, SSP-AODV and PGO-DLLA compared with IAODV and BAODV protocol .....	165
Figure 5.37. The throughput in various numbers of nodes of EAODV, SSP-AODV and PGO-DLLA compared with IAODV and BAODV protocol .....	166
Figure 5.38. Packet delivery ratio in various pause time of EAODV, SSP-AODV and PGO-DLLA compared with IAODV and BAODV protocol.....	166
Figure 5.39. The packet delivery ratio in various numbers of nodes of EAODV, SSP-AODV and PGO-DLLA compared with IAODV and BAODV protocol .....	167

Figure 5.40. The throughput in various numbers of nodes of EAODV, SSP-AODV and PGO-DLLA compared with IAODV and BAODV protocol .....	167
Figure 5.41. The system architecture of the Fuzzy-IDS protocol .....	168
Figure 5.42. Packet delivery ratio in various mobility EAODV, SSP-AODV and PGO-DLLA protocols compared with Fuzzy-IDS and BAODV protocol .....	171
Figure 5.43. Packet delivery ratio in various numbers of nodes EAODV, SSP-AODV, and PGO-DLLA protocol compared with Fuzzy-IDS and BAODV protocol.....	171
Figure 5.44. Routing overhead in various mobility of EAODV, SSP-AODV, and PGO-DLLA protocol compared with Fuzzy-IDS and BAODV protocol .....	172
Figure 5.45. Routing overhead in various number of nodes of EAODV, SSP-AODV, and PGO-DLLA protocol compared with Fuzzy-IDS and BAODV protocol.....	173
Figure 5.46. The average end-to-end delay in various mobility of EAODV, SSP-AODV, and PGO-DLLA protocol compared with Fuzzy-IDS and BAODV protocol .....	173
Figure 5.47. The average end-to-end delay in various numbers of nodes of EAODV, SSP-AODV, and PGO-DLLA protocol compared with Fuzzy-IDS and BAODV protocol .....	173
Figure 5.48. The packet delivery ratio in various numbers of attackers of EAODV, SSP-AODV, and PGO-DLLA protocol compared with SBDT protocol .....	177
Figure 5.49. The average end-to-end delay in various numbers of attackers of EAODV, SSP-AODV, and PGO-DLLA protocol compared with SBDT protocol.....	177
Figure 5.50. The packet loss results in various numbers of attackers of EAODV, SSP-AODV, and PGO-DLLA protocol compared with SBDT protocol.....	178
Figure 5.51. The packet delivery ratio in various numbers of nodes of EAODV, SSP-AODV, and PGO-DLLA protocol compared with DBST protocol .....	179

Figure 5.52. The average end-to-end delay in various numbers of nodes of EAODV, SSP-AODV, and PGO-DLLA protocol compared with DBST protocol ..... 179

Figure 5.53. The packet loss in various numbers of nodes of EAODV, SSP-AODV, and PGO-DLLA protocol compared with DBST protocol ..... 180



## List of Appendices

Appendix A	Implementation of Floyd Warshall .....	199
Appendix B	Integration of BAODV, EAODV, SSP-AODV, and PGO-DLLA Protocols to NS-2 Environment .....	205
Appendix C	Some Important Modification in NS-2 Files .....	213
Appendix D	Generated Input and Output files .....	215
Appendix E	Example of TCL, AWK script and graph file to obtain the result of the Packet Delivery Ratio (PDR) for PGO-DLLA, BAODV, and AODV protocols (see Figure 5.19, pp: 138).....	218



**UUM**  
Universiti Utara Malaysia

## List of Abbreviations

<b>AbmF</b>	Average of Best minimum Fitness
<b>ABR</b>	Associatively Based Routing
<b>AODV</b>	Ad hoc On-demand Distance Vector
<b>ARIADNE</b>	Secure On-demand Routing Protocol for Ad-hoc Network
<b>BA</b>	Bat-Inspired Algorithm
<b>BADOV</b>	Bad Ad-hoc On-demand Distance Vector
<b>BBN</b>	Backbone Network Backbone Nodes
<b>BFS</b>	Best-First Search
<b>BeeAdHoc</b>	A Secure Bee Inspired in Ad-hoc Routing Protocol
<b>BeeAIS</b>	Bee Artificial Immune System
<b>BeeHiveAIS</b>	Artificial Immune System Inspired Security Framework for Beehive
<b>BeeSec</b>	Bee Secure algorithms
<b>BP</b>	Best Path
<b>BR</b>	Best Route
<b>CA</b>	Certificate Authority
<b>CBR</b>	Constant Bit Rate
<b>CEDAR</b>	Core Extraction Distribution Ad hoc Routing protocol
<b>CGSR</b>	Cluster Gateway Switch Routing
<b>CRD</b>	Current Route Discovery
<b>DE</b>	Differential Evolution Algorithm
<b>DSN</b>	Destination Sequence Number
<b>DoS</b>	Denial of Service
<b>DRI</b>	Data Routing Information
<b>DRM</b>	Design the Research Methodology Stages
<b>DSDV</b>	Destination Sequenced Distance Vector
<b>DSR</b>	Dynamic Source Routing
<b>DYMO</b>	Dynamic MANET On-demand Routing Protocol

<b>EAODV</b>	Enhanced Ad hoc On-demand Distance Vector
<b>EDRIAODV</b>	True-link Cross-checking Enhanced AODV Protocol
<b>EtoE</b>	Average End-to-End delay
<b>FREP</b>	Further Replay
<b>FREQ</b>	Further Request
<b>Fuzzy-IDs</b>	Fuzzy Based Intrusion Detection Protocol
<b>GAODV</b>	Gray hole attack in AODV protocol
<b>GLOMOSIM</b>	Global Mobile Simulator
<b>GPS</b>	Global Positioning System
<b>GPSAL</b>	GPS/Ant-Like Algorithm
<b>HiveGuard</b>	A digital-signature-based framework security
<b>IDS</b>	Intrusion Detection System
<b>IAODV</b>	Impact Black Hole And Gray Hole Attack In AODV Protocol
<b>LAR</b>	Location-Based Algorithm
<b>LMR</b>	Lightweight Mobile Routing
<b>LT</b>	Lifetime
<b>LTL</b>	Lifetime of the Leg
<b>MAC</b>	Medium Access Control
<b>MANET</b>	Mobile Ad hoc Network
<b>MbmF</b>	Medium of Best minimum Fitness
<b>MREP</b>	Modify Route Reply
<b>MREQ</b>	Modify Route Request
<b>NRL</b>	Normalized Routing Load
<b>NS2</b>	Network Simulator 2
<b>OLSR</b>	Optimal Link Stat Routing
<b>OTCL</b>	Object Oriented of TCL
<b>PCBHA</b>	Prevention of Co-operation Black Hole Attack in MANET
<b>PD</b>	Pack Drop
<b>PDA</b>	Personal Digital Assistant

<b>PDR</b>	Packet Delivery Ratio
<b>PGO-DLLA</b>	Parallel Grid Optimization by Daddy Long-Legs Algorithm
<b>PL</b>	Packet Loss
<b>PSO</b>	Particle Swarm Optimization
<b>RD</b>	Routes Discovery
<b>RIP</b>	Restricted IP Address
<b>RFR</b>	Request Forward Rate
<b>RREP</b>	Route Request
<b>RREQ</b>	Route Replay
<b>RERR</b>	Route Error
<b>RRR</b>	Route Request Rate
<b>RTE</b>	Route Table Entry
<b>SAODV</b>	Source Ad-hoc On-demand Distance Vector Routing Protocol
<b>SBDT</b>	Swarm Based Intrusion Detection Protocol
<b>SDbmF</b>	Standard Deviation of Best minimum Fitness
<b>SP-value</b>	Shortest path value
<b>SSN</b>	Source Sequence Number
<b>SSP-AODV</b>	Shortest Secure Path for Ad-hoc On-demand Distance Vector
<b>TCL</b>	Tool Command Language
<b>TH</b>	Throughput
<b>THr</b>	Threshold
<b>TORA</b>	Temporally Ordered Routing Algorithm
<b>TTL</b>	Time To Live
<b>UDP</b>	User Datagram Protocol
<b>VDLLA</b>	Virtual Daddy Long-Legs Algorithm
<b>WRP</b>	Wireless Routing Protocol
<b>ZHLS</b>	Zone based Hierarchal Link State Routing Protocol
<b>ZRP</b>	The Zone Routing Protocol

# CHAPTER ONE

## INTRODUCTION

### 1.1 Background

During times of hardship such as wars or natural environmental disasters such as earthquakes, floods, storms, hurricanes, or volcano eruptions, Mobile Ad hoc Networks (MANETs) are considered a good alternative to other types of networks. A MANET is a wireless network that is capable of functioning without any need for base stations or infrastructures.

A MANET consists of a large number of dynamic nodes that are interconnected without any centralized controller; the nodes are always in motion and there is no router device (such as in traditional networks). Each node is a point of contact and is either an endpoint or a distributed point, depending on the network or routing protocol.

The nodes can be any active electronic devices that are installed in the network and that have the ability to forward (send) or receive information over a network channel [1]–[3]. In a MANET, routing protocols are used to route the packets of data that are moving from source to destination nodes with the nodes working as a router for forwarding the data and for discovering the path between the source node and the destination node. This requires each node to use a special mechanism, i.e., some MANET protocol to forward data packets from hop to hop [2].

MANET protocols are divided into three categories. a) Ad hoc On-demand (reactive), b) Table driven, and c) Hybrid routing protocol [2]–[4]. Figure 1.1 shows the classification of routing protocols in MANETs.

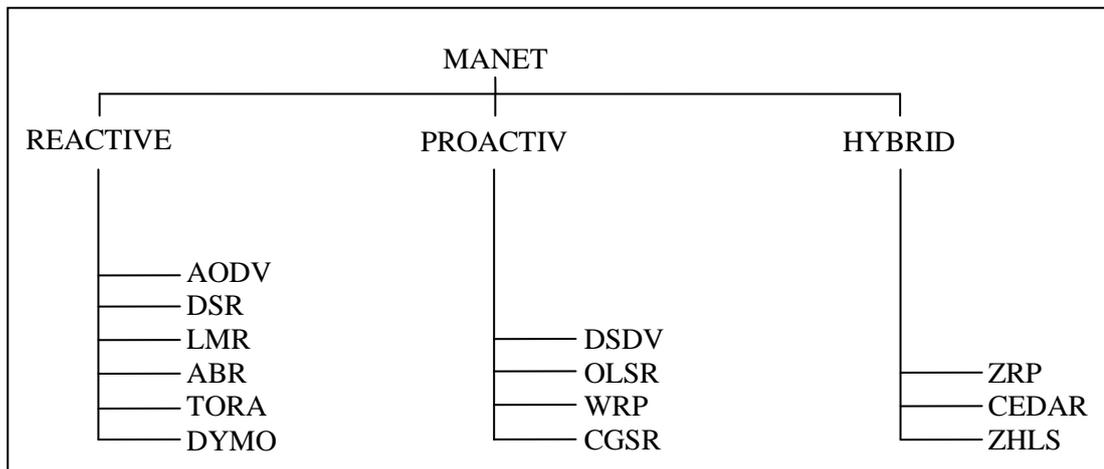


Figure 1.1. Classification of routing protocols in MANETs

Ad hoc on-demand (reactive) routing protocols [5], [6] broadcast discovery packets only when necessary to identify the route to the destination. The overall idea is to find a path between the source and destination nodes.

Recent protocols such as the Ad hoc On-demand Distance Vector (AODV) [5], Dynamic Source Routing (DSR) [7], Lightweight Mobile Routing (LMR) [8], Associatively Based Routing (ABR) [9], the Temporally Ordered Routing Algorithm (TORA) [10], and the Dynamic MANET On-demand routing protocol (DYMO) [11] are good examples of this type of protocol.

In table driven (proactive) routing protocols [12]–[14], every node maintains a routing table in which the layout of the network and information about all possible

destinations is recorded. The routing is achieved by using the routing table information which must periodically be updated by each node. Examples of this type of protocol include the Destination Sequenced Distance Vector (DSDV) [15], Optimal Link State Routing (OLSR) [16], the Wireless Routing Protocol (WRP) [17], and Cluster Gateway Switch Routing (CGSR) [18].

The hybrid (mixed) routing protocols [19] combine the basic properties of the on-demand and table driven routing protocols into a single protocol. The Zone Routing Protocol (ZRP) [20], Core Extraction Distribution Ad hoc Routing (CEDAR) [21], and the Zone-based Hierarchical Link State protocol (ZHLS) [22] are good examples of this type of protocol.

Routing protocols include a mechanism to find a path for data packets to follow from the source to the destination node [4]. The protocols used in traditional wired networks cannot be directly applied in ad hoc wireless networks, due to the unique characteristics of MANETs [23], [24]. The AODV protocol that is studied in this thesis is one of the common on-demand protocols used in MANETs.

According to [25, pp. 70], “the routing plays an important role in the security of the entire network”. Along with the benefits of using a MANET, there are also significant risks posed by vulnerabilities in and threats to MANET security [26]. MANETs are vulnerable to spoofing [25], [27], eavesdropping [28], and Denial of Service (DoS) [25], [27], [29], [30], due to the lack of infrastructure.

In addition, management messages are not authenticated, the routing control messages necessitate encryption, there are weak security practices in the routing protocols, and insufficient protection measures are not taken. In general, MANET vulnerabilities can be exploited by two types of attack: passive attacks and active attacks [25], [31].

Passive attacks are a type of eavesdropping over the network. In passive attacks, the attackers usually focus on breaching communication privacy or anonymity [32]. In contrast, active attacks may modify data, gain authentication, or procure authorization by inserting false packets into the network. Active attacks include DoS, routing table overflow [28], impersonation [28], [33], and black hole attacks [34].

A black hole attack is one of the basic type of attack in a MANET, and it is a severe attack. Black hole attacks are caused by one or more nodes within the network or in the scope of the network's broadcasting. The nodes conduct the attack by claiming that they are on the shortest route to a destination node, in order to intercept packets. In these attacks, black hole nodes need to be the first nodes to respond to a route request in order to intercept the data packets that are transmitted in the network and then keep them.

One of the common on-demand routing protocols is the AODV protocol. According to [26], AODV does not specify any security mechanisms. In the next section, we will present the main AODV protocol mechanisms for route discovery.

### 1.1.1 Ad Hoc On-Demand Distance Vector Routing Protocol (AODV)

The reactive AODV routing protocol [5] is an on-demand routing protocol used for dynamic wireless networks, where nodes can enter and leave the network at any time.

The main ideas of the AODV protocol are as follows:

- the protocol broadcasts a discovery packet only when necessary;
- the protocol identifies a route through local connectivity management neighborhood detection;
- the protocol has general topology maintenance;
- the protocol disseminates information about changes in local connectivity to those neighboring mobile nodes that are likely to need this information; and
- the protocol finds the shortest path between the source and destination nodes.

The AODV routing protocol includes two types of routing processes: route discovery and route maintenance. This research focuses on route discovery process in AODV. Figures 1.2, 1.3, and 1.4 explain the route discovery messages, route request message (RREQs), and route reply messages (RREPs) used in AODV.

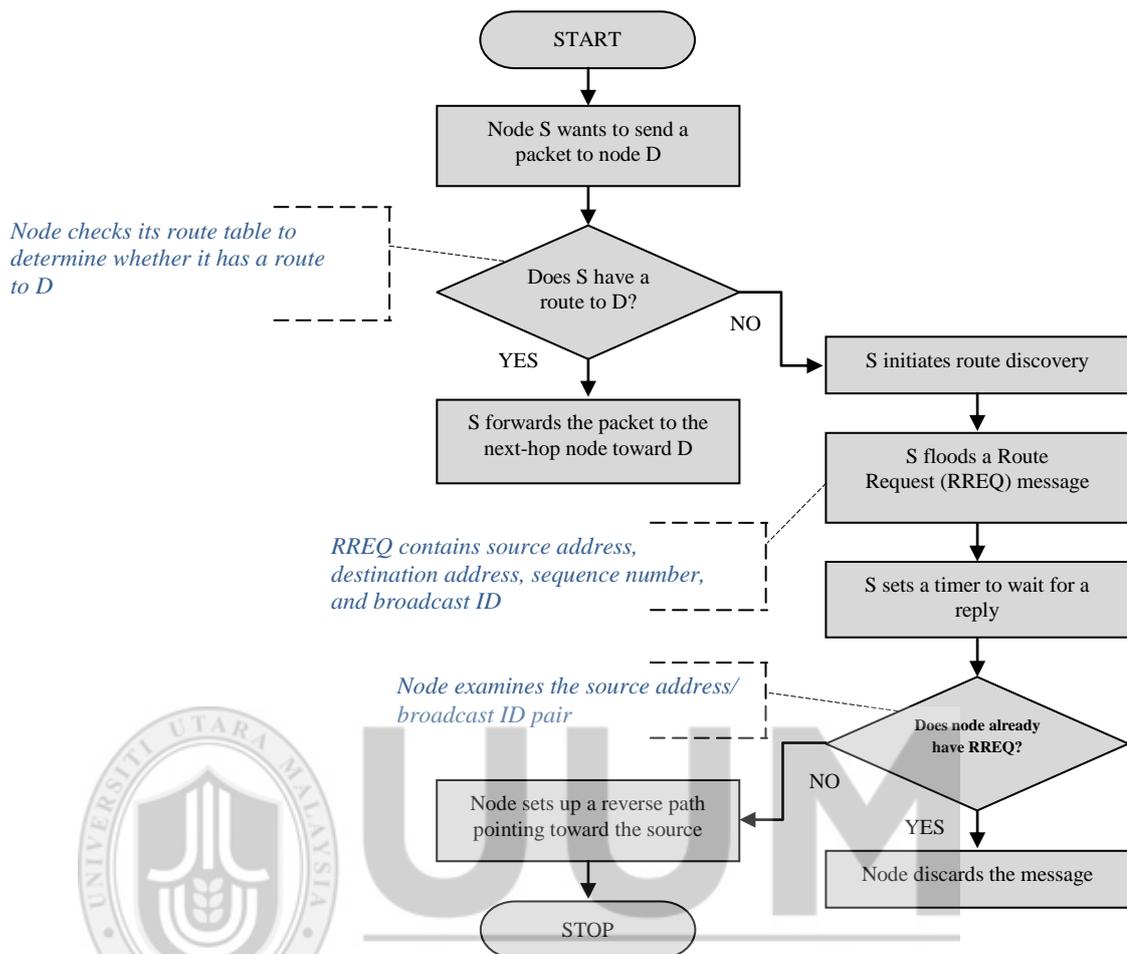


Figure 1.2. The route request message (RREQ) on AODV protocol

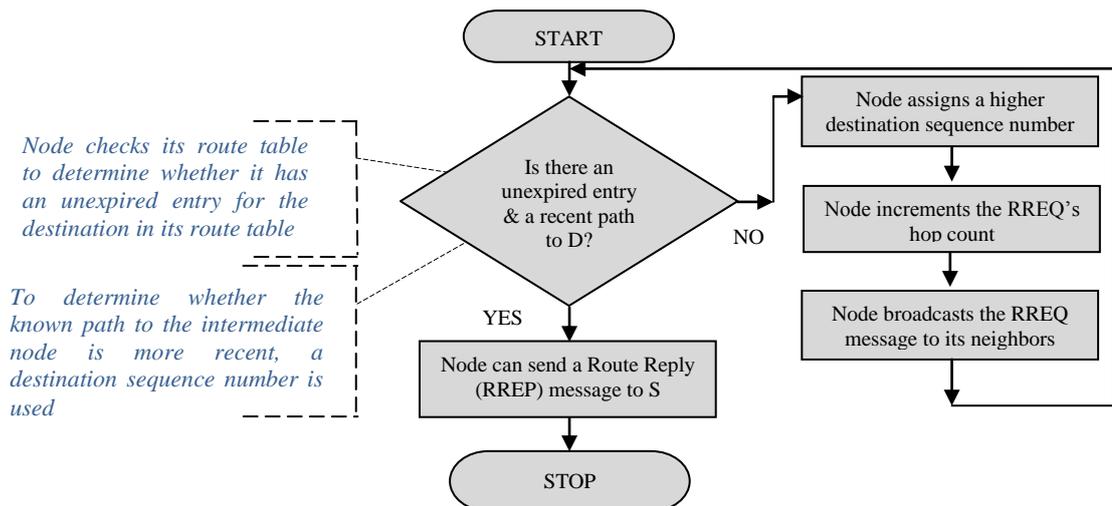


Figure 1.3. The route reply message (RREP) in the AODV protocol

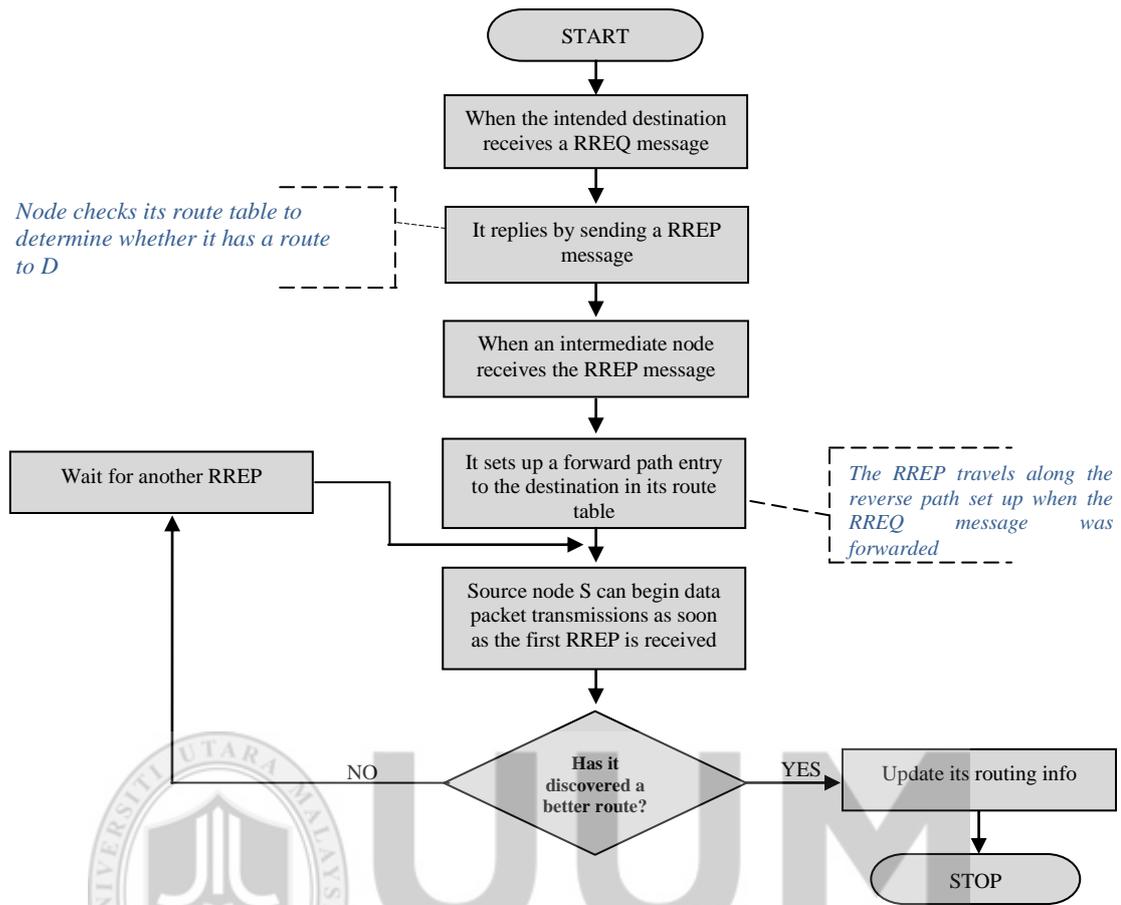


Figure 1.4. The source node forwards packets to the destination node

### 1.1.2 The Black Hole Problem in Mobile Ad Hoc Networks

A black hole attack is one of the basic attacks in MANETs and is caused by one of the nodes in the network (single black hole attack) or within the scope of a private broadcasting network (cooperative black hole attack).

The node or nodes attack the network's mechanism by sending false responses to route requests, claiming that they are on the shortest route to a destination node whose packages they want to intercept. In this process, the attacking node always tries to be

the first to respond to the request and thus intercept and keep the data packets transmitted in the network.

According to [35], a black hole attacker listens to the route requests in a flooding-based protocol. When the attacker receives a request for a route to the destination node, it creates a reply consisting of an extremely short route. If the malicious reply reaches the initiating node before the reply from the actual neighboring nodes, a fake route is created.

Once the malicious device has been able to insert itself between the communicating nodes, it can do anything with the packets passing between them. It can drop the packets passing between them to perform a DoS attack or, alternatively, use its place on the route as the first step in a man-in-the-middle attack [36].

For example, in Figure 1.5, source node S wants to send data packets to destination node D and initiates the route discovery process. We assume that node B is a malicious node that claims it has a route to the destination whenever it receives route request packets, and that it immediately sends a response to node S.

If the response from node B is the first to reach node S, then node S thinks that the route discovery is complete, it ignores all other reply messages, and it begins to send data packets to node B. As a result, all packets through the malicious node are consumed or lost.

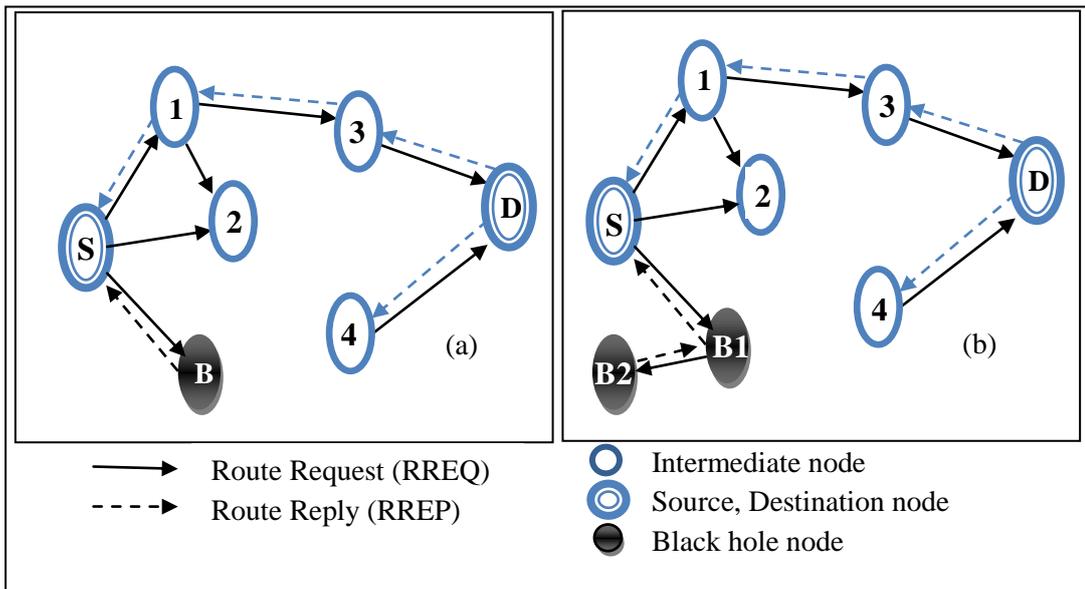


Figure 1.5. Black hole attack: (a) single black hole attack and (b) cooperative black hole attack with two malicious nodes

## 1.2 Problem Statement

A MANET is a modern, dynamic network that has a dynamic condition for auto configuration and maintenance. A MANET has flexibilities that make it a suitable type of network for many applications [32]. Many existing routing protocols in MANETs, such as AODV, are excellent in terms of performance and efficiency but lack security protection and can therefore suffer from several types of attacks [25], [37], [38]. AODV creates routes on an on-demand basis and expands the Distance Vector (DV) to eliminate loops in the routing table. A DV is very simple routing configuration in which it is easier for every node to maintain the distance to each destination than in the structure used by the Link-state algorithm, where each node maintains a view of the network topology information [39], [40].

The simplicity of the DV and the lower delay in connection setup are examples of properties of the AODV routing protocol that make it a very common protocol in MANETs. On the other hand, AODV is also subject to malicious attacks such as black hole attacks [2], [41]–[45]. These attacks have two forms: single and cooperative black hole attacks [2], [43], [44]. The route discovery in AODV uses two indicators to update information. First, it selects the route with a higher destination sequence number (DSN); this indicates the freshness of the route to the destination node. Second, the route discovery selects the route with a better metric (number of hops to the destination) in the case where the DSNs are equal [6].

In AODV protocol, when a source node does not find a path to the destination node in the routing table, it sends a RREQ to all of its neighboring nodes by flooding. In addition, AODV use routing tables to save routing information, which can be a target for attackers to read, making this a another gap that can be exploited by black hole nodes. In a single or cooperative black hole attack, one or more malicious nodes respond to the first RREQ from the source node. It is very easy for a black hole node to change the information in the routing table that comes with the RREQ and send an RREP with fake information. Unfortunately, because this RREP will have the highest DSN value and the smallest hop count, the source node will choose the malicious node as the best node to forward a packet. Black hole nodes use their higher priority to drop all the packets and thus conduct a type of DoS attack in the network. By compromising the information in a routing table and sending fake RREPs to source nodes, black hole attackers can also affect the network's performance - for example, by increasing both end-to-end delay and network overhead.

A lot of research has focused on securing the routing discovery in AODV. Many related work are presented in chapter two, it have proposed several solutions to prevent the black hole attack, and more than one of those solutions involves changing the routing mechanism in the AODV protocol. However, any change to the original route discovery mechanism will not necessarily be useful in all cases, and it may to the contrary lead to undesirable results, such as loops in the routing, a slow rate of convergence, large control overhead, and unnecessary bandwidth consumption [39], [40]. For routing security, the shortest path (least number of hops to the destination) is the best path, as it has the lowest probability of containing malicious nodes [46]–[48]. None of the existing conventional solutions focus directly on solving cooperative black hole attacks during the routing discovery phase in AODV, and most of these solutions are not sufficient to prevent the cooperative black hole attacks during the routing discovery [49], [50].

According to [49], the solutions can be classified into two groups: limited and intensive solutions. The limited approaches consume little power is consumed (as simple processes are performed) compared to what is needed for the intensive approaches (these involve complex processes that consume more power). However, while the limited approaches can offer a solution to the single black hole attack problem, most of these solutions are not accurate and are inadequate for preventing cooperative black hole attacks. In contrast, intensive solutions can be used to increase the protocol's safety and accuracy through an increased ability to prevent cooperative black hole attacks. However, there is a problem with using intensive solutions, as they consume large amounts of power and time to obtain good performance. This problem

makes the intensive solutions incapable of adapting to environments that have limited power and dynamic changes, such as MANETs.

As a consequence, there is a need for a new solution that focuses mainly on preventing black hole attacks in the AODV protocol. The solution must be an enhancement of the AODV route discovery mechanism and avoid the consumption of network resources and develops a heuristic approach that is useful for finding a shortest secure path (namely, a shortest path that is secure).

### **1.3 Research Questions**

To address the issues raised in the problem statement section of this chapter, this investigation aims to answer the following research questions:

1. How can be designed an algorithm to prevent a single and cooperative black hole attacks on the RREQs and RREPs of the AODV routing mechanism without significantly increasing the average end-to-end delay?
2. How should the proposed algorithm for preventing black hole attacks during the routing process in the AODV protocol be implemented?
3. What are the best measures for evaluating the proposed algorithm for preventing black hole attacks in the AODV protocol?

#### **1.4 Research Objectives**

The main objective of this research is to design algorithms that can prevent single black hole attacks and cooperative black hole attacks in AODV routing protocol. The sub objectives of the research are:

1. To design a new algorithm to prevent black hole attacks in the AODV protocol by:
  - a. Improving the route discovery (RREQ and RREP) through a heuristic search algorithm, to enhance the shortest path technique in the routing mechanism,
  - b. Testing the new technique with trusted and malicious environments and comparing the results with those for the original AODV, and
  - c. Improving the route discovery by using a meta-heuristic algorithm to enhance finding the shortest and most secure path to the destination;
2. To implement the proposed algorithm for preventing black hole attacks and to evaluate its performance in the simulated environment; and to evaluate the effect of the proposed algorithm in terms of validation based on the simulation results and then comparing it with the original AODV.

#### **1.5 Scope of the Research**

The scope of the research in this thesis involves the design of a new secure AODV protocol, specifically for a secure AODV routing protocol. The new protocol provides

a solution to the problem of preventing black hole attacks in the AODV route discovery. The research focuses on optimize route discovery and security. The proposed solution consists of three algorithms that can be used as solutions for the single and cooperative black hole attack problem. However, it must be noted that the scope of this research is limited to preventing black hole attacks that come from the route discovery in the AODV protocol. In order to evaluate the proposed protocol, this research uses the network simulator (NS-2) to test each algorithm under the same virtual network environment. The performance of the proposed protocol is evaluated from the five performance metrics: the packet delivery ratio (PDR), the packet loss (PL), the average end-to-end delay (EtoE), the throughput (TH), and the normalized routing load (NRL).

## **1.6 The Significance of the Research**

This research makes the following contributions:

1. Evaluate the proposed protocol with respect to preventing a black hole attack in terms of a specific set of performance metrics;
2. A modification standard AODV and proposed a new routing protocol to prevent single and cooperative black hole attacks with a heuristic search algorithm, a meta-heuristic algorithm, and swarm intelligence;
3. The new parallel algorithm proposed for optimizing routing, security with a new swarm algorithm inspired by a type of spiders found in nature.

## 1.7 Structure of the Research

This thesis has six chapters, including this introductory chapter, which presents background information about the types of black hole attacks that can occur in MANETs.

Chapter Two has two sections: the first section presents a literature review of related studies on preventing black hole attacks with limited computational solutions, while the second section focuses on studies on the use of intensive computational solutions to prevent black hole attacks.

Chapter Three presents the methodology used for conducting this research and is divided into six main sections. The first section is on the empirical experiments, while the second, third, and fourth sections are about creating solutions with artificial intelligence algorithms. The fifth section discusses heuristic search algorithms, meta-heuristic algorithms, and swarm intelligence. The last section of Chapter Three discusses the validation process for the experiments.

Chapter Four presents the design and implementation of the proposed algorithms for improving the routing security in the AODV protocol. The first section proposes an algorithm that uses heuristic search (A\*) [51], [52]. The proposed algorithm is capable of improving a route in AODV by using an estimated value to find a shortest path between the source and destination nodes. The second proposed hybrid algorithm is different from the algorithm proposed in the first section and uses the Floyd-Warshall algorithm [53] to find a distance between two nodes instead using of

estimated distances to discover the shortest secure path. The protocol proposed in this chapter can prevent a black hole attack without needing to use cryptography or authentication. The algorithm is compared with the original AODV protocol with respect to the performance metrics. To improve the local and global search in the search algorithm, we propose an algorithm with a new mechanism that works as parallel agents to optimize the shortest secure path found in the AODV protocol. Finally, all three proposed algorithms are compared with five related proposals. This chapter concludes with a discussion of how to develop the proposed algorithms in the network simulator (NS-2).

Chapter Five is divided into two parts. The first part is intended to demonstrate the performance of AODV in two different environments: regular and hostile environments, to examine the optimal setting of the AODV protocol. A regular environment is a group of nodes (i.e., computers, notepads, mobile phones, and communication devices) that communicate without any attacks. In contrast, a hostile environment is a regular environment with one or more malicious nodes (black hole nodes). This chapter also studies the important performance metrics with various settings of the pause time and different numbers of nodes to predefine an appropriate optimal setting for AODV. The study of the current behavior of AODV involves a set of simulations (empirical experiments) based on NS-2. In the second part of Chapter Five, the results of the proposed protocol performance are analyzed.

Finally, Chapter Six presents concluding remarks about the three proposed algorithms, including a description of their features, capabilities, and weaknesses. The chapter presents some recommendations that can serve as guidelines for further research on using artificial intelligence to improve routing security in MANETs.



## **CHAPTER TWO**

### **LITERATURE REVIEW**

#### **2.1 Introduction**

This chapter reviews the relevant literature related to the research field considered in the thesis. The review of literature is related to studies in preventing black hole attacks in MANETs. Sections 2.1.2 and 2.1.3 discuss the overview of MANET security attributes and types of attacks in MANET respectively. The black hole attacks are discussed in Section 2.1.4, while Section 2.1.4.3 presents the roles of Destination Sequence Number (DSN) and hop count. Section 2.2 displays the investigation of the previous techniques that are used to identify and prevent the black hole attacks at routing in AODV. Section 2.3 displays the nature-inspired algorithms in general, and displays the nature-inspired algorithms for MANET in Section 2.4. Finally, the limitation of current work is discussed in Section 2.5.

#### **2.2 Vulnerability of MANETs**

MANET is a wireless network; it has fundamental characteristics, such as open medium, dynamic topology, and constrained capability. According to [26, pp. 478], “With the network topology changing dynamically and the lack of a centralized network management functionality, these networks tend to be vulnerable to a number of attacks”. However, MANETs are vulnerable to attacks such as black hole attacks. There are several reasons to make the MANET become vulnerable, as shown in Table 2.1.

Table 2.1

*The reasons which make the vulnerabilities in MANETs.*

Reason	About the reason	Examples
Open medium [54], [55]	MANET consists of a huge number of nodes, in which external nodes can join at any time.	Mobile nodes, wireless mobile nodes
The configuration of MANETs [56]	Capable of communicating with others nodes, no infrastructure or centralized administration, difficult to put a routing security mechanism	Self-configuration, self-connected by wireless links automatically
The fundamental of MANETs [57], [58]	The malicious nodes exploit MANET properties in order to constrain its capabilities	Open medium, dynamic topology, distributed cooperation, unrestricted mobility, and connectivity to the users
The nature of nodes connections [59]	MANET nodes have an interchange of packets using hop-by-hop for any node out of range. The nodes always work as a host or router	Hop-by-hop, host or router
The MANET problems [35], [27]	Some components in MANET need to be improved to enhance the quality of service (QoS)	Routing protocols (lack in security), power constraint, limit of bandwidth, special types of attacks such as black hole attacks, fabrication attacks, flooding attacks
Hard environments [60]	MANET has been designed to work in hostile environments, some special networks can monitor animals, sea, or space	Battlefields, military applications, emergency and disaster situations

However, only one of the reasons which make the vulnerabilities in MANETs (in Table 2.1) may increase the probability of exposure to attacks. Since most security solutions in regular networks do not fit with the characteristics of MANET, although those characteristics are the reason to make MANET a very famous network. The absence of a centralized infrastructure makes it difficult to apply the methods of encryption, such as public key cryptography or certification authorities (CA). On the other hand, the node's mobility makes the security mechanisms that are applied in the stable networks not suitable to work with MANET [34].

Furthermore, the other basic characteristic of the decentralization of MANET is the decision-making, however, this may make it easier for the attacker to penetrate the part of the network as the source node. The breakthrough is held responsible for taking an important decision in an important moment by methods of deception or breach that could lead to network failure or collapse. Another problem such as relying on batteries or exhaustible means for energy may be a solution for the attacker to stop the work of the network. This problem of power loss or limitations may be caused by one of the nodes using the method of attack that leads to the depletion of energy in a certain way from a decade of catalytic. This situation lead to a breakdown in communication between them and the rest of the nodes in MANET, and stop broadcasting and data transmission over the network because the principle work in MANET is the multihop.

### 2.2.1 MANET Security

Security is the combination of specific actions of all the processes, that is used to ensure the following requirements: confidentiality, authentication, integrity, non-repudiation, availability, and access control are achieved [61]. The standard key parameter types for networks and the attributes affected by many attacks are shown in detail in Table 2.2.

Table 2.2

*Security attributes in MANET.*

Type	Effects	Details
Confidentiality [62], [63].	It is the keeping of the information unreadable to the unauthorized nodes.	The modification attacks can affect the confidentiality of MANET.
Authentication [64], [65].	Prevent unauthorized users or nodes.	In infrastructure-based network, it is possible to implement a central authority at a point such as a router, base station, or access point.
Integrity [66], [67].	The protecting of the message that is sent during transmission.	The data can be modified or deleted by malicious attackers. The malicious attackers can also resend it, which is known as replay attack.
Non-repudiation [68], [69].	If an entity sends a message, the entity cannot deny that the message was sent by it.	The entity cannot deny that the message was sent by it. The entity cannot later deny the message as well.
Availability [70], [71].	A node should maintain its ability to provide all the designed services regardless of the security state of it.	Security criterion is challenged mainly during the DoS attacks.
Access control [72], [73].	Preventing unauthorized nodes from getting access to the network.	Access control is tied to authentication attributes.

## 2.2.2 Routing Attacks in MANET

The routing attacks in MANET are classified into two types depending on its nature: passive and active attacks. The passive attacks illegitimately gain over the information by listening to traffic in the network without making any damage in the operations of routing protocol [35]. Whereas, the active attacks change the flow of data, or sometimes they insert fake packet and change its contents [74]. The active attacks are severe attacks, and are divided into two types; external and internal attacks [75]. In external attacks, some nodes do not belong to the same network with the victim nodes as they are the attackers. In contrary, in internal attacks, the attackers share the same network with the victim nodes. This attack is done by internal compromised nodes or external adversaries in external attacks. The passive and active attacks are shown in Figure 2.1.

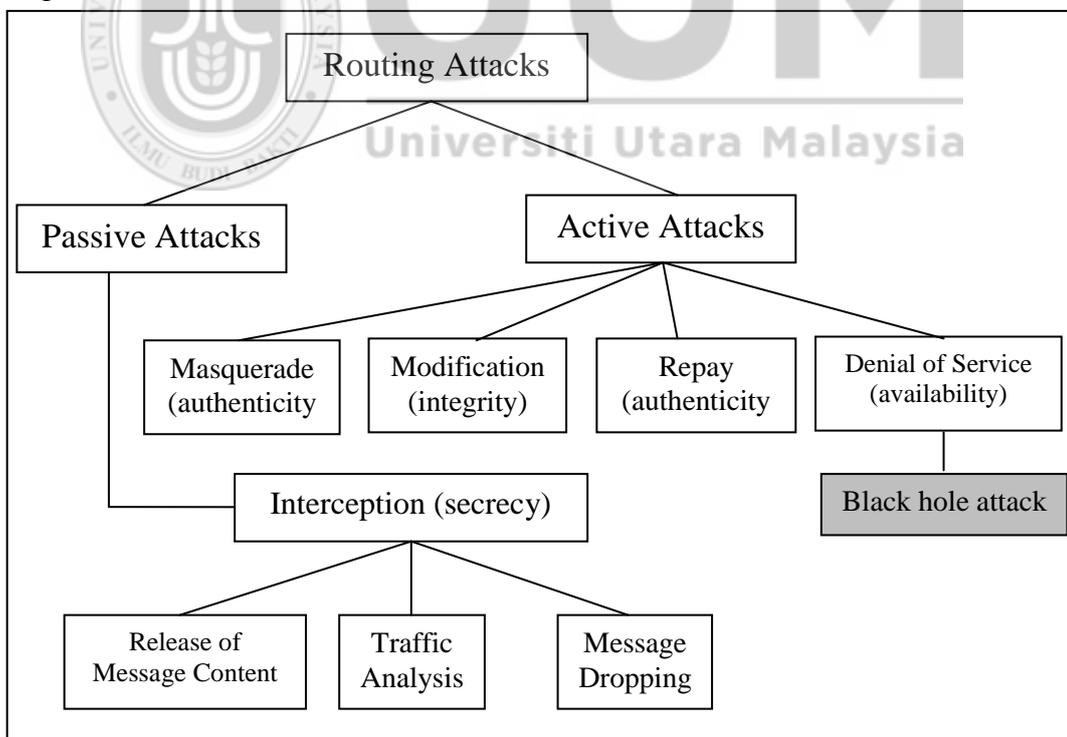


Figure 2.1. Passive and Active attacks in MANET (Adopted from [76])

### **2.2.3 The Black Hole Attack**

Black hole attack is a type of Denial of Service attack (DoS), which is a type of active attack. In a black hole attack, an attacker reduces the quantity of routing information offered to the other nodes by dropping the received routing messages.

The attack also has the action of making the target node out of reach or fall out of communication in the network. The black hole attack in a routing protocol such as AODV can appear in two forms; the black hole that is attacked by RREQ, and the black hole that is attacked by RREP.

#### **2.2.3.1 Black hole Attack Caused by RREQ**

This attack forges a RREQ message to form a black hole attack. In the forged RREQ message, the attacker pretends to rebroadcast a RREQ message with a non-existent node as the source IP address in the IP header. The originating node will update its route to go through the non-existent node to the destination node.

As a result, the route will be broken. The black hole attack can fake the RREQ message by increasing the source sequence number by at least one or decreasing the hop count between the source node and destination node by the forged RREQ message, as shown in Figure 2.2.

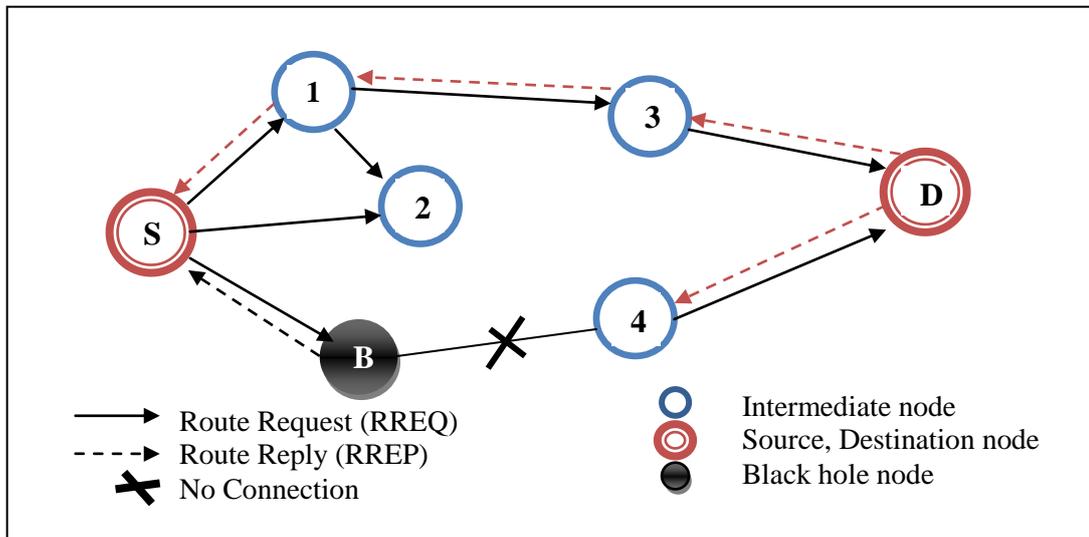


Figure 2.2. Black hole attack in the AODV routing protocol

### 2.2.3.2 Black hole Attack Caused by RREP

The attacker may create an RREP message to form a black hole as a fake RREP message to the originating node. When the originating node receives the fake RREP message, it will update its route to the destination node through the non-existent node to form a black hole, as shown in Figure 2.2.

### 2.2.4 The Roles of DSN and Hop Count

The AODV routing protocol quotes the use of the sequence number from the DSDV routing protocol to supersede stale cached routes and to prevent loops, while the discovery procedure is derived from the one adopted in the DSR routing protocol. More details about the role of DSN and hop count are explained in the following sections.

#### **2.2.4.1 Destination Sequence Number (DSN)**

AODV uses a DSN for each route entry in the routing table, as shown in Table 2.3. However, the DSN is created by the destination node to be included along with any route information it sends to requesting nodes. Using DSN ensures loop freedom, whenever an option is assumed between two routes to a destination, a requesting node is required to select the one with the greatest DSN [5], [77]. Every Route Table Entry (RTE) at each node must contain the latest information available about the sequence number for the IP address of the destination node for which the RTE is maintained. If a node receives new information about the single path from RREQ, RREP, or RERR messages that is regarding the destination, DSN is updated. A destination node increments its own sequence number in two situations:

1. Immediately before a node wants to construct a route discovery, the sequence number will be incremented to prevent the conflicts with previously established reverse routes to the source of a RREQ.
2. When a destination node responds to a RREQ and generates a RREP, it will update its own sequence number to the maximum of its current sequence number and the DSN in the RREQ packet.

#### **2.2.4.2 Hop Count**

Hop Count is a number of the hops needed to reach the destination node [66], [68]. The hop count is a very important field in the routing table that is utilized and managed by the AODV protocol when the route needs to update. For example, in the situation of the sequence numbers of intermediate nodes are equal, the hop count

plays an important role to update the path to the destination node in the routing table, where, the source node is decided to select a node with a smaller value of hop count. Table 2.3 shows the fields of the routing table in a standard AODV routing protocol.

Table 2.3

*The Contents of the Routing Table in Standard AODV Routing Protocol [5].*

The Routing Table
Destination IP Address
Destination Sequence Number (DSN)
Valid Destination Sequence Number flag
Other state and routing flags (e.g., valid, invalid, repairable, being repaired)
Network Interface
Hop Count (number of hops needed to reach destination)
Next Hop
List of Precursors
Lifetime (expiration or deletion time of the route)

### 2.3 Prevention Techniques Against Black Hole Attack

There are two main approaches to solve the black hole attack problem in MANET; simple techniques approach and complex techniques approach [49]. In the simple techniques, the computations are limited and restricted, and those techniques include simple calculations to detect the single black hole nodes. On the other hand, the complex approach uses intensive and complex techniques for protecting the routing against a single and cooperative black hole attacks. In this chapter, the review of black hole attacks is based on the taxonomy that have been proposed in the literature [49]. Figure 2.3 shows the taxonomy of the prevention techniques against the black hole attack in MANET.

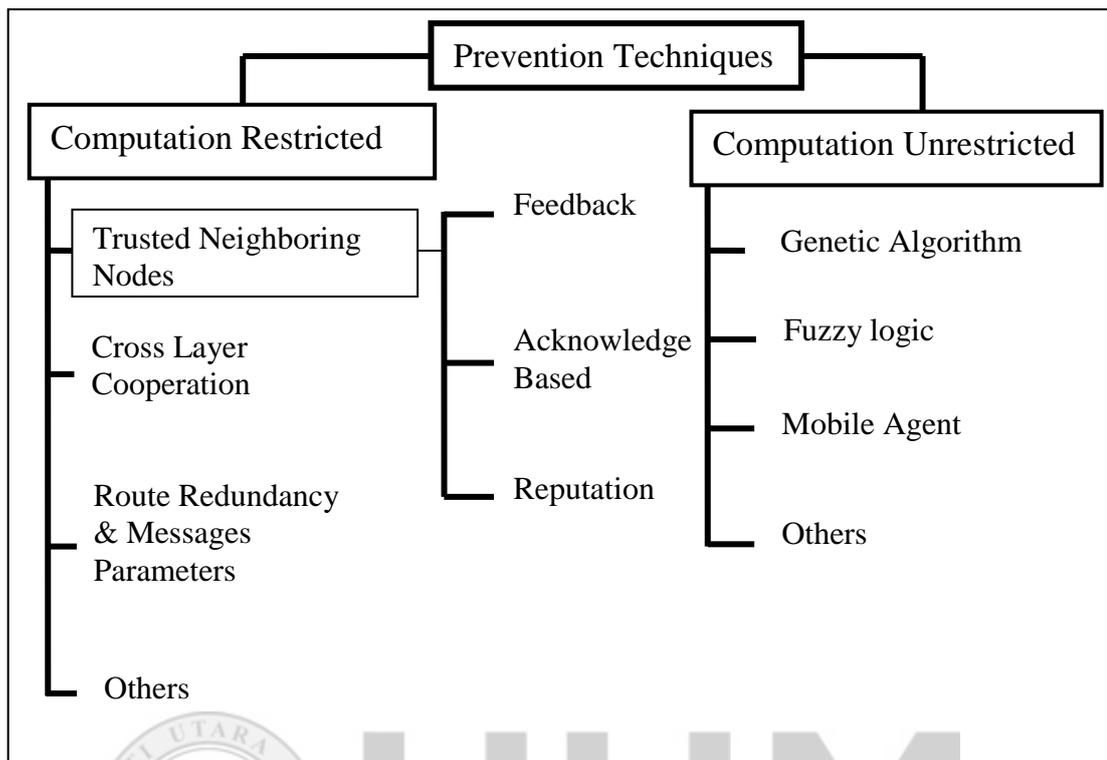


Figure 2.3. The prevention techniques against black hole attack in MANET (Adopted from [49])

### 2.3.1 The Computation Restricted Type

In this section, three main types are displayed; trusted neighboring nodes, cross layer cooperation and route redundant, as the following in the Sections 2.3.1.1, 2.3.1.2, 2.3.1.3, and 2.3.1.4.

In these solutions, there are three techniques; using feedback, using acknowledge-based, and using reputation. In the rest of the chapter, several important previous solutions to prevent a black hole attack in MANETs will be displayed, as shown in Figure 2.3.

### 2.3.1.1 Trusted Neighboring Nodes

In general, the source node uses a neighboring node to find a route to a destination and transmits the packet among the nodes. As a result, the neighboring node takes an important role in the routing, in which the same role must be taken in any new techniques to prevent AODV from a malicious attack.

In the trusted neighboring nodes, there are three techniques; using feedback, using acknowledge-based, and using reputation. Additionally, some of the important notes about these three techniques that are used in [43], [78], [79] as previous solutions to prevent a black hole attack in MANETs are displayed in the following Sections i, ii and iii.

#### i. Feedback

Many of the detection techniques have been proposed to take advantage of the feedback from the intermediate nodes. The source node judges whether a suspected node is a normal or malicious node, depending on the information from neighboring nodes.

In [43], the authors have analyzed the black hole attack and propounded that the destination sequence number must sufficiently be increased by the attacker node in order to convince the source node that the route provided is the optimum.

Based on the differences between the destination sequence numbers of the received RREPs, the authors proposed an approach that has a merit in which the attack can be

detected at a low cost without introducing extra routing traffic without modification of the existing protocol, albeit false positives is a demerit.

In [78] proposed a REAct system for detecting malicious nodes. When the destination node detects a heavy packet drop, the feedback message will be sent back to the source node to trigger the audit procedure.

Then, the source node will choose an audit node to use a bloom filter to generate behavioral proof. The source node also uses a bloom filter to produce behavioral proof and compares with the bloom filter that is generated by the audit node. However, the behavioral proof that is generated by the bloom filter contains only the information of transmitting packets, but not the information of nodes on the forwarding path.

In [79], the authors have assumed that the first RREP received by the source node is from the black hole node. A new protocol (IAODV) has been proposed to eliminate the first RREP control message that is received by the source node and then accept all other replies. However, this method may be inappropriate if there is no attack in the network or maybe not be an adequate method if there are many black hole nodes that are attacking the routing protocol cooperatively.

In general, the feedback technique in literature [43], [78], [79], contrary to perform well with the single malicious attack, it may not be appropriate in the case of collective malicious attacks. The failing to prevent a cooperative black hole attack

make these methods are acceptable methods in preventing of the single black hole attack, where, it consumes a moderate power.

## **ii. Acknowledge Based**

In [80] proposed a secure on-demand ad hoc routing protocol (ARIADNE) based on the DSR protocol [7]. The authors proposed a shared secret key between two nodes, and used a message authentication code. The study focused on using message authentication code in order to authenticate point-to-point messages between these nodes.

The proposed system compares ARIADNE with the original DSR routing protocol. The system performance reached around (41.7%) lower packet overhead compared to the optimized DSR, and about the same percentage on all other metrics. However, their scope is limited to the highly optimized version of DSR that runs in a trusted environment because they did not secure the optimization of DSR in ARIADNE, the proposed system.

However, this technique suffers from slow speed detection in case of multiple nodes of black hole that attack at the same time. This technique needs to consume more bandwidth and it is better in single than cooperative black hole attacks. It needs to send special control messages in case of centralization rather than distribution or hybridization.

### iii. Reputation

In [2] proposed a solution to collective black hole attack in MANETs called PCBHA, “Prevention of Co-operative Black Hole Attack in MANET”. They modified the basic AODV routing protocol [77] with computer simulation using GLOMOSIM (Global Mobile Simulator) [81] to achieve the required security with minimal delay and overhead.

The study focuses on making use of “fidelity tables” and assigning fidelity levels to the participating nodes. The proposed algorithm uses in the simulation a minimum threshold value, which is taken as two units as a test case. To find a valid route, the proposed solution tries up to a maximum of RREQ\_RETRIES TIMES at the maximum Time to Live (TTL) value; otherwise it is declared to have not found a valid route. The experiments are done by using the GLOMOSIM simulator and the metrics that are used to evaluate the performance are compared.

The percentage of the packet delivery ratio increased by 59% using PCBHA when compared with the standard AODV. Hence, the enhancement of a new protocol with packets received through AODV was less than 60% over their system in the presence of cooperative black hole attacks.

In addition, the average end-to-end delay was not very high. As a result, the important point in this study is that it has a solution to collective black hole attacks and make fidelity tables. However, their scope is limited to the ways to reduce the delay in the

network that is happening due to the exchange of fidelity packets in PCBHA to achieve security.

Table 2.4 shows a summary of the solutions that were proposed in this literature review using the trusted neighboring node methods.

Table 2.4

*Summary of the Computation Restricted Methods Based on The Trusted Neighboring Nodes: Feedback, Acknowledge Based, and Reputation.*

Literature	Based Protocol	Highlights	Prevention Functionality
Dynamic Updating Training [43]	AODV	<ul style="list-style-type: none"> <li>The study is focused on the changes of DSN during the routing discovery in different situations.</li> <li>Used the feedback that is coming from neighboring nodes and created some of the clusters, which are free of the black hole nodes.</li> </ul>	All nodes
REAct [78]	All	<ul style="list-style-type: none"> <li>Used the bloom filter to generate behavioral proof.</li> <li>The source node chooses an audit node from the feedback that comes from the neighboring nodes.</li> </ul>	All nodes
IAODV [79]	AODV	<ul style="list-style-type: none"> <li>Supposed a first RREP message came from a suspicious node.</li> </ul>	All nodes
Ariadne [80]	DSR	<ul style="list-style-type: none"> <li>Uses message authentication code in order to authenticate point-to-point message between the nodes.</li> <li>Uses symmetric cryptography.</li> </ul>	One compromised node
PCBHA [2]	AODV	<ul style="list-style-type: none"> <li>Uses “fidelity tables” and assigns fidelity levels to the participating nodes.</li> </ul>	All nodes

### 2.3.1.2 Cross Layer Cooperation

In [82], if both neighboring nodes and the next hop node are black hole nodes, the next hop node can respond to the source node with falsified routing information. Therefore, this scheme is still vulnerable to a cooperative black hole attack. The algorithm uses a methodology to identify multiple black hole nodes working collaboratively as a group to initiate cooperative black hole attacks. The new protocol is a slightly modified version of the AODV protocol by introducing the Data Routing Information (DRI) table and cross checking using Further Request (FREQ) and Further Reply (FREP).

In [83] the authors proposed a time-link cross checking method to prevent the routing discovery from the cooperative black hole attack. A modification towards the original AODV protocol was conducted. In this method, when a source node receives the first RREP, it does not respond directly, but it waits for a specific time to check the data routing information (DRI). A source node has a cache list to save all RREPs and all details of the next hop that are gathered from other nodes. It sends a further request to another path from a list of response paths after checking for repeating next hop nodes, otherwise it chooses a random path.

However, these methods are suffering from the increase of control overhead, due to the exchanges of DRI packets and the true-link cross-checking. The advantage of this technique reduces the network overhead by using only one hop (direct neighbor) and sending the control packets after each packet.

### 2.3.1.3 Route Redundant and Message Parameter

In this type, the nodes can use a certain message parameter such as the sequence numbers to indicate the existence of malicious nodes.

In [84], the authors proposed a verified method to check each RREP packet that is received by the destination node. The proposed method is divided into two phases: suspicion phase and confirmation phase.

The authors adopted a new change in the original AODV mechanism such as delaying the route reply RREP and checking DSN and hop count. Additionally, a control packet has been proposed to the route discovery to confirm a presence of an attacker in the suspicion phase.

In this method, a random number is inserted on the further control message MERQ to find the duplicate random number as a trusted path and send data from source to destination node. This method shows some improvement in PDR compared with the AODV protocol on the hostile environment of black hole attacks. However, the results show a slightly higher value in control overhead.

In [85], the authors proposed two solutions to prevent the black hole attack in the AODV protocol. In the first solution, the source node will wait for RREP to arrive from more than two different nodes. The source node will check for the shared hops and the nodes in this path, or to alarm the other nodes about the black hole attack.

In the second solution, they used the sequence number that was included in any packet header to extract two extra tables. The first table includes a sequence number that will be sent to every node. The second table acts to receive a sequence number from every sender. This solution has more accurate and secure results, but it has a few drawbacks such as the increase in the delay time and the routing overhead, especially in the first solution.

#### **2.3.1.4 Other Computation Restricted Types**

In [86] proposed a secure and efficient MANET routing protocol, the SAODV protocol based on the standard AODV protocol [77], and the AODV protocol with the black hole attack. The authors proposed a directly verified destination node by using the exchange of random numbers. The routing overhead as one of the important performance metrics, it was improved by (8%) in the new protocol compared with the standard AODV protocol. However, their scope is limited to the highly optimized version of AODV that runs in a trusted environment as the safety and efficiency must be better at the same time. Table 2.5 summarizes all the computation restricted methods except the trusted neighboring nodes, which are proposed in the literature review: cross layer cooperation, route redundancy, and message parameters.

Table 2.5

*The Computation Restricted Methods Based on Mechanisms: Cross Layer Cooperation, Route Redundancy, and Message Parameters.*

Literature	Based Protocol	Highlights	Prevention Functionality
DRI [82]	AODV	<ul style="list-style-type: none"> <li>• Uses Data Routing Information (DRI) table and,</li> <li>• Cross-checking using Further Request (FREQ) and Further Reply (FREP).</li> </ul>	All nodes
EDRIAODV [83]	AODV	<ul style="list-style-type: none"> <li>• Using the time-link cross checking method to prevent the routing discovery from the cooperative black hole attack.</li> </ul>	All nodes
SAODV [84]	AODV	<ul style="list-style-type: none"> <li>• Uses multiple paths to find the destination node.</li> <li>• Modify Route Request (MREQ), and Modify Route Reply (MREP).</li> <li>• Uses different random numbers in MERQ to check if any black hole nodes are found.</li> </ul>	All nodes
BAODV & SAODV [86]	AODV	<ul style="list-style-type: none"> <li>• Uses directly verifies the destination node by using the exchange of random number.</li> </ul>	All nodes

### 2.3.2 The Computation Unrestricted Type

In these methods, the researchers have attempted to use a more complex mechanism to prevent black hole attacks such as the mechanisms with fuzzy logic, genetic algorithm, clustering algorithms, and mobile agents. The following sections will show these mechanisms in detail.

### 2.3.2.1 Genetic Algorithm

In [87], the authors proposed a new technique to use the genetic algorithm for analyzing the behaviors of every node in the network by extracting the related parameters such as Pack Drop (PD), Request Forward Rate (RFR), and Route Request Rate (RRR). They built several blocks of data as a string for feeding the input patterns of the genetic algorithm. It is used to extract certain features such as the anomalous behavior of nodes to detect and prevent black hole attacks in the routing protocol. However, most of the IDS are suffering from the redundancy in the check data, as well as from the height of cost of battery consumption and bandwidth at the nodes participating in the detection process.

### 2.3.2.2 Fuzzy Logic

In [88], an IDS against black hole attacks using fuzzy logic was proposed. This system consists of four components, namely fuzzy parameter extraction, fuzzy computation, fuzzy verification module, and alarm packet generation module. On the first stage (fuzzy parameter extraction), the system will examine the behavior of the nodes according to the differentiation in the sequence numbers. In the second stage, the system will create a rule of computation to find the fidelity level of packets. The fidelity level lies between 0-10. In order to verify the fuzzy model, the last stages compare the fidelity level with the threshold fidelity level. Some improvement in the secure routing of the AODV protocol against the black hole attack is done in this method. However, this method suffers from the increase in the routing load and the increase in the end-to-end delay as a result of using the fidelity level.

### **2.3.2.3 Clustering Algorithm**

The authors in [42] used a dynamic anomaly training method, which is one of the learning methods in data mining. In this method, they have created a database that contains the features made up as a result of the black hole attacks as a first stage. In the next stage, they have been compared with the features of regular status. They have used the statistical theory to produce an anomaly threshold by measuring a projection distance. As a result, this method can detect black holes in AODV with low routing overhead, but the false positive is the main drawback of this proposed method.

### **2.3.2.4 Mobile Agents**

In [89], the authors proposed a swarm technique for detecting a malicious attack in MANETs. They used a multipath method to find a trusted connection between the source and the destination node. Ant colony optimization [90] is used to make a distributed mechanism to find a better path. For security, the authors used a trusted neighboring node to monitor an untruthful node in the network. As a swarm technique is based on distributed intrusion detection, the new method had some improvement in standard performance metrics such as packet delivery ratio (PDR) and packet loss (PL). However, the other standard performance metrics need to be improved such as normalize routing load (NRL).

### **2.3.2.5 Others Computation Unrestricted Types**

In [45], the algorithm was proposed as a type of the trusting neighboring method that is based on the feedback from the others nodes and its reputation in the network. It is

a distributed collaborative approach in ad hoc wireless networks. Each node locally and independently acts as an intrusion detection system, at the same time, the nearby nodes work together to monitor a larger area. However, each node is responsible for overseeing the activities of the local data; if an anomaly is detected in the local data, or if the evidence is not sufficient and requires a more comprehensive search, neighboring ID agents cooperate to realize the global intrusion detection. This work is focused on a trusted neighboring node, in order to add a security in the AODV routing protocol in a distributed collaborative approach. However, this algorithm suffers from the increase of the routing overhead as a result of using the anomaly detection technique.

The authors in [91] presented an approach for adding hash based function into the REAct system [78]. They argued that this system may result the source node unable identify which node on the path generates the proof. If the REAct system suffers cooperative attacks, it cannot work. Another cooperative attacker would generate the proof and transmit it to the audit node. Therefore, the proof would cheat the system that the malicious node is on another segment of the path. Hence, this system is only able to detect single malicious nodes. On the other hand, the method uses binary search to find the malicious node which may result the attacker to predict the audit node easily and change its behavior to cheat the source node. Let the behavioral proof contain both information from data traffic and forwarding paths. Both of them are taken into account to help the source node to detect malicious nodes. In other words, the overhead of the mechanism only produces when the path exists a malicious node and the malicious node begins to attack. However, these approaches still make

MANETs suffer packet loss in the initial stage and result in several harms to the network. This kind of detecting mechanism belongs to the reactive method.

In [92] proposed to use the concept of Backbone network Backbone nodes (BBN), which are a group of nodes that are powerful in terms of battery and range. Backbone network is formed with these nodes, which are permitted to allocate Restricted IP addresses (RIP) to newly arrived nodes. The authors assumed that the environment is in the Backbone network. When a source node wants to transmit data, it asks the nearest BBN for an unused RIP. Then the source node transmits RREQ to both destination and RIP. If the source node has just received RREP from destination, this situation means the network is regular and safe. If the source receives RREP from RIP; however, this situation means there are black holes in this route. Therefore, the source node sends a monitor message to alarm the neighboring nodes to go into promiscuous mode and let them start to listen to the network. The source would send dummy data packets to the destination. At the same time, the neighboring nodes can monitor the situation of the forwarding packets. If the packet loss of the monitored nodes is beyond the normal case, the neighboring nodes would notice the source node in the situation. The source node would identify the monitored node as a black hole by receiving the response messages of the neighbors. The network environment assumes that the normal nodes are more than the malicious nodes. Thus, the neighboring nodes may report a failed message when the malicious nodes are more than the normal nodes and the malicious nodes cooperate together. This results in the source node unable to know the exact location of the malicious nodes. On the other hand, the original design of MANETs does not have a Backbone network, therefore this

concept and method is only suitable for special environments. If the method of RIP is the only one used, it cannot lock the black hole and it needs to monitor and observe the suspicious nodes. Table 2.6 summarizes all of the computation unrestricted methods proposed in the literature review.

Table 2.6

*The Computation Unrestricted Methods Based on Mechanisms: Genetic Algorithm, Fuzzy Logic, Clustering Algorithm and Mobile Agents.*

Literature	Based Protocol	Highlights	Prevention Functionality
GAC-IDs [87]	AODV	<ul style="list-style-type: none"> <li>• Uses the <b>GA</b> to analyze the behaviors of every node.</li> </ul>	All nodes
Fuzzy-IDs [88]	AODV	<ul style="list-style-type: none"> <li>• Uses <b>fuzzy logic</b> in the AODV protocol to improve it against the malicious nodes.</li> <li>• Uses fidelity level.</li> </ul>	All nodes
Dynamic anomaly detection scheme [42]	AODV	<ul style="list-style-type: none"> <li>• Uses a <b>clustering algorithm</b> to collect multidimensional features based on the characteristics of attacks and utilizes the projection distance using Principle Component Analysis (PCA) based on statistics.</li> </ul>	All nodes
SBDT [89]	AODV	<ul style="list-style-type: none"> <li>• Uses <b>swarm intelligence</b>-based ant colony optimization.</li> <li>• Uses a multiple path technique to detect and prevent the black hole attack.</li> <li>• Uses the monitoring neighboring node to collect the trusted value.</li> </ul>	All nodes

Many algorithms and techniques have been investigated to highlight the advantages and disadvantages of the security issues in on-demand routing protocols. In particular,

these methods have been proposed to secure the routing of the AODV protocol against black hole attacks. It is clear from all the mentioned works that there are two types of black hole attacks in MANETs. First are single black hole attacks, and second are cooperative black hole attacks. The security issue of the two types is important, but the most important issue concerns with the second type that attacks cooperatively. If the algorithm is designed to solve the problem of cooperative black hole attacks, it may solve the problem of single and cooperative attacks together even if this led to increased costs. The following is a review of the most important ways that are used by the intensity and complex techniques in protecting the routing against black hole attacks.

## **2.4 Population Meta-heuristic Algorithms**

This section includes two subsections: optimization algorithms and the implementation of population Meta heuristic algorithms for MANET.

### **2.4.1 Optimization Algorithms**

Recently, many researchers are attracted to the intelligent collective behavior of animals or insects in nature, such as flocks of birds and colonies of ants. These phenomena can be classified as a kind of innate intelligence, which can be programmed to solve the complex problems such as the optimization problems. Several researchers in this area have proposed powerful methods. Actually, via the optimization algorithms, tough problems can be solved, but it needs simple solutions, not a complex ones. Most of the reviews with meta-heuristic and optimization are

inspired from the simple behavior of biological animals or insects. In the following sections, we will review three meta-heuristic algorithms: Particle Swarm Optimization (PSO), Differential Evolution (DE), and Bat-Inspired Algorithm (BA).

#### **2.4.1.1 Particle Swarm Optimization (PSO)**

Particle swarm optimization is a meta-heuristic optimization method which is based on swarm intelligence that solves complicated problems. The particle swarm optimization algorithm was developed by [93]. The main idea came from a research on birds and fish to conduct the flock's motion. Because of the numerous advantages including simplicity and operational, the algorithm can be used on a large scale for many applications such as speech recognition [94], image segmentation [95], table scheduling [96], road map [97] and dynamic environments [98], swarm robotics [99] etc. PSO is an intelligent algorithm that has succeeded with the scientific research; it is not similar to any other methods in swarm intelligence. It has unique properties such as simple calculations and a high speed searching. However, PSO does not include any mutation or overlapping calculations, but it is related to the movement of particles [100]. Usually, PSO easily falls in the local optima [93]. Generally, PSO starts with an initial solution. These solutions are evaluated to choose the best solution locally and best solutions globally. The solutions are changed based on the new location which highly depends on the velocity of the particles. The new solutions are also evaluated and compared with old solutions. The pseudo code of PSO [93] is illustrated below in Figure 2.4.

```

1   N=number of particle, best_particle=best fitness
2   For i=1 to n do {
3     Initialize particle
4   }
5   For i=1 to end_iteration {
6     For i=1 to n do {
7       find p=fitness (particle)
8       If p better than best_particle
9     }
10    g_best=best_fitness
11  }
12  For i=1 to n do {
13     $V_i = V_i + c_1r_1(P_i, best - P_i) + c_2r_2(g_i, best - P_i)$ 
14     $P_i = P_i + V_i$ 
15  }
16  }}

```

Figure 2.4. The pseudo code of PSO [93]

**2.4.1.2 Differential Evolution (DE)**

One of the important algorithms that have been proposed to solve the hard optimization problems such as the continuous optimization problem is Differential Evolution algorithm (DE). The main contribution of DE is to solve many hard problems in the many tasks of optimization that makes it one of the best methods in single and multi-objective optimization problems [101], [102]. This method has many advantages such as random search, a small number of parameters, and high performance. The DE algorithm can be applied to improve complex optimization problems in high-dimensional approaches; these advantages lead to the successful application of the DE algorithm in different disciplines such as image recognition [103], routing discovery [104], classification [105] etc. However, it has a few drawbacks such as the unstable convergence and the easy drop in the regional optimum. The pseudo code of DE [106] is shown in the following Figure 2.5.

```

1 initialize X=Random generated solution
2 set weight F ∈ [0,2], crossover probability Cr ∈ [0,1]
3 While stopping criterion { For i=1 to n {
4     For each xi, randomly choose 3 distinct vectors xp, xr and xq
5         vit+1 = xpt + F(xqt - xrt)
6         Generate a random index Jr ∈ [1,2,...,d] by permutation
7         Generate a randomly distributed number ri ∈ [0,1]
8     For j=1 to n do {
9         uj,it+1 = { vj,it+1 if ri ≤ Cr or j = Jr
10                { xj,it if ri ≤ Cr or j ≠ Jr
11                }
12                { uit+1 if f(uit+1) ≤ f(xit)
13                { xit otherwise
14                }
15                t=t+1
16            }

```

Figure 2.5. The pseudo code of DE [106]

### 2.4.1.3 Bat-Inspired Algorithm (BA)

This method is inspired by the bats' behavior in their search for prey through the use of sound frequencies that are reflected from the walls and the surrounding areas. It has been developed recently by [107]. This method has been successful in engineering applications [108], where it is characterized by the ease of implementation and gives a perfect result when applied to improve the optimization in complex problems [108], [109]. The echolocation behavior of bats is one of the phenomena that have been developed as a new algorithm in optimization. As swarm intelligence, bat algorithm needs to use distance and velocity to search for prey. Bat algorithm is now becoming a powerful method for solving many tough optimization problems. Some new algorithms have been proposed to avoid a non-active bat algorithm in high dimensions

[106], [107], [110]. The pseudo code of bat algorithm [107] is illustrated in the following Figure 2.6.

```

1   initialize population  $x_i$  ( $I = 1, 2, \dots, n$ ) and  $v_i$ 
2   initialize frequencies  $f_i$ , pulse rate  $r_i$  and the loudness  $A_i$ 
3   While ( $t < \text{Max number of iterations}$ )
4     generate new solutions by adjusting frequency,
    and update velocities and locations/solutions
         $f_i = f_{\min} + (f_{\max} - f_{\min})\beta$ ,
         $v_i^t = v_i^{t-1} + (x_i^t - x_*)f_i$ ,
         $x_i^t = x_i^{t-1} + v_i^t$ ,
5     if ( $\text{rand} > r_i$ )
6         select a solution among the best solutions
7     Generate a local solution around the selected best solution
8     end if
9     Generate a new solution by flying randomly
10    if ( $\text{rand} < A_i \& f(x_i) < f(x^*)$ )
11        accept the new solutions
12    increase  $r_i$  and reduce  $A_i$ 
13    end if
14    rank the bats and find the current best  $x^*$ 
15    end while

```

Figure 2.6. The pseudo code of bat algorithm [107]

## 2.5 Nature-Inspired Algorithms for MANET

In this section, there will be a review of the specific classes of nature-inspired algorithms that have been proposed to discover the shortest paths in the routing protocols in MANETs: the ant and bee colonies.

In [111], the authors have proposed an extended mechanism for the AODV routing protocol named Ant-AODV. They used a new method to take advantage of the ant colony algorithm and the on-demand routing table in the AODV protocol, to make a new routing discovery.

In the conventional ant-based routing techniques, a large number of data packets are being dropped due to the route break, and at the same time, the source node still keeps on sending data packets while being unaware of the link breakage. At the suggestion of the proposed Ant-AODV, some adaptive features come from the advantages in the AODV mechanism such as the local connectivity maintenance, and from the advantages of ant-based routing algorithm such as the shortest path selection in the route discovery process of ants. However, the experiment results of Ant-AODV show an improvement in terms of End-to-End delay and routing latency compared with the standard AODV and ant-based routing, yet the packet delivery ratio is still low.

The authors in [112] proposed another algorithm to improve the shortest path selection in the route discovery using the ACO algorithm in MANET. In this work, and as an agent for global search, it takes advantage of the Global Positioning System (GPS) device in order to receive locations of nodes which act as a local search in a specific area. The experiment results of GPS/Ant-Like Algorithm (GPSAL) show an improvement in terms of routing overhead compared with Location-Based Algorithm (LAR), but it has several drawbacks such as the limited size of the area that is covered as a work area of the GPS.

In [113], [114], the authors have used the artificial immune system to propose a security framework for beehive (BeeHiveAIS), and a digital-signature-based framework security (HiveGuard). In this proposed framework, each MANET node has a software model that consists of three parts: dance floor, packing floor, and trance floor, named (hive).

They have been using the new proposal to analyze the security threats in MANET. Moreover, they have used a digital signature as a security framework and artificial immune system models for securing an ad hoc system, named Secure Bee Inspired in Ad hoc Routing Protocol (BeeAdHoc). However, the experiment results of the Bee Secure algorithms (BeeSec) and Bee Artificial Immune System (BeeAIS) based on BeeAdHoc shown an improvement in the network overhead compared with the AODV and DSR routing protocols. Table 2.7 summarizes all the nature-inspired algorithms for MANET which are selected in this chapter as an example of the literature of the shortest path selection for routing discovery in MANETs.

Table 2.7

*The Nature-Inspired Algorithms for MANET: Ant and Bee Colonies.*

Literature	Based Method	Highlights	Security
Ant-AODV [111]	AODV/Ant colony	<ul style="list-style-type: none"> <li>• Uses the Ant colony and on-demand on AODV to build the routing discovery of Ant-AODV.</li> </ul>	None
GPSAL [112]	ACO	<ul style="list-style-type: none"> <li>• Uses Ant agent for global search.</li> <li>• Uses a GPS device to receive locations of nodes as a local search in a specific area.</li> </ul>	None
BeeSec [113], [114]	BeeAdHoc	<ul style="list-style-type: none"> <li>• Uses the public key encryption for authentication.</li> </ul>	Yes
BeeAIS [113], [114]	BeeAdHoc	<ul style="list-style-type: none"> <li>• Uses the artificial immune system for the development of an AIS-based security framework, BeeAIS.</li> <li>• Evaluation of the security features of three protocols: BeeAdHoc, BeeSec and BeeAIS.</li> </ul>	Yes

## 2.6 Limitation of Current Work

To prevent the black hole attack in mobile ad hoc networks, two approaches are proposed: computation restricted and computation unrestricted. Each proposed approach aims to either effectively prevent black hole nodes to packet dropping or effectively detect the black hole nodes.

However, the methods that have been proposed in [43], [78], [79] depend on the feedback from neighboring nodes, it works with single black hole attacks, but it may fail to prevent a cooperative black hole attack.

In [43], the authors have proposed a new approach to detect the attack at a low cost without introducing an extra routing traffic and without the modification of the existing protocol; however, the false positives are a drawback here.

In [78], the authors have proposed that the source node will choose an audit node to use a bloom filter to generate behavioral proof; however, the behavioral proof that is generated by the bloom filter contains only the information of transmitted packets, and not the information of nodes on the forwarding path.

In [79], the authors have proposed a method for preventing the cooperative black hole attack, however, this method inappropriate if there is no attack in the network or may not be an adequate method.

In addition, the Acknowledge-Based technique [80] suffers from slow speed detection in case of multiple nodes of black holes that attack at the same time.

This technique needs to consume more bandwidth and it is better to prevent a single black hole attack rather than a cooperative black hole attack. Using the Reputation approach as proposed in [2], it has a limitation in their scope because it needs to reduce the delay in the network caused by the exchange of fidelity packet in PCBHA to achieve security.

In Cross Layer Cooperation [82], [83], these methods suffer from the increase of control overhead, due to the exchanges of DRI packets and the true-link cross-checking. The route redundant and message parameter techniques in [84], [85], also have drawbacks in time delay and increasing routing overhead.

As for the computation unrestricted type in [42], [87], [88], an Intrusion Detection System is works to analyze an anomaly for malicious activities and produce reports about this behavior. However, most of the IDS are suffering from the redundancy in the check data, as well as from the height of cost of battery consumption and bandwidth at the nodes which are participating in the detection process [45].

## **CHAPTER THREE**

### **RESEARCH METHODOLOGY**

#### **3.1 Introduction**

This chapter presents the research methodology in order to provide an understanding of the representation and validation steps in this thesis. Design the research methodology (DRM) is used in this thesis [115].

The DRM consists of four stages: research clarification (Section 3.2), descriptive study I (Section 3.3), prescriptive study (Section 3.4), and descriptive study II (Sections 3.5 and 3.6). Details about the research methodology stages of this thesis will be provided in the following sections.

Figure 3.1 shows the research methodology stages, which include the initial studies concerning the standard AODV protocol, the formulation of an algorithm for preventing black hole attacks, and the evaluation of its performance metrics.

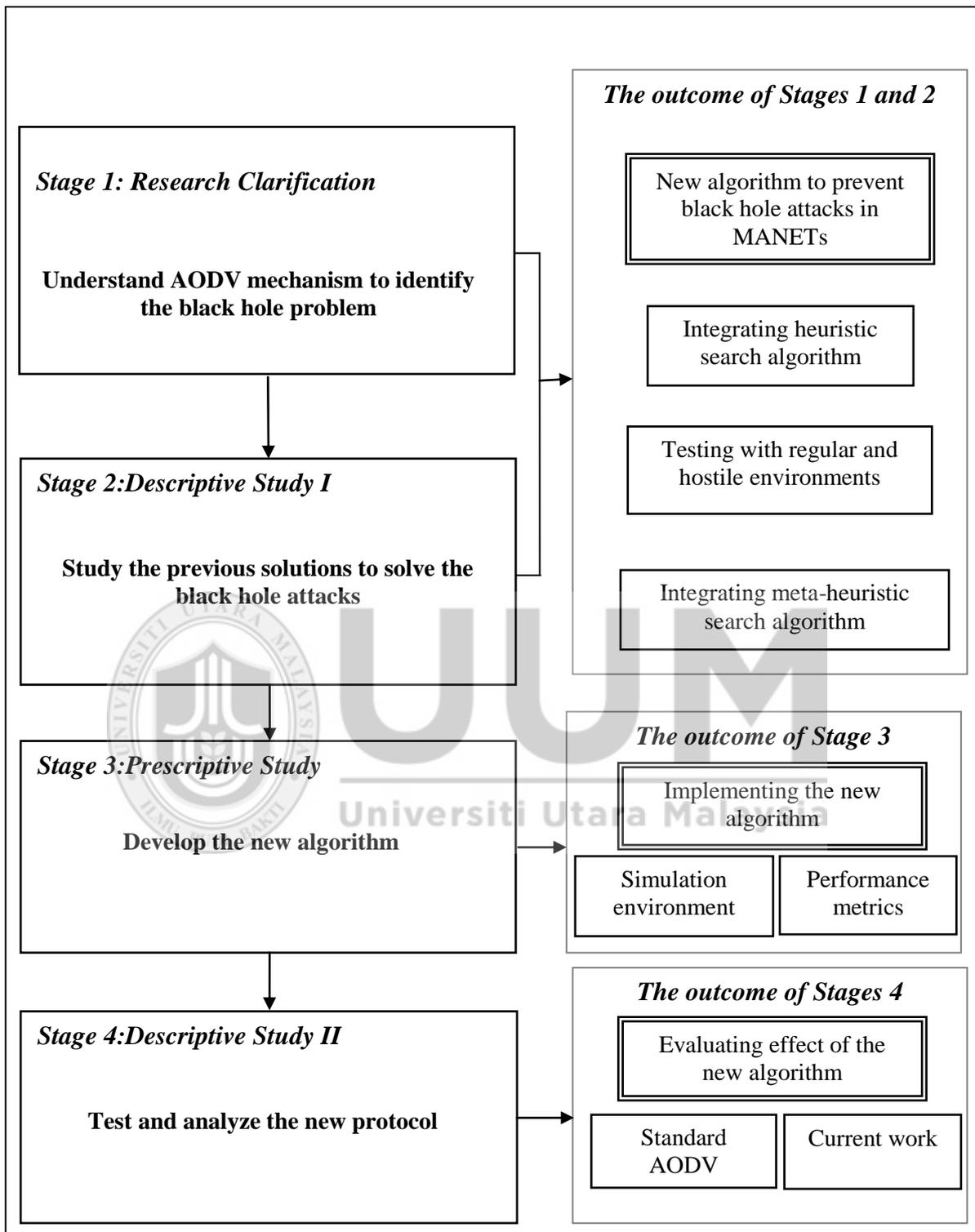


Figure 3.1. Research methodology stages

### 3.2 Research Clarification

This stage aims to identify the vulnerability and security problems in the AODV routing protocol. The main goal is to understand the AODV routing protocol; we will analyze the problems in the AODV mechanism by studying previous research. A large amount of data and information has been collected to identify why the AODV routing protocol is vulnerable to the malicious attacks. In this stage, we will also analyze the problems in the AODV mechanism by a study of malicious attacks and the behavior of black hole nodes during the routing discovery, with the aim of understanding how the route request (RREQ, broadcast packets; see Figure 3.2) and route reply (RREP, unicast to a single neighbor; see Figure 3.3) control messages are involved. We will study in detail the role of the destination sequence number (DSN) and hop count indicators in the routing mechanism as a weakness that is exploited by attackers.

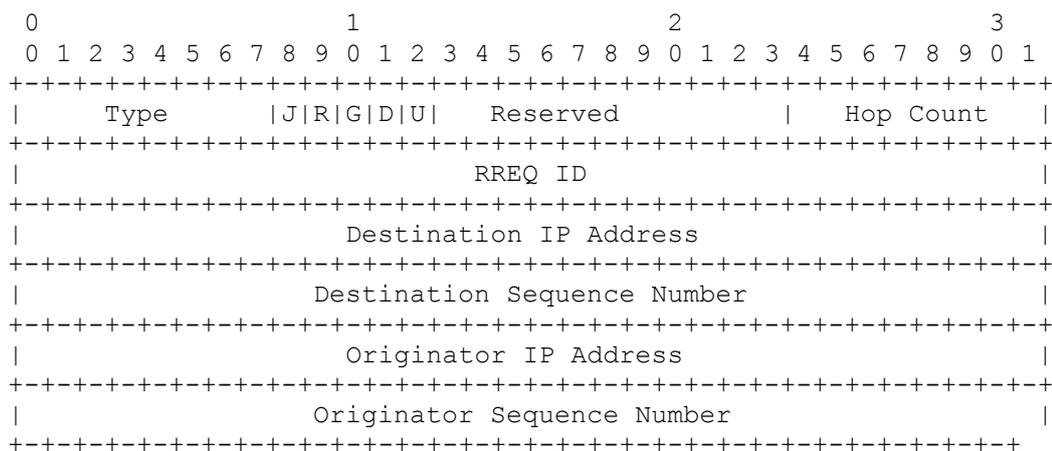


Figure 3.2. RREQ packet format in AODV routing protocol [77]

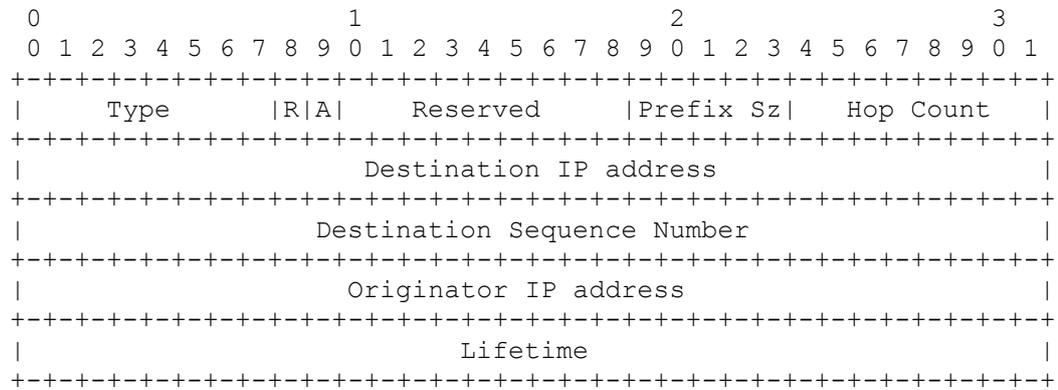
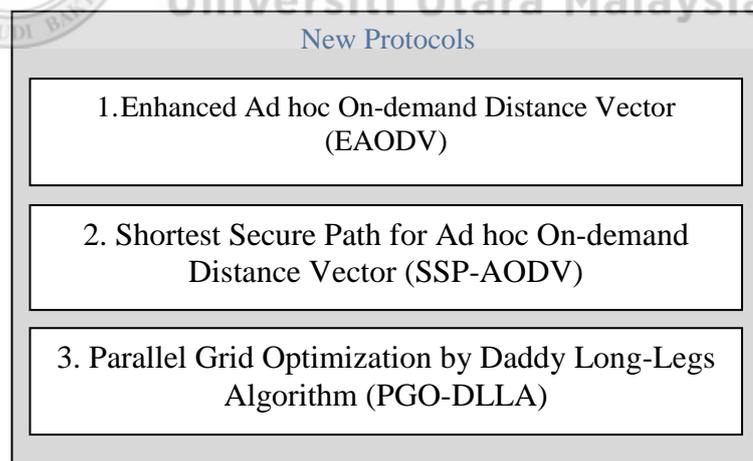


Figure 3.3. RREP packet format in AODV routing protocol [77]

### 3.3 Descriptive Study I

The main objective of this thesis is to propose a new protocol to prevent black hole attacks by using heuristic and meta-heuristic algorithms. Most of the related studies use classical security methods such as cryptography and authentication to prevent black hole attacks in the AODV protocol, and most of these methods suffer from high end-to-end delays and network overhead. The routing algorithms have a scheme for transferring a message from one node to another, and the quality of the path affects the algorithm performance. The balance of traffic over the source and destination nodes as routers nodes and the intermediate node as hosts may contribute to the improvement of performance metrics such as packet loss, end-to-end delays, and throughput. On the one hand, the shortest path is less likely to be attacked than the longer paths. In addition, some changes always occur during the attacks, i.e., destination sequence numbers or hop counts are changed.

However, the first outcome of this stage is to try to optimize a shortest path (with the fewest hops) and thus improve the route discovery. As shown in Figure 3.5, the proposed heuristic search algorithm integrates the A\* and Floyd-Warshall algorithms, and the results are compared with standard AODV in environments with different levels of hostility. The A\* heuristic search algorithm is used for optimizing the shortest path, while the Floyd-Warshall algorithm is used to make the route discovery more dynamic at the outset. However, while this improvement of the route discovery process can reduce the severity of black hole attacks, it cannot completely eliminate them. The protocol proposed in Stages 1 and 2 needs to be improved to make it work in very difficult environments where cooperative black hole attacks can occur. However, in this section, we are determining the initial definitions in order to develop appropriate solutions to prevent black hole attacks in AODV. The proposed algorithms are shown in Figure 3.4.



*Figure 3.4.* The proposed new protocols

### 3.3.1 Enhanced Ad hoc On-demand Distance Vector (EAODV)

As noted in the previous chapter, we are studying how artificial intelligence techniques can be used to reduce the distance that data need to travel during transmission. We expect the potential of attack to be reduced when artificial intelligence algorithms are used, because reducing the travel distance from the source to the destination node will reduce the chance of single and cooperative black hole attacks in the network. A heuristic search can improve search efficiency by providing an estimate of the remaining yet unexplored distance to a goal. Neither depth-first search, breadth-first search, nor Dijkstra's algorithm take advantage of such estimates, and these algorithms are therefore called uninformed search algorithms. A\* is different from these algorithms in that it performs a heuristic search. The heuristic A\* search finds a shortest path by combining greedy and uniform cost searches, in a sense. It uses cost and evaluation functions to determine its ordering. The idea of the A\* search algorithm is to find the path from the source to the destination node that has the least cost. It uses a distance function  $g(n)$  and a cost heuristic function  $h(n)$  to determine the order in which it visits the nodes in a network. The A\* algorithm is outlined in Figure 3.5. It is utilized in many applications and has proven successful in problem solving. Equation (3.1) defines the evaluation function  $f(n)$  of the original A\* heuristic search algorithm [52], [116]:

$$f(n) = g(n) + h(n) \quad (3.1)$$

where  $n$  is the node,  $g(n)$  is the distance,  $h(n)$  is the estimated cost from  $n$  to the goal, and  $f(n)$  is the estimated total cost of the path through  $n$  to the goal.

```

Function A* Heuristic Search( ) return BestPath
1:Begin
2:Inputs:HopCount, Estimated Time, Current, Temp
3:Local variables: FN, GN, HN
4:Temp=GN(Source)
5:For i =1 to Current do
6:   GN(i) = HopCount ( i ) + HN ( i )
7:   If GN ( i ) < Temp
8:     Then Temp = GN(i)
     End If
9:   Else:
     End For
10:BestPath = Temp
11:End Function

```

Figure 3.5. Pseudo code for A\* function in the EAODV routing protocol

All the communication takes place in two phases: the control information phase and the transmission phase. The first phase sets the route and specifications, while the second phase is the factual flowing of information. For control information, we will use the same control messages RREQ and RREP that are used in the AODV routing protocol. Our change is to the second phase, where we use the A\* algorithm. At the beginning, the position of each node is calculated by its Cartesian coordinates along the three axes (see Equation 3.2), to form the estimated values that will serve as the initial values to start the algorithm on a grid system:

$$D = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \quad (3.2)$$

### 3.3.2 Shortest Secure Path for Ad hoc On-demand Distance Vector (SSP-AODV)

In EAODV, the route discovery technique uses the estimated values calculated by Equation 3.2. The algorithm is static, as the rounded values or routes do not change as long as the node stays within the specific grid. We can find the shortest distance between a pair of nodes using a single source shortest paths algorithm such as best-first search or Dijkstra's algorithm, but this requires passing through the entire graph several times. The best solution is to apply the all-pairs shortest path Floyd-Warshall algorithm. This algorithm can be used to determine the length of the shortest path between two nodes in any graph. Figure 3.6 shows pseudo code for the Floyd-Warshall algorithm.

```
Procedure Floyd-Warshall( int n, int w[1..Node i,1..Node j])
1:Begin
2:array d[1..n, 1..n]
3:For i = 1 to n do // Phase One //
4:For j = 1 to n do
5:    d[Node i, Node j] = w[Node i, Node j]
6:    d[Node i, Node j] = ∞ } // If {node i , node j} is not a path //
7:End For i, End For j
8:For i = 1 to n do // Phase Two //
9: For j = 1 to n do
10:For k = 1 to n do
11:  If (d[Node j, Node i] + d[Node i, Node k] < d[Node j, Node k]) then
12:    d[Node j, Node k] = d[Node j, Node i] + d[Node i, Node k]
    //new shorter path length//
13:End if
14:return d
15:End For k, End For j, End For i
16:End Procedure
```

Figure 3.6. Pseudo code for Floyd-Warshall function in SSP-AODV routing protocol

### 3.3.3 Parallel Grid Optimization by Daddy Long-Legs Algorithm (PGO-DLLA)

Improvement of the routing alone may not succeed in highly dynamic environments, which are subject to more severe types of attacks, as we pointed out in preceding sections of this chapter. AODV may be augmented with a new phase to improve security with classical ciphers, but this may increase the network overhead. We suggest solving this problem for SSP-AODV by adding a new security layer, using a delay before sending the packet to the first node that replies with a RREP in order to check the safety of the path. We expected the integration of the two previous techniques to improve the prevention of single black hole attacks. The dynamic routing discovery can be done by integrating a meta-heuristic search algorithm into the protocol.

The techniques to prevent cooperative black hole attacks must be more accurate than the existing proposals, must be heuristic, and must not be one of the classical solutions. Some restricted solutions might be useful for preventing single black hole attacks, but they are not enough to prevent cooperative black hole attacks. Solutions that use complex computations may be more active and more accurate and can deal with severe black hole attacks. The use of artificial intelligence techniques may be the solution. A natural inspiration comes from a research area for information models that studies the collective behavior of insect or animal swarms. In summary, this research propose a new protocol to address black hole attacks through new search algorithms and improving routing security with swarm intelligence.

### 3.4 Prescriptive Study

The aim of this section is to find the optimal settings for the AODV protocol. This section includes two different scenarios: an empirical experiment with regular environments that tests the standard AODV protocol with different numbers of nodes and an empirical experiment with hostile environments to determine the appropriate pause time. Figure 3.7 gives an overview of the empirical experiments for AODV.

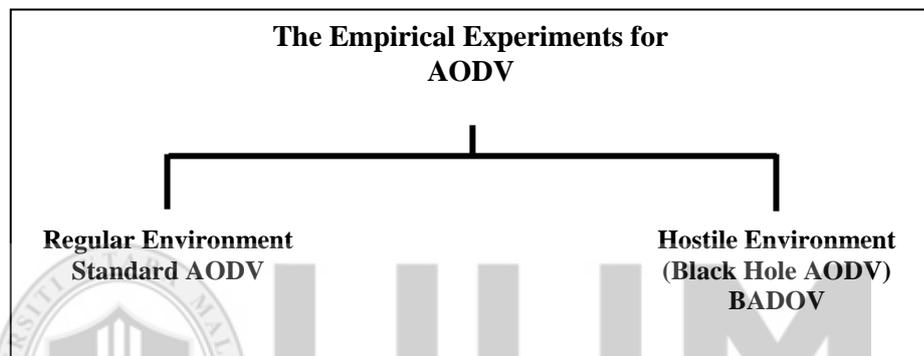


Figure 3.7. Empirical experiments for AODV

In these scenarios, we will set up different parameters for standard AODV with various numbers of nodes (10, 20, 50, and 100) and different environments. CBR is used as a traffic model; different numbers of nodes are distributed within a random waypoint model for mobility. The map area of simulation is 800 x 800 m is a square area is a perfect area to test the empirical experiments in MANETs. The transmission range is 250 m nominal bit rate of 2 MB/Sec and a nominal transmission range of 250 meters with an omni directional antenna. The rest of the simulation parameters is indicated in Table 3.1.

Table 3.1

*Simulation Parameters for Regular Environment.*

Parameter	Regular Scenario
<i>Simulation Time</i>	1000 sec.
<i>Number of Nodes</i>	10, 20, 50, 100
<i>Routing Protocol</i>	AODV
<i>Traffic Model</i>	CBR (UDP)
<i>Pause Time</i>	0
<i>Maximum Mobility</i>	60 m/sec.
<i>No. of sources</i>	1
<i>Map area</i>	800m x 800m
<i>Transmission Range</i>	250m
<i>Malicious Node</i>	0

Some of the parameters that depict the distinctive behavior of an ad hoc network are varied as follows:

1. network size: determines the connectivity by the number of nodes that are moving inside the area of networking. The size of network is important parameter in MANET because is effect of the routing of network.
2. mobility: the rate of topological changes. The mobility is the depends on nodes speed, it is affect the performance metrics in MANET.
3. offered network load: the rate of packet transmission.

### **3.5 Descriptive Study II**

This section presents the implementation of a new routing protocol in NS-2 to simulate the black hole behavior.

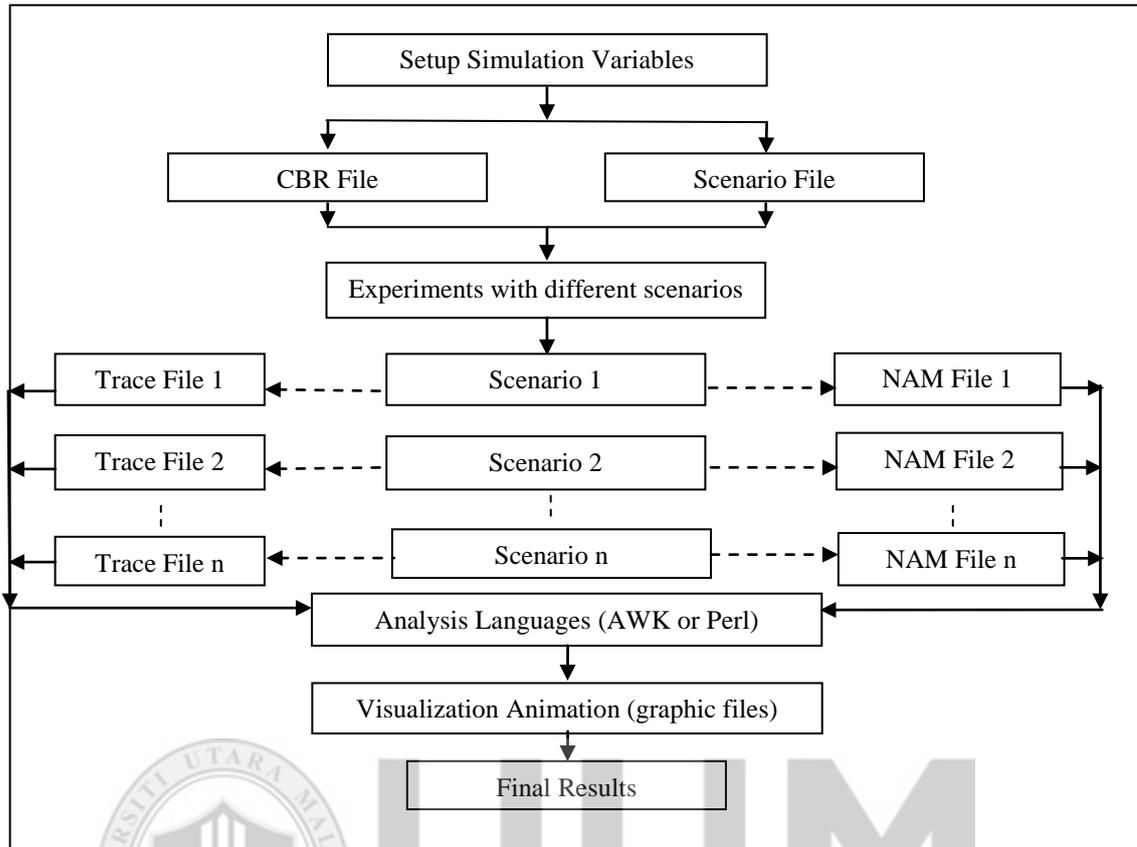


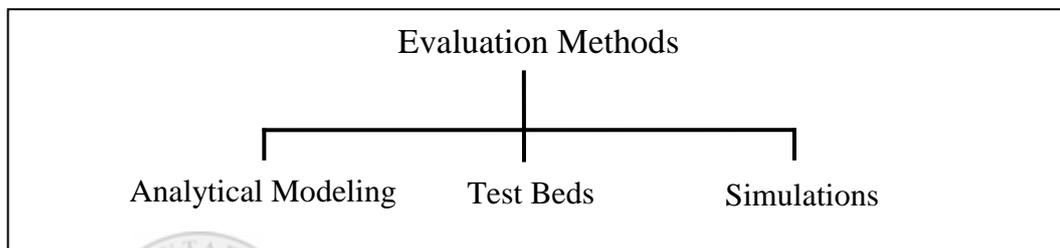
Figure 3.8. Flow diagram for NS-2 scenario implementation

The simulator uses two main languages, object-oriented TCL (OTCL) and C++. OTCL is used for writing simulation scripts, and it has an interpreter for translating and running a simulation step by step. The C++ language provides the ability to update an existing protocol or develop a new one. The two languages are fully compatible. OTCL is used to write a script and setup the simulation variables. Before NS-2 starts a compilation, two files must be input. The implementation is done using the AODV protocol to add the nodes that exhibit the black hole behavior. The new routing protocols are implemented using the high-level language C++ to compile in the NS-2 network simulator. It can support all requirements of design and comparing

a new protocol with the others related protocols in wireless high mobility environments.

### 3.6 Evaluation of Network Performance

Three popular research methodologies are used in network research to evaluate performance: analytical modeling, test beds, and simulations (see Figure 3.9).



*Figure 3.9.* The methods that are used to evaluate performance in networks

Simulations will be used as the performance evaluation method in this research. It is necessary to understand that, in the case of mobile networks, many factors are involved in developing an analytical modeling, and the relations between these factors are still not perfectly understood. Such factors include mobility speed, traffic load, and network size. Moreover, the exact effects of each factor on network performance are not accurately understood, which provides further justification for using simulations to study mobile networks [117].

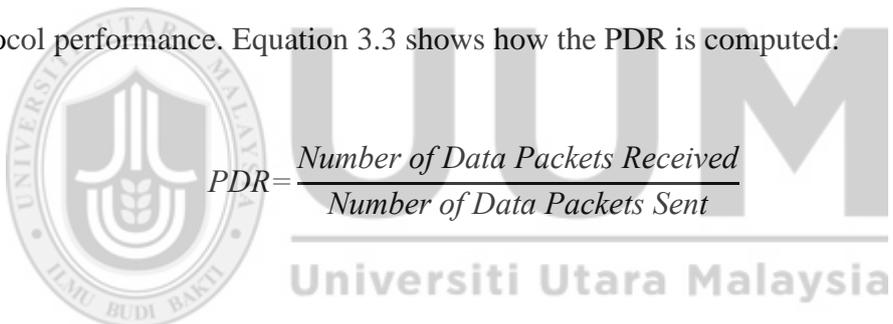
#### 3.6.1 Performance Metrics

Five performance indicators are used to measure the performance of the proposed algorithms for evaluation purposes [6], [50], [118]–[120]: the packet delivery ratio

(PDR), the packet loss (PL), the average end-to-end delay (EtoE), the throughput (TH), and the normalized routing load (NRL). The details of these performance metrics are given in the following subsections.

### 3.6.1.1 The Packet Delivery Ratio (PDR)

The PDR is defined as the number of data packets received by the destination divided by the number of data packets sent [6], [121]–[123]. This metric indicates the amount of data that is actually received by the destination. The PDR depends on the packet size, the network load, and the change of topology. The PDR is important for measuring the best traffic effort. A larger package delivery ratio indicates better protocol performance. Equation 3.3 shows how the PDR is computed:


$$PDR = \frac{\text{Number of Data Packets Received}}{\text{Number of Data Packets Sent}} \quad (3.3)$$

### 3.6.1.2 Packet Loss (PL)

The PL is the percentage of packets that are lost during the simulation. A lower PL rate indicates better protocol performance [118], [123]. Equation 3.4 shows how PL is computed:

$$PL = \frac{\text{Number of Packets Sent} - \text{Number of Packets Received}}{\text{Number of Packets Sent}} \quad (3.4)$$

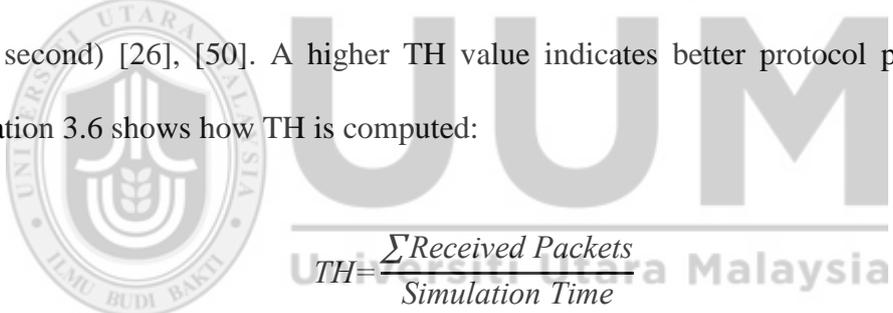
### 3.6.1.3 The End-to-End Delay (EtoE)

The EtoE is the average time taken for data packets to reach the destination. Only the data packets that are successfully addressed and delivered are counted [6], [50], [123]. A lower EtoE indicates better performance. Equation 3.5 shows how the EtoE is computed:

$$EtoE = \frac{\sum \text{Packets Arrival Time} - \text{Transmission Time}}{\sum \text{Connections}} \quad (3.5)$$

### 3.6.1.4 Throughput (TH)

TH is the number of packets received (bytes delivered) per unit of simulation time (per second) [26], [50]. A higher TH value indicates better protocol performance. Equation 3.6 shows how TH is computed:


$$TH = \frac{\sum \text{Received Packets}}{\text{Simulation Time}} \quad (3.6)$$

### 3.6.1.5 Normalized Routing Load (NRL)

The NRL is the number of routing packets such as RREQ and RREP that move per data packet delivered to the destination [6]. The NRL is an important metric as a measure to determine the efficiency of consumed network resources. Equation 3.7 shows how the NRL is computed:

$$NRL = \frac{\sum \text{Sent Packets}}{\sum \text{Data Packets Received}} \quad (3.7)$$

### 3.7 Summary

The main objective of this thesis is to construct a protocol for preventing black hole attacks in MANETs based on artificial intelligence techniques. The methodology that is used in this thesis is an experimental methodology called DMR, which includes four stages: research clarification, descriptive study I, prescriptive study, and descriptive study II.

The methodological stages of this thesis have been explained in detail in this chapter. The proposed artificial intelligence techniques for the AODV mechanism were described in Section 3.3.

Three main phases in the prevention algorithm design are proposed: enhanced ad hoc on-demand distance vector using the A\* algorithm, shortest secure path for the ad hoc on-demand distance vector using the Floyd-Warshall algorithm, and parallel grid optimization using the daddy long-legs algorithm.

The new routing protocols are implemented using NS-2 network simulator. It can support all requirements of design and comparing a new protocol with the others related protocols in wireless high mobility environments. Thus, we choose NS-2, because it is suitable for the design and implementation of the new protocol. Furthermore, NS-2 is a discrete event-driven simulator open source, free distribution, and it has high flexibility.

# **CHAPTER FOUR**

## **DESIGN AND IMPLEMENTATION OF THE PROPOSED PROTOCOL**

### **4.1 Introduction**

The previous chapter presented the research methodology for the design of the proposed protocols. In this chapter, the proposed design and implementation of the new algorithms will be displayed and discussed. This chapter presents the heuristic and meta-heuristic search algorithms that are used to prevent the black hole attacks in the AODV protocol in order to provide an understanding of the representation of design and implementation in NS-2.

As the first algorithm is integrated in the routing discovery of AODV, Sections 4.2 provide the elements of the EAODV protocol design. The proposed SSP-AODV is presented in Section 4.3. Section 4.4 provides the nature-inspired technique that is proposed as a solution to prevent the black hole attacks in PGO-DLLA. The implementation of EAODV, SSP-AODV PGO-DLLA and BAODV is discussed in Section 4.5. Figure 4.1 shows the design of the proposed algorithms' taxonomy, which includes the heuristic A\* in the EAODV protocol, Floyd-Warshall algorithm in the SSP-AODV protocol, the new nature inspired algorithm termed as Virtual Daddy Long-Legs Algorithm (VDLLA), and the integration of the VDLLA algorithm with the AODV routing protocol, termed as a Parallel Grid Optimization algorithm in (PGO-DLLA).

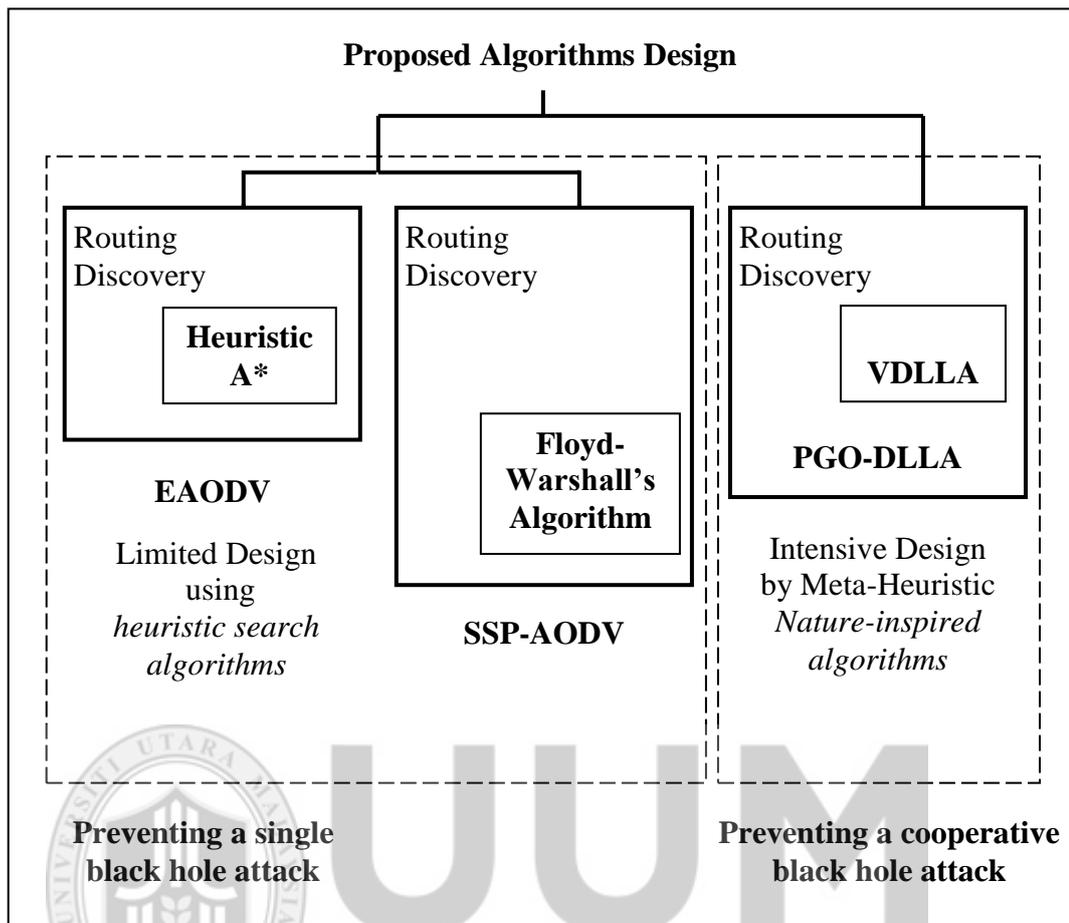


Figure 4.1. The taxonomy of the proposed algorithms design

#### 4.2 The Design of EAODV Routing Protocol

The main idea for the EAODV protocol is to improve the route discovery as compared with the standard AODV routing protocol and the testing impact of the integrating heuristic A\* with performance metrics. Therefore, various artificial intelligence concepts are integrated into the routing mechanism for the shortest path to destination. The A\* algorithm needs the estimated values of the node distances and uses the values of DSNs and hop counts that are obtained from the control messages. Figure 4.2 shows the elements of the EAODV protocol: protocol message format, data structure of new protocol, and routing discovery mechanism.

In the EAODV routing protocol, extra fields are added to the routing messages and the routing table to support the shortest path to destination. These messages, the routing table and the operations of EAODV are described in the following sections.

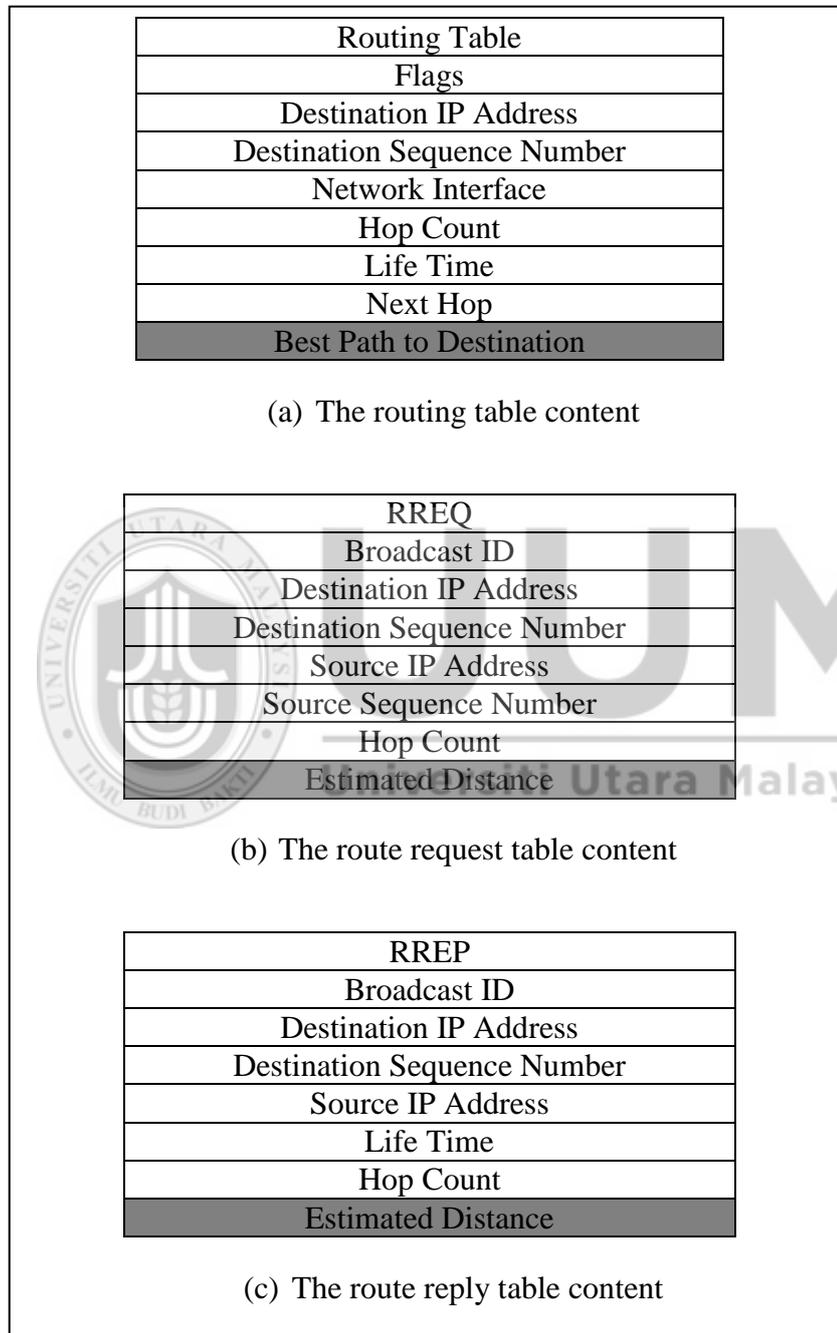


Figure 4.2. The elements of EAODV routing protocol

#### **4.2.1 EAODV Messages Format**

AODV can be modified to enhance the security in routing discovery by implementing a black hole prevention mechanism in the AODV routing protocol.

The finding of the shortest path is done by selecting a short path during the route discovery and save the result in the Best Path field (BP). There are two types of control messages that have changes in its content (see Figure 4.2). The routing messages, Route Request (RREQ) and Route Reply (RREP), have a new field that does not exist in the original AODV protocol.

#### **4.2.2 Data Structure of EAODV**

In the EAODV protocol, each node keeps and maintains a neighbor table, exactly similar to the original AODV routing protocol in the structure. As a mobile ad hoc network node, it has a source broadcast ID table and a reverse route table. The former, source broadcast ID table, is used to record the neighbor information and the address of source node and broadcast ID. The routing table is used to store the routing information and the update of the routing information.

#### **4.2.3 Route Discovery of EAODV Protocol**

Every node can calculate the estimated distance by using this information in the routing table. EAODV shares the original AODV's on-demand characteristics in that it also discovers routes on needed similar route discovery process. EAODV is a process of finding a route between two nodes when available. However, EAODV adopts a different mechanism in the routing discovery by using the heuristic A\*.

In AODV, the route discovery process uses a traditional routing table, one entry per destination, and relies on routing table entries to propagate a RREQ to route data packets to the destination and RREP back to the source node if there is no route to the destination in the source routing table. The mechanism of EAODV is shown in the flowchart in Figure 4.3.

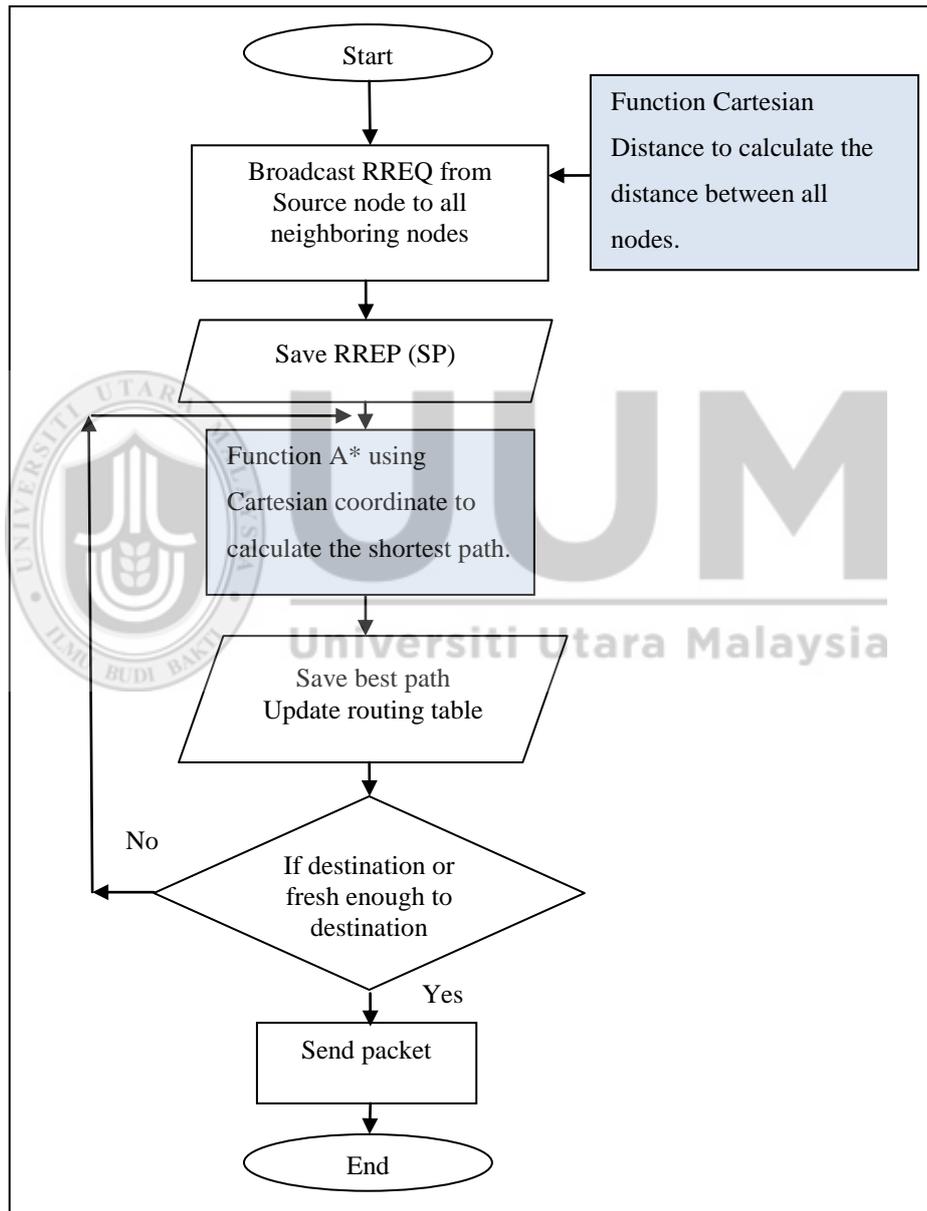


Figure 4.3. The flowchart of proposed EAODV

In EAODV, the function A\* using Cartesian Distance to calculate the shortest path is proposed. Where, it is supposed that  $g(n)$  equals to the hop count in the routing discovery, while the  $h(n)$  equals the estimated values to destination node and  $f(n)$  refers to the estimated total cost of path through  $n$  to the goal. Equation 4.1 shows the objective function of A\* in the EAODV protocol.

$$f'(n) = g(n) + h(n) \quad (4.1)$$

To calculate the value of  $h(n)$ , Cartesian Distance (CD) is used as shown in Equation 4.2, where in this situation; it is required to determine the initial values of each node position in the network.

$$\text{Cartesian Distance(CD)}(\text{node1, node2}) = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \quad (4.2)$$

where,  $(x_1, y_1)$  and  $(x_2, y_2)$  are the position of first and second nodes.

In order to describe how the heuristic A\* finds the shortest path in the route discovery of the EAODV protocol, Example I describes the flow of A\* process.

#### 4.2.3.1 Example I: Process of A\* in EAODV

In this example, it is supposed that there are six nodes (a, b, c, d, e, and f) in a simple graph as shown in Figure 4.4, Where (a) is a source node, (f) is a destination node, and the rest are intermediate nodes. The goal is to find a best path from the source node (a) to the destination node (f). Suppose all the nodes know about its neighbors from its routing table. In general, the heuristic A\* operates to find the path with the least cost. In this example, firstly, calculate the estimated distance between each node

and destination node using Equation 4.2 as shown below. Table 4.1 shows the results of the estimated values.

$$CD(a, f) = \sqrt{(7 - 2)^2 + (1 - 1)^2} = \sqrt{5^2 + 0^2} = \sqrt{25} = 5$$

$$CD(b, f) = \sqrt{(7 - 4)^2 + (1 - 2)^2} = \sqrt{3^2 + (-1)^2} = \sqrt{9 + 1} = \sqrt{10} = 3.2$$

$$CD(c, f) = \sqrt{(7 - 3)^2 + (1 - 0)^2} = \sqrt{4^2 + (1)^2} = \sqrt{16 + 1} = \sqrt{17} = 4.1$$

$$CD(d, f) = \sqrt{(7 - 6)^2 + (1 - 2)^2} = \sqrt{1^2 + (-1)^2} = \sqrt{2} = 1.4$$

$$CD(e, f) = \sqrt{(7 - 5)^2 + (1 - 0)^2} = \sqrt{2^2 + (1)^2} = \sqrt{4 + 1} = \sqrt{5} = 2.2$$

Table 4.1

*Example of Estimated Distances.*

Source Nodes	Estimated Distances
a	5.0
b	3.2
c	4.1
d	1.4
e	2.2
f	0.0

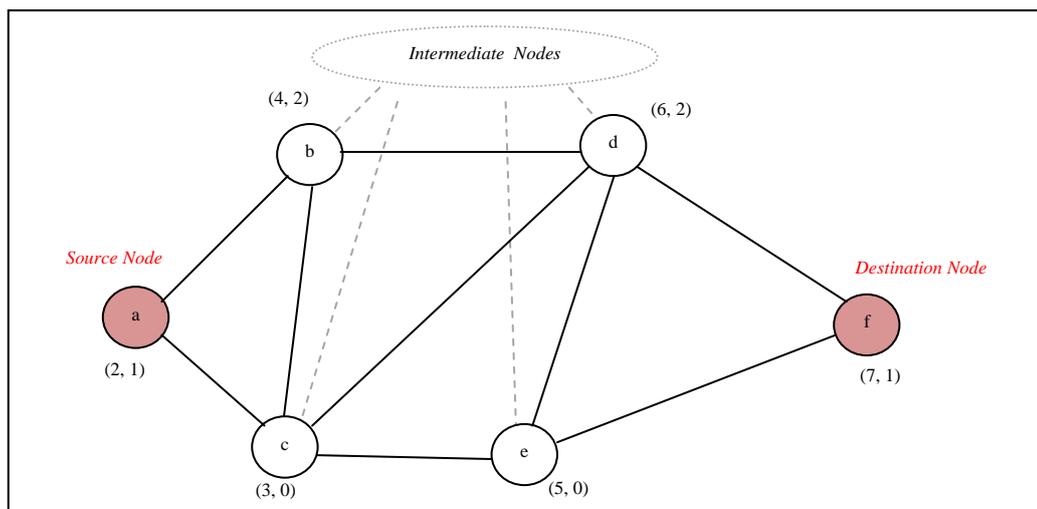


Figure 4.4. Example of six nodes

After that, the calculation of the cost between nodes is carried out using Equation 4.1, where  $g(n)$  is the number of hop counts, and  $h(n)$  is the estimated distance of  $n$  that can be derived from Table 4.1,  $f(n)$  is the best cost value. Hence, the nodes (b) and (c) are the closest nodes to the source node, where in this situation; there are two routes for the source node (a) to choose between them; path 1 from node (a) to node (b) with a cost equal to 4.2 and path 2 from node (a) to node (c) with a cost equal to 5.1. So the best route with the minimum cost is route  $\{a \rightarrow b\}$  as shown in Figure 4.5.

**From node (a) to node (b)  $\{a \rightarrow b\}$ ,**

$$\begin{aligned} f(b) &= g(b) + h(b) \\ &= 1.0 + 3.2 \\ &= 4.2 \end{aligned}$$

**From node (a) to node (c)  $\{a \rightarrow c\}$ ,**

$$\begin{aligned} f(c) &= g(c) + h(c) \\ &= 1.0 + 4.1 \\ &= 5.1 \end{aligned}$$

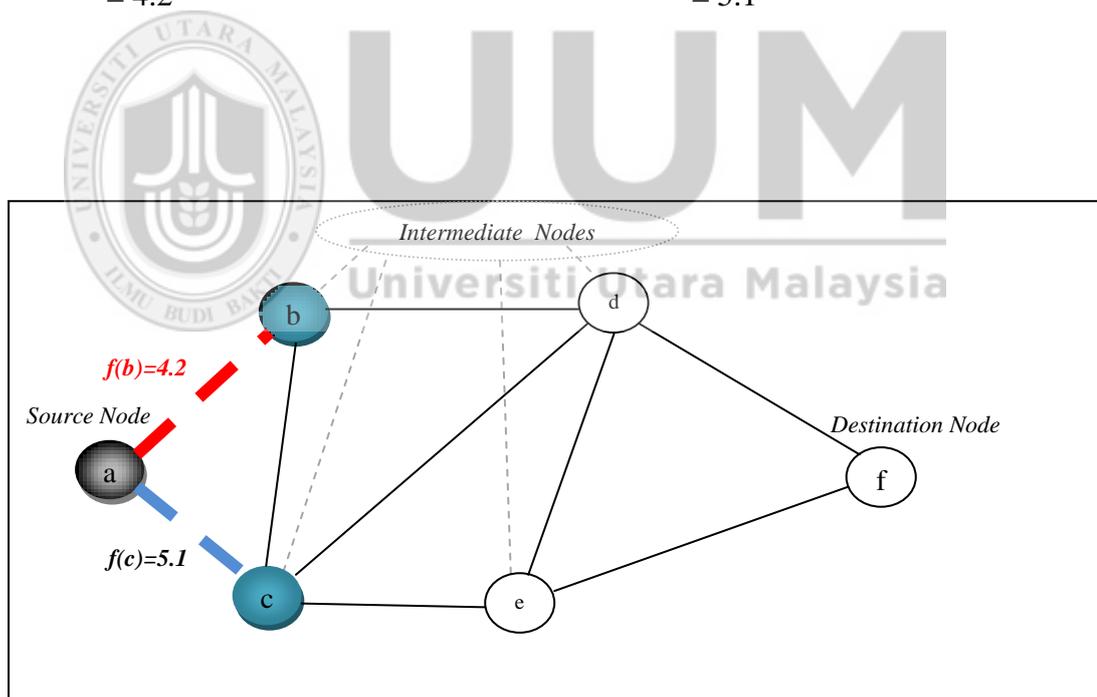


Figure 4.5. An example of best route between node (a) and node (b)

Afterwards, node (b) takes the role of source node. Node (b) has three routes to destination node (f); from node (b) to node (d), from node (b) to node (c) or return to node (a) from node (b).

<p><b>From node (b) to node (d) { b→d},</b>  <math>f(d) = g(d) + h(d)</math>  <math>= 2.0 + 1.4</math>  <math>= 3.4</math></p>	<p><b>From node (b) to node (c) {b→c},</b>  <math>f(c) = g(c) + h(c)</math>  <math>= 2.0 + 4.1</math>  <math>= 6.1</math></p>	<p><b>From node (b) to node (a) {b→a},</b>  <math>f(a) = g(a) + h(a)</math>  <math>= 2.0 + 5.0</math>  <math>= 7</math></p>
--	---	---

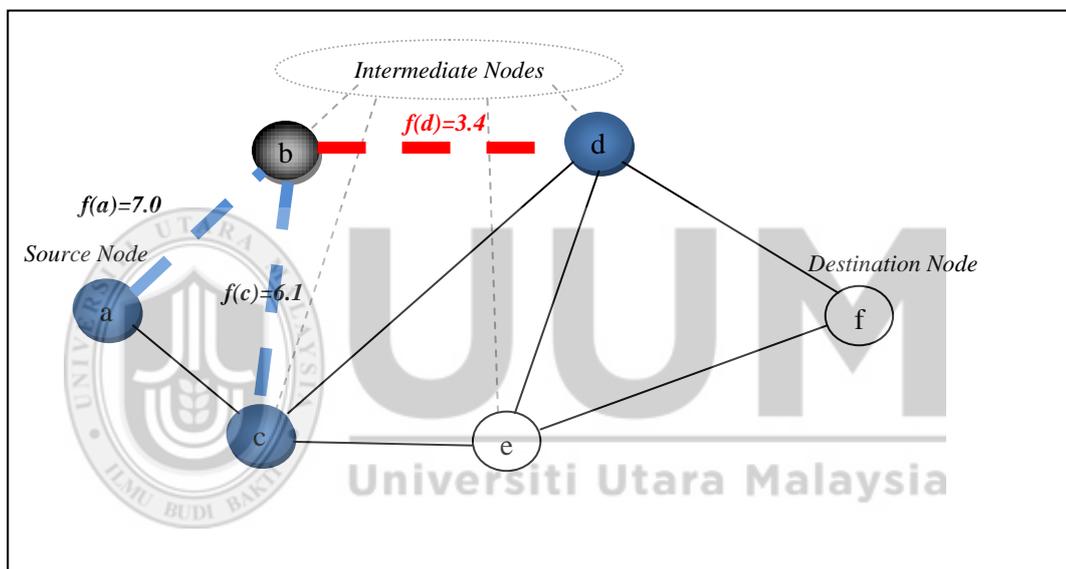


Figure 4.6. An example of best route between node (b) and node (d)

As illustrated in Figure 4.6 an example of finding the best route between node (b) and node (d) is shown, where the best route is the route between {b→d} which has a heuristic value equal to 3.4.

In the next step, the node (d) becomes as source node. It has four routes to destination node (f); from node (d) to node (c), from node (d) to node (e), from node (d) to node (f), and from node (d) to node (b).

From node (d) to node (c) {d→c}	From node (d) to node (e) {d→e}	From node (d) to node (f) {d→f}	From node (d) to node (b) {d→b}
$f(c) = g(c) + h(c)$ $= 3.0 + 4.1$ $= 7.1$	$f(e) = g(e) + h(e)$ $= 3.0 + 2.2$ $= 5.2$	$f(f) = g(f) + h(f)$ $= 3.0 + 0.0$ $= 3.0$	$f(b) = g(b) + h(b)$ $= 3.0 + 3.2$ $= 6.2$

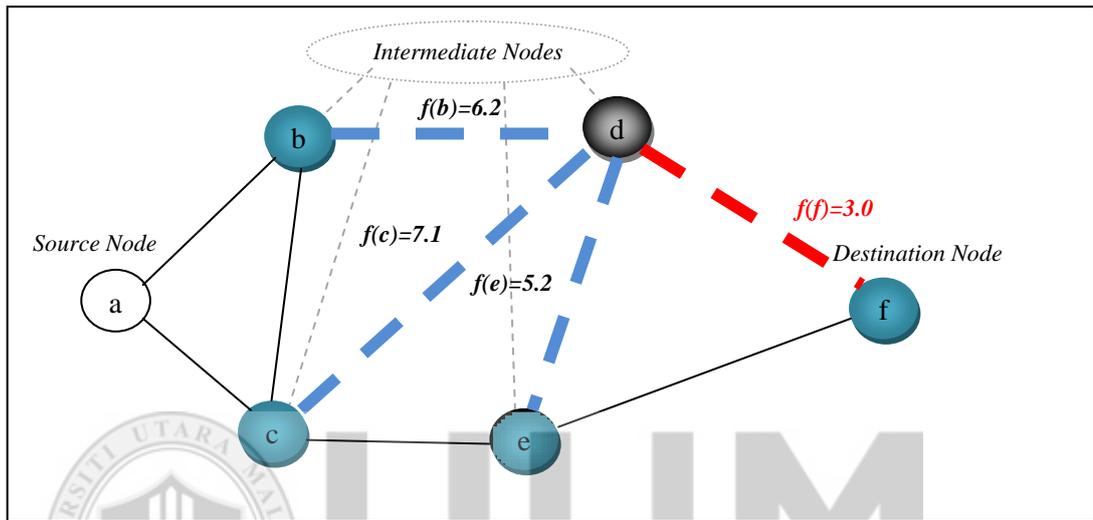


Figure 4.7. An example of best route between node (d) and node (f)

The best route is the shortest route from node (d) to node (f) {d→f} with a heuristic value equal to 3 as shown in Figure 4.7. Finally, the best route from source node (a) to destination node (f) is {a→b→d→f} as shown in Figure 4.8.

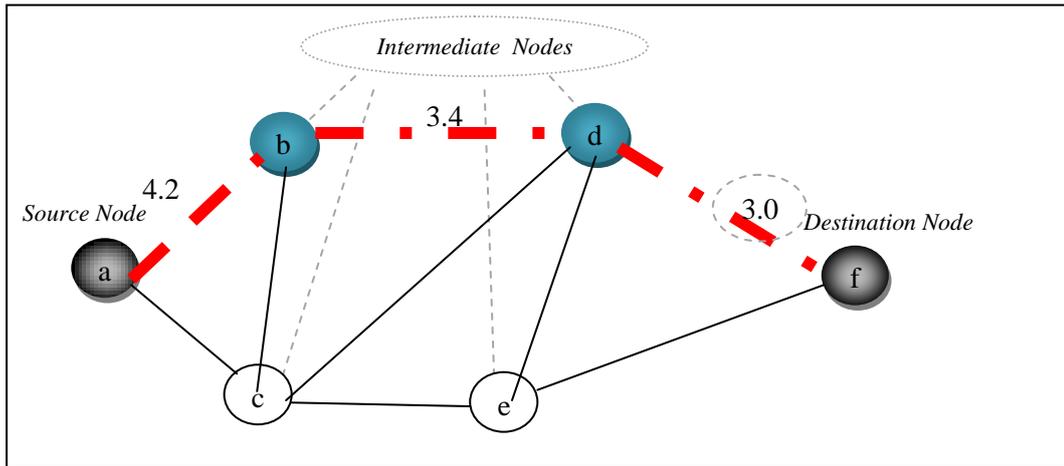


Figure 4.8. An example of the best route {a→b→d→f}

### 4.3 A Dynamic Heuristic Search Algorithm

The A\* algorithm is optimally efficient when applied with routing, but it operates in a static topology and it still requires exponential time and space in general. The A\* function estimates the number of moves required to modify the current state into a goal state [124].

When integrating A\* algorithm with AODV, it still has a few drawbacks; a) the estimated distances among nodes to calculate a path as an initial value, b) the enhanced routing of AODV but not to improve the routing security. To overcome these drawbacks, an intelligent technique that integrates A\* algorithm with Floyd-Warshall's algorithm in AODV is proposed for improving the routing protocol to find the shortest path. In order to improve the routing security, adopt the technique that is used in [2] which works as the source node and does not respond directly to the first RREP (i.e. sending the data packet) and check more than one path arrives to source node. The proposed routing protocol, termed as Shortest Secure Path for AODV

(SSP-AODV) integrates the heuristic A\* and Floyd-Warshall's algorithms in AODV to achieve the shortest and secure path in a dynamic environment against the black hole attack. In the following sections, there are several explanations about the proposed shortest path algorithm.

#### 4.3.1 The Proposed SSP-AODV

There are many algorithms that are used to find the shortest path between two nodes in a graph. Dijkstra's algorithm is one of these algorithms, which determines the distance from a specific node to all other nodes. In Dijkstra's algorithm, the distance and the least cost from source to destination node will be saved. It starts to work by creating four matrices as initial; adjacency matrix for cost, distance matrix, path matrix and include matrix. Figure 4.9 shows the pseudo code of Dijkstra's algorithm.

```

Procedure Dijkstra's Algorithm();
1:dist[s] ← 0 // set 0 distance to source vertex //
2:For all v ∈ V-{s}
3:Do dist[v] ← ∞ // set ∞ to all other distances //
4:S ← ∅ // S, the set of visited vertices is initially empty //
// Q, the queue initially contains all vertices //
6:While Q ≠ ∅ // while the queue is not empty //
7:Do u ← mindistance(Q,dist) // select the element of Q with the min. Distance//
8:S ← S ∪ {u} // add u to list of visited vertices //
9:For all v ∈ neighbors[u]
10:Do If dist[v] > dist[u] + w(u, v) // if new shortest path found //
then d[v] ← d[u] + w(u, v) // set new value of shortest path //
// if desired, add traceback code //
11:Returndist;
12:End

```

Figure 4.9. Pseudo code of Dijkstra's algorithm [116]

The shortest distance between a pair of nodes can be found using a single source shortest path algorithm such as BFS or Dijkstra's algorithm, but this would pass the entire graph several times. The best solution is the application of all pairs of shortest path using Floyd-Warshall's algorithm. It can be used to determine the length of the shortest path between two nodes in any graph net (in the computer network by using the values of a weighted connection between them). In spite of this, Floyd-Warshall's algorithm is not a heuristic search algorithm, in which it cannot be used alone to find the shortest path, so by integrating it with heuristic A\* algorithm to achieve the best shortest path. In order to address the problem that has been described in Chapter One, each node has a routing table which includes the information, such as hop count, destination sequence number (DSN), life time, source IP address, and so on. Every node can calculate the estimated distance by using this information in the routing table. SSP-AODV has one field is added in the routing table more than an EAODV routing table, which is shortest path value (SP-value), As shown in Figure 4.10.

<b>RREQ-AODV Table</b> <b>Broadcast ID</b>	<b>RREP-AODV Table</b>	<b>SSP-AODV Table</b>
Destination IP Address	Destination IP Address	Destination IP & DSN
Destination Sequence Number	Destination Sequence Number	DSN-Flags
Source IP Address	Source IP Address	Flags
Source Sequence Number	Life Time	Network Interface
Hop Count	Hop Count	Hop Count
Estimated Distance	Estimated Distance	Next Hop
SP- Value	SP- Value	Life Time
		Best Path to Destination

Figure 4.10. The routing table: SSP-AODV, RREQ-AODV and RREP-AODV

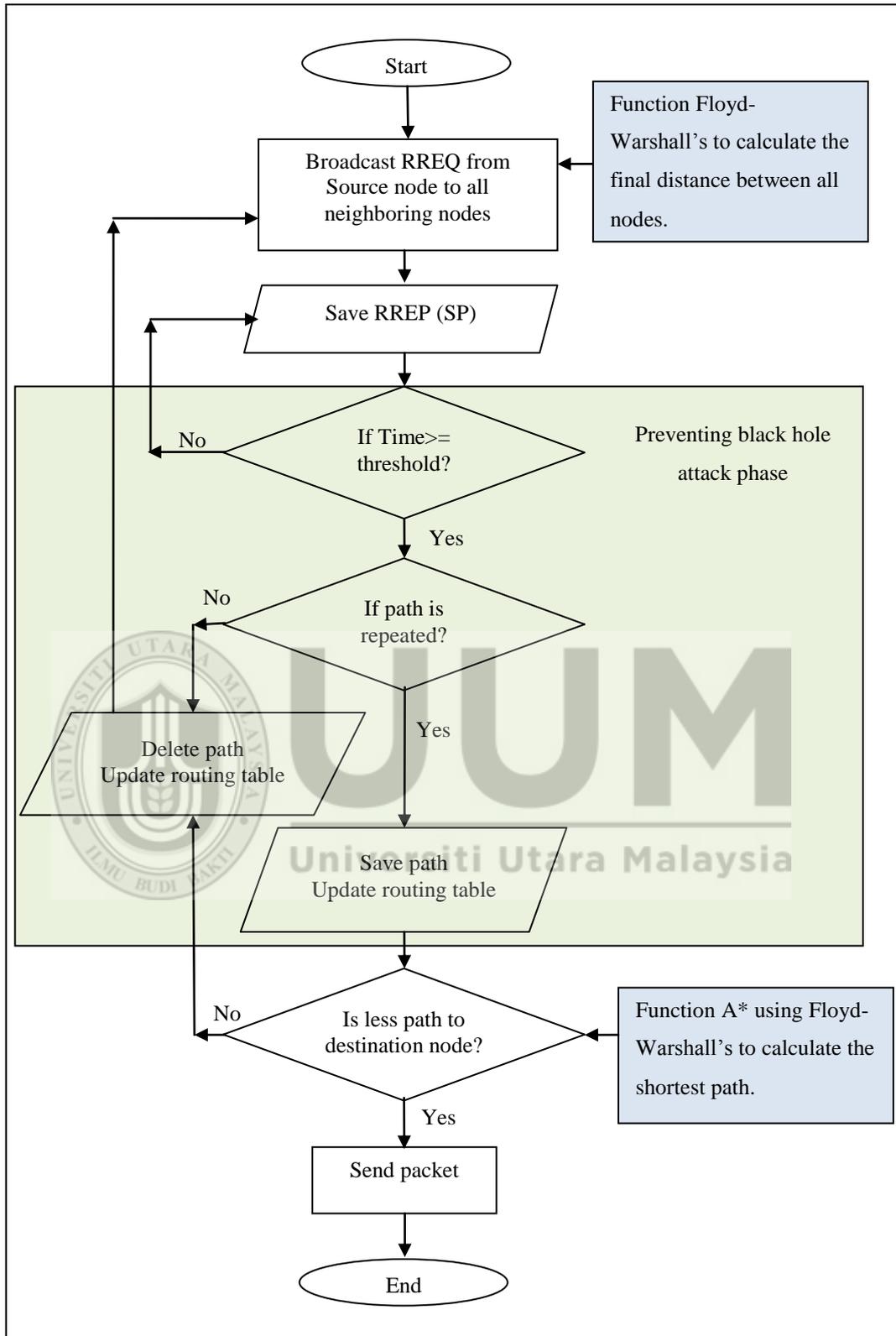


Figure 4.11. The flowchart of the proposed SSP-AODV

The estimated distance values are calculated using Equation 4.2 and SP-value can be computed by A\* function. Figure 4.11 shows the flowchart of the proposed SSP-AODV. The proposed SSP-AODV has two phases: calculation of the shortest path, and checking for a secure path.

#### 4.3.1.1 Phase One: Shortest Path

In this phase, the Floyd-Warshall's algorithm combined with A\* algorithm is used to find the value of the shortest path. The same example in Section 4.2.3.1 in Figure 4.4 is used to explain the process of Floyd-Warshall's algorithm. As mentioned previously, node (a) is a source node, node (f) is a destination node, and (b), (c) and (e) are intermediate nodes. The estimated distance value for each node to destination node is calculated using Equation 4.2. Table 4.1 shows the estimated distance value for each node of route discovery to destination node (f). The initial phase of Floyd-Warshall's algorithm can be computed using Equation 4.3 and the value is derived from Table 4.1.

$$ED(n_1, n_2) = ET(n_1) + ET(n_2) \quad (4.3)$$

where, ED is the estimated distance between two nodes, ET is the estimated distance value of the node derived from Table 4.1.

After the last phase of Floyd-Warshall's algorithm, the A\* algorithm is executed, where the value of  $g(n)$  represents the number of hop count, while the value of  $h(n)$  represents the distance from node (n) to the destination that can be retrieved from the

last phase of Floyd-Warshall's algorithm. In the following Example II, the A\* algorithm using Floyd-Warshall's algorithm with six nodes topology is explained.

#### 4.3.1.2 Example II: Process of A\* in SSP-AODV

Figure 4.12 shows an example of the topology of six nodes that includes the estimated distance among nodes that is calculated using Equation 4.3 and Table 4.1.

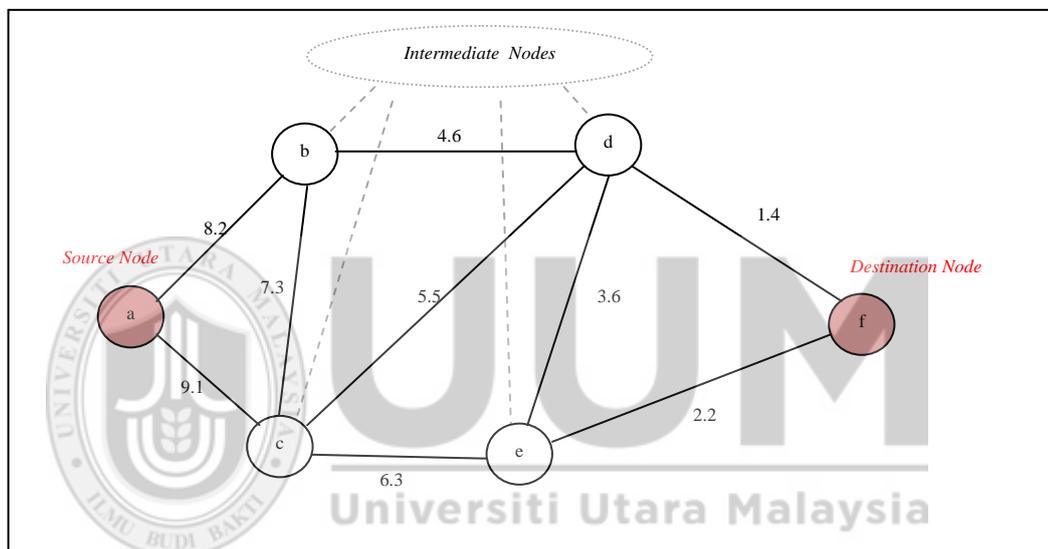


Figure 4.12. An example of six nodes topology with estimated distance

$$ED(a, b) = ET(a) + ET(b) = 5 + 3.2 = 8.2$$

$$ED(a, c) = ET(a) + ET(c) = 5 + 4.1 = 9.1$$

$$ED(b, d) = ET(b) + ET(d) = 3.2 + 1.4 = 4.6$$

$$ED(b, c) = ET(b) + ET(c) = 3.2 + 4.1 = 7.3$$

$$ED(c, d) = ET(c) + ET(d) = 4.1 + 1.4 = 5.5$$

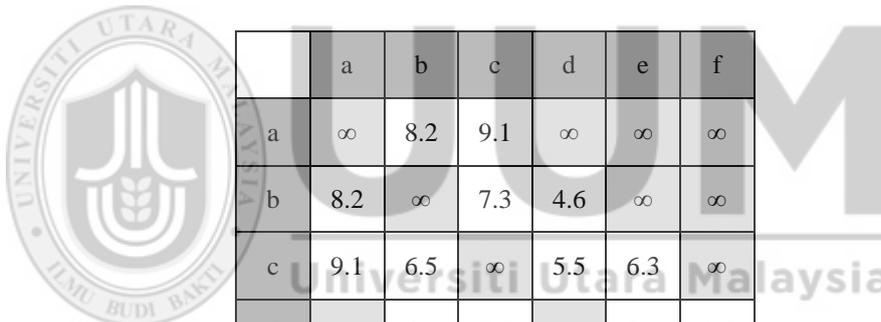
$$ED(c, e) = ET(c) + ET(e) = 4.1 + 2.2 = 6.3$$

$$ED(d, f) = ET(d) + ET(f) = 1.4 + 0.0 = 1.4$$

$$ED(d, e) = ET(d) + ET(e) = 1.4 + 2.2 = 3.6$$

$$ED(e, f) = ET(e) + ET(f) = 2.2 + 0.0 = 2.2$$

To construct the first phase of Floyd-Warshall's algorithm, the topology in Figure 4.12, example with six nodes, is converted into matrix (6x6). The rows and columns of the matrix represent the nodes and the intersection between nodes is the weight value that connects the two nodes in the graph (estimated distance in this situation), but if the connection between two nodes does not exist, the value is equal to  $\infty$  [125]. Figure 4.13 shows an example of the first phase of Floyd-Warshall's algorithm.



	a	b	c	d	e	f
a	$\infty$	8.2	9.1	$\infty$	$\infty$	$\infty$
b	8.2	$\infty$	7.3	4.6	$\infty$	$\infty$
c	9.1	6.5	$\infty$	5.5	6.3	$\infty$
d	$\infty$	4.6	5.5	$\infty$	3.6	1.4
e	$\infty$	$\infty$	6.3	3.6	$\infty$	2.2
f	$\infty$	$\infty$	$\infty$	1.4	2.2	$\infty$

Figure 4.13. An example of the first phases of Floyd-Warshall's algorithm

The second phase of applying Floyd-Warshall's algorithm is illustrated in Figure 4.14(a). After six iterations (number of iteration based on number of nodes), the final phase can be found as shown in Figure 4.14(b). The implementation of Floyd-Warshall's algorithm in this example undergoes 28 phases as shown in Appendix A.



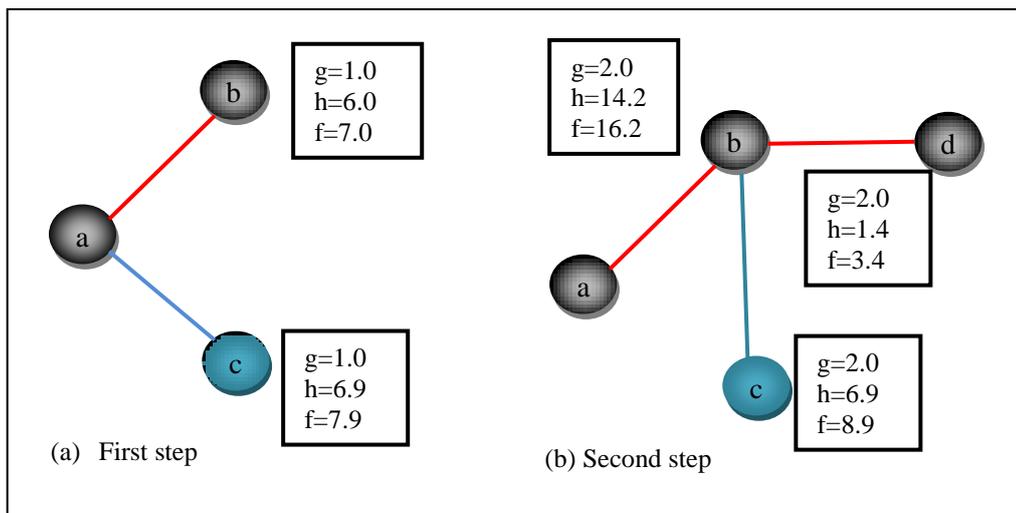


Figure 4.15. The first and second steps of Floyd-Warshall's with A\* algorithm

So the best route is shortest route  $\{a \rightarrow b\}$ . In Figure 4.15(b), there are three routes to destination node (f): from node (b) to node (d), from node (b) to node (c), and from node (b) back to node (a).

**From node (b)  
to node (d)  $\{b \rightarrow d\}$ ,**

$$\begin{aligned} f(d) &= g(d) + h(d) \\ &= 2.0 + 1.4 \\ &= 3.4 \end{aligned}$$

**From node (b)  
to node (c)  $\{b \rightarrow c\}$ ,**

$$\begin{aligned} f(c) &= g(c) + h(c) \\ &= 2.0 + 6.9 \\ &= 8.9 \end{aligned}$$

**From node (b)  
to node (a)  $\{b \rightarrow a\}$ ,**

$$\begin{aligned} f(a) &= g(a) + h(a) \\ &= 2.0 + 14.2 \\ &= 16.2 \end{aligned}$$

The best route is shortest route  $\{b \rightarrow d\}$ . In the next step as shown in Figure 4.16, there are four routes: from node (d) to node (c), from node (d) to node (e), from node (d) to destination node (f), and from node (d) back to node (b).

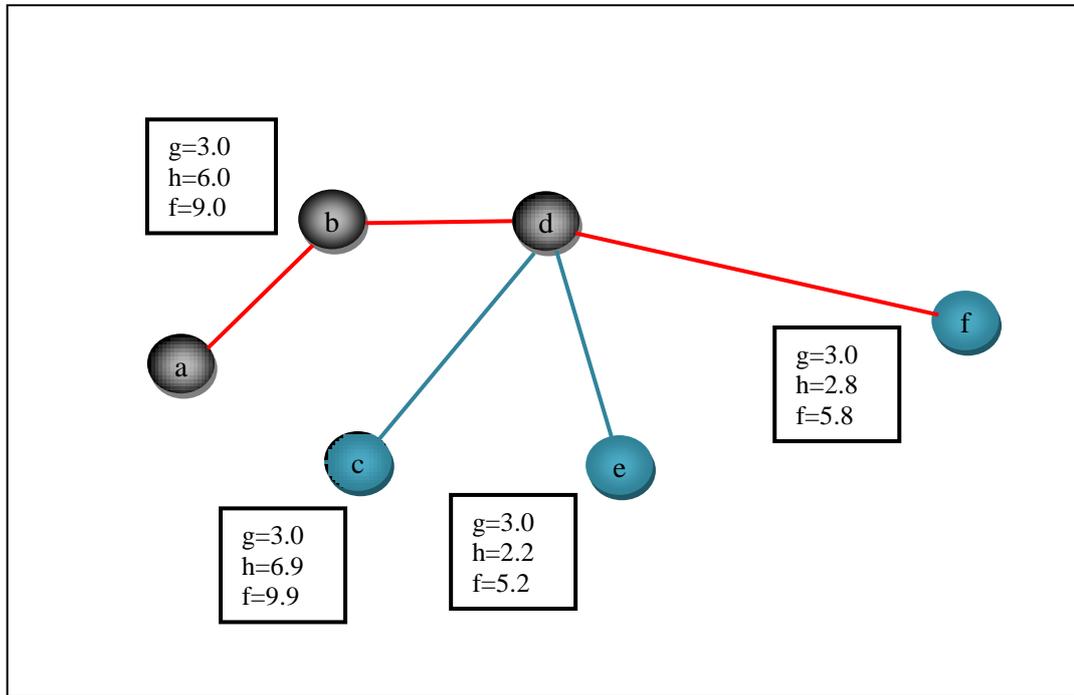


Figure 4.16. Third step of Floyd-Warshall's with A\* algorithm

From node (d) to node (c) { d→c }	From node (d) to node (e) { d→e }	From node (d) to node (f) { d→f }	From node (d) to node (b) { d→b }
$f(c) = g(c) + h(c)$ $= 3.0 + 6.9$ $= 9.9$	$f(e) = g(e) + h(e)$ $= 3.0 + 2.2$ $= 5.2$	$f(f) = g(f) + h(f)$ $= 3.0 + 2.8$ $= 5.8$	$f(b) = g(b) + h(b)$ $= 3.0 + 6$ $= 9.0$

The best route is shortest route {d→e}, but there is also route {d→f} that reaches destination f, so in this situation, this route is chosen. Finally, the best route from source node (a) to destination node (f) is {a→b→d→f}.

#### 4.3.1.3 Phase Two: Prevent Black Hole Attack

Phase two is the working on preventing the black hole attack; when the source node receives one or more RREP from the neighboring nodes, it will not send a packet

directly, but it waits until the routing time expires. The source node will wait for a short time, then, it will check based on threshold ( $THr = 10, 20, \dots, 100$  second), if the waiting time is greater or equal than  $THr$ , then it will check the hop count value of the new path with all RREP/RREQ tables. The new path is saved as a secure path if it has repeated path to the destination, or else the path will be removed from the routing table. The shortest secure path will chosen from the SP-value fields of the RREQ/RREQ tables. Finally, the packet will be sent to the destination node.

#### **4.4 PGO-DLLA to Prevent Black Hole Attack**

Nature-inspired algorithm is a research area for information models that studies the collective behavior of insects or animal swarms. Some algorithms have been proposed to address black hole attacks through new protocols and improved routing security with swarm intelligence. In this section, a parallel grid algorithm for MANETs is proposed that optimizes both routing discovery and security in an Ad Hoc On-Demand Distance Vector (AODV). The new algorithm, called Parallel Grid Optimization by the Daddy Long-Legs Algorithm (PGO-DLLA), simulates the behavior of the biological spiders known as daddy long-legs spiders [126].

##### **4.4.1 Parallel Grid Optimization by Daddy Long-Legs Algorithm (PGO-DLLA)**

PGO-DLLA is based on the idea of the VDLLA optimization algorithm (see section 4.4.1.1), that is based on the choosing the least path to the destination and the nature behavior of daddy long-legs spiders of catching the prey or avoiding attacks in a spider web.

#### 4.4.1.1 Daddy Long-Legs Spider

Globally, there are more than 30,000 kinds of spiders, which are characterized by a unique way of hunting prey. Most types of spiders respond to any vibrations that come from their web. Spiders have special methods for quick access to prey and capture them as soon as possible. Several vibrations coming from the web may signal a source of danger, and changing strategies is essential to avoid the threat [127]. The new proposed algorithm is based on the behavior of spiders in nature, namely daddy long-legs spiders. This type of spider responds to the first vibration that comes from the web and chooses the shortest path to catch the prey without giving it a chance to escape from the trap [126]. Spiders have a huge number of strategies to capture prey, such as trapping the prey in a sticky web [126], [127]. In the case of daddy long-legs spiders, all paths in the web are available for access to a destination, because daddy long-legs spiders do not use the glue that other spiders use. The absence of glue on the yarns of daddy long-legs spiders provides them with unique features, such as the ability to change their location in the web to avoid any dangers coming from outside the web. In addition, when there is more than one source of vibrations, the daddy long-legs spider chooses the smallest vibration frequency value to avoid the risk. This spider is sometimes called a hopper spider because it generates inverse artificial vibrations [126], which can be useful to tighten restrictions on its prey or to discard the prey when it is an unwanted kind. Daddy long-legs spiders are slightly different from other spiders because they have high sensing precision using their eight legs, which act like sensors or agents to receive signals or to discover their prey's position.

#### 4.4.1.2 The Spider's Leg Behaviors

The main principles of the leg behavior are as presented in Figure 4.17.

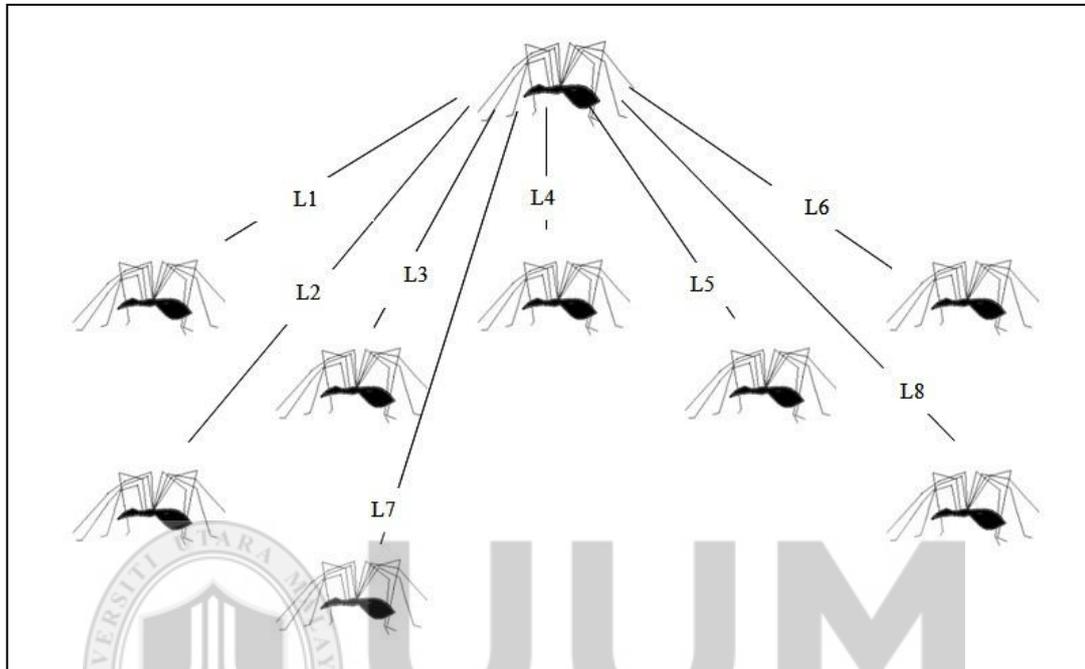


Figure 4.17. The 8-legs in VDLLA

- **Homogeneity:** Each spider leg (8-legs) has the same behavior model.
- **Locality:** The motion of each leg is only influenced by its nearest path mates.
- **Collision Avoidance:** Automatically avoids collision with nearby legs.
- **Body Centering:** Attempts to stay close to nearby prey mates.
- **Parallelism:** Eight local search agents are working in parallel, for a single spider, the search for global solution is at the same time.
- **Speed of Converge:** Even though the global optima are not close to the best leg, it is possible for the spider to explore other areas.

- **Solve local optima problem:** The  $globalposition_{new} = bestposition$  and the *legpositions* change based on the position of the body.

#### 4.4.2 Virtual Daddy Long-Legs Algorithm (VDLLA)

The nature-inspired algorithms are based on a biological behavior of insects or animals. Many meta-heuristic optimization algorithms have the ability to find near optimal global solutions; however, these algorithms are weak in searching for local solution.

This means that meta-heuristic optimization is strong in exploration but poor in exploitation. The success of any Meta heuristic optimization algorithm depends on the balance between exploration and exploitation. So, in this section, a new nature-inspired algorithm is presented that is based on daddy long-legs spider as a new optimization algorithm with virtual behavior, which is named Virtual Daddy Long-Legs Algorithm (VDLLA).

In VDLLA, each agent (spider) has nine positions (solutions) which represent the eight legs of the spider and the current body position. The proposed VDLLA will be tested in the next chapter on four standard functions using average fitness, medium fitness and standard deviation.

Experimented results are later compared against Particle Swarm Optimization (PSO), Differential Evolution (DE) and Bat-Inspired Algorithm (BA). Additionally, the statistical analysis of T-Test will be conducted to verify the results to learn that the

proposed VDLLA demonstrates promising results on the benchmark test functions for unconstrained optimization problems and has significantly improved other nature-inspired algorithms.

In this section, the operational principles from VDLLA used a guideline to develop a new inspired optimization algorithm. VDLLA assume that:

- The entire search space is a network content from nodes in the search space.
- All the legs of one spider interact with each other.
- Each solution with in the search space represents a spider position.
- Every leg receives a weight according to the fitness value.
- The algorithm models eight different search agents (legs).
- Depending on the legs, each individual uses the same evaluation function.
- The algorithm starts by one spider (one body and eight legs).

#### **4.4.2.1 The Motivations of the Design of VDLLA**

In 2002, Kennedy and Mendes [128] developed a new version of PSO based on the standard PSO [93], to solve the problem of trapping in local optima, and to put more balance between exploration (global search) and exploitation (local search).

In [128], the authors have suggested to put a new technique in PSO, by adding two new comparisons (lbest on the right side and lbest on the left side). In the other words, the new algorithm can locate the global optimal with greater chance but the convergence will be slower.

On the contrary, VDLLA as a new algorithm has a robust simple mechanism to solve all the disadvantages in the old previous swarm algorithms such as attracted to the best part of the search space in PSO. Depending on these disadvantages and the standard behavior of the individuals, the higher priority rule is the attempt to find a minimum distance between themselves and others at all times [129].

Furthermore, to avoid being isolated, the individuals tend to be attracted towards other individuals and align themselves with neighbors [129]. In VDLLA, with the existence of the eight legs, eight positions plus the position of the spider body can be compared. Hence, the old behavior of the previous swarm algorithms can be changed to simulate a different biological behavior of daddy long-legs.

In this behavior, it is unnecessary to avoid isolating any individual by being attracted to the best solution. The spider can avoid any dangerous path by isolating the leg without any effect of its mechanism.

#### **4.4.2.2 The Process of VDLLA**

VDLLA is a swarm of spiders. It is assumed that each spider has nine positions represented as a  $3 \times 3$  matrix in a grid space, where eight of the positions are for the spider's eight legs and the center position is for the spider's body.

Each spider evaluates the nine positions based on the objective function and determines the best location from the nine positions. The best position for each spider

is then evaluated to choose a global position. Table 4.2 shows the positions of each agent in VDLLA.

Table 4.2

*The positions of each agent (spider).*

Leg5= (X-0.1, Y+0.1)	Leg1 = (X,Y+0.1)	Leg6 =(X+0.1,Y+0.1)
Leg3=(X-0.1,Y)	body = (X,Y)	Leg4 = (X+0.1,Y)
Leg7 =(X-0.1,Y-0.1)	Leg2=(X,Y-0.1)	Leg8 =(X+0.1, Y-0.1)

Some of the test functions we have used are the bivariate (Rosenbrock [130], Michalewicz [131], Eggcrate [130], and Beal [130] functions) after implementation to validate the VDLLA algorithm. For example the Rosenbrock function  $f(x, y) = (1-x)^2 + 100(y-x)^2$  is a non context function used a classic performance test function in optimization theory. In contrast to many derived-free optimizers, Rosenbrock's function can be efficiently optimized by adapting appropriate coordinate system without using any gradient information and without building local approximation models. Sometimes referred as a Rosenbrock's banana function due to the shape of its contour lines. Rosenborck's function is a continuous, differentiable, non-separable, scalable, and unimodal function [130]. The equation of Rosenborck's function is Equation 4.3, and the graph of the Rosenborck's function is illustrated in Figure 4.18.

$$f(x) = \sum_{i=1}^{D-1} [100(x_{i+1} - x_i^2)^2 + (x_{i-1})^2] \quad (4.3)$$

Subject to  $-30 \leq x_i \leq 30$ , the global minimum is located at  $x^* = f(1, \dots, 1)$ ,  $f(x^*) = 0$ .

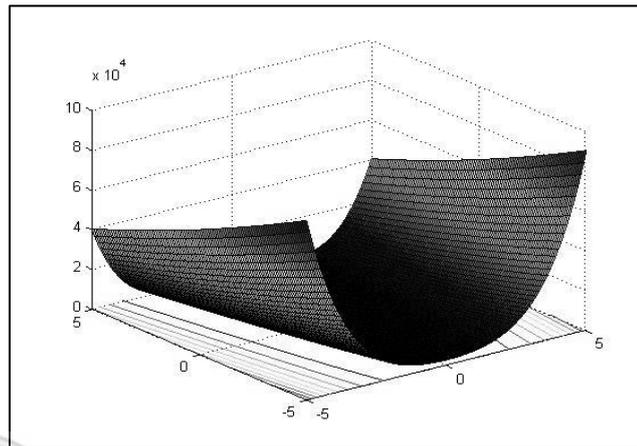


Figure 4.18. The Rosenbrock's function [130]

The Michalewicz's function [131] is as shown in Equation 4.4, and the graph of the Michalewicz's function is illustrated in Figure 4.19.

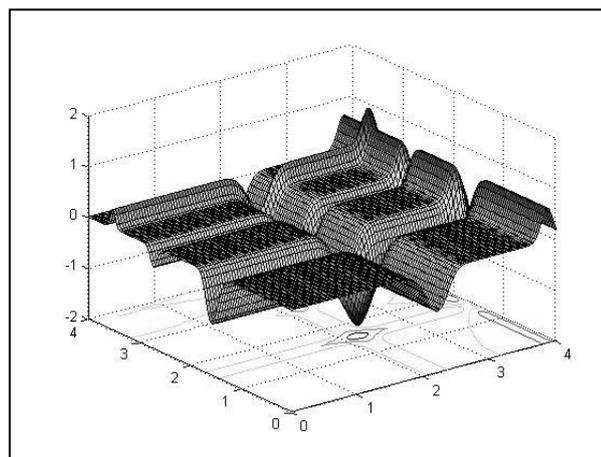


Figure 4.19. The Michalewicz's function [131]

Subject to  $0 \leq x_i \leq \pi$ . The global minimum is located at  $x^* = f(1, 2, \dots, d)$ ,

$f(x^*) = -1.801$  for  $d = 2$ , while  $f(x^*) = -4.6877$  for  $d = 5$ .

$$f(x) = \sum_{i=1}^d \sin(x_i) \left[ \sin\left(\frac{ix_i^2}{\pi}\right) \right]^2, m, m = 10 \quad (4.4)$$

The EggCrate's function is presented in Equation 4.5, and its graph is illustrated in Figure 4.20.

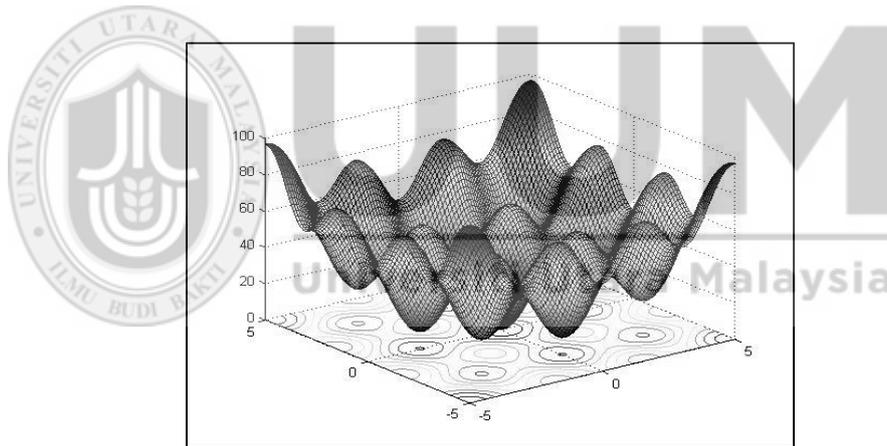


Figure 4.20. The EggCrate's function [130]

It is a continuous, non-separable, and non-scalable function [130].

$$f(x_i) = x_1^2 + x_2^2 + 25(\sin^2(x_1) + \sin^2(x_2)) \quad (4.5)$$

Subject to  $-5 \leq x_i \leq 5$ , the global minimum is located at  $x^* = f(0, 0)$ ,  $f(x^*) = 0$ . And finally, the Beal's function is as shown in Equation 4.6, and the graph of the Beal's

function is illustrated in Figure 4.21. It is a continuous, non-differentiable, non-separable, non-scalable, and unimodal function [130].

$$f(x) = (1.5 - x_1 + x_1x_2)^2 + (2.25 - x_1 + x_1x_2^2)^2 + (2.625 - x_1 + x_1x_2^3)^2 \quad (4.6)$$

Subject to  $-4.5 \leq x_i \leq 4.5$ , the global minimum is located at  $x^* = (3, 0.5)$ ,  $f(x^*) = 0$ .

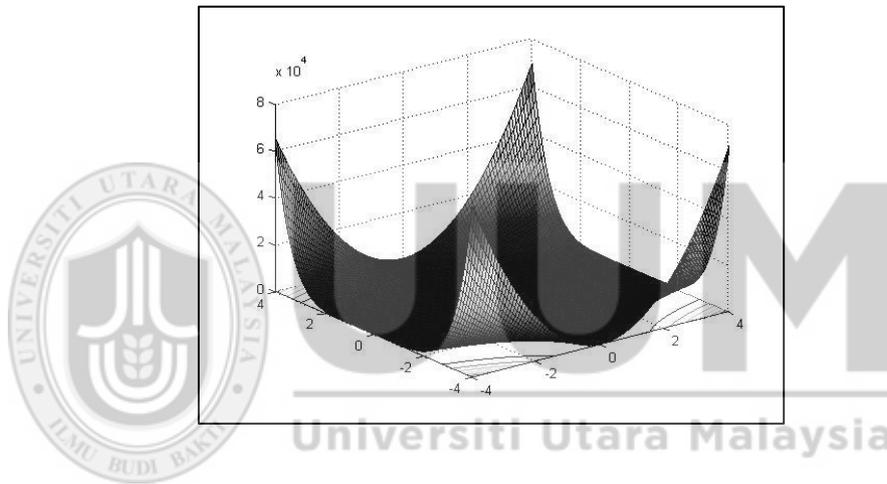


Figure 4.21. The Beal's function [130]

#### 4.4.2.3 Implementation of VDLLA

We suppose that each spider has nine positions as matrix 3x3 in grid space where eight legs and the center position are for the spider's body of spider as shows in Table 4.2. In start search space, each spider evaluates the nine positions randomly to determine the initial solution, for example body = (1.2, 1.5). The best position for each spider is evaluated to choose a global position as a minimum value. The computational procedure of VDLLA algorithm can be summarized as follows:

**Virtual Daddy Long-Legs Algorithm (VDLLA) return BestPath**

- 1: Begin**
- 2:Input:***N, body position, up, down, left, right, upleft, upright, downleft, downright*
- 2:Generate Initial population** of spider members, considering N as the total number of members
- 3:Generate Initial location** for each body of spider members randomly, and then calculate the legs' position based on body position:  
Assume the **body position** = (X, Y), the legs' position is eight directions where: from **up** = (X,Y+0.1), from **down** =(X,Y-0.1), from **left** =(X-0.1,Y), from **right** = (X+0.1,Y), from **up left** = (X-0.1, Y+0.1), **up right** =(X+0.1,Y+0.1), from **down left** =(X-0.1,Y-0.1) and **downright** =(X+0.1, Y-0.1) as shown in Table 4.2
- 4: Evaluate the fitness** for each agent (spider) where the evaluation includes all positions of the agent (*body + legs*) using **Equations 4.3, 4.4, 4.5 or 4.6.**
- 5:Select the best fitness** for each agent (spider) and save the position as best position
- 6:Select the global fitness** from all best fitness and save the position as global position
- 7:Do while** global fitness greater than tolerance value (tolerance value is based on objective function)
- 8:Find new position** for each agent where the body moves to best position and legs' position changes based on the body
- 9:Find new best fitness and new global fitness**
- 10:If** *new global fitness less than global fitness*
- 11:***Global fitness = new global fitness*
- 12:Else if** *new global equal global fitness*
- 13:Change the global position** using Equation 4.5 as shown below
$$Gpos_{new} = Gpos_{old} + 0.01(RND(1,d)) \quad (4.5)$$

*// Where, d is the dimension of objective function.//*
- 14:***iteration=iteration +1*
- 15:End while**
- 16:** *Bestpath=Gpos<sub>new</sub>*
- 17:End**

Figure 4.18. Pseudo code of VDLLA

#### 4.4.3 Problem Formulation and Solution Representation

As mention in Chapter one, MANETs suffer from several limitations, such as short battery lives, limited capacities, and vulnerability to malicious behaviors. A black hole is one type of attack that occurs in MANETs. Black hole nodes attack routing protocols such as the AODV protocol [77], causing network packets to be dropped. The main goal of the AODV protocol is to find a path from a source to its destination node and then to forward the packets. The routing mechanism in AODV uses route requests (RREQs; for discovering routes) and route replies (RREPs; for receiving paths). However, this mechanism is vulnerable to attacks by malicious black hole nodes that can easily adjust the values of routing table fields such as hop count and DSN in order to deceive the source node after sending a RREQ, a source node will respond to the first RREP it receives. This RREP may be from a black hole node, and the source will not reply to other intermediate nodes. As a result, the cooperative work in the MANET may be terminated [2], [41]–[45]. Intensive computations are required to make AODV secure against black hole attacks [132]. Most of the proposed solutions with limited computations such as trusting neighboring nodes, using cross-layer cooperation, or allowing route redundancy fail to detect cooperative black hole attacks [2], [43], [44]. However, the use of intensive computations as a solution to cooperative black hole attacks may lead to the depletion of the limited energy of batteries. In this mechanism, methods are developed to find the shortest secure path and to reduce overhead using the information that is available in the routing tables. However, this information is used as an input to propose a more complex algorithm using swarm intelligence. Mathematical formulas such as Hooke's law [133],

Newton's second and third laws [134] are utilized to evaluate the route reply and choose the best path. For example, the vibration between two nodes, depending on Hooke's law.

Swarm behavior in animals or insects is an intelligent behavior of their biological group. The study of swarm intelligence is aimed at understanding the behavior of a group in nature. Biological scientists have found that many models can mimic the living systems of animals or insects such as the Ant colony optimization algorithm mimic the behavior of ant in finding the best path.

Most spiders do not live in communities, so swarm intelligence does not reflect the collective behavior directly, rather, in this research, the sensitive behavior of spider legs are considered to represent the collective performance. This approach is a relatively new orientation in the area of swarm intelligence.

The developing a new frameworks, which may be very useful in highly dynamic routing networks in this area. The new algorithm is applied to MANETs to address the problem of black hole attacks in the AODV routing protocol. The new proposal is based on the daddy long-legs spider's behavior in nature, as described in the next section.

#### **4.4.3.1 The Proposed PGO-DLLA Algorithm**

In the AODV routing protocol, each node has a routing table which includes the information, such as hop count, destination sequence number (DSN), life time, and

source IP address. PGO-DLLA have three routing tables; the first table (L1) contains a Source Sequence Number (SSN), Destination Sequence Number (DSN) and Lifetime of the Leg (LTL1). The second table (L2) contains SSN, DSN, and the force (F). The third table is the routing table that contains all Route Discovery (RD), Current Route Discovery (CRD), Lifetime (LT), and the Best Route (BR) to the destination node. Figure 4.19 illustrates the PGO-DLLA routing tables.

L1	L2	Routing Table										
<table border="1"> <tr><td>SSN</td></tr> <tr><td>DSN</td></tr> <tr><td>LTL1</td></tr> </table>	SSN	DSN	LTL1	<table border="1"> <tr><td>SSN</td></tr> <tr><td>DSN</td></tr> <tr><td>F</td></tr> </table>	SSN	DSN	F	<table border="1"> <tr><td>RD</td></tr> <tr><td>CRD</td></tr> <tr><td>LT</td></tr> <tr><td>BR</td></tr> </table>	RD	CRD	LT	BR
SSN												
DSN												
LTL1												
SSN												
DSN												
F												
RD												
CRD												
LT												
BR												

Figure 4.19. PGO-DLLA routing tables

The route discovery in PGO-DLLA is shown in Figure 4.20. The spider sends an agent (L1), to neighboring nodes to discover the route to the destination (prey). After broadcasting the legs to all neighboring nodes, the spider (source) waits for a lifetime (LT) for receive (L2), if the source receives L2 that means this node has a route to the destination or it is a destination. Then, the source node evaluates all route replies that come from neighboring nodes using Equation 4.9, to find the best move and select the next path.

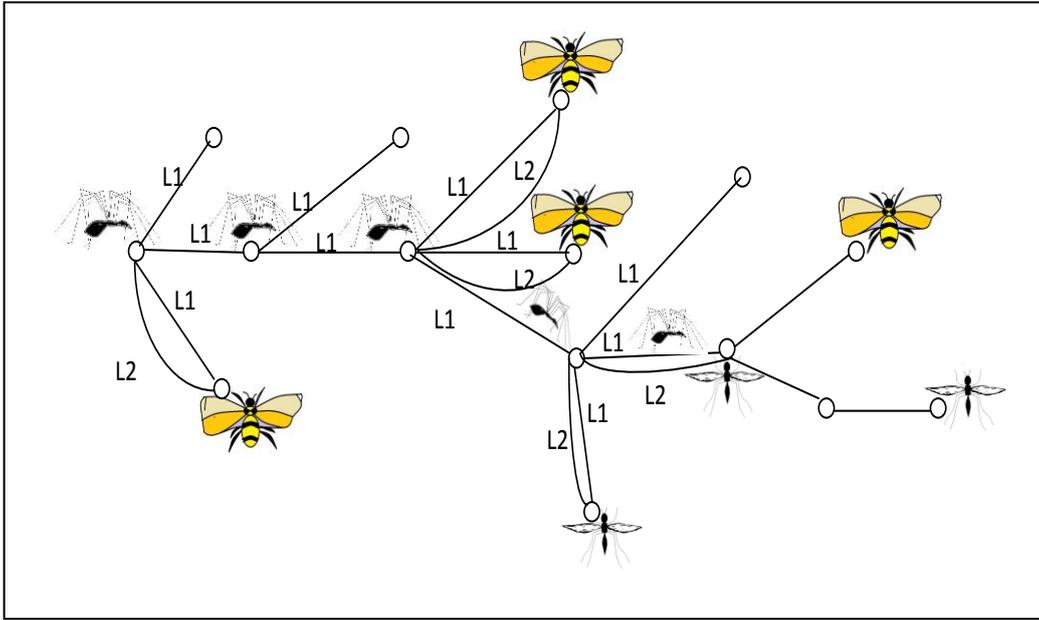


Figure 4.20. The route discovery in VDLA

Newton's second law is computed the force. According to [135], Newton's second law is stated as "The vector sum of the forces on an object is equal to the total mass of that object multiplied by the acceleration of the object". Equation 4.6 shows the original Newton's second law.

$$F_{net} = ma \quad (4.6)$$

where,  $m$  is the mass,  $a$  is the acceleration where it can also be calculated by Newton's second law in Equation 4.7.

$$a = (F_{net} / m) \quad (4.7)$$

Depending on Hook's law [133], it is stated that, "The force exerted by the spring which is proportional to the length of stretch or compression of the spring and opposite in direction to the direction of the stretch or the compression". Equation 4.8 shows the original Hook's law.

$$F = -kx \quad (4.8)$$

where,  $k$  is constant,  $x$  is displacement. By replacing Equation 4.7 with Equation 4.8, the acceleration is equal to Equation 4.9.

$$a = -\left(\frac{k}{m}\right)x \quad (4.9)$$

In this study, it is supposed that  $m$  equals to DSN, and  $k$  is the constant number which is set to 0.1.

#### 4.4.3.2 Solution Representation

The PGO-DLLA algorithm has one main goal ( shortest secure path ). The main goal can be achieved by using the objective function that includes two sub goals: shortest path and secure path. The shortest secure path in PGO-DLLA from source to destination can be calculated by the following process as shown in Figure 4.20.

```

PGO-DLL() return BestPath
#Parallel Grid Optimization based on Virtual Daddy Long-Legs Algorithm
(PGO-DLLA) to prevent black hole attacks in MANETs
Begin
1:Input:N, body position, up, down, left, right, upleft, upright, downleft, downright
2:Distribute one agent to every node that is a central station to its neighbors, and
this is done by checking the table of each node separately
3:Each agent simultaneously (applied at the same time)

4:Create two tables for each agent

    a) The distances table which represents the distance between agent and
        neighboring nodes.

    b) The acceleration table which represents the evaluation function for the
        agent to choose the best path.

5:Find the result of evaluation function for the agent using Equation 4.10.

$$a = \frac{kx}{m} \quad (4.10)$$

6:Create an ascending table for the (a) values (ListMin)
7:Calculate the value of threshold as Equation 4.11

$$Th_{Dynamic} = \frac{kx}{DSN(6\%)} \quad (4.11)$$

// Where, (6%) of the end of maximum sequence number to define black hole node//
8:ForListMin (node)
If ListMin (node) <= ThDynamic
Then select Path
Exit For
Else
delete Path from routing table
9:End For (new node)
10: BestPath=ListMin(node)
10:End

```

Figure 4.20. The pseudo code of parallel grid optimization based on virtual daddy long-legs algorithm (PGO-DLLA)

#### **4.5 Implementation of EAODV, SSP-AODV, PGO-DLLA and BAODV**

This section is focused on the development of Enhanced Ad hoc On-demand Distance Vector (EAODV) protocol, Shortest Secure Path for Ad hoc On-demand Distance Vector (SSP-AODV) protocol, Parallel Grid Optimization Daddy Long Legs Algorithm (PGO-DLLA) protocol, and the Black hole Ad hoc On-demand Distance Vector (BAODV) protocol.

##### **4.5.1 Implementing Black Hole Behavior Protocol in NS-2**

This section presents the implementation of a new routing protocol in NS-2 to simulate the black hole behavior. The implementation can be done using the AODV protocol to add the nodes that exhibit the black hole behavior. The new routing protocol is named as Black hole Ad hoc On-demand Distance Vector (BADOV). The integration of BAODV, EAODV, SSP-AODV, and PGO-DLLA protocols (protocol agent file, protocol makes agent file, and dropping or accepting packet's file) to NS-2 environment are shown in Appendix B. Some important modification of NS-2 files that need to be modified on its directory to integrate the new protocols are shown in Appendix C. To generate a new scenario, two files are created to be as an input, which are called random traffic and node movements. Then, the output that result from run the scenario are analyzed by AWK commands and drawing by graph commands. All input files and execution commands are illustrated in Appendix D.

#### 4.6 Summary

The chapter illustrated the heuristic and meta-heuristic search which explains how the new concept is going to convert the AODV protocol into a secure routing protocol in order to enhance the performance of the protocol.

This chapter explained how the proposed protocol discovers the malicious node black hole that is attacking the route discovery and make a denial of service. Furthermore, this chapter discussed the artificial intelligence elements of the EAODV protocol and the SSP-AODV protocol in detail.

Each element has an objective to enhance the performance of the AODV protocol. The first element of the EAODV and the SSP-AODV protocols is the shortest path by using an A\* heuristic search algorithm and Floyd-Warshall's to enhance the routing discovery; the second element is by using of the prevent technique to enhance the secure path to the destination in the SSP-AODV routing protocol.

After that, the chapter introduced the design of the VDLLA optimization algorithm and the PGO-DLLA algorithm design. This proposes a defense mechanism against a cooperative black hole attack in a MANET that relies on the AODV routing protocol.

The new method is called the PGO-DLLA protocol, which modifies the standard AODV and optimizes the routing process. The idea is inspired by a spider called daddy long-legs, which is a new technique for finding suspicious nodes and avoiding black hole attacks.

As a swarm algorithm, PGO-DLLA can consolidate the routing mechanism. Some changes are made in the routing tables to store the shortest and secure path from source to destination node. The main objective in this method is to avoid black hole attacks without causing delays in the routing protocol.

The experimental results show that PGO-DLLA is able to improve the performance of the network with respect to most of the performance metrics examined. Moreover, this chapter presented the implementation of black hole nodes in the Ad hoc On-demand Distance Vector (BAODV) protocol, and integrated it in network simulator NS-2.

Furthermore, this chapter presented the implementation of three new protocols that are used to achieve the shortest and secure routing in AODV. Firstly, the implementation of Enhanced Ad hoc On-demand Distance Vector (EAODV) algorithm has been presented and the steps of the integration of the protocol in NS-2 are illustrated.

In the same way, the implementation of the Shortest Secure Path Ad hoc On-demand Distance Vector (SSP-AODV) algorithm and Parallel Grid Optimization Daddy Long-Legs Algorithm are presented in this chapter.

Additionally, the modification of some important files of NS-2 is presented to integrate a new protocol in the simulator. Finally, the simulation script of NS-2 is illustrated by the flowchart to present the flow of the NS-2 scenarios.

## **CHAPTER FIVE**

### **THE RESULTS AND PERFORMANCE ANALYSIS**

#### **5.1 Introduction**

This chapter intends to demonstrate the performance of AODV in two different environments: regular and hostile environments. Regular environment is a group of nodes (i.e. computers, A personal digital assistant (PDA), mobile, and any communication devices) that communicate between them without any attacks. In contrast, hostile environment is an environment with one or more malicious nodes (black hole node). In addition, this chapter will study the important performance metrics with various settings of pause time and different numbers of nodes to predefine an appropriate optimal setting for AODV. The studying of the current behavior of AODV will involve a set of the simulations (empirical experimental) based on NS-2 as mentioned in the Chapter Three.

#### **5.2 Empirical Experiments for AODV**

The aim of this section is to explore the optimal setting for the AODV protocol. This section includes two different scenarios: empirical experiment on regular environments that tests the standard AODV with different numbers of nodes, and an empirical experiment on hostile environments to setup the appropriate pause time. Figure 5.1 illustrates the empirical experiment of AODV.

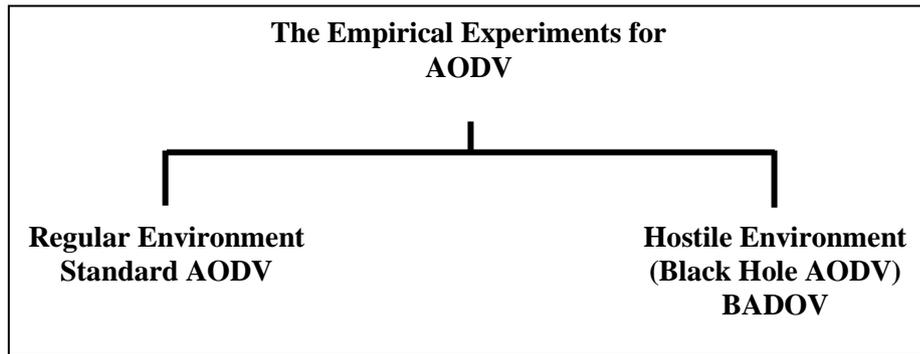


Figure 5.1. Empirical experiment of AODV

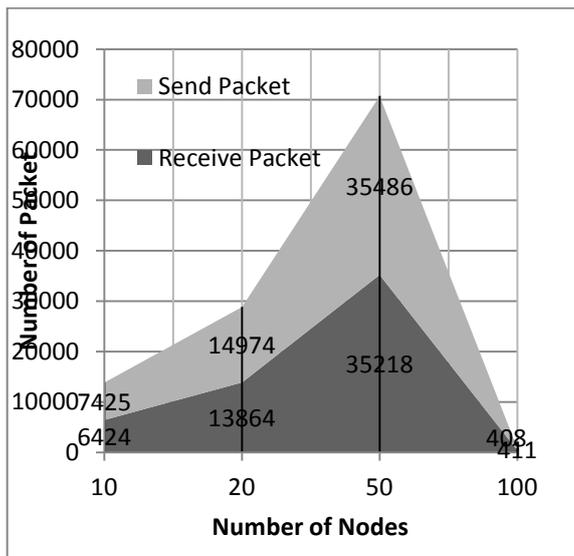
### 5.2.1 Empirical Experiment on Regular Environment

In this scenario, different parameters are set up for the standard AODV with various numbers of nodes (10, 20, 50 and 100). CBR is used as a traffic model; different numbers of nodes are distributed within a random waypoint model for mobility. The map area of simulation is 800 x 800 m. The transmission range is 250 m. The rest of the simulation parameters is indicated in Chapter three Table 3.1. The experiment is conducted four times based on the number of nodes to obtain the trace file. The trace file is analyzed six times in the AWK file based on the number of performance metrics to find the final results. Figure 5.2 illustrates the results of the performance metrics with different numbers of nodes. In Figure 5.2 (c), can be seen that the best Throughput Ratio is 13.86 and 6.34 when the number of nodes is 20 and 10 respectively. Based on the literature, the highest Throughput Ratio means the best performance for the protocol [67].

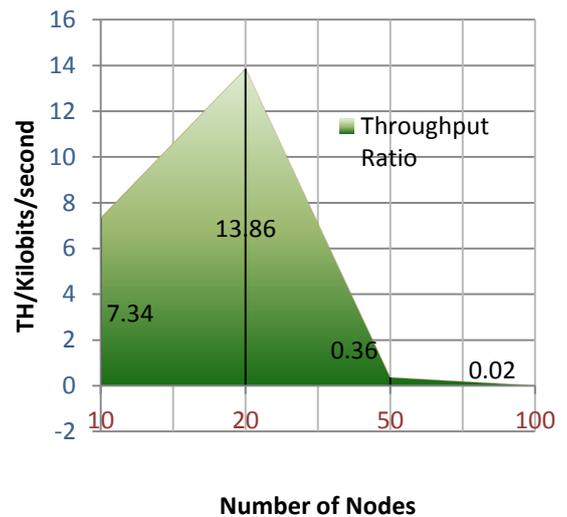
In the case of Packet Delivery Ratio (PDR), the best value in Figure 5.2 (e) is 92.88, 99.54 and 99.57 when the number of nodes is 20, 50 and 100. The largest ratio of PDR denotes better performance for the protocol [68].

The Normalized Routing Load as in Figure 5.2 (f) is 0.09 and 0.15 better in nodes 10 and 20, and depending on the literature, the smallest value of Normalized Routing Load means the best performance for the protocol [67], [68]. Additionally, the best value of Average End-to-End in Figure 5.2 (g) is 34.93 and 33.64 obtained in 50 and 100 nodes. The smallest value of End-to-End means the best protocol performance [67], [68].

In addition, the Packet Loss Ratio in Figure 5.2 (h) is 0.75 and 0.72 generated in nodes 50 and 100. However when the total node is 20, can be seen that the acceptable value is better than node 10. When Packet Loss Ratio is small, this means a good result for protocol performance [68].



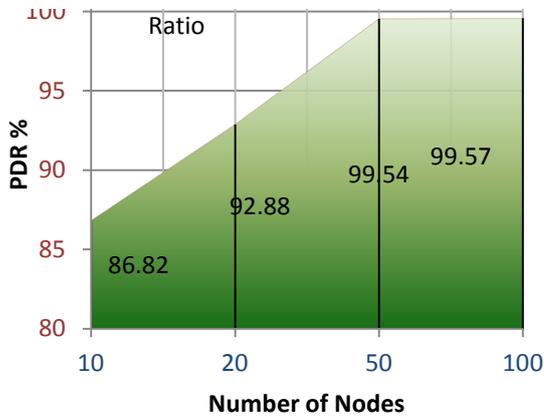
(a) Send and Receive Packet with Different



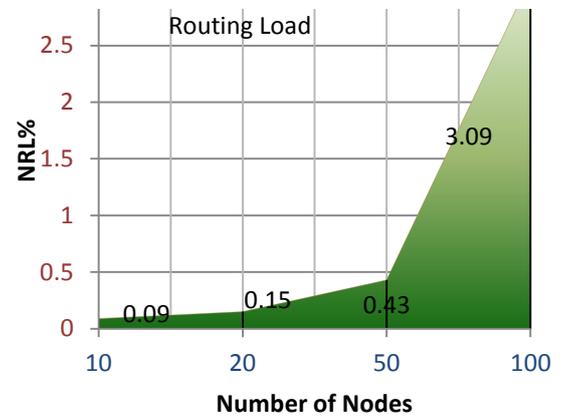
(b) Throughput Ratio with Different No.

Figure 5.2. continued

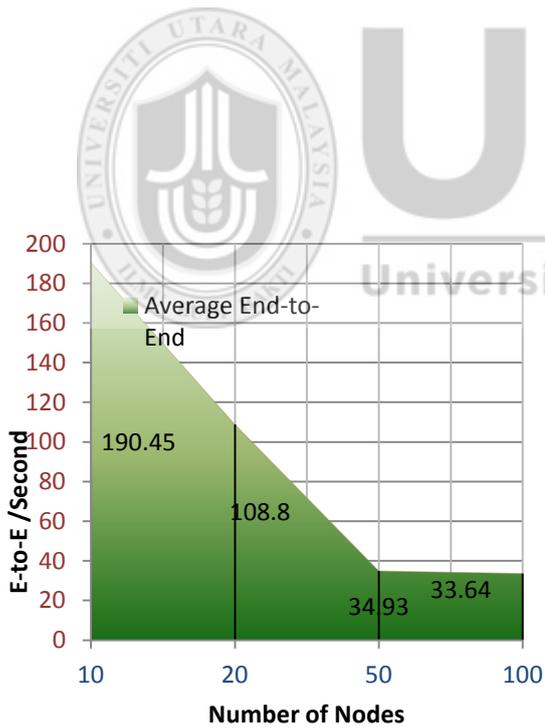
Figure 5.2. The performance metrics of standard AODV with various numbers of nodes



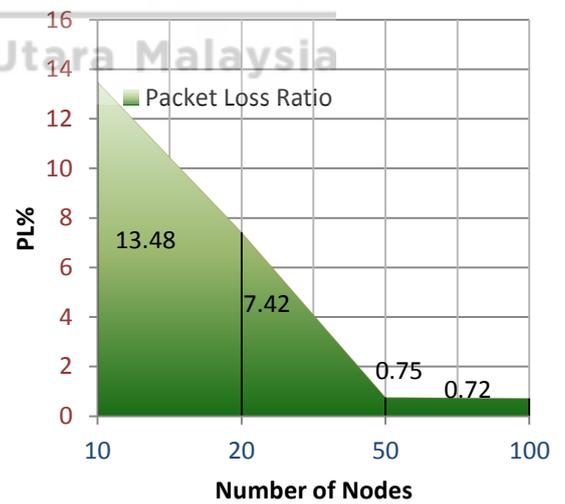
(c) PDR with Different No. of Nodes



(d) Normalization with Different No. of Nodes



(e) Average EtoE with Different No. of Nodes



(f) PL with Different No. of Nodes

In Table 5.1, the performance metric results are compared together to find the best number of nodes and this result is applied in the second simulation. As a result, the best number of node is 20 nodes based on the all of the performance metrics.

Table 5.1

*The comparison of performance metrics result for AODV.*

Performance Metrics	The Number of Nodes			
	10	20	50	100
Throughput Ratio%	√	√	X	X
Packet Delivery Ratio%	X	√	√	√
Normalized Routing Load	√	√	√	X
Average End-to-End	X	√	√	√
Packet Loss Ratio%	X	√	√	√

Based on [77] the comparison of three performance metrics: the packet loss percentage, normalized routing load, and throughput of AODV, are best to show the efficiency protocol. Figure 5.3 illustrates the curves of three chosen performance. It can be concluded from Figure 5.3 that the best number of node is 20 nodes.

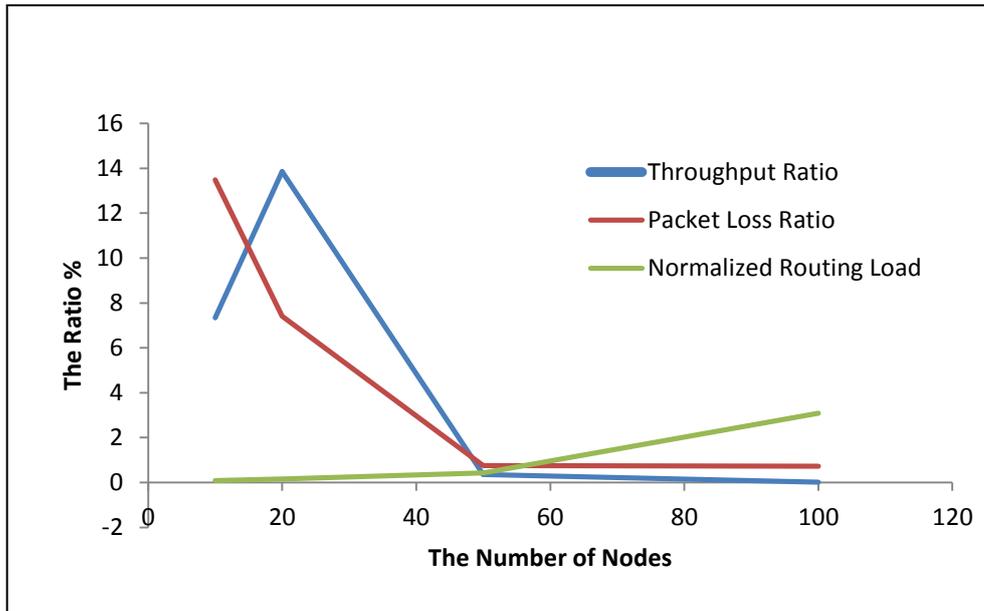


Figure 5.3. Comparison result of the performance metrics TH, PL, and NRL

From the previous comparison result, the number 20 is concluded as the best setting of node in AODV that is used with the other experiments in this chapter.

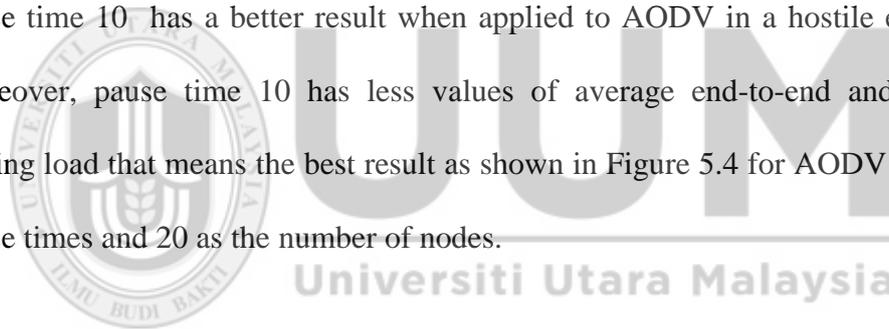
### 5.2.2 Empirical Experiment on Hostile Environment

The term "hostile environment" means unreliable environment which has some unfriendly nodes. The main function of nodes in AODV is being cooperative with each other in a meaningful way. Unfortunately, if at least one of the nodes want to get away without cooperating with its community, this changes the current environment to a hostile environment.

In this section, some of the scenarios have been simulated which include a black hole environment using NS-2 to integrate the black hole nodes in the AODV protocol; a new simulation with malicious nodes has been implemented that has the ability to

drop data packets after attracting them to itself. The second scenario has been done with the same parameter setting, except for the number of nodes which are set to 20, and with various pause times (10, 20, 40, 60, 80, and 100) as shown in Chapter three Table 3.1.

The simulation period of 100 seconds with a hostile environment means using a black hole node as a malicious node to attack the AODV protocol. This experience has been re-implemented with six pause times which is equivalent to 12 scenarios in each experiment. The protocol name and pause time are varied, while the rest of parameters are fixed. In this scenario, the packet delivery ratio and throughput for pause time 10 has a better result when applied to AODV in a hostile environment. Moreover, pause time 10 has less values of average end-to-end and normalized routing load that means the best result as shown in Figure 5.4 for AODV with various pause times and 20 as the number of nodes.



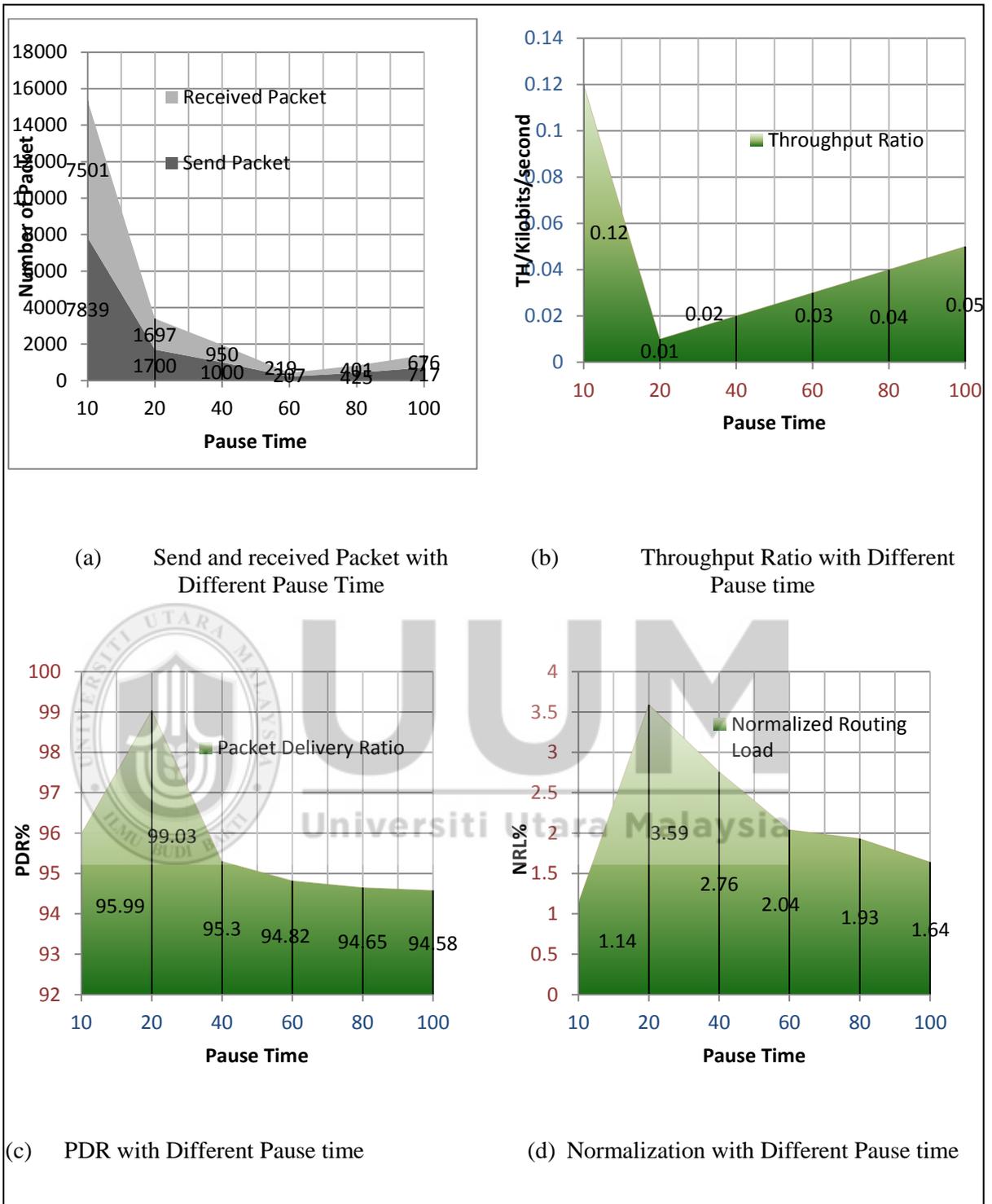


Figure 5.4. The performance metrics of standard AODV with various pause time

Figure 5.4. continued

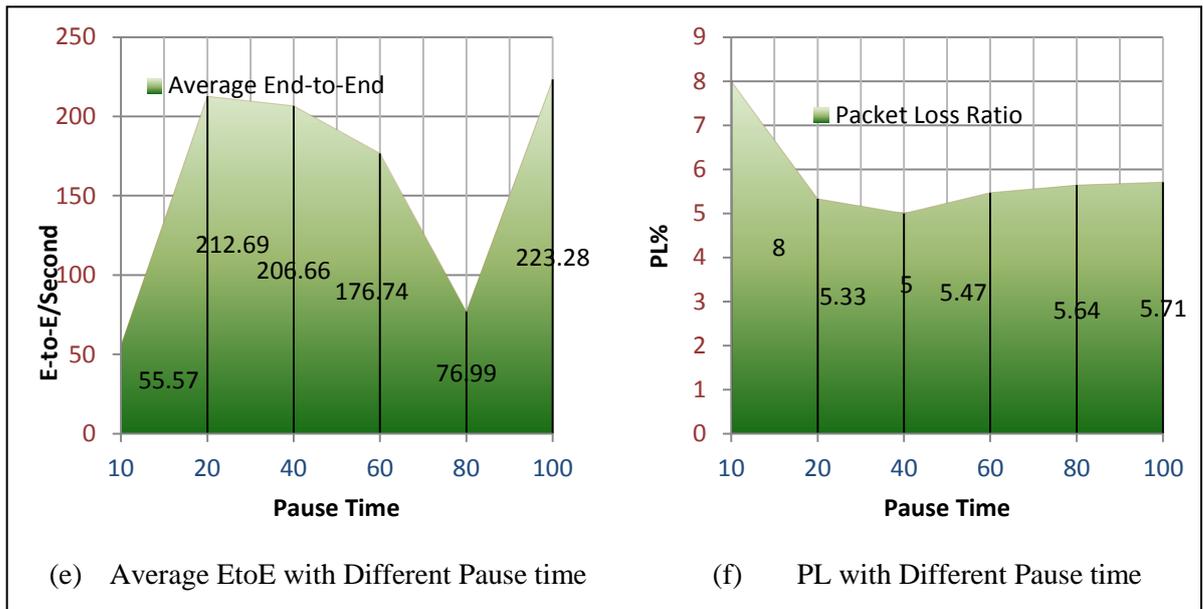


Figure 5.4 shows the simulation result of AODV in a hostile environment with black hole nodes and the same parameters of the second scenario. The simulations have been conducted 36 times to find the result for six performance metric (TH, PDR, NRL, E-to-E, PL, and send received packets) with six pause time (10, 20, 40, 60, 80, and 100) in Figure 5.5.

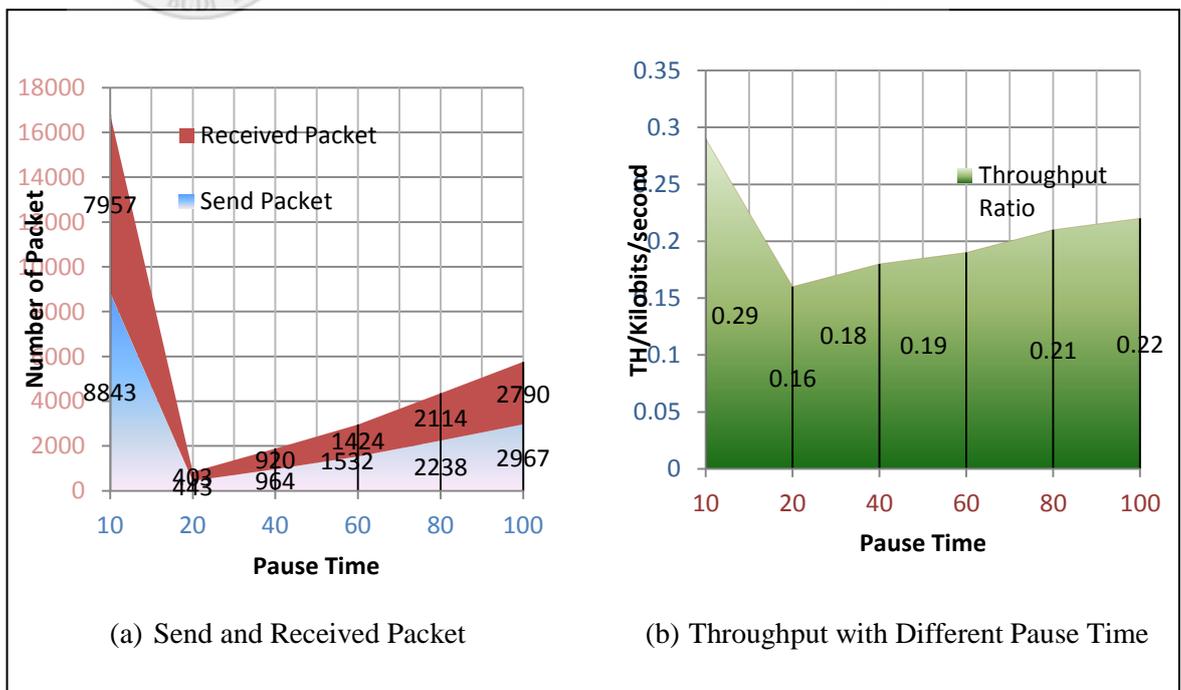


Figure 5.5. The performance metrics of BAODV with various pause time

Figure 5.5. Continued

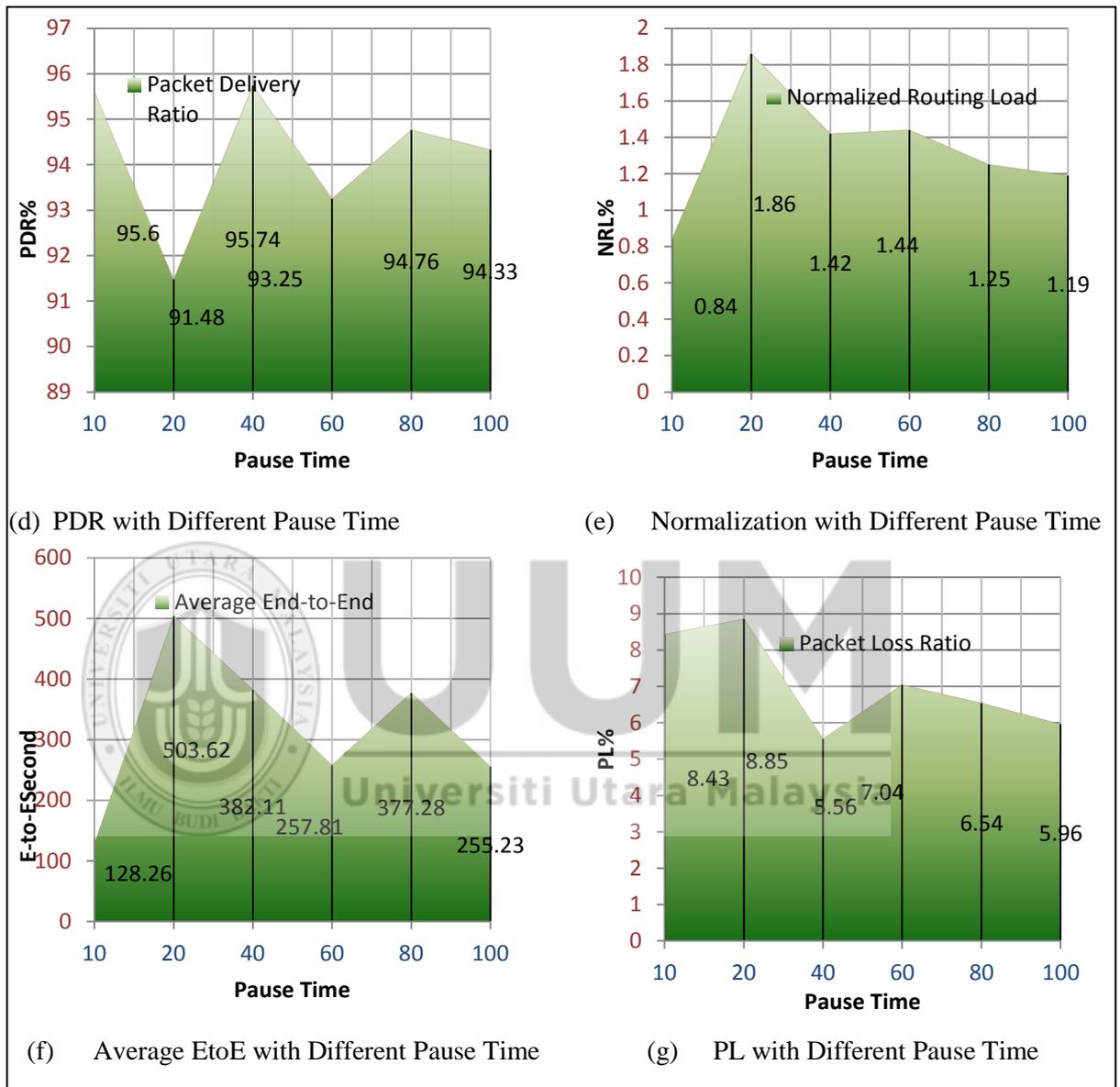


Table 5.2 shows the result of comparing various pause times based on the performance metrics to find the best pause time setting for the next simulations. The result shows that the AODV protocol is better than the BAODV protocol (60, 80, and 100) pause times, which is because of the low mobility and the none existence black hole nodes in the network.

High mobility in (10, 20, and 40) pauses times and black hole nodes were affecting AODV and made it worse when compare with BADOV.

As a result, the pause time with  $t=10$  will be chosen to check the bad condition with black hole as the hostile environment, or  $t=100$  as a good condition for regular AODV without black hole nodes.

Table 5.2

*The Performance Metrics for AODV and BAODV.*

<b>Performance Metrics</b>	<b>t=10</b>	<b>t=20</b>	<b>t=40</b>	<b>t=60</b>	<b>t=80</b>	<b>t=100</b>
Throughput Ratio	X	X	X	X	X	X
Packet Delivery Ratio	√	√	X	√	-	√
Normalized Routing Load	X	X	X	X	√	√
Average End-to-End	√	√	√	√	√	√
Packet Loss Ratio	√	√	√	√	√	√

### 5.3 Experiments Setup, Results and Analysis of EAODV Protocol

NS-2 simulator version 2.33 [136], [137] has been used to experiment three scenarios. Scenario 1 is to test the original AODV, Scenario 2 is to test the black hole attack in AODV and Scenario 3 is to test the execution of the proposed EAODV in finding the shortest secure path and securing the AODV protocol. The Simulation Parameters for Scenarios 1,2 and 3 are setting such as the standard table in Chapter three Table 3.1, except the size of network change to 50 nodes as a huge scenario and as show in Figure 5.3 that is comparison result of the performance metrics TH, PL, and NRL, The three metrics are the most important for best-effort traffic [6]. The rest of the simulation parameters is indicated in Table 5.3.

Table 5.3

*Simulation Parameters for EAODV, BAODV, and Standard AODV.*

Parameter	<i>Scenario1</i>	<i>Scenario2</i>	<i>Scenario3</i>
<i>Simulation Time</i>	1000 sec.	1000 sec.	1000 sec.
<i>Number of Nodes</i>	50	50	50
<i>Routing Protocol</i>	AODV	BAODV	EAODV
<i>Traffic Model</i>	CBR(UDP)	CBR(UDP)	CBR(UDP)
<i>Pause Time</i>	10, 20, 40, 60, 80 and 100 sec.	10, 20, 40, 60, 80 and 100 sec.	10, 20, 40, 60, 80 and 100 sec.
<i>Maximum Mobility</i>	60 m/sec.	60 m/sec.	60 m/sec.
<i>Map area</i>	800m x 800m	800m x 800m	800m x 800m
<i>Transmission Range</i>	250m	250m	250m
<i>Malicious Node</i>	1	1	1

### 5.3.1 Packet Loss: Results and Discussion

In Figure 5.6, three scenarios: original AODV, black hole AODV and EAODV, are compared. The packet loss ratio increases in black hole AODV, which degrades the performance of the protocol and causes many packet losses.

As a result, it triggers a DoS attack in MANET. The proposed EAODV minimize the packet loss and improves the network performance as compared to the original AODV. Packet loss is 21.41% in AODV, but it increases in black hole AODV, which is 28.32%, and after implementing EAODV, the percentage is improved, to 24.96%. The decrease of packet loss with black hole AODV is compared with the result of EAODV; this means some improvement has been conducted in avoiding the black hole attack. After the original AODV packet loss was increased 7.8 % with black hole AODV, the packet loss was decreased to 3.36 % with EAODV.

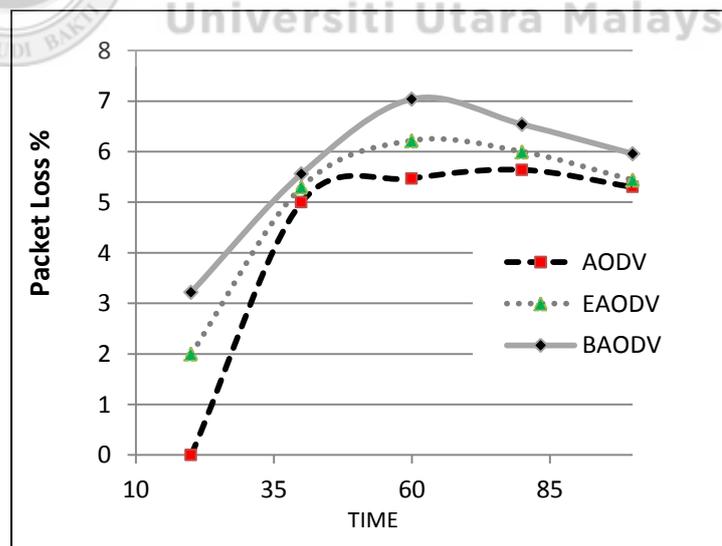


Figure 5.6. Packet loss percentage for AODV, black hole AODV and EAODV

### 5.3.2 Average End-to-End Delay: Results and Discussion

Figure 5.7 shows the comparison of the average End-to-End delay of the three scenarios. The average End-to-End Delay increases with the existence of black holes. This delay degrades the performance of the network and causes more delay time when the packets try to reach the destination node.

Furthermore, when the original AODV is compared with the proposed protocol EAODV, the result indicated that EAODV minimizes the Average End-to-End Delay and improves the network performance. The percentage of delay was 29% with black hole node compared with the original AODV. This percentage is about 11.09% with EAODV.

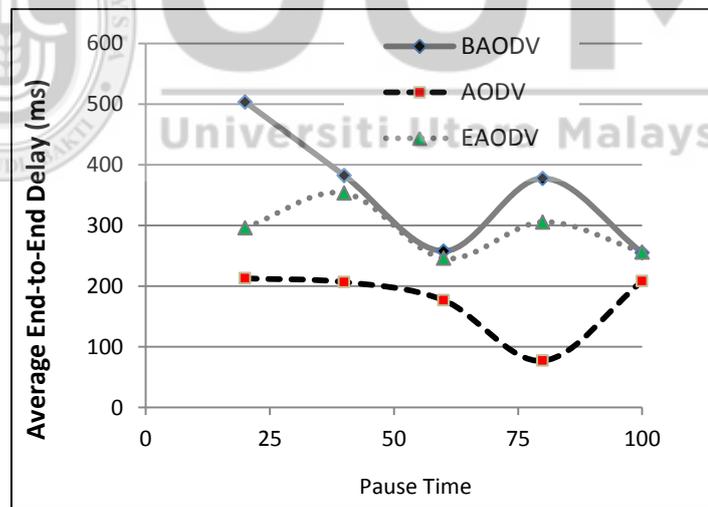


Figure 5.7. The average end-to-end delay for AODV, black hole AODV and EAODV

### 5.3.3 Packet Delivery Ratio (PDR) Results and Discussion

Figure 5.8 displays the PDR for the three scenarios. It can be seen from the graphs that the packet delivery ratio does not increase with the existence of the black holes in the network. The packets that reached the destination from the source node was 479.77 in total for standard AODV, 469.56 for AODV with black hole nodes, and 447.43 for EAODV. Therefore, it can be seen that the overall PDR for EAODV does not degrade significantly due to the implementation of security algorithm.

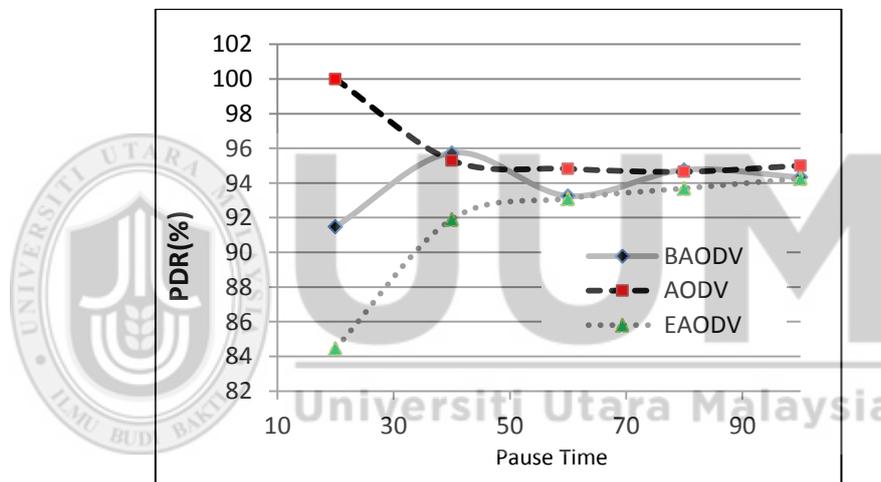


Figure 5.8. The packet delivery ratio for AODV, black hole AODV and EAODV

### 5.3.4 The Normalized Routing Load (NRL) Results and Discussion

In Figure 5.9, several increase of the routing load can be seen in the proposed EAODV protocol as compared to the standard AODV due to the new mechanism to find the shortest path from source to destination node and the lack of strategy to prevent the black hole attack. The increase in NRL for BAODV is very close to the proposed protocol due to the black hole nodes that try to drop the packets.

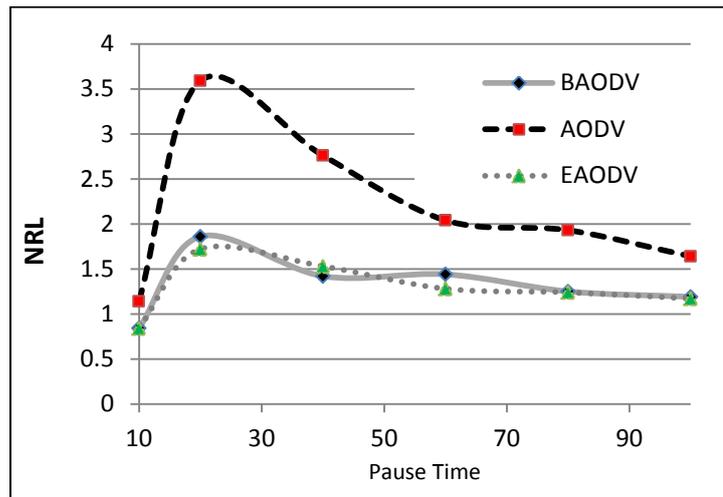


Figure 5.9. The normalized routing load for AODV, black hole AODV and EAODV

### 5.3.5 The Throughput Ratio (TH): Results and Discussion

In Figure 5.10, it displays the comparison result between the proposed EAODV protocol with black hole AODV protocol and the standard AODV protocol in terms of throughput. The results show that the EAODV protocol outperformed the other two protocols. In EAODV, the use of the heuristic A\* to routing discovery is made to find the shortest path from source to destination node. As the new mechanism discovers the best path with the lesser number of hops, this gives effect in avoiding the black hole attack and dropping packets. The TH in EAODV increases as a result to the improving route discovery.

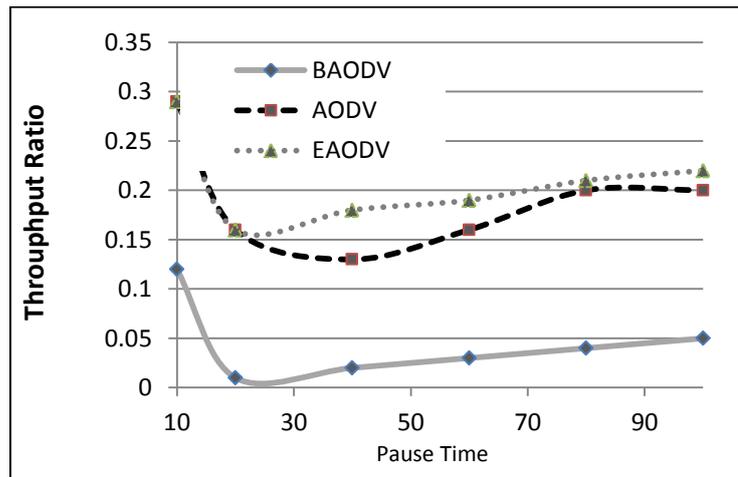


Figure 5.10. The throughput ratio for AODV, black hole AODV and EAODV

#### 5.4 Experiments Setup, Results and Analysis of SSP-AODV Protocol

The NS-2 simulator version 2.33 is used in this experiment as in the previous experiments discussed in Sections 5.2 and 5.3. Three scenarios are conducted: the first scenario tests the standard AODV routing protocol, the second scenario tests the black hole AODV (BAODV) routing protocol, and the third scenario tests the proposed SSP-AODV routing protocol in finding the shortest and secure path.

The size of network and mobility are constant, where the number of nodes is equal to 50 and the maximum mobility is set to 60 m/second. However, the pause time is varied. The rest of the simulation parameters in the first scenario, second scenario, and third scenario are shown in Table 5.4.

Table 5.4

*Simulation Parameters for three Scenarios with SSP-AODV, BAODV and Standard AODV Routing protocol.*

<b>Parameter</b>	<b>Simulation1</b>	<b>Simulation2</b>	<b>Simulation3</b>
<i>Simulation Time</i>	1000 sec.	1000 sec.	1000 sec.
<i>Number of Nodes</i>	50	50	50
<i>Routing Protocol</i>	AODV	Black Hole-AODV	SSP-AODV
<i>Traffic Model</i>	CBR(UDP)	CBR(UDP)	CBR(UDP)
<i>Pause Time</i>	10, 20, 40, 60,	10, 20, 40, 60,	10, 20, 40, 60,
<i>Maximum Mobility</i>	80 and 100 sec.	80 and 100 sec.	80 and 100 sec.
<i>No. of sources</i>	60 m/sec.	60 m/sec.	60 m/sec.
<i>Map area</i>	1	1	1
<i>Transmission Range</i>	800m x 800m	800m x 800m	800m x 800m
<i>Malicious Node</i>	250m	250m	250m
	1	1	1

#### 5.4.1 Packet Loss (PL) Results and Discussion

Three scenarios, original AODV, black hole AODV and SSP-AODV, are compared. The packet loss ratio increases in black hole AODV, which degrades the performance of the protocol and causes many packet losses, which in turn, may trigger a DoS attack.

Compared to the original AODV, the proposed SSP-AODV indicates that SSP-AODV minimizes the packet loss and improves the network performance. Packet loss is 21.41% in AODV, but increases with black hole to 28.32%, and with SSP-AODV, the percentage increases to 22.98%, as shown in Figure 5.11.

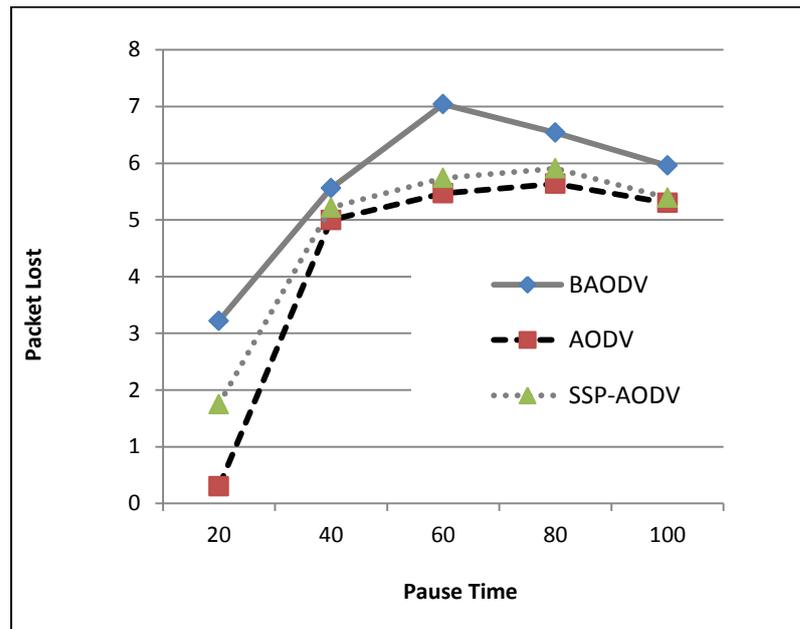


Figure 5.11. The packet loss percentage for AODV, BAODV and SSP-AODV

#### 5.4.2 Average End-to-End Delay: Results and Discussion

Figure 5.12 shows the comparison of the average End-to-End delay of the three scenarios. The average End-to-End Delay increases with the existence of black holes. This delay degrades the performance of the network and causes more delay time when packets try to reach the destination node.

Furthermore, when the original AODV is compared with the proposed SSP-AODV protocol, the result indicated that SSP-AODV minimizes the Average End-to-End Delay and improves the network performance. The percentage of delay is 29% with black hole node as compared to the original AODV. This percentage is about 11.09% with SSP-AODV.

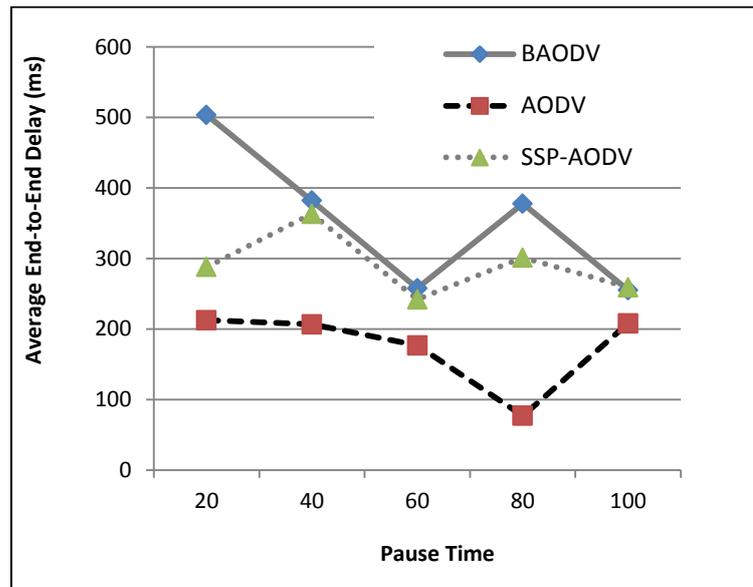


Figure 5.12. The average end-to-end delay for AODV, BAODV and SSP-AODV

### 5.4.3 Packet Delivery Ratio (PDR) Results and Discussion

Figure 5.13 the demonstrated PDR for the three scenarios. It can be seen from the graphs that the packet delivery ratio does not increase with the existence of the black holes in the network. The packets that reach to the destination from the source node is 479.77 in total for standard AODV, 469.56 for AODV with black hole nodes, and 447.43 for SSP-AODV. Thus, it can be seen that the overall PDR of SSP-AODV does not degrade significantly due to the implementation of security algorithm.

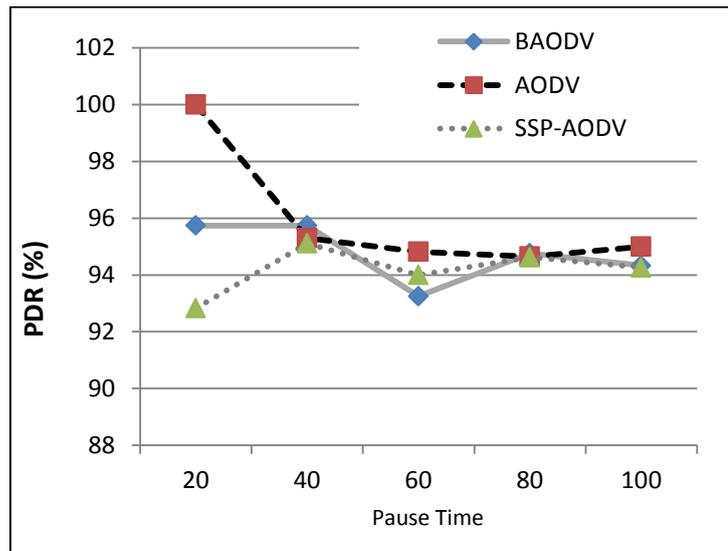


Figure 5.13. The packet delivery ratio for AODV, BAODV and SSP-AODV

#### 5.4.4 The Normalized Routing Load (NRL): Results and Discussion

In Figure 5.14, the decreasing NRL in the proposed SSP-AODV protocol is compared with the black hole AODV and the standard AODV due to the new mechanism to find the shortest path from source to destination node and the strategy to prevent the black hole attack. The increasing NRL of BAODV is very high due to the black hole nodes that try to drop the packets and the increasing link failure. As the simulation experiments for the standard AODV have conducted in a regular environment without any black hole node attacking the routing mechanism, the NRL is not very high and is very close to SSP-AODV. Some improvement has been done in the NRL performance metric for the prevention of the black hole attacks and zero network overhead.

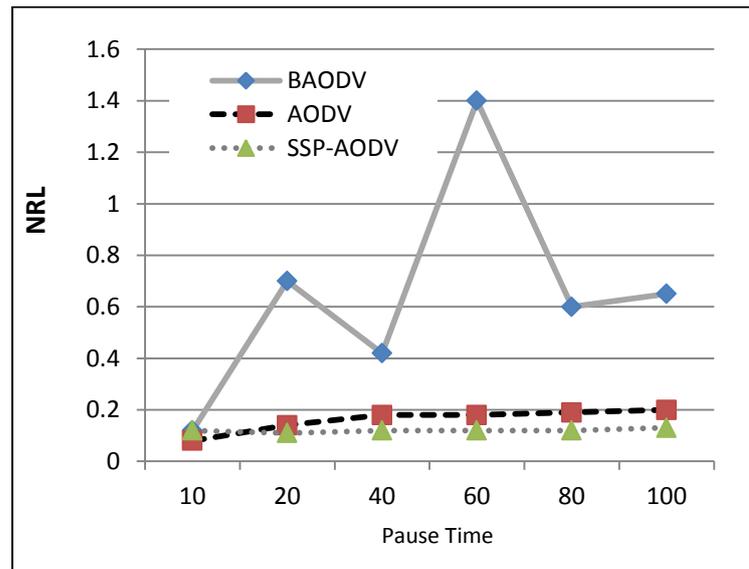


Figure 5.14 . The normalized routing load for AODV, BAODV and SSP-AODV

#### 5.4.5 The Throughput Ratio (TH): Results and Discussion

Figure 5.15 demonstrates the result of the comparison between the proposed SSP-AODV protocol with the black hole AODV protocol and the standard AODV protocol in terms of throughput. The results show that the SSP-AODV protocol outperformed the other two protocols. In SSP-AODV, the integration of the heuristic A\* and Floyd-Warshall's algorithm to routing discovery is performed to find the shortest path from source to destination node. As the new mechanism discovers the best path with the lesser number of hops and the technique prevents the black hole attack, some improvement for the TH performance metric are done by preventing the damages that occur due to the black hole attacks and dropping of packets. The TH in SSP-AODV increases as a result to the improving route discovery.

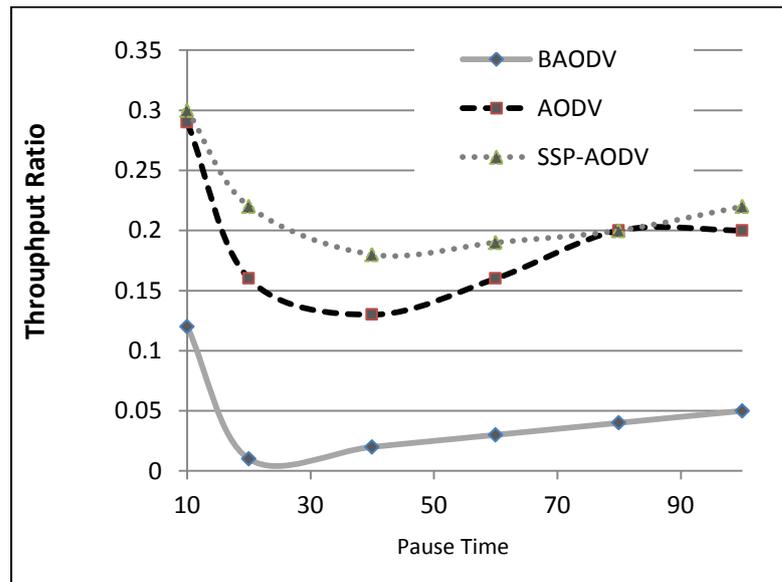


Figure 5.15. The throughput ratio for AODV, BAODV and SSP-AODV

### 5.5 Evaluation of Virtual Daddy Long-Legs Algorithm

The proposed algorithm is tested on four standard functions that have been extensively used in the evaluation of a new optimization algorithm. The standard functions Rosenbrock's function [130], Michalewicz's function [131], EggCrate's function [130], and Beal's function [130] are shown in Table 5.5 includes the performance result of the proposed VDLLA, Differential Evolution (DE), Particle Swarm Optimization (PSO), and BAT algorithm (BA). The data indicates that VDLLA has 4.78E-06, the smallest Average of Best minimum Fitness, as compared to BA which has 4.92E-06, PSO for Rosenbrock's function, 5.30E-06, and DE 5.74E-06. Furthermore, for the four functions, VDLLA generates 4.22E-06, the smallest value of Medium of Best minimum Fitness, against BA which produces 4.62E-06, PSO with 5.04E-06 and DE that produces 6.18E-06. The Standard Deviation of Best minimum

Fitness for VDLLA is  $2.52E-06$ , smaller than BAT which is  $2.69E-06$ , DE  $2.76E-06$  and PSO  $3.04E-06$ .

This means the proposed VDLLA algorithm generates the best result for the Rosenbrock's Function. It can be seen that VDLLA performs  $-1.80046$  much better than BA  $-1.70237$  and PSO  $-1.55568$  algorithms, while the DE algorithm  $-1.85158$  is much superior to the other algorithms in terms of the smallest Average of Best minimum Fitness with the Michalewicz's function. However, the proposed VDLLA generates  $-18013$  and  $5.64E-04$  in terms of Medium of Best minimum Fitness and Standard Deviation of Best minimum Fitness respectively, much better than BA that produces  $-1.80065$  and  $3.09E-01$ , DE generates  $-1.8001$ , and  $7.49E-02$ , and finally PSO  $-1.7899$ , and  $3.55E-01$ , respectively.

For the EggCrate's function, the proposed VDLLA generates  $5.68E-06$  for the smallest Average of Best minimum Fitness compared to BA which has  $5.33E-06$ , PSO has  $2.846593$ , while DE has  $4.81E-06$ , better than the other algorithms (refer to VDLLA, BA and PSO). Furthermore, for the other functions, VDLLA generates  $5.51E-06$ , the smallest value of Medium of Best minimum Fitness, against BA which produces  $5.57E-06$ , PSO produces  $7.43E-06$ , while DE produces  $4.53E-06$ . The Standard Deviation of Best minimum Fitness for VDLLA  $2.50E-06$  is smaller than BA which is  $2.54E-06$ , DE  $2.96E-06$ , and PSO  $4.34E-06$ . This means the proposed VDLLA algorithm generated the best result in terms of the Standard Deviation of Best minimum fitness in the EggCrate's function. For the Beale's function, the proposed

VDLLA generated  $4.15E-06$ , the smallest Average of Best minimum Fitness, as compared to BA which has  $1.07E-01$ , PSO has  $2.66E-01$  and, DE has  $4.66E-06$ .

Furthermore, for the similar function, VDLLA generates  $3.30E-06$ , the smallest value of Medium of Best minimum Fitness against BA which produces  $6.14E-06$ , PSO produces  $8.48E-06$  and, DE produces  $5.33E-06$ . The Standard Deviation of Best minimum Fitness for VDLLA  $2.82E-06$  is smaller than BA which is  $2.20E-01$ , and PSO  $3.77E-01$ , while DE is  $2.74E-06$ .

This means the proposed VDLLA algorithm generated the best result in the Beale's function in terms of the smallest Average of Best minimum Fitness (AbmF) and the smallest value of Medium of Best minimum Fitness (MbmF).



Table 5.5

*Performance Results of proposed VDLLA vs. DE vs. PSO vs. BA, bold values mean best result, Average of Best minimum Fitness =  $AbmF$ , Medium of Best minimum Fitness =  $MbmF$ , Standard Deviation of Best minimum Fitness =  $SDbmF$ .*

Functions	Performance Index	DE	PSO	BA	VDLLA
Rosenbrock's	$AbmF$	5.74E-06	5.30E-06	4.92E-06	<b>4.78E-06</b>
	$MbmF$	6.18E-06	5.04E-06	4.62E-06	<b>4.22E-06</b>
	$SDbmF$	2.76E-06	3.04E-06	2.69E-06	<b>2.52E-06</b>
Michalewicz's	$AbmF$	<b>-1.85158</b>	-1.55568	-1.70237	-1.80046
	$MbmF$	-1.8001	-1.7899	-1.80065	<b>-1.8013</b>
	$SDbmF$	7.49E-02	3.55E-01	3.09E-01	<b>5.64E-04</b>
EggCrate's	$AbmF$	<b>4.81E-06</b>	2.846593	5.33E-06	5.68E-06
	$MbmF$	<b>4.53E-06</b>	7.43E-06	5.57E-06	5.51E-06
	$SDbmF$	2.96E-06	4.348236	2.54E-06	<b>2.50E-06</b>
Beale's	$AbmF$	4.66E-06	2.66E-01	1.07E-01	<b>4.15E-06</b>
	$MbmF$	5.33E-06	8.48E-06	6.14E-06	<b>3.30E-06</b>
	$SDbmF$	<b>2.74E-06</b>	3.77E-01	2.20E-01	2.82E-06

Table 5.6 tabulates the two-tailed p-value test provided by SPSS 17 that is generated by hypothesis testing of differences of means for independent samples T-Test comparing VDLLA vs. BA, VDLLA vs. PSO, and VDLLA vs. DE over the average of best minimum fitness value ( $AbmF$ ) from Table 5.5.

Independent samples T-Test is a non-parametric statistical approach used to test significant proof between the means of two algorithms [108]. In this test, two

hypotheses are supposed: null hypothesis (H0), where there is no significant difference between the mean values of two algorithms, and alternative hypothesis (H1), where there is a significant difference between the mean values of two algorithms.

**H0:** No significant difference between the mean values (AbmF) of two algorithms.

**H1:** Significant difference between the mean values of (AbmF) of two algorithms.

Table 5.6

*P-value generated by T-Test for independent samples comparing VDLLA vs. BA, VDLLA vs. PSO, and VDLLA vs. DE, over the average of best minimum Fitness (AbmF) value from Table 5.5, bold value means significant proof.*

Functions	VDLLA vs. BA		VDLLA vs. PSO		VDLLA vs. DE	
	Assuming equal variance	Assuming unequal variance	Assuming equal variance	Assuming unequal variance	Assuming equal variance	Assuming unequal variance
Rosenbrock's	0.415	0.415	0.2405	0.2405	<b>0.085</b>	<b>0.085</b>
Michalewicz's	0.465	0.49	<b>0.000</b>	<b>0.0005</b>	<b>0.005</b>	<b>0.005</b>
EggCrate's	0.2955	0.2955	<b>0.0005</b>	<b>0.0005</b>	0.114	0.114
Beale's	<b>0.006</b>	<b>0.007</b>	<b>0.000</b>	<b>0.0005</b>	0.2415	0.2415

In Table 5.6, by using the Rosenbrock's function, the p-values of the test via the equal variance assumption between VDLLA vs. PSO, VDLLA vs. BA and VDLLA vs. DE are 0.2405, 0.415 and 0.085 respectively, and the unequal variance assumption between VDLLA vs. PSO, VDLLA vs. BA and VDLLA vs. DE are 0.2405, 0.415 and 0.085 respectively. Since the values are greater than  $\alpha = 0.05$ , the null hypothesis H0 is accepted and it can be concluded that there is no significant difference between the mean values of VDLLA vs. PSO, VDLLA vs. BA and VDLLA vs. DE.

Furthermore, in Table 5.6, by using the Michalewicz's function, it can be noticed that the p-values of the test assuming equal variance between VDLLA vs. PSO and VDLLA vs. DE are 0.000 and 0.005 respectively, and the p-values of the test assuming unequal variance between VDLLA vs. PSO and VDLLA vs. DE are 0.0005 and 0.005. Since both values are less than  $\alpha = 0.05$ , the alternative hypothesis H1 is accepted and it can be concluded that there is a significant difference between the mean values of VDLLA vs. PSO and VDLLA vs. DE. The p-values of the test assuming equal variance between VDLLA vs. BA is 0.465, and assuming unequal variance is 0.49. Since the value is greater than  $\alpha = 0.05$ , the null hypothesis H0 is accepted and it can be concluded that there is no significant difference between the mean values of VDLLA vs. BA.

In addition, in Table 5.6, by using the EggCrate's function, it can be noticed that in the two-tailed p-value test between VDLLA vs. PSO, the p-value of the test assuming equal variance is 0.0005 and assuming unequal variance is 0.0005. Since both p-values are less than  $\alpha = 0.05$ , the alternative hypothesis H1 is accepted and it can be concluded that there is a significant difference between VDLLA and PSO. The p-values of the test assuming equal variance between VDLLA vs. DE and VDLLA vs. BA are 0.114 and 0.2955, and assuming unequal variance between VDLLA vs. DE and VDLLA vs. BA are 0.114 and 0.2955 respectively. Since both values are greater than  $\alpha = 0.05$ , the null hypothesis is accepted.

and it can be concluded that there is no significant difference between the mean values of VDLLA vs. DE and VDLLA vs. BA.

Moreover, in Table 5.6, by using the Beale's function, it can be seen that the p-values of the test assuming equal variance between VDLLA vs. PSO and VDLLA vs. BA are 0.000 and 0.0065, and assuming unequal variance between VDLLA vs. PSO and VDLLA vs. BA are 0.0005 and 0.007 respectively. Since both values are less than  $\alpha = 0.05$ , the alternative hypothesis  $H_0$  is accepted and it can be concluded that there is a significant difference between the mean values of VDLLA vs. PSO and VDLLA vs. BA. The p-value of the test assuming equal variance between VDLLA vs. DE is 0.2415, and assuming unequal variance is 0.2415. Since the value is greater than  $\alpha = 0.05$ , the null hypothesis is accepted and it can be concluded that there is no significant difference between the mean values of VDLLA vs. DE.

A new Virtual Daddy Long-Legs Algorithm (VDLLA) has been successfully formulated for solving optimization tasks and adjusting the balance between exploration and exploitation. The VDLLA algorithm is based on the simulation of the behavior of a daddy long-legs spider based on the biological laws of hunting the prey on spider webs. According to most of the existing swarm algorithms, the success of any optimization algorithm depends on the balance between exploration and exploitation. In VDLLA, each spider has nine positions as matrix (3x3) in a grid space, where eight positions are for eight legs and one center position for the body of the spider. Each spider evaluates the nine positions based on the objective function and determines the best location from nine positions.

The best position for each spider is evaluated to choose a global position. The performance of VDLLA has been compared with DE, PSO and BA, four swarm

algorithms. Additionally, the T-Test has been conducted to show the significant difference between the proposed algorithm and other algorithms. VDLA showed very promising results on the benchmark test functions for unconstrained optimization problems and also significantly improved the original swarm algorithms.

## **5.6 Experimental Results of PGO-DLLA Protocol**

Simulation 1 tests the original AODV, Simulation 2 tests the black hole AODV, Simulation 3 tests the AntNet algorithm [138], [139], and Simulation 4 tests the proposed PGO-DLLA in discovering the shortest secure path. The parameters for Simulations 1 to 4 are same to the parameters that are shown in Table 5.4, except the number of malicious nodes is set to two nodes in BAODV, AntNet, and PGO-DLLA to test the performance the new algorithms in hostile environments. The experiments involve two parts; the comparison of performance results of PGO-DLLA with AntNet, and the comparison of performance results of PGO-DLLA with AODV and BAODV.

### **5.6.1 Results and Discussion of Comparison between PGO-DLLA and AntNet**

In this scenario, the pause time varies from 0 to 100 seconds as shown in the parameters of scenario in Table 5.9. In Figure 5.24, the PDR for the PGO-DLLA algorithm is relatively better when compared with the AntNet algorithm for most sets of connections. This is because the new routing characteristics of the proposed PGO-DLLA algorithm is better, in which it has the ability to find the shortest route between source and destination nodes with minimum number of hops. Generally, when the algorithm selects a route based on less hop counts, it chooses the shortest path to the

destination node and avoids a link failure. Therefore, some important performance metrics, such as average end-to-end delay, may decrease [140].

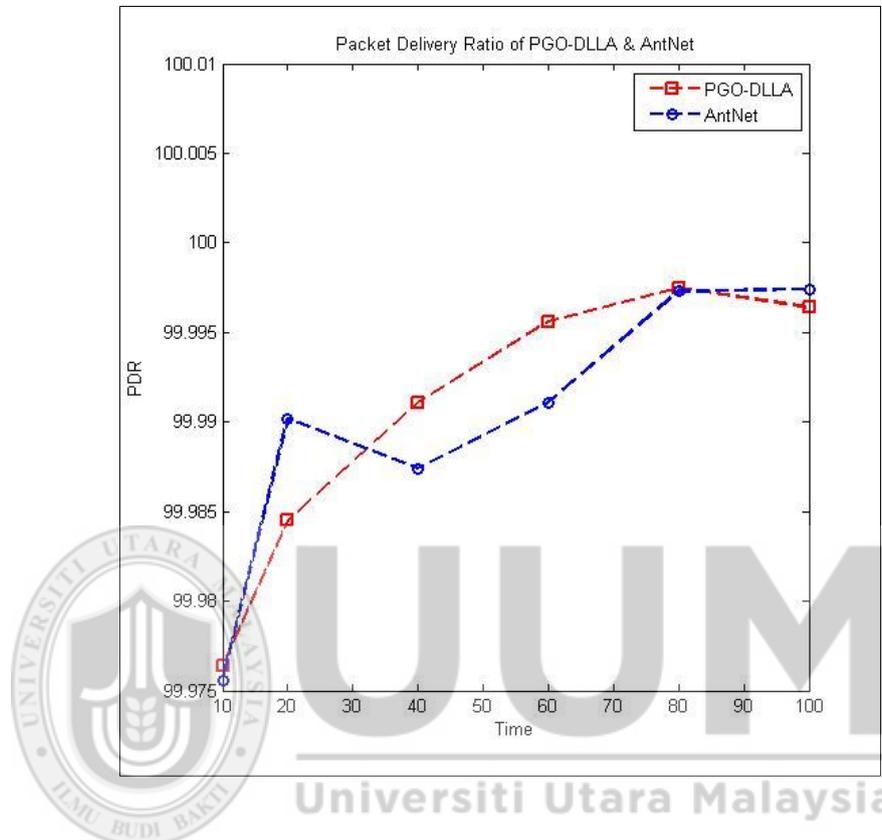


Figure 5.16. PDR results: PGO-DLLA vs. AntNet

In Figure 5.16, the value of End-to-End for the PGO-DLLA algorithm is slightly higher than the AntNet algorithm. One of the reasons is the calculation that it is needed to find the new route and the strategy of avoiding attacks. The throughput depends on the number of receiving packets from the source to the destination node. If any delay occurs as a result of a complex routing or updating route, it will decrease the throughput to the least value [141].

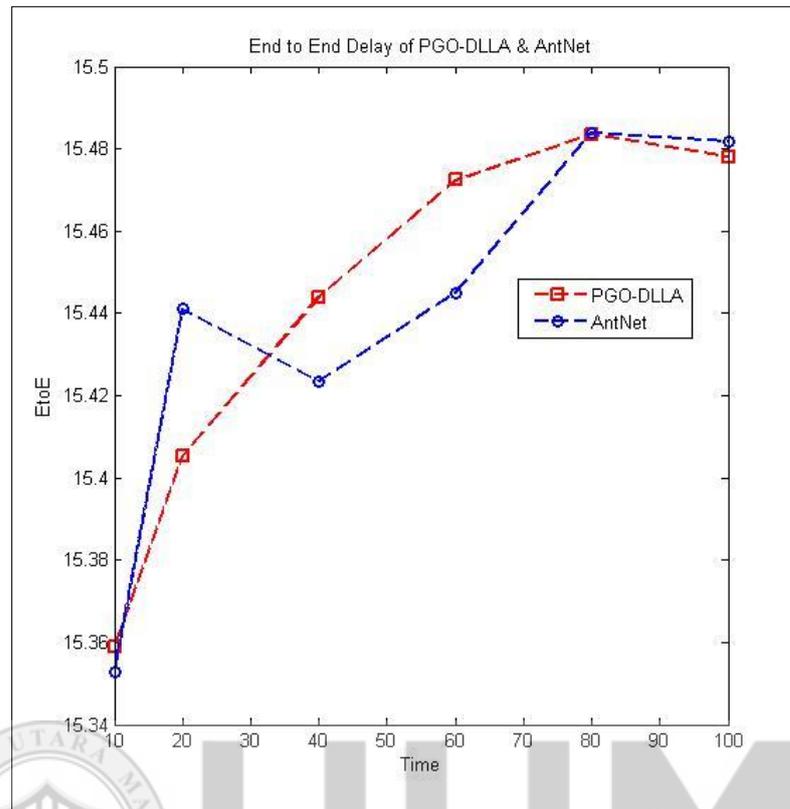


Figure 5.16. End-to- End results PGO-DLLA vs. AntNet

In Figure 5.17, the PGO-DLLA algorithm has a better throughput from the AntNet algorithm in the first four period times ( $t=10, 20, 30, 40$ ), then, the throughput value of the AntNet algorithm becomes better.

Nonetheless, the throughput values of PGO-DLLA and AntNet algorithms are increasing to higher values across time. In some cases, when the routing decisions are done in intermediate nodes such as the self-adaptive algorithms, they are updating the routing after each iteration. In these types of algorithms the routing discovery is not done by the source node and most of these algorithms are designed to work in dynamic environments.

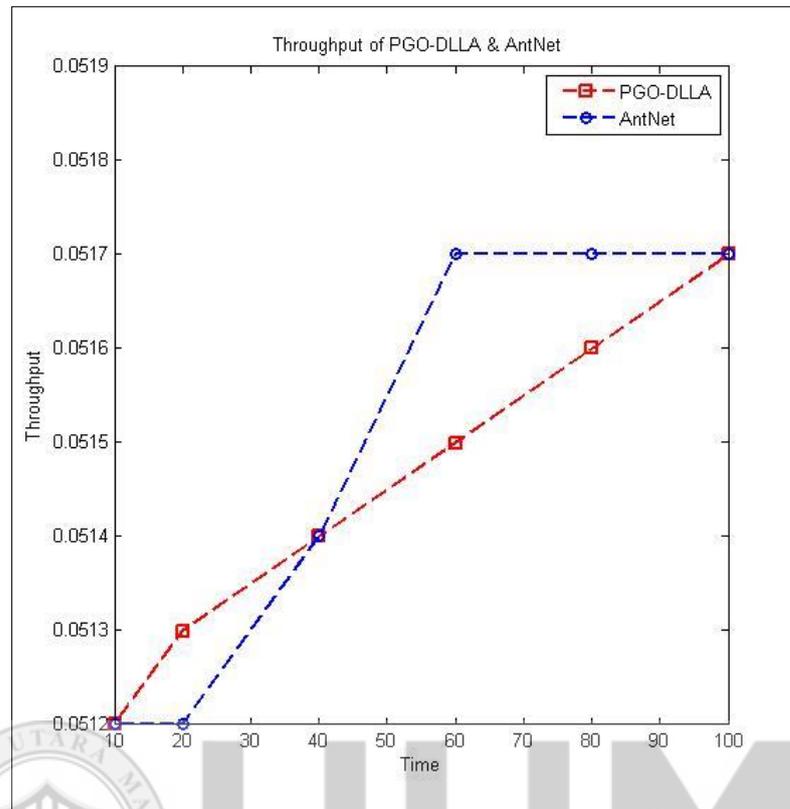


Figure 5.17. Throughput results PGO-DLLA vs. AntNet

As a result, the rate of data dropping will be increased and this will lead to the increase in packet loss rate. However, PGO-DLLA is more stable than AntNet in packet loss rate because of its special characteristics of routing to the destination node, as shown in Figure 5.18.

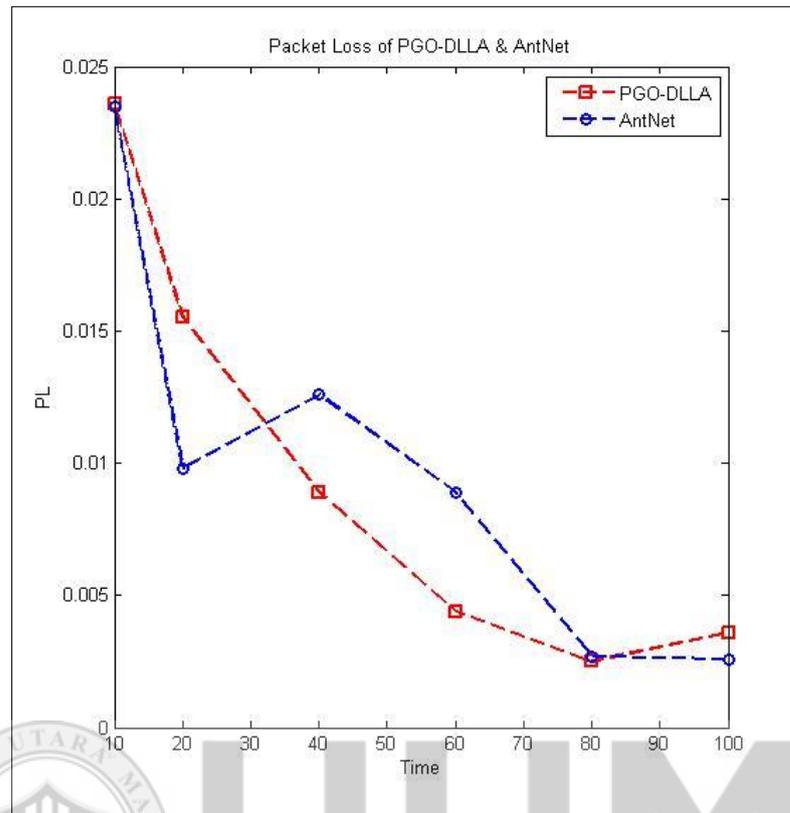


Figure 5.18. Packet Loss results PGO-DLLA vs. AntNet

### 5.6.2 Comparison of PGO-DLLA with AODV and BAODV

The rate of throughput is small, which happens as there is no pause time (continuous motion) [50]. In this situation, the rate of PDR may be not affected because the new algorithm has more than one strategy to ensure all packets are received by the destination nodes.

In Figure 5.19, it can be seen that there is some decrease in the PDR rates on BADOV and standard AODV, due to the effect of black hole node attacks as a malicious node. In contrary, the PDR rate of PGO-DLLA is increased due to the strategy of avoiding black hole attacks with keeping the shortest path to the destination node.

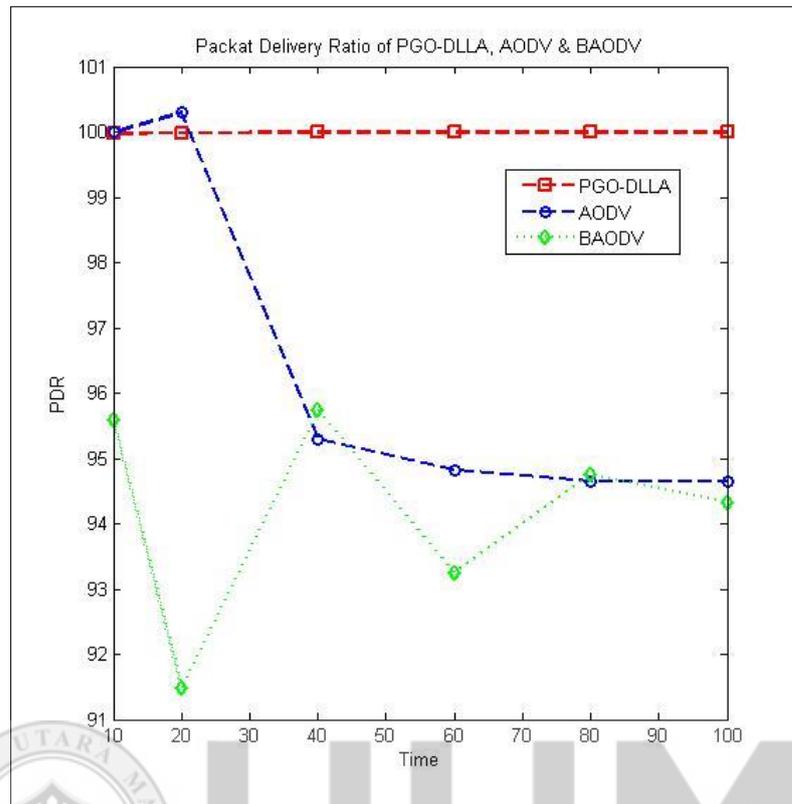


Figure 5.19. PDR results: PGO-DLLA vs. AODV vs. BAODV

Figure 5.20 presents a pictorial representation of Average End-to-End delay between PGO-DLLA, BADOV and standard AODV. In this figure, it can be seen that PGO-DLLA has a lesser rate, due to the strategy to change the route when it is broken as a result of misbehaving nodes. In contrary, the Average rate of EtoE on BADOV is increased because of the effects of the black hole attacks.

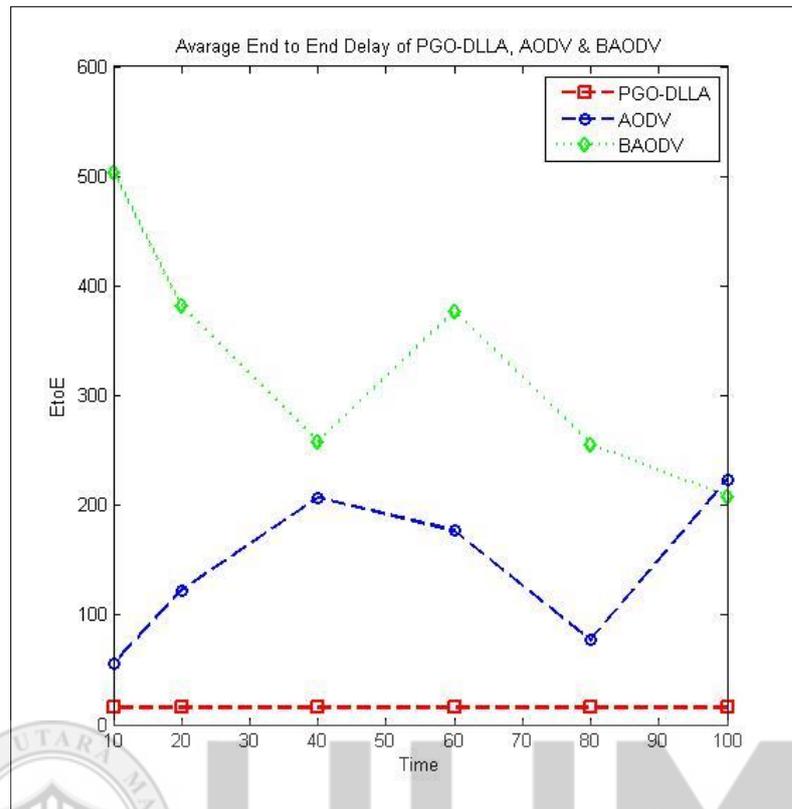


Figure 5.20. End-to-End results PGO-DLLA vs. AODV vs. BAODV

Throughput results among PGO-DLLA, BADOV and standard AODV are presented in a graphical representation in Figure 5.21. As can be seen in Figure 5.21, the PGO-DLLA algorithm has a higher rate of throughput due to the ability of avoiding black holes' dropping of packets and the mechanism of changing the route to the destination if any disconnection by the attackers is found.

In turn, the throughput on BADOV is very low; this is because BADOV does not include any strategy to avoid the black hole attacks.

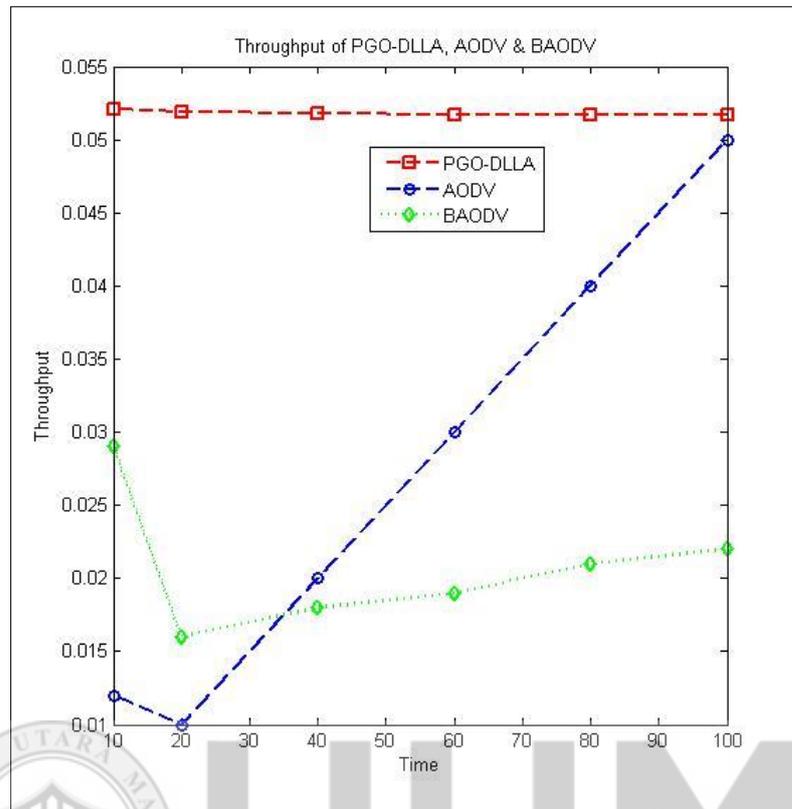


Figure 5.21. Throughput results PGO-DLLA vs. AODV vs. BAODV

Figure 5.22 shows the rate of packet loss results in a graphical representation among PGO-DLLA, BADOV and standard AODV. In this figure, it can be observed that the curve of packet loss result of BADOV is higher than PGO-DLLA and standard AODV, due to the mechanism of the BADOV routing protocol that does not contain a black hole avoiding technique.

Meanwhile, the rate of packet loss result in PGO-DLLA is very low (better) after 60 seconds and better than BADOV and standard AODV.

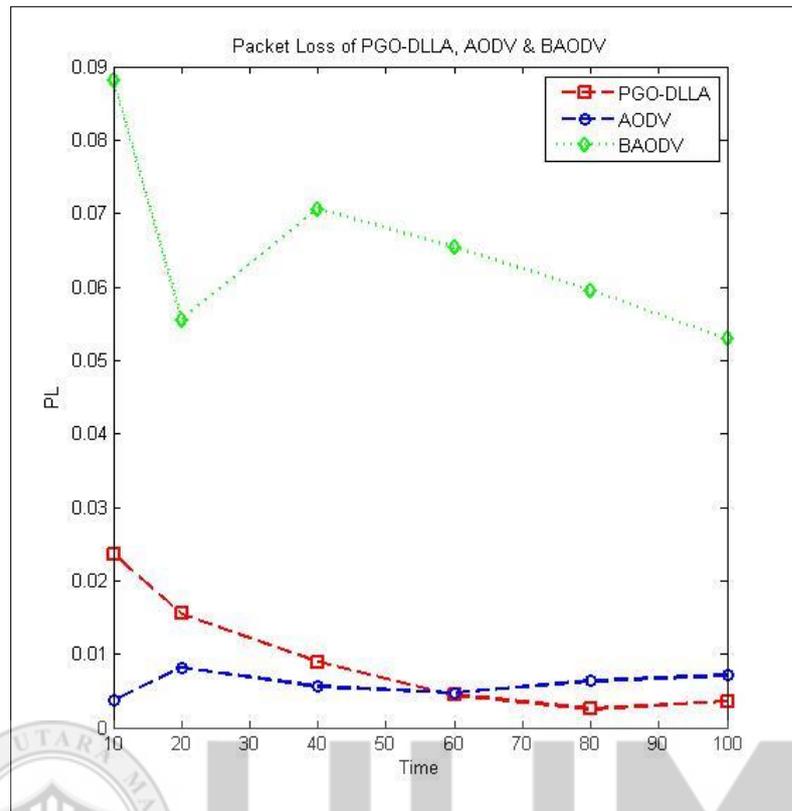


Figure 5.22. Packet Loss results PGO-DLLA vs. AODV vs. BAODV

In conclusion, the proposed PGO-DLLA protocol demonstrates some improvement in the performance metrics, namely PDR, EtoE, throughput and PL against a cooperative black hole attack in a MANET that relies on the AODV routing protocol.

The proposed PGO-DLLA modifies the standard AODV and optimizes the routing process by a technique of finding a prey and avoiding attacks that is inspired from the spider named daddy long-legs. The inspired technique of PGO-DLLA enhances the routing mechanism of AODV. Some changes in the routing tables is conducted to help in avoiding black hole attacks and avoiding the delay in routing.

The experimental results show that PGO-DLLA has improved the securing of AODV in preventing black hole attacks which leads to better performance metrics against standard AODV and BAODV.

### **5.7 Performance Comparison of Proposed Protocols with Current Work**

This section presents the comparison of EAODV, SSP-AODV and PGO-DLLA protocols with the current prevention protocols against black hole attacks, which are based on the AODV protocol. The next sections discuss the theoretical framework and compare the graphical results [121], [122] of the proposed protocol with the current work.

#### **5.7.1 True-link Cross-checking Enhanced AODV Protocol (EDRIAODV)**

This section presents the first comparison between the proposed protocols ( EAODV, SSP-AODV and PGO-DLLA) and true-link cross-checking enhanced AODV protocol (EDRIAODV) [83]. DRIAODV and EDRIAODV protocols have been proposed to enhance the performance of the AODV protocol in terms of the prevention of the cooperative black hole attack. However, the results of the proposed protocols (PGO-DLLA, SSP-AODV and EAODV) are compared to DRIAODV and EDRIAODV protocols in terms of graphical frameworks. The main goal of DRIAODV and EDRIAODV protocols is to prevent the cooperative black hole attacks in order to reduce the End-to-End delay and routing overhead via modification in DRI-based cross-checking with true-link rendezvous phase by changing the existing AODV protocol scheme. They have proposed a new methodology by introducing Data

Routing Information (DRI) Table and cross-checking with true-link concept in preventing the cooperative black hole attacks with the modified AODV protocol.

Figure 5.23 shows the system architecture of the DRIAODV and EDRIAODV protocols.

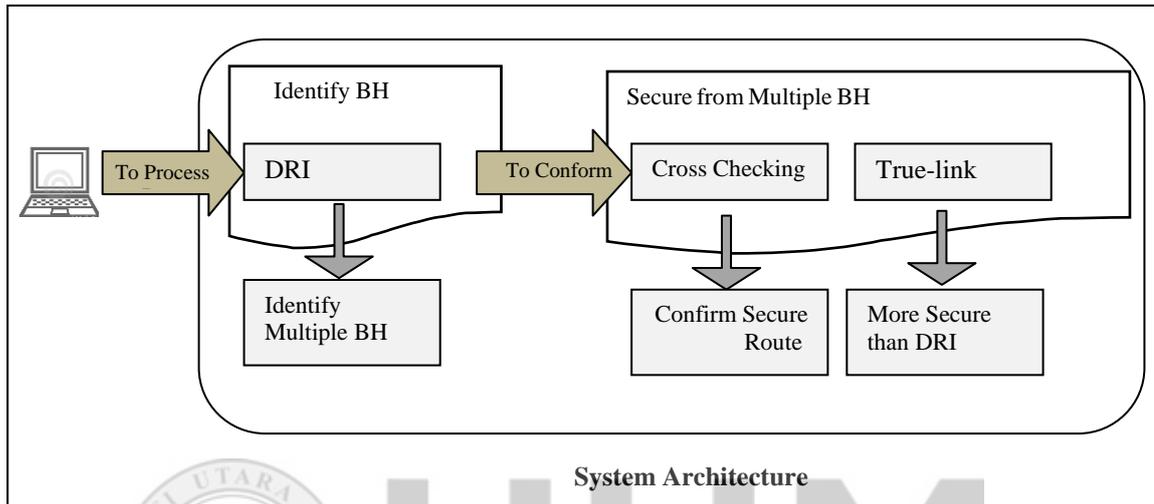


Figure 5.23. The system architecture of the DRIAODV and EDRIAODV [83].

The increase of routing overhead via the process of cross-checking the intermediate nodes is the one of the drawbacks of this methodology.

### 5.7.1.1 Comparison of Proposed Protocols with EDRIAODV Protocol

In this section, the evaluation results and performance analysis are graphically compared with the same input parameter values and metrics such as packet delivery ratio, end-to-end delay, throughput and normalized routing load. The network simulator NS-2.33 [136], [142] is used to conduct the experiments of three scenarios. The first scenario is to test the original EAODV, the second scenario is to test SSP-AODV, and the third scenario is to test PGO-DLLA. These scenarios are developed to carry out the tests using the mobility of the nodes in the network. The simulation

parameters for Scenario 1,2 and 3 are identical to the simulation parameters in [83] as shown in Table 5.7.

Table 5.7

*The PGO-DLLA, SSP-AODV, and EAODV Scenarios identical to the Scenarios in the Simulation of DRIAODV and EDRIAODV Protocols [83].*

<b>Parameter</b>	<b>Scenario 1</b>	<b>Scenario 2</b>	<b>Scenario 3</b>
<b>Simulation Time</b>	200 sec.	200 sec.	200 sec.
<b>Number of Nodes</b>	50	50	50
<b>Routing Protocol</b>	EAODV	SSP-AODV	PGO-DLLA
<b>Traffic Model</b>	CBR(UDP)	CBR(UDP)	CBR(UDP)
<b>Pause Time</b>	1 sec.	1 sec.	1 sec.
<b>Mobility Model</b>	Random way point	Random way point	Random way point
<b>No. of sources</b>	3	3	3
<b>Map area</b>	1000m x 1000m	1000m x 1000m	1000m x 1000m
<b>MAC Type</b>	802.11	802.11	802.11
<b>Malicious Node</b>	Black hole	Black hole	Black hole

The evaluation of the three routing protocols, EAODV, SSP-AODV and PGO-DLLA, includes two steps: firstly, the comparison between EAODV, SSP-AODV and PGO-DLLA, and secondly, the comparison between EAODV, SSP-AODV and PGO-DLLA with DRIAODV, EDRIAODV and AODV is using four metrics: End-to-End, Packet Delivery Ratio, Throughput and Normalized Routing Load. Figure 5.24 illustrates the throughput result of six protocols EAODV, SSP-AODV, PGO-DLLA, DRIAODV and EDRIAODV [83]. It can be seen that PGO-DLLA increases the throughput when it increases the mobility. When EAODV and SSP-AODV are in a

lower mobility (0-10), they show better throughput than SSP-AODV, but in higher mobility (10-30) they show worst throughput as compared to SSP-AODV. In Figure 5.24 can be noticed that PGO-DLLA is better than the original AODV, but approximately has the same result with DRIAODV and EDRIAODV.

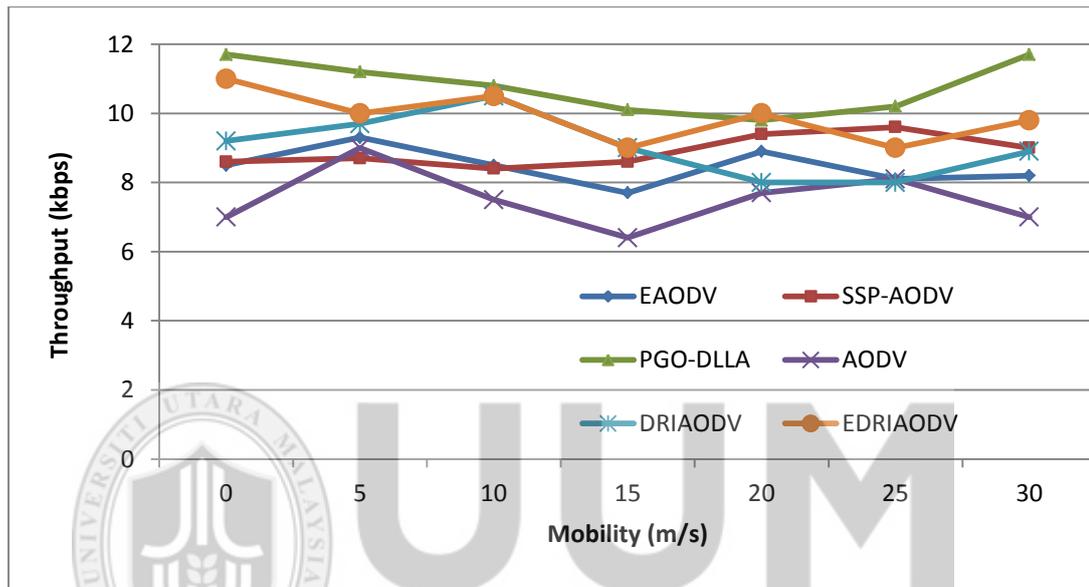


Figure 5.24. The throughput in various mobility models of proposed protocols EAODV, SSP-AODV and PGO-DLLA that are compared with DRIAODV, EDRIAODV and standard AODV protocol

Figure 5.25 shows the Packet Delivery Ratio result of six protocols, the comparison of PGO-DLLA, SSP-AODV and EAODV protocols, in which it can be seen that PGO-DLLA increases the PDR value when it increases the mobility. However, EAODV in a lower mobility (0-15) shows the worst PDR than SSP-AODV, but in a higher mobility (20-30), it shows the best PDR matches to the value of SSP-AODV. Figure 5.25 presents the comparison between three protocols, namely DRIAODV, EDRIAODV, and AODV. It can be noticed that PGO-DLLA, SSP-AODV and

EAODV are better than the original AODV and have a better result from DRIAODV and EDRIAODV.

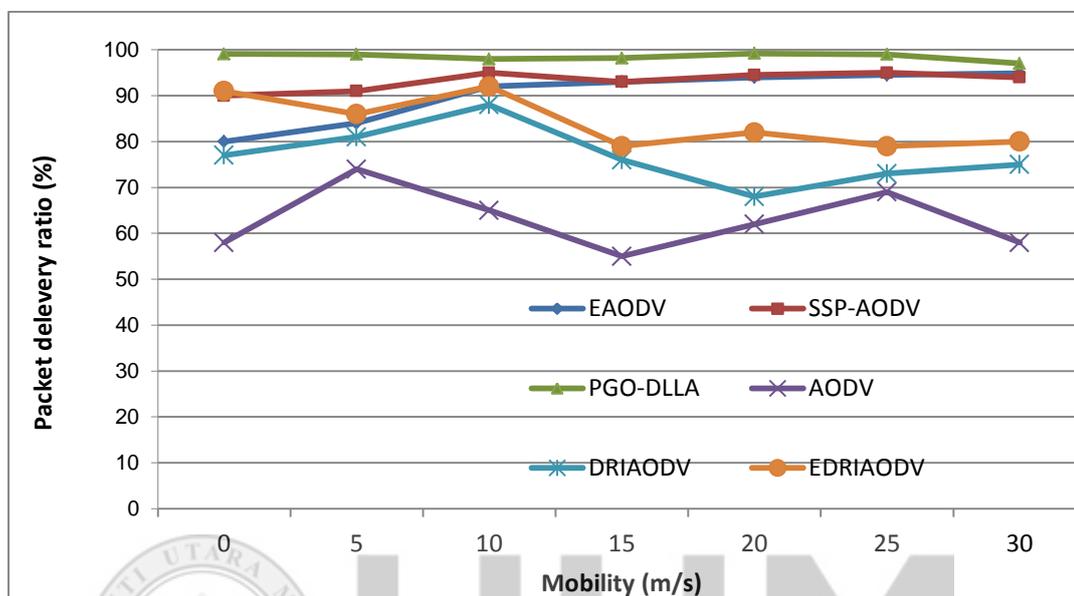


Figure 5.25. The packet delivery ratio in various mobility models of proposed protocols EAODV, SSP-AODV and PGO-DLLA compared with DRIAODV, EDRIAODV and standard AODV protocol

Figure 5.26 illustrates the Normalized Routing Load result of the protocols PGO-DLLA, EAODV, and SSP-AODV. It can be seen that the two protocols, EAODV and SSP-AODV, increase the NRL value when the mobility increases. When PGO-DLLA is in a lower mobility, shows a lower NRL than EAODV and SSP-AODV. EAODV is better from SSP-AODV in the period between (0-5), but in a higher mobility (20-25), it shows the highest NRL as compared to SSP-AODV. Figure 5.26 demonstrate the comparison between three protocols, namely DRIAODV, EDRIAODV, and AODV.

it can be noticed that PGO-DLLA is better than the three protocols, AODV, DRIAODV and EDRIAODV; and EAODV and SSP-AODV are better as well, especially in high mobility (25-30).

Clearly this is because of the security that is used here, delaying respond of RREQ control messages in SSP-AODV, it can be stated that the NRL has some increase in the performance of the two protocols.

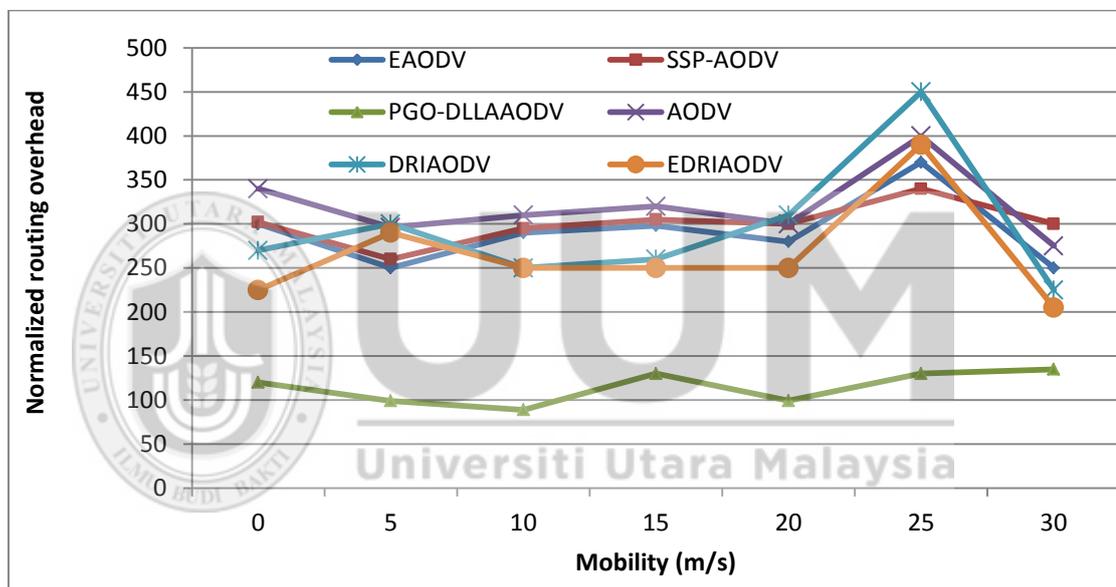


Figure 5.26. The normalized routing overhead in various mobility models of proposed protocols EAODV, SSP-AODV and PGO-DLLA compared with DRIAODV, EDRIAODV and the standard AODV protocol

Figure 5.27 illustrates the End-to-End delay result of five protocols, it can be seen that the three protocols, namely PGO-DLLA, EAODV and SPP-AODV, keep a lower delay time in spite of the increase of the mobility (0-30) in EAODV and SSP-AODV.

This is because of the heuristic search that is used as a new technique in these two protocols. Figure 5.27 displays the comparison between three protocols, DRIAODV, EDRIAODV, and AODV. It can be noticed that PGO-DLLA is still a better protocol and the two protocols EAODV and SSP-AODV (the delay is limited between (0-400)) are better than the three protocols AODV, DRIAODV and EDRIAODV (the delay is not limited (100-1800)).

As a result, Figures 5.25, 5.26, and 5.27 shows the comparative results of Throughput, PDR, NRL, and End-to-End delay. In terms of normalized routing load, PDR, and the average End-to-End delay. It can be stated that there has been some improvement in PGO-DLLA, EAODV and SSP-AODV protocols in low mobility and the average delay in most changes of mobility as shown in Figure 5.27. The results of the comparison between the two protocols, DRIAODV and EDRIAODV, with the standard AODV protocol in term of average delay. As a result, it triggers a black hole attack in MANET. The three protocols, PGO-DLLA, EAODV and SSP-AODV, has improved the network performance compared with the original AODV and the DRIAODV and EDRIAODV protocols.

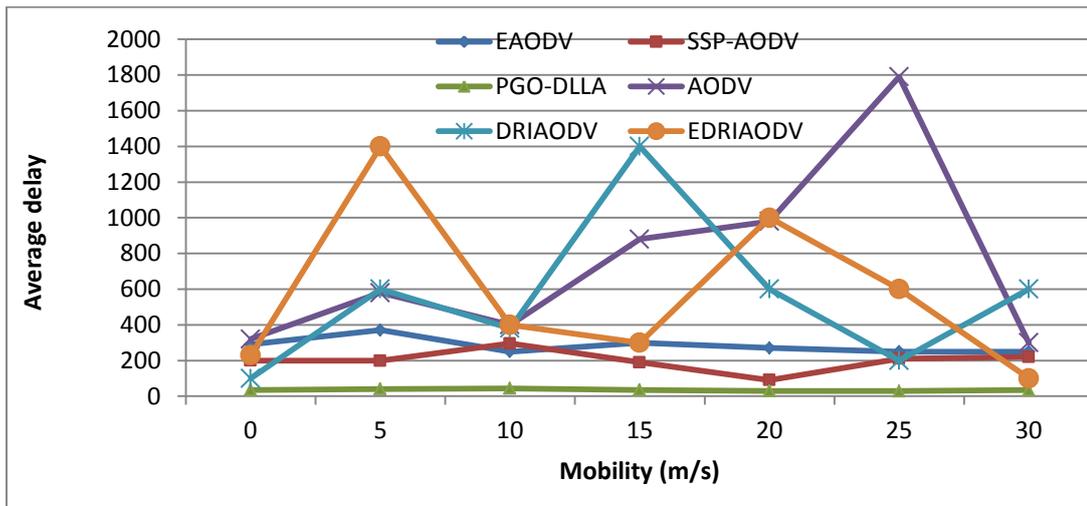


Figure 5.27. The average delay in various mobility models of proposed protocols EAODV, SSP-AODV and PGO-DLLA Compared with DRIAODV, EDRIAODV and standard AODV protocol

### 5.7.2 Secure Ad hoc On-demand Distance Vector Protocol (SAODV)

This section presents the second comparison between the proposed protocols (EAODV, SSP-AODV and PGO-DLLA) and one of the black hole attack prevention methodology that is proposed in enhancing the AODV routing protocol named Secure Ad-hoc On-demand Distance Vector (SAODV), and Bad Ad-hoc On-demand Distance Vector (BAODV) [84].

In this method, a proposed secure ad hoc on-demand distance vector (SAODV) [84], uses multiple paths to find the destination node. The source node stores several information for analysis using some new fields in the routing table.

The proposed method has two phases, suspicion phase and confirmation phase. In the first phase, the authors stand on two facts; unusually higher DSN, and fast respond by RREP without delay. Depending on the average values of all response times to the

RREP by intermediate nodes, the authors have compared the anomaly of the delay time to respond in RREP and makes the suspect node list.

In the second phase, the authors created the new control messages named; Modify Route Request (MREQ), and Modify Route Reply (MREP). Using the different random numbers that are inserted in MERQ, the new method checks if any black hole node is found inside the network. In addition, the method also suffers the delay when the source node is waiting to make a decision on selecting the proper path to the destination node.

#### **5.7.2.1 Comparison of Proposed Protocols with SAODV Protocol**

In this section, the evaluation results and performance analysis are graphically compared with the same input parameter values and metrics such as packet delivery ratio, and throughput. The network simulator NS-2.33 [136], [142] is used to conduct the experiments of three scenarios. The first scenario is to test the original EAODV, the second scenario is to test SSP-AODV, and the third scenario is to test PGO-DLLA. These scenarios are developed to carry out the tests using various numbers of nodes and various speeds of the nodes in the network. The simulation parameters for Scenarios 1, 2 and 3 are identical to the simulation parameters in [84] as shown in Table 5.8.

Table 5.8

*The PGO-DLLA, SSP-AODV, and EAODV Scenarios Identical to Scenario in the Simulation of SAODV Protocol [84].*

<b>Parameter</b>	<b>Scenario 1</b>	<b>Scenario 2</b>	<b>Scenario 3</b>
<b>Simulation Time</b>	50 sec.	50 sec.	50 sec.
<b>Number of Nodes</b>	20	20	20
<b>Routing Protocol</b>	EAODV	SSP-AODV	PGO-DLLA
<b>Traffic Model</b>	CBR(UDP)	CBR(UDP)	CBR(UDP)
<b>Pause Time</b>	1 sec.	1 sec.	1 sec.
<b>Mobility Model</b>	Random way point	Random way point	Random way point
<b>No. of sources</b>	1	1	1
<b>Map area</b>	750m x 750m	750m x 750m	750m x 750m
<b>MAC Type</b>	802.11	802.11	802.11
<b>Malicious Node</b>	Black hole	Black hole	Black hole

The evaluation of three routing protocols, namely EAODV, SSP-AODV and PGO-DLLA, includes two steps: firstly, the comparison between EAODV, SSP-AODV and PGO-DLLA, and secondly, the comparison between EAODV, SSP-AODV and PGO-DLLA with SAODV, BAODV and AODV is using two metrics, Packet Delivery Ratio and Throughput.

Figure 5.28 illustrates the PDR result of three proposed and the comparison between SAODV and BAODV. It can be seen that PGO-DLLA increases the PDR when the number of nodes increases. When EAODV and SSP-AODV are in a lower number of nodes (0-30), they show better PDR, while in a higher number of nodes (40-50), they shows the worst PDR compared to PGO-DLLA. It can be noticed that PGO-DLLA is better than the original AODV, but approximately has the same result with SAODV.

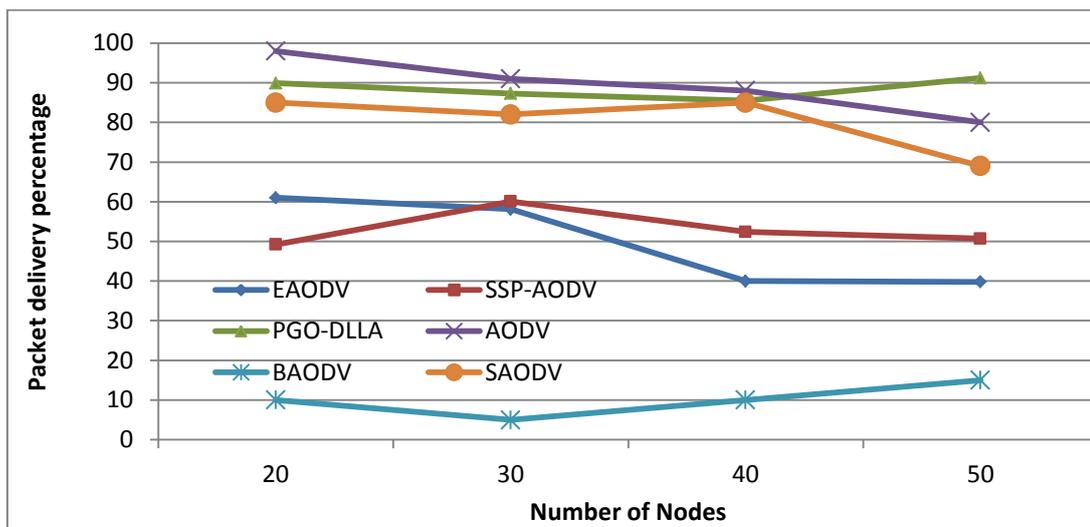


Figure 5.28. The packet delivery ratio in various numbers of nodes in proposed protocols EAODV, SSP-AODV and PGO-DLLA to compare with SAODV, BAODV and standard AODV protocol

Figure 5.29 shows the throughput of six protocols, where in Figure 5.29, it shows the comparison of PGO-DLLA, SSP-AODV and EAODV protocols. It can be seen that the three protocols, PGO-DLLA, SSP-AODV and EAODV, are increase the TH when the number of nodes increases.

However, EAODV and SSP-AODV in a lower TH (40-60) kbps between (25-30) numbers of nodes is the worst TH than PGO-DLLA's (80-120) kbps. The higher TH in the three protocols is registered at (40-50) number of nodes. It can be noticed that the TH in PGO-DLLA is between (80-160) kbps, SSP-AODV (60-80) kbps and EAODV (40-65) kbps, which are better than BAODV's (20-40) kbps, very close to SAODV (100-160) kbps, except the original AODV, which is between (110-190) kbps.

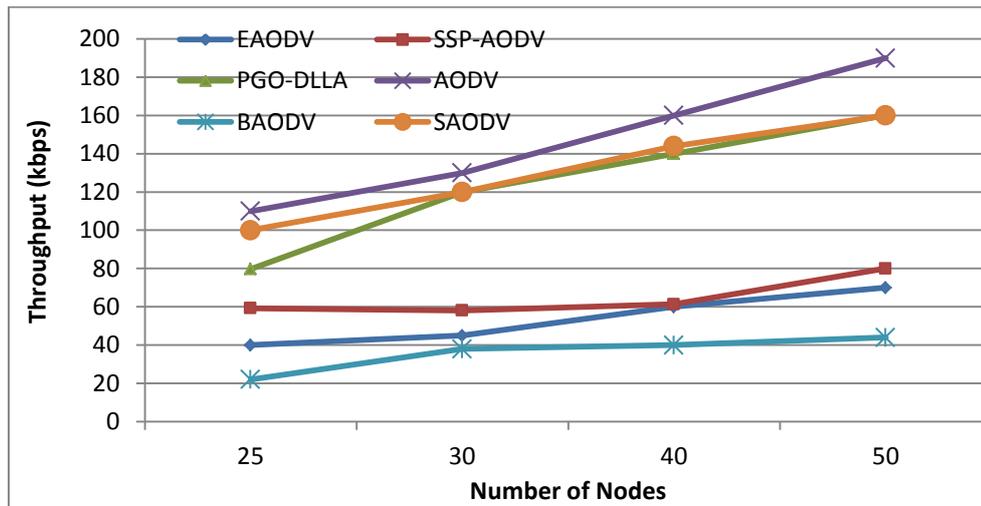


Figure 5.29. Throughput in various numbers of nodes of proposed protocols EAODV, SSP-AODV and PGO-DLLA compared with SAODV, BAODV and standard AODV protocol

Figure 5.30 shows the packet delivery ratio of six protocols with various node speeds. It can be seen that the three protocols, namely PGO-DLLA, SSP-AODV and EAODV, increase the PDF when the speed of nodes increases because of the use of the heuristic search algorithm.

However, EAODV and SSP-AODV in lower PDF (80%-90%) between (0-5) m/s is the worst PDF than PGO-DLLA (95%-99%). The higher PDF in the three protocol is registered at (10-30) m/s.

It can be seen that the PDF in PGO-DLLA is between (95%-99%), SSP-AODV (90%-95%) and EAODV (80%-95%), which are better than BAODV (10%-30%), very close to SAODV (79%-90%) and the standard AODV which is between (85%-96%).

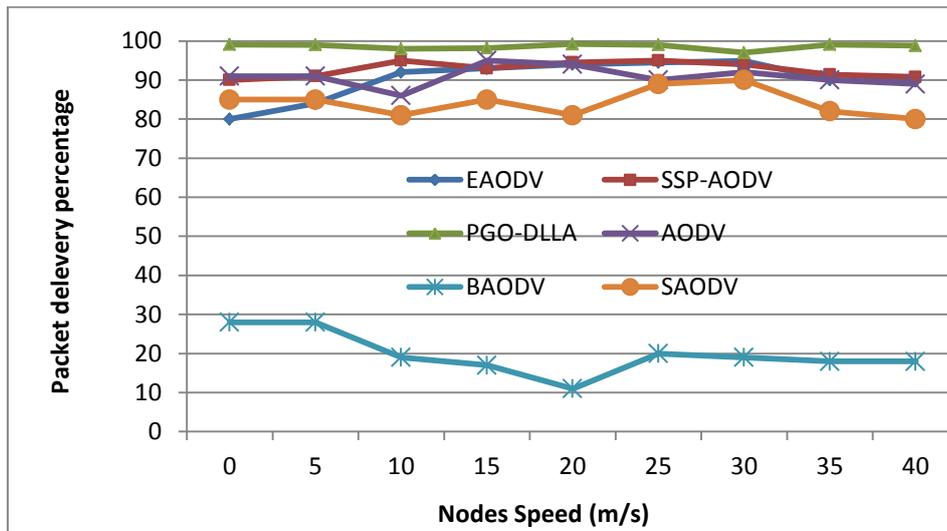


Figure 5.30. The packet delivery percentage in various nodes speed of proposed protocols EAODV, SSP-AODV and PGO-DLLA compared with SAODV, BAODV and standard AODV protocol

Figure 5.31 shows the TH of six protocols with various node speeds. It can be seen that not all the three protocols increase the TH when the speed of nodes increases (only the PGO-DLLA increases the TH) because of the use of a meta-heuristic search algorithm.

However, EAODV and SSP-AODV in a lower TH (30-50) kbps between (35-40) m/s is the worst TH than PGO-DLLA's (60-85) kbps. The higher TH in the two protocols (EAODV and SSP-AODV) is registered at (35-40) m/s for EAODV and at (5-10) m/s for SSP-AODV. Figure 5.31 shows the comparison between the three protocols (SAODV, BAODV, and AODV). It can be seen that the TH in PGO-DLLA, SSP-AODV and EAODV are better than BAODV's (15-25) kbps, and only PGO-DLLA is very close to SAODV's (60-80) kbps, and the standard AODV's TH is between (75-90) kbps.

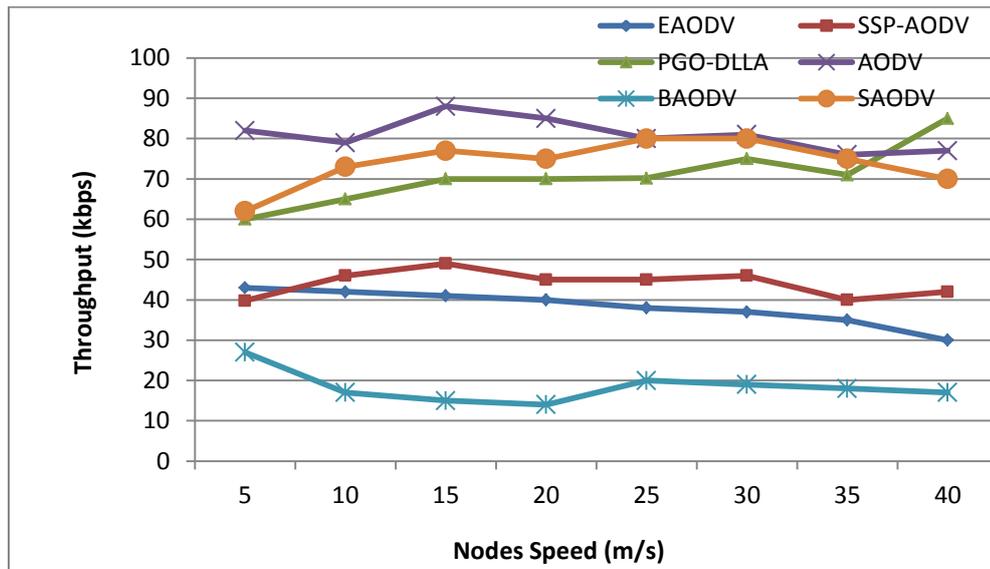


Figure 5.31. The Throughput in various nodes speed of proposed protocols EAODV, SSP-AODV and PGO-DLLA compared with SAODV, BAODV and standard AODV protocol

### 5.7.3 Impact of Black Hole and Gray Hole Attacks in AODV Protocol (IAODV)

In this work, a real time anomaly detection based algorithm is implemented. The authors proposed an improvement in the AODV routing protocol in preventing black and gray hole attacks.

It is a type of malicious node attack in MANET, it should be honest in at the first time, and then it changes behavior as a malicious node such as black hole attack. The authors concluded that in order to stop the black and the gray hole attacks, the prevention techniques should be implemented to identify the malicious node at route identification phase only. The adaptive protocol (IAODV) is proposed to implement the black and the gray hole attacks in the AODV protocol, and it is designed to prevent the malicious node attack.

This technique depends on the well-known fact that the malicious node tries to take the advantage by sending the first RREP with a higher destination sequence number and a minimum hop count. In reality, the IAODV protocol and AODV are the same protocol with some changes in the original mechanism of route discovery by not responding to the first RREP that comes to the source node in the IAODV protocol [79].

### **5.7.3.1 Comparison of Proposed Protocols with IAODV Protocol**

In this section, the evaluation results and performance analysis are graphically compared with the same input parameter values and metrics such as packet drop ratio, throughput and normalized routing load.

The first scenario is to test the original EAODV, the second scenario is to test SSP-AODV, and the third scenario is to test PGO-DLLA. These scenarios are developed to carry out the tests using the mobility of the nodes in the network. The simulation parameters for Scenario 1,2 and 3 are identical to the simulation parameters in [79] as shown in Table 5.9.

Table 5.9

*The PGO-DLLA, SSP-AODV, and EAODV Scenarios identical to the Scenarios in the Simulation of IAODV Protocol [79].*

<b>Parameter</b>	<b>Scenario 1</b>	<b>Scenario 2</b>	<b>Scenario 3</b>
<b>Simulation Time</b>	500 sec.	500 sec.	500 sec.
<b>Number of Nodes</b>	15,30,45,60,75	15,30,45,60,75	15,30,45,60,75
<b>Routing Protocol</b>	EAODV	SSP-AODV	PGO-DLLA
<b>Traffic Model</b>	CBR(UDP)	CBR(UDP)	CBR(UDP)
<b>Pause Time</b>	0, 30, 60, 90, 120, 150 sec.	0, 30, 60, 90, 120, 150 sec.	0, 30, 60, 90, 120, 150 sec.
<b>Mobility Model</b>	Random way point	Random way point	Random way point
<b>Max. Speed</b>	5, 10, 20, 30, 40 m/s	5, 10, 20, 30, 40 m/s	5, 10, 20, 30, 40 m/s
<b>Map area</b>	750m x 750m	750m x 750m	750m x 750m
<b>MAC Type</b>	802.11	802.11	802.11
<b>Malicious Node</b>	Black hole	Black hole	Black hole

The evaluation of the three routing protocols, EAODV, SSP-AODV and PGO-DLLA, includes two steps: firstly, the comparison between EAODV, SSP-AODV and PGO-DLLA, secondly, the comparison between EAODV, SSP-AODV and PGO-DLLA with IAODV and AODV using four metrics: Packet Drop Ratio, Throughput and Normalized routing load.

Figure 5.32 illustrates the PDR result of the three proposed protocols, and shows the comparison between IAODV and BAODV. It can be seen that PGO-DLLA increases the PDR when the speed increases except when the speed is (5-10) and (30-40). PGO-DLLA, EAODV and SSP-AODV have better PDR values than the PDR of BAODV, whereas only PGO-DLLA is better than IAODV. The PDR values of EAODV and SSP-AODV are compared with IAODV, which have approximately the same result.

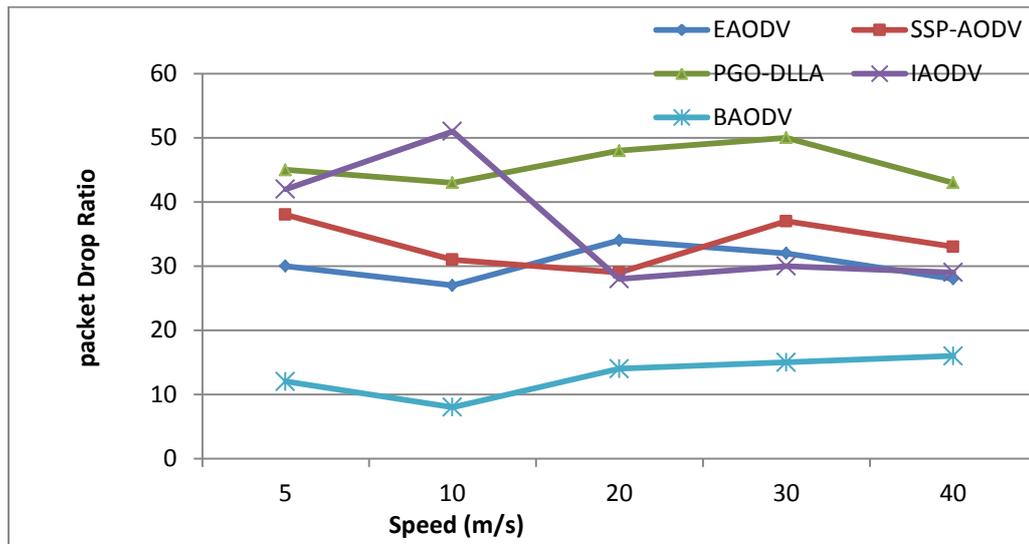


Figure 5.32. The packet delivery ratio in various nodes speed of proposed protocols EAODV, SSP-AODV and PGO-DLLA compared with IAODV and BAODV protocol

Figure 5.33 shows the NRL of six protocols with various node speed, and shows the comparison of PGO-DLLA, SSP-AODV and EAODV protocols. It can be seen that not all the three protocols increases the NRL when the speed of nodes increases. However, EAODV and SSP-AODV in lower NRL between 2.4, 2.6 at 10 m/s is the worst NRL than PGO-DLLA 0.9 at 20 m/s. The higher NRL 3.1 in the two protocols EAODV and SSP-AODV is registered at 5 m/s because the speed is very slow, as for PGO-DLLA, the higher NRL 1.4 is found at the speed 30 m/s because the ability of dynamic adaptive to recalculate a new path. Figure 5.33 shows the comparison between two protocols IAODV and BAODV, and the result shows that the IAODV protocol outperforms the BAODV protocol when it registers the very low NRL as compared to the black hole AODV. However, it can be seen that there is some improvement in all of the proposed protocols with the BAODV protocol. They have

better NRL than BAODV, and the NRL in SSP-AODV and EAODV are close to the IAODV protocol even in the high speed because of the heuristic algorithm. Figure 5.33 shows the graphical comparison between the proposed protocols with various speed and the impact of black hole and gray hole attacks in the AODV protocol IAODV in terms of NRL.

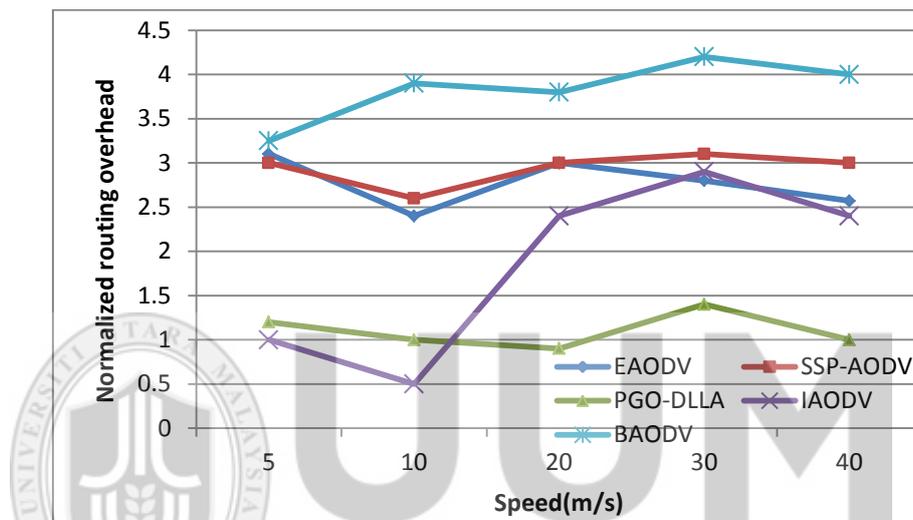


Figure 5.33. Normalized routing load in various speed of proposed protocols EAODV, SSP-AODV and PGO-DLLA compared with IAODV and BAODV protocol

Figure 5.34 shows the TH of five protocols with various node speed. In Figure 5.34 shows the comparison of PGO-DLLA, SSP-AODV and EAODV protocols, it can be seen that not all the three protocols increase the TH when the speed of nodes increases. However, EAODV and SSP-AODV in lower TH in high speed between (30-40) m/s, while PGO-DLLA in higher TH are between the same time of speed improves the ability to be adaptive with high speed nodes.

The higher TH 45 kbps of the two protocols EAODV and SSP-AODV is registered between 10-30 m/s. Figure 5.34 shows the comparison between the two protocols IAODV and BAODV, and the result shows that the IAODV protocol outperforms the BAODV protocol when it registers the very high TH as compared to the black hole AODV. However, When the proposed protocols PGO-DLLA, SSP-AODV and EAODV with the same metric (TH), it can be seen that there some improvement in all of the proposed protocols with the BAODV protocol. They have better TH than BAODV at the beginning of the simulation and less down at the final simulation. PGO-DLLA performs in TH at time 40 of speed because the parallel swarm optimization algorithm avoids the malicious nodes, except in the beginning of the simulation IAODV protocol is better.

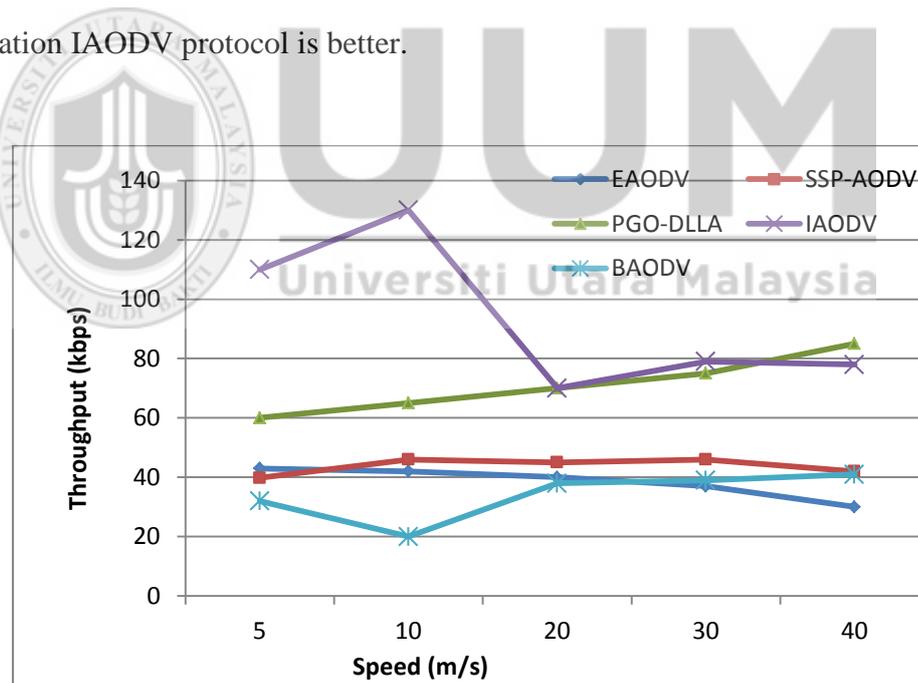


Figure 5.34. The throughput in various speed of proposed protocols EAODV, SSP-AODV and PGO-DLLA compared with IAODV and BAODV protocol

Figure 5.34 shows the graphical comparison between the proposed protocols with various speed and the impact of black hole and gray hole attacks in the AODV protocol IAODV in terms of TH. The next comparison with the IAODV protocol in the same performance metrics with various pause timed and number of nodes are shown in Table 5.10.

Table 5.10

*Comparison Result of EAODV, SSP-AODV, and PGO-DLLA Protocols with IAODV and Standard AODV Protocol* ■ Best ■ Average ■ Bad.

	<i>Speed</i>					<i>No. of Nodes</i>					<i>Pause Time</i>				
	<i>EA</i>	<i>SSP</i>	<i>PGO</i>	<i>IA</i>	<i>BA</i>	<i>EA</i>	<i>SSP</i>	<i>PGO</i>	<i>IA</i>	<i>BA</i>	<i>EA</i>	<i>SSP</i>	<i>PGO</i>	<i>IA</i>	<i>BA</i>
<i>PDR</i>			Best	Average	Bad			Best	Average	Bad			Best		
<i>NRL</i>			Best	Average	Bad			Best	Average	Bad			Best		
<i>TH</i>			Best	Average	Bad			Best	Average	Bad			Best		

Table 5.10 displays the comparison of the proposed protocols with the current work of the IAODV protocol, where the pink color shows the best protocol performance, the green color shows the average protocol performance or close to the best protocol performance, and the gray color is the bad result of protocol performance. In the experiments of various number of nodes in Figures 5.35, 5.36, and 5.37 respectively, Figures 5.35 is in terms of PDR, and Figures 5.37 is in terms of TH, which shows that PGO-DLLA is the best protocol while Figures 5.36 shows that PGO-DLLA is very close to the best protocol performance of the IAODV protocol's performance.

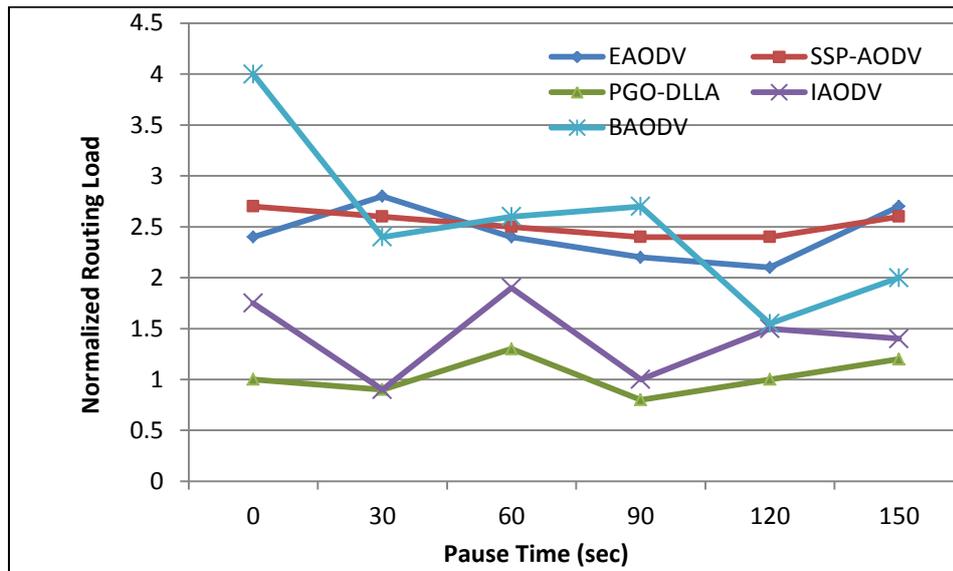


Figure 5.35. Normalized routing load with various pause times of EAODV, SSP-AODV and PGO-DLLA Compared with IAODV and BAODV Protocol

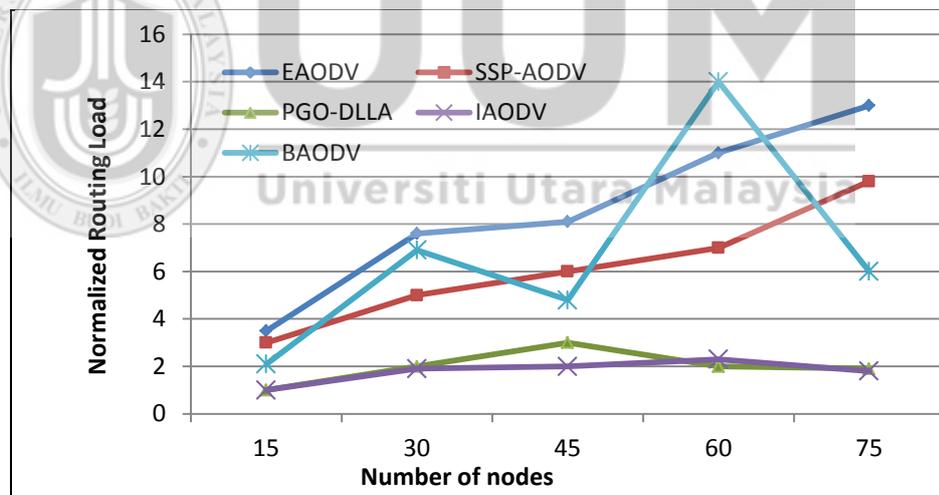


Figure 5.36. Normalized routing load with various number of nodes of EAODV, SSP-AODV and PGO-DLLA compared with IAODV and BAODV protocol

In the experiments of various numbers of pause times, Figures 5.35, 5.36, and 5.37 are in terms of PDR, NRL and TH, which show that PGO-DLLA is the best protocol, the IAODV protocol is very close to the best protocol performance, and the BAODV

protocol is the bad protocol performance because of the existence of black hole nodes without the security algorithm.

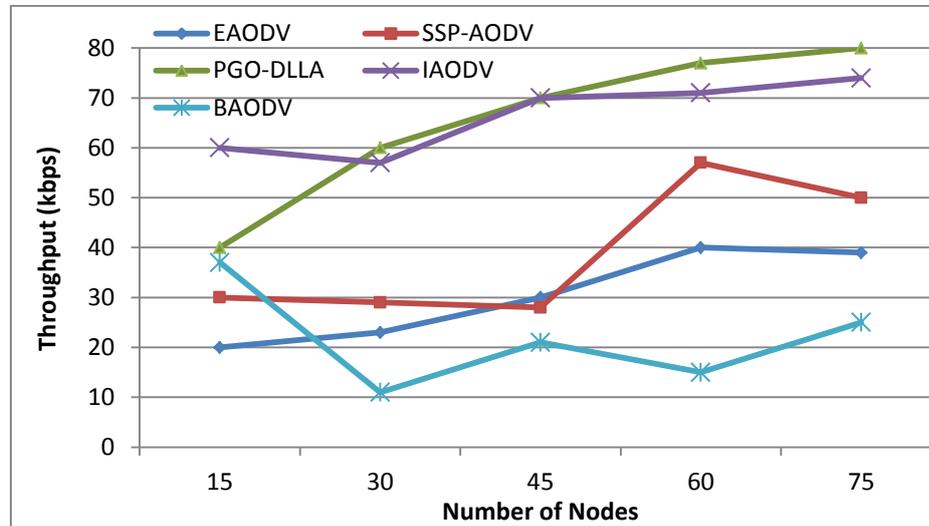


Figure 5.37. The throughput in various numbers of nodes of EAODV, SSP-AODV and PGO-DLLA compared with IAODV and BAODV protocol

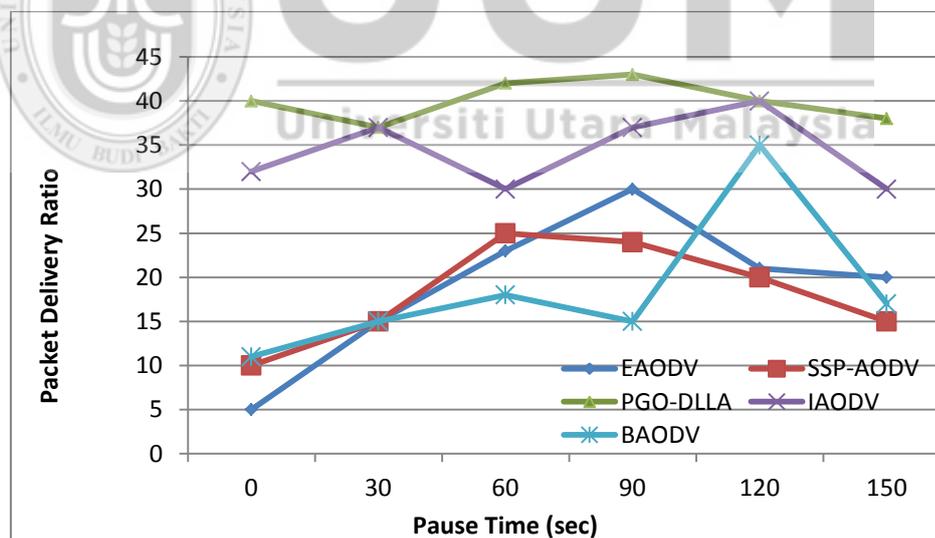


Figure 5.38. Packet delivery ratio in various pause time of EAODV, SSP-AODV and PGO-DLLA compared with IAODV and BAODV protocol

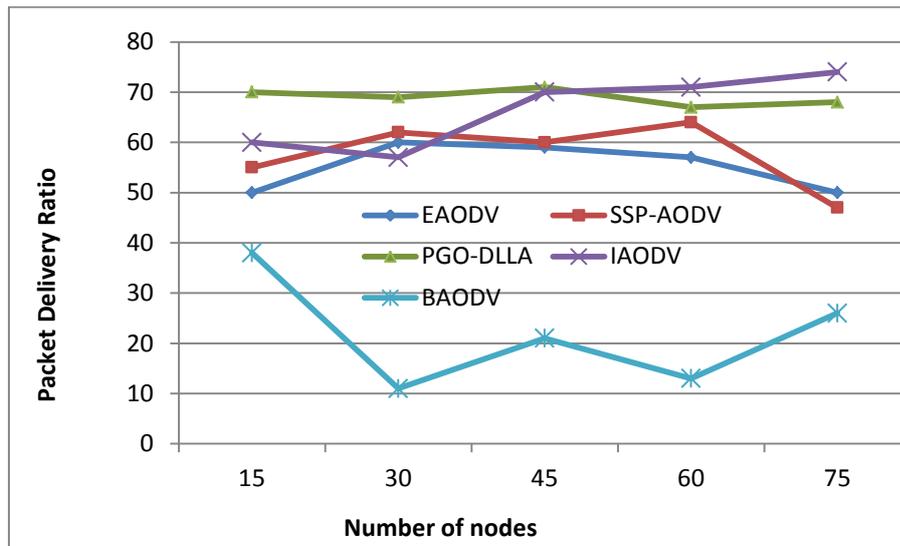


Figure 5.39. The packet delivery ratio in various numbers of nodes of EAODV, SSP-AODV and PGO-DLLA compared with IAODV and BAODV protocol

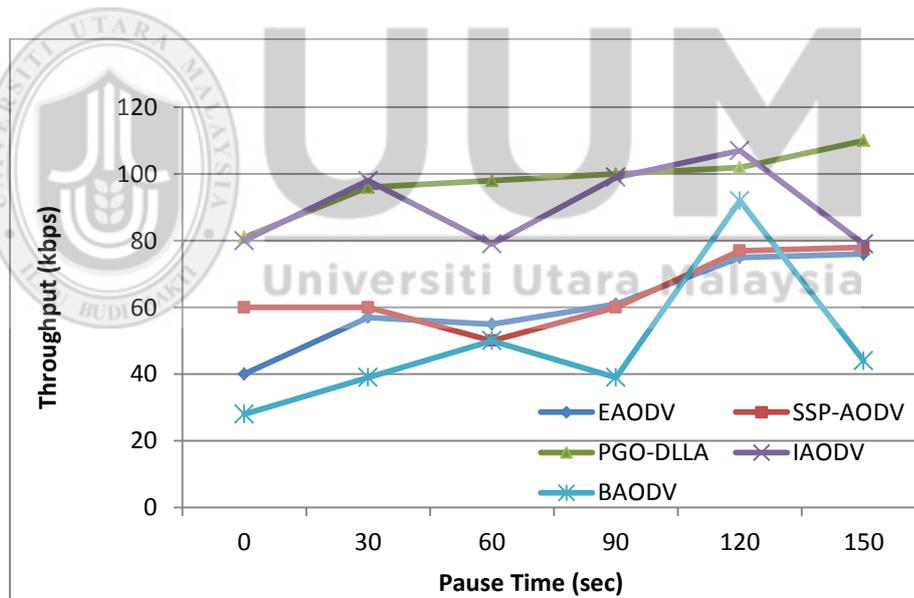


Figure 5.40. The throughput in various numbers of nodes of EAODV, SSP-AODV and PGO-DLLA compared with IAODV and BAODV protocol

#### 5.7.4 Fuzzy-Based Intrusion Detection Protocol (Fuzzy-IDS)

In this paper, the authors proposed an intrusion detection system to prevent the black hole attacks in MANET. The new proposed system is integrated with fuzzy logic in the AODV protocol to improve it against the malicious nodes in the black hole attack. They suggested a new fuzzy model that consists of four parts: part one is a parameter extraction, part two is a computation part, the verification is part three, and part four is the alarm packet generation as shown in Figure 5.41.

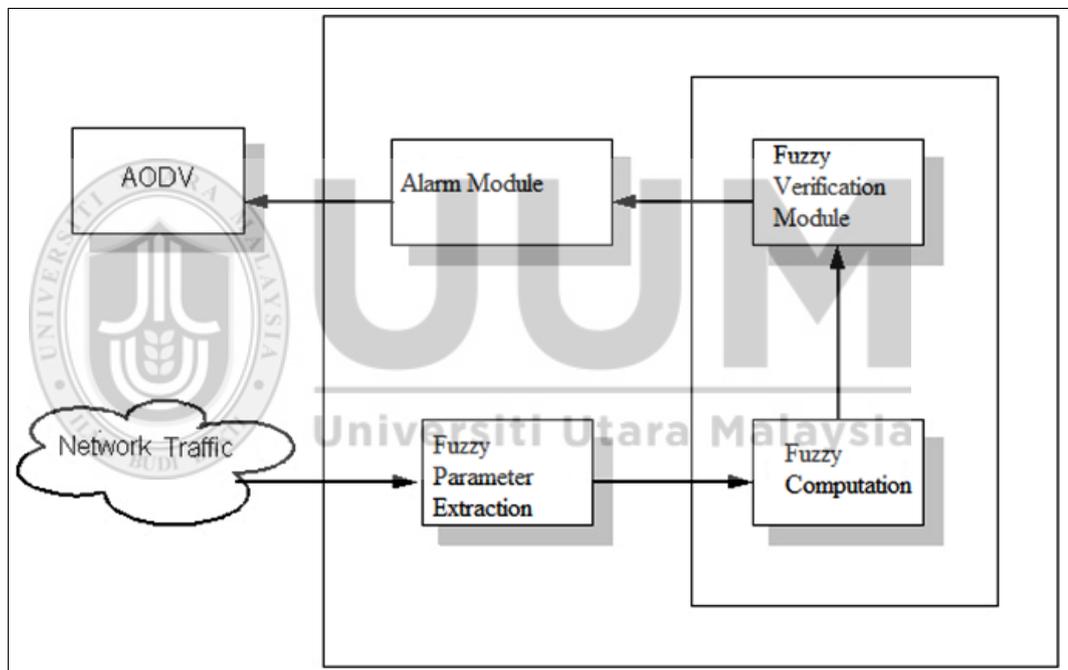


Figure 5.41. The system architecture of the Fuzzy-IDS protocol (adopt from [88])

In this method, the four components cooperate to detect the black hole attack. In the first part, the fuzzy parameter extraction extracts the information from the monitoring neighboring nodes to collect the parameters that are required for the analysis of the network traffic. Also, in this part, the neighbor table has fields for storing the values

of forward packet ratio, average destination sequence and number of fidelity level. In the second part, fuzzy computation is extracted to the fidelity level of the information that comes from the first part (fuzzy parameter extraction). Many rule bases will be used to evaluate the fidelity level values in the table of fuzzy rule base as shown in Table 5.11.

Table 5.11

*The Fuzzy Rule Base* (adopted from [88]).

	<b>Forward Packet Ratio</b>	<b>Average Destination Sequence Number</b>	<b>Fidelity Level</b>
1	LOW	LOW	LOW
2	LOW	MEDIUM	LOW
3	LOW	HIGH	LOW
4	MEDIUM	LOW	MEDIUM
5	MEDIUM	MEDIUM	MEDIUM
6	MEDIUM	HIGH	LOW
7	HIGH	LOW	HIGH
8	HIGH	MEDIUM	HIGH
9	HIGH	HIGH	LOW

In fuzzy rule base, Rule 1, for example, extracted the low fidelity level because the forward packet ratio and average destination sequence number are low. While, in Rule 8, the fidelity level is high because the forward packet ratio is high and the average destination sequence number is medium.

#### 5.7.4.1 Comparison of Proposed Protocols with Fuzzy-IDs Protocol

In this section, a comparison between the proposed protocols (AODV, SSP-AODV, and PGO-DLLA) and the current work named Fuzzy-IDs protocol is performed. The evaluation results and performance analysis are compared graphically with the same

simulation parameter of Fuzzy-IDs protocol and the same performance metrics which had been used in these experiments (packet delivery ratio, routing overhead, and end-to-end delay). The network simulator NS-2.33 [136], [142] is used to conduct the experiments of three scenarios. The first scenario is to test the original EAODV, the second scenario is to test SSP-AODV, and the third scenario is to test PGO-DLLA. These scenarios are developed to carry out the tests using the various speed of mobility and number of nodes. The simulation parameters for Scenario 1,2 and 3 are identical to the simulation parameters in [88] as shown in Table 5.12.

Table 5.12

*The PGO-DLLA, SSP-AODV, and EAODV Scenarios identical to the Scenarios in the Simulation of Fuzzy-IDs Protocol [88].*

<b>Parameter</b>	<b>Scenario 1</b>	<b>Scenario 2</b>	<b>Scenario 3</b>
Simulation Time	200 sec.	200 sec.	200 sec.
Number of Nodes	50	50	50
Routing Protocol	EAODV	SSP-AODV	PGO-DLLA
Traffic Model	CBR(UDP)	CBR(UDP)	CBR(UDP)
Pause Time	1 sec.	1sec.	1 sec.
Mobility Model	Random way point	Random way point	Random way point
Max. Speed	10 to 70 m/s	10 to 70 m/s	10 to 70 m/s
Map area	1000m x 1000m	1000m x 1000m	1000m x 1000m
MAC Type	802.11	802.11	802.11
Malicious Node	Black hole	Black hole	Black hole

The evaluation of the three routing protocols, namely EAODV, SSP-AODV and PGO-DLLA, includes two steps: firstly, the comparison between EAODV, SSP-AODV and PGO-DLLA, and secondly, the comparison between EAODV, SSP-AODV and PGO-DLLA with Fuzzy-IDs and standard AODV protocols is using three metrics: packet delivery ratio, routing overhead, and the end-to-end delay. Figures

5.42 and 5.43 illustrate the PDR result of the three proposed protocols and Fuzzy-IDs protocol where the mobility and number of nodes vary.

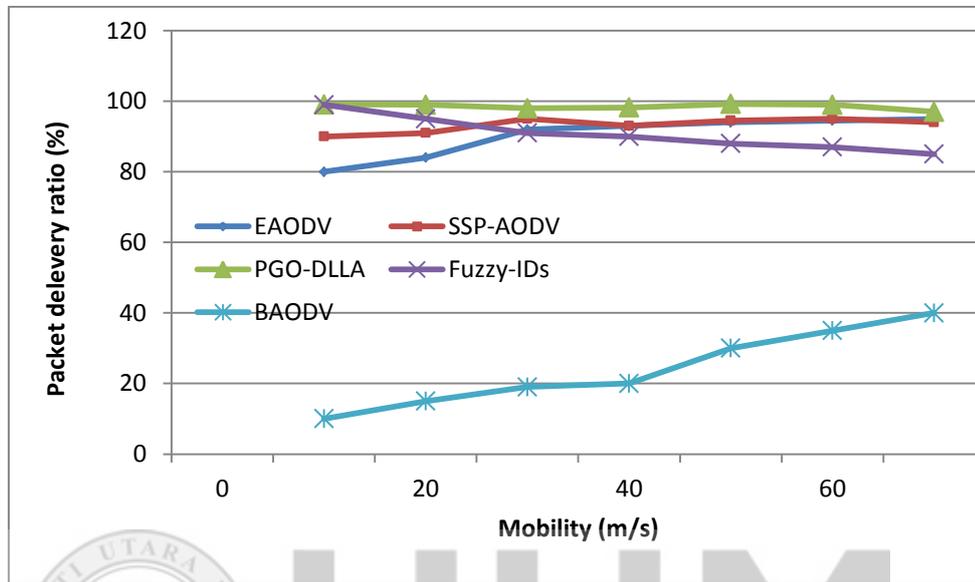


Figure 5.42. Packet delivery ratio in various mobility EAODV, SSP-AODV and PGO-DLLA protocols compared with Fuzzy-IDs and BAODV protocol

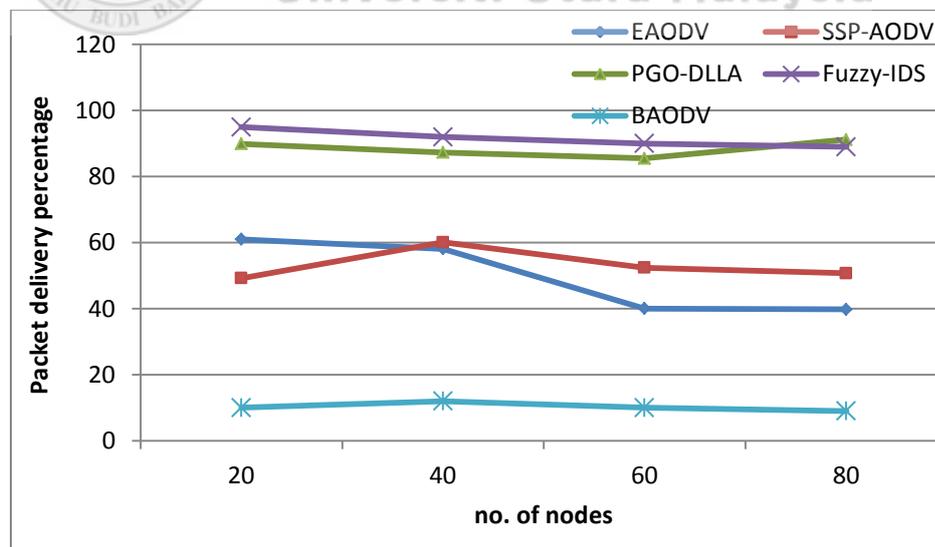


Figure 5.43. Packet delivery ratio in various numbers of nodes EAODV, SSP-AODV, and PGO-DLLA protocol compared with Fuzzy-IDs and BAODV protocol

Figures 5.44 and 5.45 illustrate the routing overhead result of the three proposed protocols and Fuzzy-IDs protocol where the mobility and number of nodes vary.

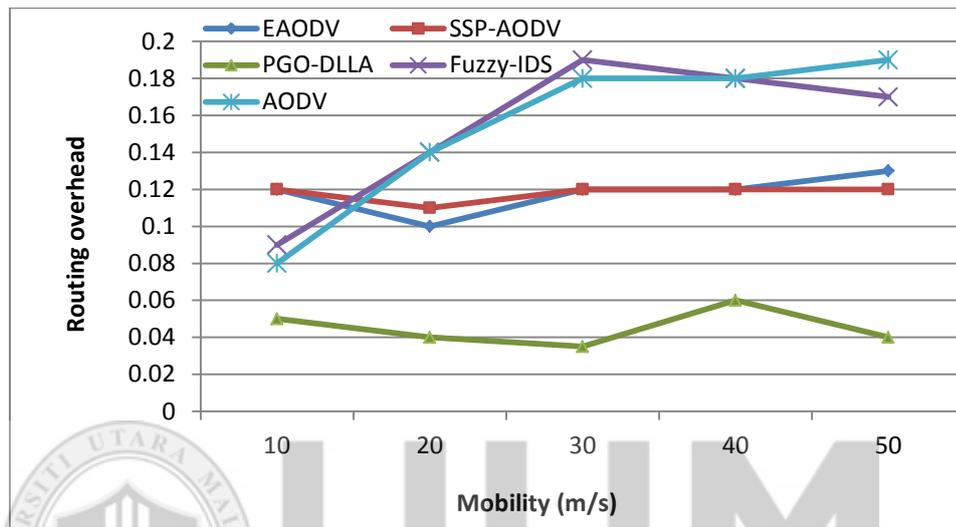


Figure 5.44. Routing overhead in various mobility of EAODV, SSP-AODV, and PGO-DLLA protocol compared with Fuzzy-IDS and BAODV protocol

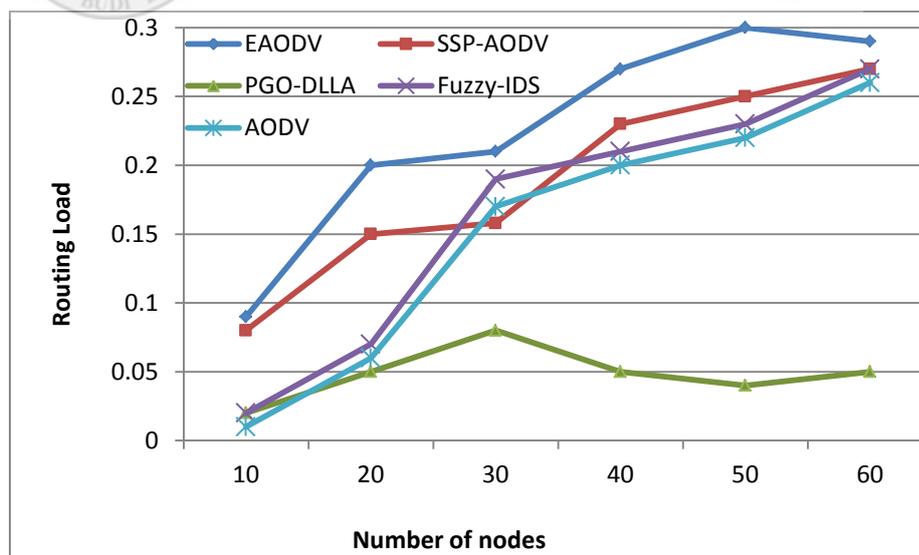


Figure 5.45. Routing overhead in various number of nodes of EAODV, SSP-AODV, and PGO-DLLA protocol compared with Fuzzy-IDS and BAODV protocol

Figures 5.46 and 5.47 illustrate the End-to-End delay result of the three proposed protocols and Fuzzy-IDS protocol where the mobility and number of nodes vary.

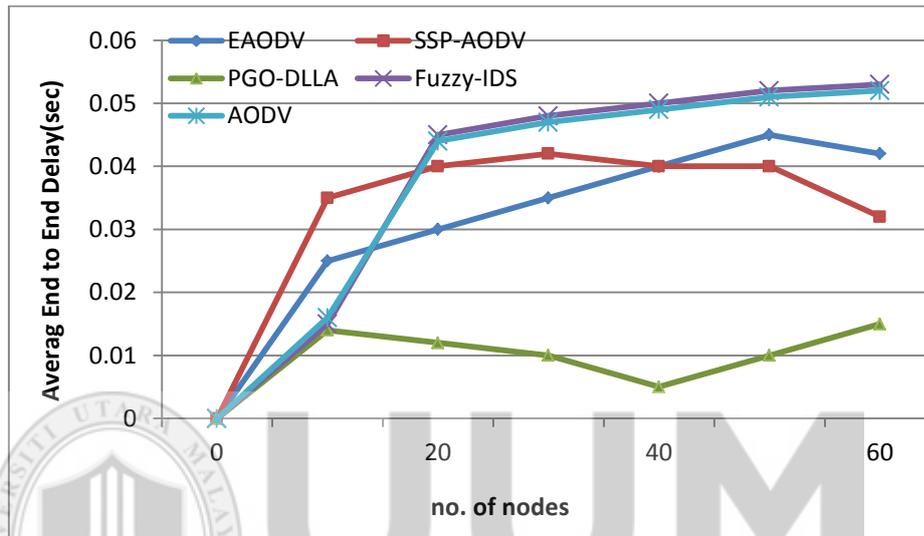


Figure 5.46. The average end-to-end delay in various mobility of EAODV, SSP-AODV, and PGO-DLLA protocol compared with Fuzzy-IDS and BAODV protocol

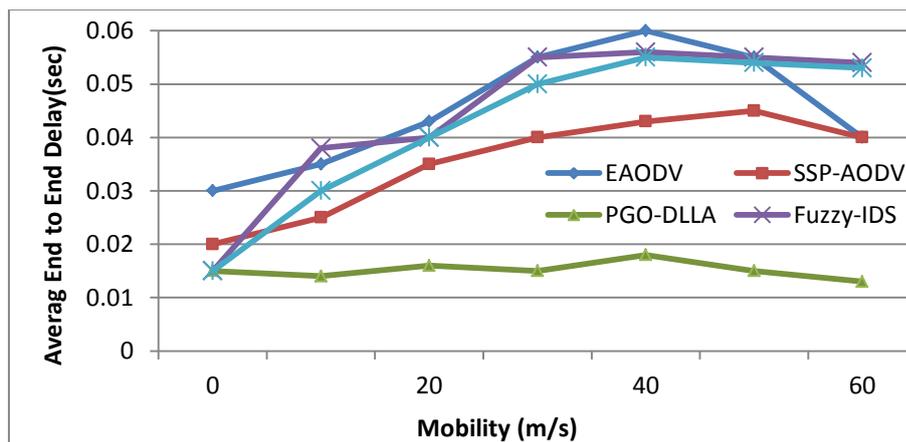


Figure 5.47. The average end-to-end delay in various numbers of nodes of EAODV, SSP-AODV, and PGO-DLLA protocol compared with Fuzzy-IDS and BAODV protocol

Table 5.13 demonstrates the comparison of the proposed protocols with the current work of the Fuzzy-IDs protocol, where the pink color shows the best protocol performance, the green color shows the average protocol performance or close to the best protocol performance, and the gray color is the bad result of protocol performance. In the experiments of packet delivery ratio with various mobility and various numbers of nodes in Figures 5.42 and 5.43 respectively, the proposed protocol PGO-DLLA is the only one best for various mobility. Figures 5.44 and 5.45 show the routing overhead protocol performance with various mobility and various numbers of nodes respectively. The proposed protocol PGO-DLLA is the best protocol. Figures 5.46 and 5.47, in terms of End-to-End Delay, show that PGO-DLLA is the best protocol performance of Fuzzy-IDs protocol performance.

Table 5.13

*Comparison Result of EAODV, SSP-AODV, and PGO-DLLA Protocols with FUZZY-IDs and Standard AODV Protocol* ■ Best ■ Average ■ Bad

	<i>Speed</i>					<i>No. of Nodes</i>				
	<i>EAODV</i>	<i>SSP-AODV</i>	<i>PGO-DLLA</i>	<i>FUZZY</i>	<i>BAD</i>	<i>EAODV</i>	<i>SSP-AODV</i>	<i>PGO-DLLA</i>	<i>FUZZY</i>	<i>BAD</i>
<i>PDR</i>			Best		Bad			Average	Best	Bad
<i>RO</i>			Best		Bad			Best	Average	Bad
<i>E2E</i>			Best		Bad			Best	Average	Bad

### **5.7.5 Swarm Based Intrusion Detection Protocol (SBDT)**

Swarm based Intrusion Detection and Defense Technique (SBDT) [89] is one of the last published works and has been chosen in order to compare the results of the implementation proposal with the SBDT of previous experiments in the same environment. In this technique, the nodes with the highest trust value, residual bandwidth and residual energy are selected as active nodes using swarm intelligence-based ant colony optimization. In this paper, the authors proposed a multiple path technique to detect and prevent the black hole attack by transmitting data using the swarm intelligence of ant colony optimization. They used the monitoring neighboring nodes to collect the trusted value for each node inside the transmission range. In this technique, the malicious nodes have a minimum trusted value compared with the threshold. The threshold value will be ordered to select the legitimate path (safety path without black hole attack) to the destination node. The authors used a threshold cryptography to find the trusted value of the neighboring nodes. However, this technique may suffer from the higher routing overhead because the keys are generated and sub keys are used to make cryptography for trusting level.

#### **5.7.5.1 Comparison of Proposed Protocols with SBDT Protocols**

In this section, a comparison between the proposed protocols (AODV, SSP-AODV, and PGO-DLLA) is performed and the current work of the SBDT protocol. The evaluation results and performance analysis are compared graphically with the same simulation parameter of the SBDT protocol and the same performance metrics which had been used in these experiments (packet delivery ratio, packet lost, and end-to-end

delay). The network simulator NS-2.33 [136], [142] is used to conduct the experiments of three scenarios. The first scenario is to test the original EAODV, the second scenario is to test SSP-AODV, and the third scenario is to test PGO-DLLA. These scenarios are developed to carry out the tests using the various numbers of attackers and various numbers of nodes. The simulation parameters for Scenario 1,2 and 3 are identical to the simulation parameters in [89] as shown in Table 5.14.

Table 5.14

*The PGO-DLLA, SSP-AODV, and EAODV Scenarios identical to the Scenarios in the Simulation of SBDT [89] .*

<b>Parameter</b>	<b>Scenario 1</b>	<b>Scenario 2</b>	<b>Scenario 3</b>
Simulation Time	50 sec.	50 sec.	50 sec.
Number of Nodes	20, 40, 60, 80, and 100	20, 40, 60, 80, and 100	20, 40, 60, 80, and 100
Routing Protocol	EAODV	SSP-AODV	PGO-DLLA
Traffic Model	CBR(UDP)	CBR(UDP)	CBR(UDP)
Pause Time	1 sec.	1sec.	1 sec.
Mobility Model	Random way point	Random way point	Random way point
Max. Speed	10 m/s	10 m/s	10 m/s
Map area	1000m x 1000m	1000m x 1000m	1000m x 1000m
MAC Type	802.11	802.11	802.11
Number of Attackers	1, 2, 3, 4, and 5	1, 2, 3, 4, and 5	1, 2, 3, 4, and 5

The evaluation of the three routing protocols, namely EAODV, SSP-AODV and PGO-DLLA, includes two steps: firstly, the comparison between EAODV, SSP-AODV and PGO-DLLA, and secondly, the comparison between EAODV, SSP-AODV and PGO-DLLA with the SBDT protocol and the black hole AODV protocol using three metrics: packet delivery ratio, packet lost, and the end-to-end delay.

Figures 5.48, 5.49 and 5.50 illustrate the PDR, PL, and EtoE delay results of the three proposed protocols and the SBDT protocol where the numbers of attackers vary.

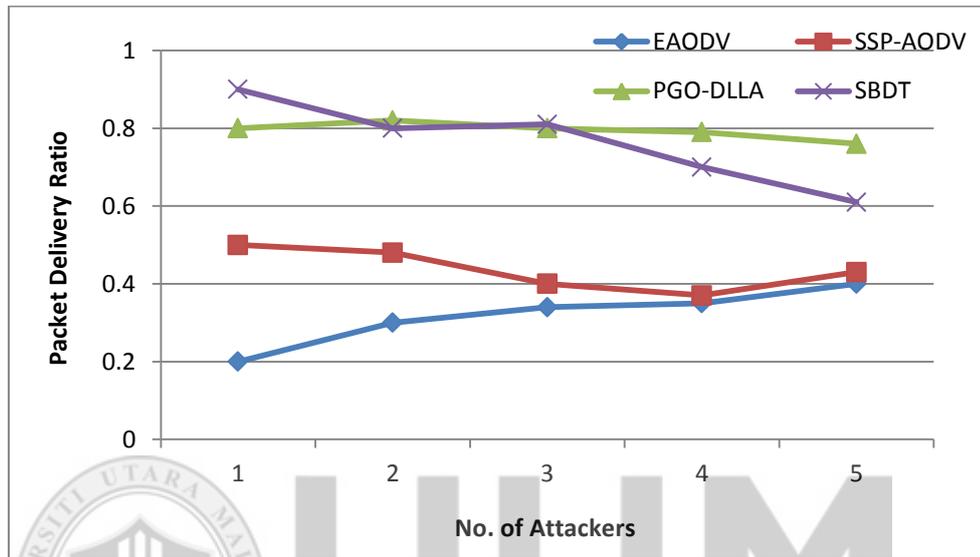


Figure 5.48. The packet delivery ratio in various numbers of attackers of EAODV, SSP-AODV, and PGO-DLLA protocol compared with SBDT protocol

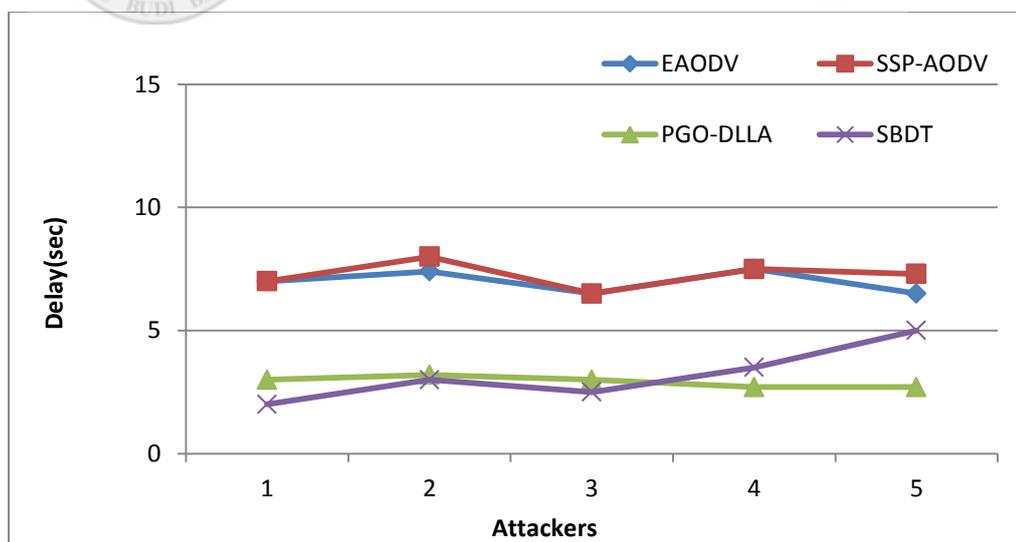
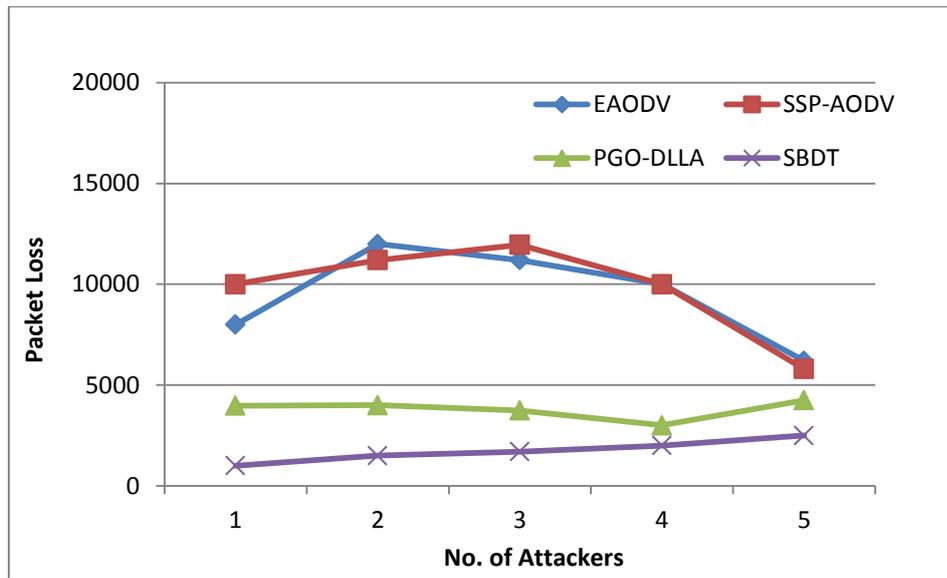


Figure 5.49. The average end-to-end delay in various numbers of attackers of EAODV, SSP-AODV, and PGO-DLLA protocol compared with SBDT protocol



*Figure 5.50.* The packet loss results in various numbers of attackers of EAODV, SSP-AODV, and PGO-DLLA protocol compared with SBDT protocol

The evaluation of EAODV, SSP-AODV and PGO-DLLA protocols with the SBDT protocol includes two graphical comparisons. The first comparison is the comparison between EAODV, SSP-AODV and PGO-DLLA, with SBDT in various numbers of attackers as shown in Figures 5.48, 5.49, and 5.50. The second comparison is the comparison between EAODV, SSP-AODV and PGO-DLLA, with SBDT in various numbers of nodes as shown in Figures 5.51, 5.52, and 5.53. The comparison among which the best, average and worst protocol performance between EAODV, SSP-AODV and PGO-DLLA with the SBDT protocol and the black hole AODV protocol uses three metrics: packet delivery ratio, packet lost, and the end-to-end delay which are illustrated in Table 5.15.

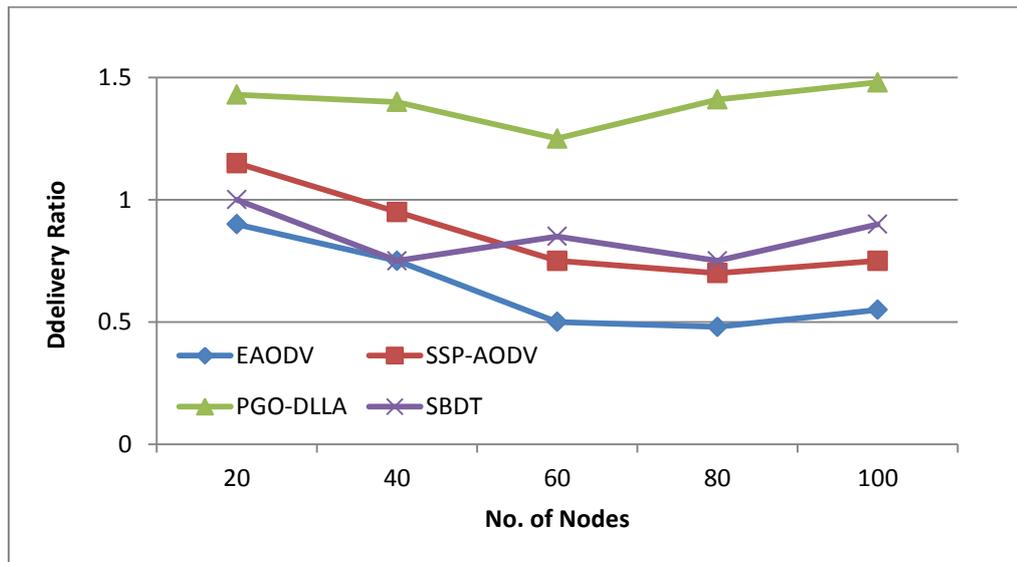


Figure 5.51. The packet delivery ratio in various numbers of nodes of EAODV, SSP-AODV, and PGO-DLLA protocol compared with DBST protocol

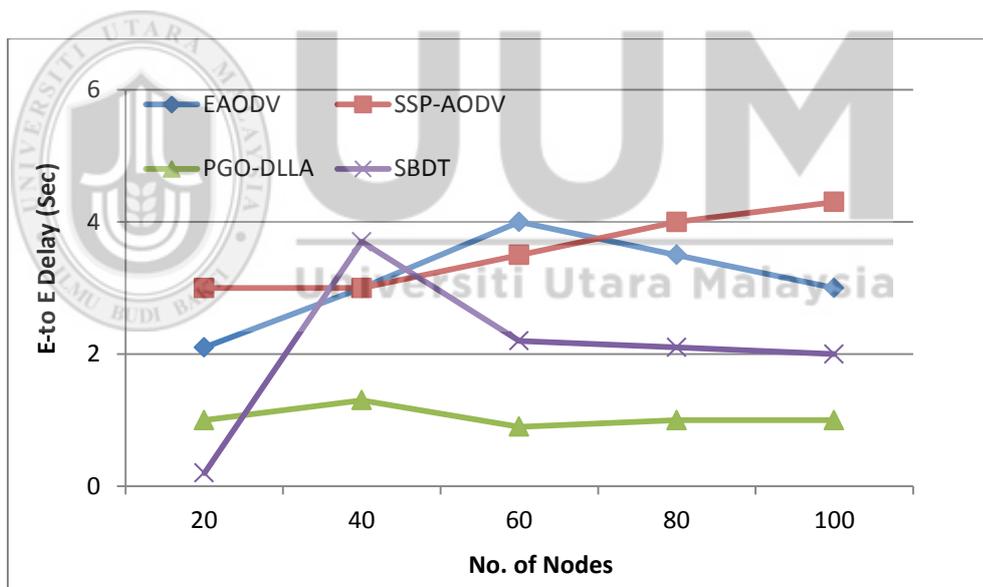


Figure 5.52. The average end-to-end delay in various numbers of nodes of EAODV, SSP-AODV, and PGO-DLLA protocol compared with DBST protocol

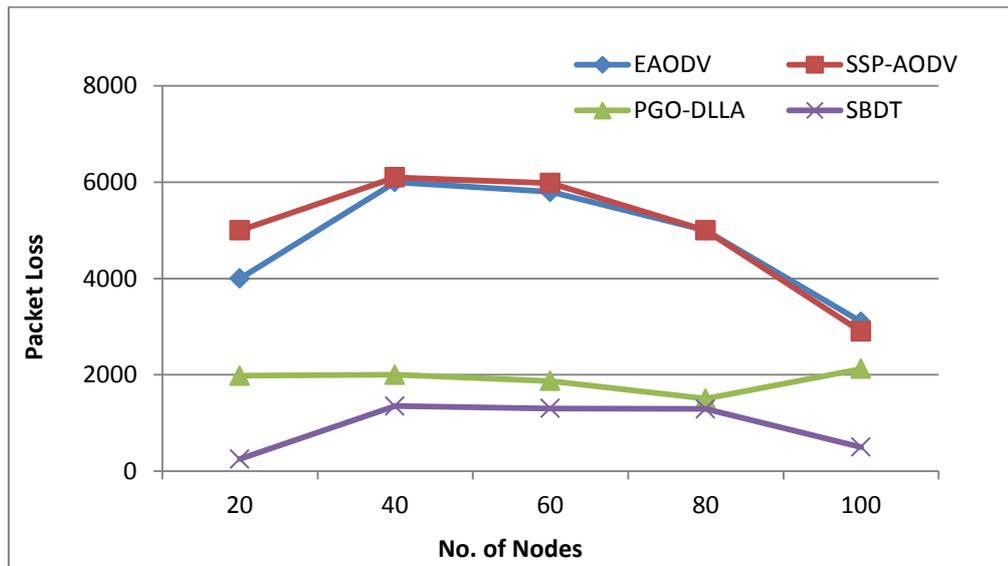


Figure 5.53. The packet loss in various numbers of nodes of EAODV, SSP-AODV, and PGO-DLLA protocol compared with DBST protocol

Table 5.15 demonstrates the comparisons of the proposed protocols with the current work of the SBDT protocol, where the pink color shows the best protocol performance, the green color shows the average protocol performance or close to the best protocol performance, and the gray color is the worst result of protocol performance. In the experiments of packet delivery ratio with various numbers of attackers and various numbers of nodes in Figures 5.48 and 5.51 respectively, the proposed protocol PGO-DLLA is the best protocol performance. Figures 5.50 and 5.51 shows the packet loss of protocol performance during the simulation with various numbers of attackers' mobility and various numbers of nodes respectively. The DBST protocol is the best protocol and the proposed protocol PGO-DLLA had an average result to the DBST protocol. Figures 5.49 and 5.52 in terms of End-to-End Delay show that PGO-DLLA is the best protocol performance from the DBST protocol performance.

Table 5.15

Comparison Result of EAODV, SSP-AODV, and PGO-DLLA Protocols with SBDS and Standard AODV Protocol ■ Best ■ Average ■ Bad

	<i>Number of Attackers</i>					<i>No. of Nodes</i>				
	<i>EAODV</i>	<i>SSP-AODV</i>	<i>PGO-DLLA</i>	<i>SBDS</i>	<i>BAD</i>	<i>EAODV</i>	<i>SSP-AODV</i>	<i>PGO-DLLA</i>	<i>SBDS</i>	<i>BAD</i>
<i>PDR</i>										
<i>PL</i>										
<i>Delay</i>										

### 5.8 Summary

This chapter provides an experimental trial in the parameters that affect the characteristic behaviors of a mobile ad hoc network; the network size and mobility. The pre-test or experiment for the size of the network has achieved in terms of the number of nodes' optimal setting for the standard AODV by performance metrics that affect network size. The mobility has been discussed in various experiments with different metrics and pause times to examine the effect of dynamic topology. The offered network load is defined as the actual load that supports the network. End-to-End delay and packet delivery ratio are compared in order to determine the best effort traffic. The routing load metric, such as normalized routing load, evaluates the efficiency of the routing protocol. However, after the tests of the experimental AODV with different models, the results of the optimal setting of network parameters showed the improvement of the shortest and secure routing from black hole attack in MANET.

## CHAPTER SIX

### THE CONCLUSION AND FUTURE WORK

#### 6.1 Introduction

Designing a new AODV protocol based on artificial intelligence techniques in preventing black hole attacks is the main goal of this thesis. The work can be divided into two parts to achieve the main goal; first, this thesis used the heuristic search algorithm as a local search to improve the shortest path from source to destination node.

Second, this thesis used the parallel technique to achieve the prevention of black hole nodes and improve the routing security. The next section summarizes the conclusion drawn from the experiments and results discussed in Chapter Six, while the following section suggests the directions for future work.

#### 6.2 Research Contribution

There are three basic contributions of this thesis. The first contribution is finding the best search method to enhance the shortest path to the destination. The hop count is the number of hops needed to reach the destination, so they minimize the hop count, which means the shortest path.

Basically, when there are two paths to the destination, the shortest one is the safest path. The combination of the heuristic search algorithm with the AODV route

discovery as the search method is the best approach in finding the shortest path to the destination. The first algorithm is the heuristic search algorithm A\*.

However, this search algorithm does not give the best performance in the dynamic search environment because it used an estimated location value in the initial start and it needs to a classical method such as hash chain function to enhance the prevention against the black hole nodes.

Using the Floyd-Warshall's algorithm will enhance the dynamic search discovery, but it will increase the routing overhead. The second proposed SSP-AODV protocol is based on a new technique by waiting the duplicated response from intermediate nodes about the path to the destination to avoid the black hole attacks.

Experiments in Chapter Five found that SSP-AODV performed better than EAODV in preventing the black hole attack, but it still needs to improve the search discovery. One of the important contributions of this thesis is the proposed improved routing discovery based on the meta-heuristic algorithms.

VDLLA is an inspired algorithm from the nature of a spider named daddy long-legs, which works as a swarm of agents (8-legs) algorithm. The third proposal is a parallel grid optimization using the virtual daddy long-legs algorithm (PGO-DLLA), VDLLA is integrated with the AODV routing protocol to optimize the shortest and secure path. As a conclusion, PGO-DLLA is able to improve the route discovery against the black hole attacks in AODV. Experiments in this thesis have shown that PGO-DLLA is able to reduce the normalized routing load, end-to-end delay, and packet loss and has a

good throughput and packet delivery ratio when compared with the standard AODV protocol, BAODV protocol, and five current protocols that enhanced the routing security of the AODV protocols.

### **6.3 Objectives of the Research**

The objectives of this research are to investigate the issues related to AODV's current mechanism in terms of secure routing and to develop a new mechanism to prevent single and cooperative black hole attacks in AODV's route discovery.

1. First objective: To design a new algorithm to prevent single and cooperative black hole attacks in the AODV protocol by achieving three sub objectives:

a) Improving the route discovery (RREQ and RREP) through a heuristic search algorithm, to enhance the shortest path technique in the routing mechanism.

This sub-objective is achieved by developing the Enhanced Ad hoc On-demand Distance Vector protocol (EAODV), as Chapter Four Section 4.2 has explained.

b) Testing the new technique with trusted and malicious environments and comparing the results with those of the original AODV. To achieve this sub objective, pre-test experiments of the AODV protocol in hostile and peaceful environments has been conducted, as Chapter Five Section 5.2 has explained.

c) Improving the route discovery by using a meta-heuristic algorithm to enhance finding the shortest and most secure path to the destination. This sub objective

is integrating the meta-heuristic search algorithm in routing discovery to prevent the cooperative black hole attacks.

This sub objective is achieved by developing the Shortest Secure Path for Ad hoc On-demand Distance Vector protocol (SSP-AODV), as Chapter Four Section 4.3 has explained, and the Parallel Grid Optimization by Daddy Long-Legs Algorithm (PGO-DLLA), as Chapter Four Section 4.4 has explained.

2. Second objective: To implement the proposed algorithm in preventing black hole attacks and to evaluate its performance in the simulated environment. The second objective is related to implementing the developed protocol.

To achieve this objective, the experiments using NS2 simulator have been conducted to check the EAODV and SSP-AODV protocols with five performance metrics (PL, EtoE, PDR, NRL, and TH), as Chapter Five Sections 5.3 and 5.4 have explained.

In addition, more than one hundred and eighty experiments using the NS2 simulator have been carried out to check the PGO-DLLA protocol with the performance metrics (PL, EtoE, PDR, and TH), as Chapter Five Sections 5.6.1 and 5.6.2 have explained.

3. Third objective: To evaluate the effect of the proposed algorithm in terms of validation based on the simulation results and then comparing it with the original AODV.

4. This objective is achieved by the results gained from the simulation experiments of the developer protocol and the comparisons of EAODV, SSP-AODV, and PGO-DLLA and related works as well, as Chapter Five Sections 5.5, 5.6, and 5.7 have explained.

#### **6.4 Future Work**

In future, it is suggested to test the performance of preventing the malicious attacks on most difficult environments in MANETs, with huge numbers of attackers, since this study has only focused on medium difficult environments. More difficult environments may cause more routing overhead.

On the other hand, a study on the enhancement of the route discovery in the AODV routing protocol could also be done in the route discovery on all the on-demand routing protocols, especially that they share the same properties and the same malicious attacks.

Future work also can enhance the proposed AODV, SSP-AODV, and PGO-DLLA protocols to cope with other attacks such as worm hole attack or gray hole attack that are also shared in some features of attacks with black hole attacks.

Although the proposed protocols have been designed to prevent black hole attacks in the AODV protocol using some of the heuristic search algorithms, it is also interesting to adapt other techniques to prevent black hole attacks.

For future work, it is suggested to apply VDLLA to solve a few NP-hard optimization problems (such as traveling salesman), besides the ability to gain a good result in some fields such as the security optimization problems, network routing algorithm, and dynamic grid computing algorithm.

However, it has been planned to examine the enforcement of additional complex attacks. Furthermore, it has been planned to apply PGO-VDLLA in the fields of robotic research such as the robotic system in medicine, engineering, and space discovery.



## REFERENCES

- [1] J. Zutt, A. J. C. van Gemund, M. M. de Weerdt, and C. Witteveen, "Dealing with Uncertainty in Operational Transport Planning," in *Intelligent Infrastructures*, vol. 42, R. R. Negenborn, Z. Lukszo, and H. Hellendoorn, Eds. Springer, 2010, pp. 355–382.
- [2] L. Tamilselvan and V. Sankaranarayanan, "Prevention of Co-operative Black Hole Attack in MANET," *J. Networks*, vol. 3, no. 5, pp. 13–20, May 2008.
- [3] D. Medhi and K. Ramasamy, *Network Routing Algorithms, Protocols, and Architectures*. Morgan Kaufmann Publishers is an imprint of Elsevier, 2010, p. 848.
- [4] S. Agrawal, S. Jain, and S. Sharma, "A Survey of Routing Attacks and Security Measures in Mobile Ad-hoc Networks," *arXiv Prepr. arXiv1105.5623*, 2011.
- [5] C. Perkins, Elding-Royer, and S. Das, "Ad hoc On-Demand Distance Vector (AODV) Routing," *IETF RFC 3561*, pp. 1–37, 2003.
- [6] C. Perkins, E. Royer, S. R. Das, and M. K. Marina, "Performance Comparison of Two On-demand Routing Protocols for Ad hoc Networks," *IEEE Pers. Commun.*, vol. 8, no. 1, pp. 16–28, 2001.
- [7] D. B. Johnson and D. A. Maltz, *Dynamic Source Routing in Ad Hoc Wireless Networks*. Springer, 1996, pp. 153–181.
- [8] L. Zhou and Z. J. Haas, "Securing Ad Hoc Networks," *Network, IEEE*, vol. 13, no. 6, pp. 24–30, 1999.
- [9] C.-K. Toh, "Associativity-based Routing for Ad Hoc Mobile Networks," *Wirel. Pers. Commun.*, vol. 4, no. 2, pp. 103–139, 1997.
- [10] V. D. Park and M. S. Corson, "A Highly Adaptive Distributed Routing Algorithm for Mobile Wireless Networks," in *INFOCOM'97. Sixteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Driving the Information Revolution., Proceedings IEEE*, 1997, vol. 3, pp. 1405–1413.
- [11] I. Chakeres and C. Perkins, "Dynamic MANET On-demand (DYMO) Routing," *Draft. (work progress)*, 2008.
- [12] E. M. Royer and C.-K. Toh, "A Review of Current Routing Protocols for Ad Hoc Mobile Wireless Networks," *Pers. Commun. IEEE*, vol. 6, no. 2, pp. 46–55, 1999.
- [13] S.-J. Lee, M. Gerla, and C.-K. Toh, "A Simulation Study of Table-driven and On-demand Routing Protocols for Mobile Ad Hoc Networks," *Network, IEEE*, vol. 13, no. 4, pp. 48–54, 1999.
- [14] K. Akkaya and M. Younis, "A Survey on Routing Protocols for Wireless Sensor Networks," *Ad hoc networks, Elsevier*, vol. 3, no. 3, pp. 1–38, 2010.

- [15] C. E. Perkins and P. Bhagwat, "Highly Dynamic Destination-sequenced Distance-vector Routing (DSDV) for Mobile Computers," in *Proc. ACM Conference on Communications Architectures, Protocols and Applications (SIGCOMM'94)*, 1994, vol. 24, no. 4, pp. 234–244.
- [16] T. Clausen, G. Hansen, L. Christensen, and G. Behrmann, "The Optimized Link State Routing Protocol, Evaluation Through Experiments and Simulation," in *IEEE Symposium on Wireless Personal Mobile Communications*, 2001.
- [17] S. Murthy and J. J. Garcia-Luna-Aceves, "An Efficient Routing Protocol for Wireless Networks," *Mob. Networks Appl. Springer*, vol. 1, no. 2, pp. 183–197, 1996.
- [18] C.-C. Chiang, H.-K. Wu, W. Liu, and M. Gerla, "Routing in Clustered Multihop, Mobile Wireless Networks With Fading Channel," in *proceedings of IEEE SICON*, 1997, vol. 97, no. 1997.4, pp. 197–211.
- [19] M. Abolhasan, T. Wysocki, and E. Dutkiewicz, "A Review of Routing Protocols for Mobile Ad Hoc Networks," *Ad hoc networks, Elsevier*, vol. 2, no. 1, pp. 1–22, 2004.
- [20] Z. J. Haas, M. R. Pearlman, and P. Samar, "The Zone Routing Protocol (ZRP) for Ad Hoc Networks," *Draft. txt*, 2002.
- [21] R. Sivakumar, P. Sinha, and V. Bharghavan, "CEDAR: A Core-extraction Distributed Ad Hoc Routing Algorithm," *Sel. Areas Commun. IEEE J.*, vol. 17, no. 8, pp. 1454–1465, 1999.
- [22] T. Hamma, T. Katoh, B. B. Bista, and T. Takata, "An Efficient Zhls Routing Protocol for Mobile Ad Hoc Networks," in *Database and Expert Systems Applications, 2006. DEXA '06. 17th International Workshop on*, 2006, pp. 66–70.
- [23] H. Weerasinghe and H. Fu, "Preventing Cooperative Black Hole Attacks in Mobile Ad Hoc Networks: Simulation Implementation and Evaluation," in *Future Generation Communication and Networking (FGCN 2007)*, 2007, vol. 2, pp. 362–367.
- [24] C. S. R. Murthy and B. S. Manoj, *Ad Hoc Wireless Networks Architectures and Protocols*. Prentice Hall, 2004.
- [25] H. Deng, W. Li, and D. P. Agrawal, "Routing Security in Wireless Ad Hoc Networks," *Commun. Mag. IEEE*, vol. 40, no. 10, pp. 70–75, 2002.
- [26] M. Ilyas and R. C. Dorf, Eds., *The Handbook of Ad Hoc Wireless Networks*. Boca Raton, FL, USA: CRC Press, Inc., 2003.
- [27] B. Kannhavong, H. Nakayama, Y. Nemoto, N. Kato, and A. Jamalipour, "A Survey of Routing Attacks in Mobile Ad Hoc Networks," *IEEE Wirel. Commun.*, vol. 14, no. 5, pp. 85–91, 2007.

- [28] B. Wu, J. Chen, J. Wu, and M. Cardei, "A Survey of Attacks and Countermeasures in Mobile Ad Hoc Networks," in *Wireless Network Security*, Springer, 2007, pp. 103–135.
- [29] Y.-R. Tsai and S.-J. Wang, "Two-tier Authentication for Cluster and Individual Sets in Mobile Ad Hoc Networks," *Comput. Network, Elsevier North-Holland, Inc.*, vol. 51, no. 3, pp. 883–900, 2007.
- [30] P. Narula, S. K. Dhurandher, S. Misra, and I. Woungang, "Security in Mobile Ad-hoc Networks Using Soft Encryption and Trust-based Multi-path Routing," *Comput. Commun., Elsevier Sci. Publ. B. V.*, vol. 31, no. 4, pp. 760–769, 2008.
- [31] Y. Liang, H. V. Poor, and L. Ying, "Secrecy Throughput of MANETs Under Passive and Active Attacks," *Inf. Theory, IEEE Trans.*, vol. 57, no. 10, pp. 6692–6702, 2011.
- [32] X. Wang, Ed., *Mobile Ad-Hoc Networks: Applications*. InTech, 2011.
- [33] D. Glynos, P. Kotzanikolaou, and C. Douligeris, "Preventing Impersonation Attacks in MANET with Multi-factor Authentication," in *Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks, 2005. WIOPT 2005. Third International Symposium on*, 2005, pp. 59–64.
- [34] S. Misra, I. Woungang, and S. C. Misra, *Guide to Wireless Ad hoc Networks*. Springer, 2009.
- [35] A. K. Rai, R. R. Tewari, and S. K. Upadhyay, "Different Types of Attacks on Integrated MANET-Internet Communication," *Int. J. Comput. Sci. Secur.*, vol. 4, no. 3, pp. 265–274, 2010.
- [36] Y. Desmedt, "Man-in-the-middle attack," in *Encyclopedia of Cryptography and Security*, Springer, 2011, p. 759.
- [37] M.-Y. Su, "Prevention of Selective Black Hole Attacks on Mobile Ad Hoc Networks Through Intrusion Detection Systems," *Comput. Commun. Elsevier B.V.*, vol. 34, no. 1, pp. 107–117, 2011.
- [38] S. Tan and K. Kim, "Secure Route Discovery for Preventing Black Hole Attacks on AODV-based MANETs," in *Proceedings - 2013 IEEE International Conference on High Performance Computing and Communications, HPCC 2013 and 2013 IEEE International Conference on Embedded and Ubiquitous Computing, EUC 2013*, 2014, pp. 1159–1164.
- [39] Z. Xu, S. Dai, and J. J. Garcia-Luna-Aceves, "A More Efficient Distance Vector Routing Algorithm," in *MILCOM 97 Proceedings*, 1997, vol. 2, pp. 993–997.
- [40] R. V Boppana and S. P. Konduru, "An Adaptive Distance Vector Routing Algorithm for Mobile Ad Hoc Networks," in *INFOCOM 2001. Twentieth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, 2001, vol. 3, pp. 1753–1762.

- [41] F.-H. Tseng, L.-D. Chou, and H.-C. Chao, "A Survey of Black Hole Attacks in Wireless Mobile Ad Hoc Networks," *Human-centric Comput. Inf. Sci. Springer Open Ltd*, vol. 1, no. 1, p. 4, 2011.
- [42] H. Nakayama and S. Kurosawa, "A Dynamic Anomaly Detection Scheme for AODV-based Mobile Ad Hoc Networks," *IEEE Trans. Veh. Technol.*, vol. 58, no. 5, pp. 2471–2481, 2009.
- [43] S. Kurosawa, H. Nakayama, and N. Kato, "Detecting Blackhole Attack on AODV-based Mobile Ad Hoc Networks by Dynamic Learning Method," *Int. J. Netw. Secur.*, vol. 5, no. 3, pp. 338–346, 2007.
- [44] X. Zhang, Y. Sekiya, and Y. Wakahara, "Proposal of a Method to Detect Black Hole Attack in MANET," in *Proceedings - 2009 International Symposium on Autonomous Decentralized Systems, ISADS 2009*, 2009, pp. 149–154.
- [45] Y. Zhang and W. Lee, "Intrusion Detection in Wireless Ad-hoc Networks," in *Proceedings of the 6th annual international conference on Mobile computing and networking - MobiCom '00*, 2000, pp. 275–283.
- [46] K. Sanzgiri, B. Dahill, B. N. Levine, C. Shields, and E. M. Belding-Royer, "A Secure Routing Protocol for Ad Hoc Networks," in *10th IEEE International Conference on Network Protocols, 2002. Proceedings.*, 2002, pp. 78–87.
- [47] S. Yi, P. Naldurg, and R. Kravets, "A Security-aware Routing Protocol for Wireless Ad Hoc Networks," *Urbana*, vol. 51, p. 61801, 2002.
- [48] H. S. Jassim, S. Yussof, T. S. Kiong, S. P. Koh, and R. Ismail, "A Routing Protocol based on Trusted and shortest Path selection for Mobile Ad hoc Network," in *Communications (MICC), 2009 IEEE 9th Malaysia International Conference on*, 2009, pp. 547–554.
- [49] A. Sherif, M. Elsabrouty, and A. Shoukry, "A Novel Taxonomy of Black-Hole Attack Detection Techniques in Mobile Ad-hoc Network (MANET)," in *2013 IEEE 16th International Conference on Computational Science and Engineering*, 2013, pp. 346–352.
- [50] S. K. Tiong and H. S. Jassim, "EMNet: Electromagnetic-like Mechanism Based Routing Protocol for Mobile Ad Hoc Network," *Trendas Appl. Sci. Res.*, vol. 7, no. 11, pp. 881–900, Nov. 2012.
- [51] G. F. Luger, *Artificial Intelligence: Structures and Strategies for Complex Problem Solving*. Pearson education, 2005.
- [52] S. Russell and P. Norvig, *A Modern Approach*, vol. 25. Englewood Cliffs, New Jersey: Alan Apt, 1995.
- [53] R. W. Floyd, "Algorithm 97: Shortest Path," *Commun. ACM*, vol. 5, no. 6, p. 345, 1962.

- [54] A. Singh, C. R. Ramakrishnan, and S. A. Smolka, "A Process Calculus for Mobile Ad Hoc Networks," *Sci. Comput. Program. Elsevier*, vol. 75, no. 6, pp. 440–469, 2010.
- [55] H. Yang, H. Luo, F. Ye, S. Lu, and L. Zhang, "Security in Mobile Ad Hoc Networks: Challenges and Solutions," *Wirel. Commun. IEEE*, vol. 11, no. 1, pp. 38–47, 2004.
- [56] P. Papadimitratos and Z. J. Haas, "Secure Routing for Mobile Ad Hoc Networks," in *Proceedings of the SCS Communication Networks and Distributed Systems Modeling and Simulation Conference (CNDS)*, 2002, pp. 193–204.
- [57] D. W. Carman, P. S. Kruus, and B. J. Matt, "Constraints and Approaches for Distributed Sensor Network Security," *DARPA Proj. report, (Cryptographic Technol. Group, Trust. Inf. Syst. NAI Labs)*, vol. 1, p. 1, 2000.
- [58] K. Lakshmi, S. Manju Priya, A. Jeevarathinam, K. Rama, and K. Thilagam, "Modified AODV Protocol Against Blackhole Attacks in MANET," *Int. J. Eng. Technol.*, vol. 2, no. 6, pp. 444–449, 2010.
- [59] H. Xiao, W. K. G. Seah, A. Lo, and K. C. Chua, "A Flexible Quality of Service Model for Mobile Ad-hoc Networks," in *Vehicular Technology Conference Proceedings, 2000. VTC 2000-Spring Tokyo. 2000 IEEE 51st*, 2000, vol. 1, pp. 445–449.
- [60] B. Sun, L. Osborne, Y. Xiao, and S. Guizani, "Intrusion Detection Techniques in Mobile Ad Hoc and Wireless Sensor Networks," *Wirel. Commun. IEEE*, vol. 14, no. 5, pp. 56–63, 2007.
- [61] D.-Z. Du Yang Xiao, Xuemin (Sherman) Shen, *Wireless Network Security*. Boston, MA: Springer US, 2007.
- [62] W. Lou, W. Liu, and Y. Fang, "SPREAD: Enhancing Data Confidentiality in Mobile Ad Hoc Networks," in *INFOCOM 2004. Twenty-third Annual Joint Conference of the IEEE Computer and Communications Societies*, 2004, vol. 4, pp. 2404–2413.
- [63] K. S. Ng and W. K. G. Seah, "Routing Security and Data Confidentiality for Mobile Ad Hoc Networks," in *Vehicular Technology Conference, 2003. VTC 2003-Spring. The 57th IEEE Semiannual*, 2003, vol. 3, pp. 1821–1825.
- [64] A.-S. K. Pathan, *Security of Self-organizing Networks: MANET, WSN, WMN, VANET*. CRC press, 2010.
- [65] V. R. Ghorpade, Y. V. Joshi, and R. R. Manthalkar, "Efficient Public Key Authentication in MANET," in *Proceedings of the International Conference on Advances in Computing, Communication and Control*, 2009, pp. 106–112.
- [66] T. R. Andel and A. Yasinsac, "On the Credibility of MANET Simulations," *Comput. IEEE Comput. Soc.*, vol. 39, no. 7, pp. 48–54, 2006.

- [67] R. B. Bobba, L. Eschenauer, V. Gligor, and W. Arbaugh, "Bootstrapping Security Associations for Routing in Mobile Ad-hoc Networks," in *Global Telecommunications Conference, 2003. GLOBECOM'03. IEEE*, 2003, vol. 3, pp. 1511–1515.
- [68] M. G. Zapata, "Secure Ad Hoc On-demand Distance Vector Routing," *ACM SIGMOBILE Mob. Comput. Commun. Rev.*, vol. 6, no. 3, pp. 106–107, 2002.
- [69] G. Vigna, S. Gwalani, K. Srinivasan, E. M. Belding-Royer, and R. A. Kemmerer, "An Intrusion Detection Tool for AODV-Based Ad Hoc Wireless Networks," in *Computer Security Applications Conference, 2004. 20th Annual*, 2004, pp. 16–27.
- [70] J.-M. Kim and J.-W. Jang, "AODV Based Energy Efficient Routing Protocol for Maximum Lifetime in MANET," in *Telecommunications, 2006. AICT-ICIW'06. International Conference on Internet and Web Applications and Services/Advanced International Conference on*, 2006, p. 77.
- [71] T. Hara, "Quantifying Impact of Mobility on Data Availability in Mobile Ad Hoc Networks," *Mob. Comput. IEEE Trans.*, vol. 9, no. 2, pp. 241–258, 2010.
- [72] S. Kumar, V. S. Raghavan, and J. Deng, "Medium Access Control Protocols for Ad Hoc Wireless Networks: A Survey," *Ad hoc networks, Elsevier*, vol. 4, no. 3, pp. 326–358, 2006.
- [73] S.-L. Wu, Y.-C. Tseng, and J.-P. Sheu, "Intelligent Medium Access for Mobile Ad Hoc Networks with Busy Tones and Power Control," *Sel. Areas Commun. IEEE J.*, vol. 18, no. 9, pp. 1647–1657, 2000.
- [74] P. M. Jawandhiya, M. M. Ghonge, M. S. Ali, and J. S. Deshpande, "A Survey of Mobile Ad Hoc Network Attacks," *Int. J. Eng. Sci. Technol.*, vol. 2, no. 9, pp. 4063–4071, 2010.
- [75] A. Mishra, K. Nadkarni, and A. Patcha, "Intrusion Detection in Wireless Ad Hoc Networks," *Wirel. Commun. IEEE*, vol. 11, no. 1, pp. 48–60, 2004.
- [76] Y. Xiao, X. S. Shen, and D.-Z. Du, Eds., *Wireless Network Security*. Boston, MA: Springer US, 2007.
- [77] C. Perkins and E. Royer, "Ad-hoc On-demand Distance Vector Routing," in *Proceedings WMCSA '99. Second IEEE Workshop on Mobile Computing Systems and Applications*, 1999, pp. 90–100.
- [78] W. Kozma and L. Lazos, "REAct: Resource-efficient Accountability for Node Misbehavior in Ad Hoc Networks Based on Random Audits," in *Proceedings of the second ACM conference on Wireless network security*, 2009, pp. 103–110.
- [79] K. Chadha and S. Jain, "Impact of Black Hole and Gray Hole Attack in AODV Protocol," in *International Conference on Recent Advances and Innovations in Engineering (ICRAIE-2014)*, 2014, pp. 1–7.

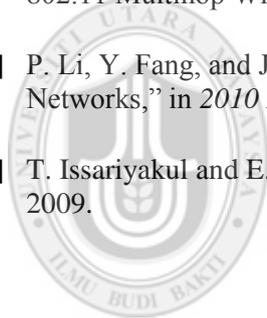
- [80] Y.-C. Hu, A. Perrig, and D. B. Johnson, "Ariadne: A Secure On-Demand Routing Protocol for Ad Hoc Networks," *Wirel. networks, Springer-Verlag New York, Inc.*, vol. 11, no. 1–2, pp. 21–38, Jan. 2005.
- [81] X. Zeng, R. Bagrodia, and M. Gerla, "GloMoSim: A Library for Parallel Simulation of Large-scale Wireless Networks," in *Parallel and Distributed Simulation, 1998. PADS 98. Proceedings. Twelfth Workshop on*, 1998, pp. 154–161.
- [82] H. W. Hesiri Weerasinghe and H. F. Huirong Fu, "Preventing Cooperative Black Hole Attacks in Mobile Ad Hoc Networks: Simulation Implementation and Evaluation," *Int. J. Softw. Eng. its Appl.*, vol. 2, no. 3, pp. 39–54, 2008.
- [83] G. Wahane, A. M. Kanthe, and D. Simunic, "Technique for Detection of Cooperative Black Hole Attack Using True-link in Mobile Ad-hoc Networks," in *Information and Communication Technology, Electronics and Microelectronics (MIPRO), 2014 37th International Convention on*, 2014, pp. 1428–1434.
- [84] R. Yerneni and A. K. Sarje, "Enhancing Performance of AODV Against Black Hole Attack," in *Proceedings of the CUBE International Information Technology Conference*, 2012, pp. 857–862.
- [85] N. Sharma and A. Sharma, "The Black-hole Node Attack in MANET," in *Advanced Computing & Communication Technologies (ACCT), 2012 Second International Conference on*, 2012, pp. 546–550.
- [86] S. Lu, L. Li, K.-Y. Lam, and L. Jia, "SAODV: A MANET Routing Protocol that can Withstand Black Hole Attack," in *2009 International Conference on Computational Intelligence and Security*, 2009, vol. 2, pp. 421–425.
- [87] K. S. Sujatha, V. Dharmar, and R. S. Bhuvaneshwaran, "Design of Genetic Algorithm Based IDS for MANET," in *Recent Trends In Information Technology (ICRTIT), 2012 International Conference on*, 2012, pp. 28–33.
- [88] Kulbhushan and J. Singh, "Fuzzy Logic Based Intrusion Detection System against Blackhole Attack on AODV in MANET," *IJCA Spec. Issue Netw. Secur. Cryptogr.*, no. 2, pp. 28–35, 2011.
- [89] G. Indirani and K. Selvakumar, "Swarm based Intrusion Detection and Defense Technique for Malicious Attacks in Mobile Ad Hoc Networks," *Int. J. Comput. Appl.*, vol. 50, no. 19, pp. 1–7, 2012.
- [90] M. Dorigo and L. M. Gambardella, "Ant Colony System: A Cooperative Learning Approach to the Traveling Salesman Problem," *Evol. Comput. IEEE Trans.*, vol. 1, no. 1, pp. 53–66, 1997.
- [91] W. Wang, B. Bhargava, and M. Linderman, "Defending Against Collaborative Packet Drop Attacks on MANETs," in *2nd International Workshop on Dependable Network Computing and Mobile Systems (DNCMS 2009)(in Conjunction with IEEE SRDS 2009), New York, USA, 2009*, vol. 27.

- [92] K. Vishnu and A. J. Paul, "Detection and Removal of Cooperative Black/Gray hole attack in Mobile AdHoc Networks," *Int. J. Comput. Appl.*, vol. 1, no. 22, pp. 40–44, 2010.
- [93] J. Kennedy and R. Eberhart, "Particle Swarm Optimization," *Proc. ICNN'95 - Int. Conf. Neural Networks*, vol. 4, pp. 1942–1948, 1995.
- [94] A. Chatterjee, K. Pulasinghe, K. Watanabe, and K. Izumi, "A Particle-swarm-optimized Fuzzy-neural Network for Voice-controlled Robot Systems," *IEEE Trans. Ind. Electron.*, vol. 52, no. 6, pp. 1478–1489, 2005.
- [95] K. Manikantan, A. B. V, and D. K. S. Yaradoni, "Optimal Multilevel Thresholds based on Tsallis Entropy Method using Golden Ratio Particle Swarm Optimization for Improved Image Segmentation," *Procedia Eng. Elsevier*, vol. 30, pp. 364–371, Jan. 2012.
- [96] F. Zhao, J. Tang, J. Wang, and Jonrinaldi, "An Improved Particle Swarm Optimization with Decline Disturbance Index (DDPSO) for Multi-objective Job-shop Scheduling Problem," *Comput. Oper. Res. Elsevier*, vol. 45, pp. 38–50, 2014.
- [97] Y. G. Petalas, K. E. Parsopoulos, and M. N. Vrahatis, "Improving Fuzzy Cognitive Maps Learning Through Memetic Particle Swarm Optimization," *Soft Comput. Springer*, vol. 13, no. 1, pp. 77–94, 2009.
- [98] X. Li, J. Branke, and T. Blackwell, "Particle Swarm with Speciation and Adaptation in a Dynamic Environment," in *Proceedings of the 8th annual conference on Genetic and evolutionary computation - GECCO '06*, ACM Press, 2006, p. 51.
- [99] W. Jatmiko, K. Sekiyama, and T. Fukuda, "A PSO-based Mobile Robot for Odor Source Localization in Dynamic Advection-diffusion with Obstacles Environment: Theory, Simulation and Measurement," *Comput. Intell. Mag. IEEE*, vol. 2, no. 2, pp. 37–51, 2007.
- [100] Q. Bai, "Analysis of Particle Swarm Optimization Algorithm," *Comput. Inf. Sci.*, vol. 3, no. 1, pp. 180–184, 2010.
- [101] Y.-C. Wu, W.-P. Lee, C.-W. Chien, and others, "Modified the Performance of Differential Evolution Algorithm with Dual Evolution Strategy," in *International Conference on Machine Learning and Computing, IPCSIT*, 2011, vol. 3, pp. 57–63.
- [102] R. Storn and K. Price, "Differential Evolution – A Simple and Efficient Heuristic for global Optimization over Continuous Spaces," *J. Glob. Optim. Springer Kluwer Acad. Publ.*, vol. 11, no. 4, pp. 341–359, 1997.
- [103] S. Das and A. Konar, "Automatic Image Pixel Clustering with an Improved Differential Evolution," *Appl. Soft Comput. Elsevier*, vol. 9, no. 1, pp. 226–236, 2009.

- [104] C. Erbao and L. Mingyong, "A Hybrid Differential Evolution Algorithm to Vehicle Routing Problem with Fuzzy Demands," *J. Comput. Appl. Math. Elsevier B.V.*, vol. 231, no. 1, pp. 302–310, 2009.
- [105] U. Maulik and I. Saha, "Modified Differential Evolution Based Fuzzy Clustering for Pixel Classification in Remote Sensing Imagery," *Pattern Recognition, Elsevier Sci. Inc.*, vol. 42, no. 9, pp. 2135–2149, 2009.
- [106] X. Yang, *Nature-Inspired Metaheuristic Algorithms Second Edition*. Luniver Press, 2010, p. 115.
- [107] X. S. Yang, "A New Metaheuristic Bat-inspired Algorithm: Nature Inspired Cooperative Strategies for Optimization," in *NICSO 2010, Springer*, 2010, pp. 65–74.
- [108] X. S. Yang and X. He, "Bat Algorithm: Literature Review and Applications," *Int. J. Bio-Inspired Comput.*, vol. 5, no. 3, p. 141, 2013.
- [109] E. Cuevas and M. Cienfuegos, "A New Algorithm Inspired in the Behavior of the Social-Spider for Constrained Optimization," *Expert Syst. with Appl. Elsevier Ltd*, vol. 41, no. 2, pp. 412–425, 2014.
- [110] I. Fister, D. Fister, and X.-S. Yang, "A Hybrid Bat Algorithm," *Electrotech. Rev. ArXiv*, vol. 80, no. 1–2, pp. 1–7, 2013.
- [111] S. Marwaha, C. K. Tham, and D. Srinivasan, "Mobile Agents Based Routing Protocol for Mobile Ad Hoc Networks," in *Global Telecommunications Conference, 2002. GLOBECOM'02. IEEE*, 2002, vol. 1, pp. 163–167.
- [112] D. Câmara and A. A. F. Loureiro, "A Novel Routing Algorithm for Ad Hoc Networks," in *System Sciences, 2000. Proceedings of the 33rd Annual Hawaii International Conference on*, 2000, p. 8–pp.
- [113] N. Mazhar and M. Farooq, "Vulnerability Analysis and Security Framework (BeeSec) for Nature Inspired MANET Routing Protocols," in *Proceedings of the 9th annual conference on Genetic and evolutionary computation*, 2007, pp. 102–109.
- [114] N. Mazhar and M. Farooq, "BeeAIS: Artificial Immune System Security for Nature Inspired, MANET Routing Protocol, BeeAdHoc," in *Artificial Immune Systems*, Springer, 2007, pp. 370–381.
- [115] L. T. M. Blessing and A. Chakrabarti, *DRM, a Design Research Methodology*. London: Springer London, 2009.
- [116] S. Edelkamp and S. Schroedl, *Heuristic Search: Theory and Applications*. Elsevier Inc, 2012.
- [117] K. Herrmann and M. A. Jaeger, "PayFlux - Secure Electronic Payment in Mobile Ad Hoc Networks," J. Lopez, S. Qing, and E. Okamoto, Eds. Springer Berlin Heidelberg, 2004, pp. 66–78.

- [118] R. Baumann, S. Heimlicher, M. Strasser, and A. Weibel, "A Survey on Routing Metrics," *TIK Rep.*, vol. 262, 2007.
- [119] S. R. Das, R. Castaneda, J. Y. J. Yan, and R. Sengupta, "Comparative Performance Evaluation of Routing Protocols for Mobile Ad Hoc Networks," *Proc. 7th Int. Conf. Comput. Commun. Networks (Cat. No.98EX226)*, pp. 153–161, 1998.
- [120] S. Corson and J. Macker, "Mobile Ad hoc Networking (MANET): Routing Protocol Performance Issues and Evaluation Considerations", RFC 2501," 1999.
- [121] N. A. Husieen, "A Reliable Multipath Dynamic Source Routing Protocol for Multimedia Applications in Mobile Ad-hoc Networks," PhD Thesis, Universiti Utara Malaysia, 2013.
- [122] S. M. Abdule, "Predictive Divert Failure Route Protocol For Mobile Ad-hoc Networks Based on Ad-hoc On Demand Distance Vector," PhD Thesis, Universiti Utara Malaysia, 2012.
- [123] H. Simaremare and R. Sari, "Performance Evaluation of AODV variants on DDOS, Blackhole and Malicious attacks," *Int. J. Comput. Sci. Netowrk Secur.*, vol. 11, no. 6, pp. 277–287, 2011.
- [124] C. Grosan and A. Abraham, *Intelligent Systems*, vol. 17. Springer Berlin Heidelberg, 2011, pp. 131–147.
- [125] W.-K. Chen, *Theory of nets: Flows in networks*. Wiley-Interscience, 1990, p. 493.
- [126] A. E. Wignall and M. E. Herberstein, "Male Courtship Vibrations Delay Predatory Behaviour in Female Spiders.," *Sci. Rep.*, vol. 3, p. 3557, 2013.
- [127] E. Bechinski, D. Schotzko, and C. Baird, "Homeowner Guide to Spiders Around the Home and Yard," 2010.
- [128] J. Kennedy and R. Mendes, "Population Structure and Particle Swarm Performance," in *Proceedings of the 2002 Congress on Evolutionary Computation CEC '02, IEEE*, 2002, vol. 2, pp. 1671–1676.
- [129] C. Grosan, A. Abraham, and M. Chis, *Swarm Intelligence in Data Mining*. Springer Berlin Heidelberg, 2006.
- [130] M. Jamil and X. S. Yang, "A Literature Survey of Benchmark Functions for Global Optimisation Problems," *Int. J. Math. Model. Numer. Optimisation, Inderscience Publ. Ltd*, vol. 4, no. 2, p. 150, 2013.
- [131] X.-S. Yang, *Nature-Inspired Optimization Algorithms*. Elsevier, 2014.
- [132] R. Mitchell and I.-R. Chen, "A Survey of Intrusion Detection in Wireless Network Applications," *Comput. Commun. Elsevier B.V.*, vol. 42, pp. 1–23, 2014.

- [133] S. Horibe, "Robert Hooke, Hooke's Law & the Watch Spring," 2011.
- [134] M. H. Niemz, *Laser-Tissue Interactions*. Springer Berlin Heidelberg, 2007, p. 218.
- [135] G. L. Lucas, F. W. Cooke, and E. A. Friis, *A Primer of Biomechanics*. New York, NY: Springer New York, 1999.
- [136] K. Fall and K. Varadhan, "The Network Simulator (NS-2)," URL [http://www. isi.edu/nsnam/ns](http://www.isi.edu/nsnam/ns), 2007.
- [137] T. Issariyaku and E. Hossain, *Introduction to Network Simulator NS2*. 2009. Springer.
- [138] G. Di Caro and M. Dorigo, "AntNet: Distributed Stigmergetic Control for Communications Networks," *J. Artif. Intell. Res.*, vol. 9, pp. 317–365, 1998.
- [139] H. Huang, H.-B. Xie, J.-Y. Guo, and H.-J. Chen, "Ant Colony Optimization-based Feature Selection Method for Surface Electromyography Signals Classification," *Comput. Biol. Med. Elsevier*, vol. 42, no. 1, pp. 30–8, 2012.
- [140] C. Sarr and I. Guérin-Lassous, "Estimating Average End-to-End Delays in IEEE 802.11 Multihop Wireless Networks," *INRIA TR*, 2007.
- [141] P. Li, Y. Fang, and J. Li, "Throughput, Delay, and Mobility in Wireless Ad Hoc Networks," in *2010 Proceedings IEEE INFOCOM*, 2010, no. 1, pp. 1–9.
- [142] T. Issariyakul and E. Hossain, *Introduction to Network Simulator NS2*. Springer US, 2009.



UUM  
Universiti Utara Malaysia

## Appendix A

### Implementation of Floyd Warshall

(1)							(2)						
	a	b	c	d	e	f		a	b	c	d	e	f
a	Inf	8.2	9.1	Inf	Inf	Inf	a	Inf	8.2	9.1	Inf	Inf	Inf
b	8.2	16.4	7.3	4.6	Inf	Inf	b	8.2	16.4	7.3	4.6	Inf	Inf
c	9.1	7.3	Inf	5.5	6.3	Inf	c	9.1	7.3	18.2	5.5	6.3	Inf
d	Inf	4.6	5.5	Inf	3.6	1.4	d	Inf	4.6	5.5	Inf	3.6	1.4
e	Inf	Inf	6.3	3.6	Inf	2.2	e	Inf	Inf	6.3	3.6	Inf	2.2
f	Inf	Inf	Inf	1.4	2.2	Inf	f	Inf	Inf	Inf	1.4	2.2	Inf

(3)							(4)						
	a	b	c	d	e	f		a	b	c	d	e	f
a	16.4	8.2	9.1	Inf	Inf	Inf	a	16.4	8.2	9.1	12.8	Inf	Inf
b	8.2	16.4	7.3	4.6	Inf	Inf	b	8.2	16.4	7.3	4.6	Inf	Inf
c	9.1	7.3	18.2	5.5	6.3	Inf	c	9.1	7.3	18.2	5.5	6.3	Inf
d	Inf	4.6	5.5	Inf	3.6	1.4	d	Inf	4.6	5.5	Inf	3.6	1.4
e	Inf	Inf	6.3	3.6	Inf	2.2	e	Inf	Inf	6.3	3.6	Inf	2.2
f	Inf	Inf	Inf	1.4	2.2	Inf	f	Inf	Inf	Inf	1.4	2.2	Inf

(5)							(6)						

	a	b	c	d	e	f
a	16.4	8.2	9.1	12.8	Inf	Inf
b	8.2	16.4	7.3	4.6	Inf	Inf
c	9.1	7.3	14.6	5.5	6.3	Inf
d	Inf	4.6	5.5	Inf	3.6	1.4
e	Inf	Inf	6.3	3.6	Inf	2.2
f	Inf	Inf	Inf	1.4	2.2	Inf

	a	b	c	d	e	f
a	16.4	8.2	9.1	12.8	Inf	Inf
b	8.2	16.4	7.3	4.6	Inf	Inf
c	9.1	7.3	14.6	5.5	6.3	Inf
d	12.8	4.6	5.5	Inf	3.6	1.4
e	Inf	Inf	6.3	3.6	Inf	2.2
f	Inf	Inf	Inf	1.4	2.2	Inf

(7)							(8)						
	a	b	c	d	e	f		a	b	c	d	e	f
a	16.4	8.2	9.1	12.8	Inf	Inf	a	16.4	8.2	9.1	12.8	15.4	Inf
b	8.2	16.4	7.3	4.6	Inf	Inf	b	8.2	16.4	7.3	4.6	Inf	Inf
c	9.1	7.3	14.6	5.5	6.3	Inf	c	9.1	7.3	14.6	5.5	6.3	Inf
d	12.8	4.6	5.5	9.2	3.6	1.4	d	12.8	4.6	5.5	9.2	3.6	1.4
e	Inf	Inf	6.3	3.6	Inf	2.2	e	Inf	Inf	6.3	3.6	Inf	2.2
f	Inf	Inf	Inf	1.4	2.2	Inf	f	Inf	Inf	Inf	1.4	2.2	Inf

(9)							(10)						
	a	b	c	d	e	f		a	b	c	d	e	f
a	16.4	8.2	9.1	12.8	15.4	Inf	a	16.4	8.2	9.1	12.8	15.4	Inf
b	8.2	14.6	7.3	4.6	Inf	Inf	b	8.2	14.6	7.3	4.6	13.6	Inf
c	9.1	7.3	14.6	5.5	6.3	Inf	c	9.1	7.3	14.6	5.5	6.3	Inf
d	12.8	4.6	5.5	9.2	3.6	1.4	d	12.8	4.6	5.5	9.2	3.6	1.4
e	Inf	Inf	6.3	3.6	Inf	2.2	e	Inf	Inf	6.3	3.6	Inf	2.2
f	Inf	Inf	Inf	1.4	2.2	Inf	f	Inf	Inf	Inf	1.4	2.2	Inf

(11)							(12)						

	a	b	c	d	e	f
a	16.4	8.2	9.1	12.8	15.4	Inf
b	8.2	14.6	7.3	4.6	13.6	Inf
c	9.1	7.3	14.6	5.5	6.3	Inf
d	12.8	4.6	5.5	9.2	3.6	1.4
e	15.4	Inf	6.3	3.6	Inf	2.2
f	Inf	Inf	Inf	1.4	2.2	Inf

	a	b	c	d	e	f
a	16.4	8.2	9.1	12.8	15.4	Inf
b	8.2	14.6	7.3	4.6	13.6	Inf
c	9.1	7.3	14.6	5.5	6.3	Inf
d	12.8	4.6	5.5	9.2	3.6	1.4
e	15.4	13.6	6.3	3.6	Inf	2.2
f	Inf	Inf	Inf	1.4	2.2	Inf

(13)						
	a	b	c	d	e	f
a	16.4	8.2	9.1	12.8	15.4	Inf
b	8.2	14.6	7.3	4.6	13.6	Inf
c	9.1	7.3	14.6	5.5	6.3	Inf
d	12.8	4.6	5.5	9.2	3.6	1.4
e	15.4	13.6	6.3	3.6	12.6	2.2
f	Inf	Inf	Inf	1.4	2.2	Inf

(14)						
	a	b	c	d	e	f
a	16.4	8.2	9.1	12.8	15.4	14.2
b	8.2	14.6	7.3	4.6	13.6	Inf
c	9.1	7.3	14.6	5.5	6.3	Inf
d	12.8	4.6	5.5	9.2	3.6	1.4
e	15.4	13.6	6.3	3.6	12.6	2.2
f	Inf	Inf	Inf	1.4	2.2	Inf

(15)						
------	--	--	--	--	--	--

(16)						
------	--	--	--	--	--	--

	a	b	c	d	e	f
a	16.4	8.2	9.1	12.8	15.4	14.2
b	8.2	9.2	7.3	4.6	13.6	Inf
c	9.1	7.3	14.6	5.5	6.3	Inf
d	12.8	4.6	5.5	9.2	3.6	1.4
e	15.4	13.6	6.3	3.6	12.6	2.2
f	Inf	Inf	Inf	1.4	2.2	Inf

(17)

	a	b	c	d	e	f
a	16.4	8.2	9.1	12.8	15.4	14.2
b	8.2	9.2	7.3	4.6	8.2	6
c	9.1	7.3	14.6	5.5	6.3	Inf
d	12.8	4.6	5.5	9.2	3.6	1.4
e	15.4	13.6	6.3	3.6	12.6	2.2
f	Inf	Inf	Inf	1.4	2.2	Inf

(18)

	a	b	c	d	e	f
a	16.4	8.2	9.1	12.8	15.4	14.2
b	8.2	9.2	7.3	4.6	8.2	6
c	9.1	7.3	11	5.5	6.3	Inf
d	12.8	4.6	5.5	9.2	3.6	1.4
e	15.4	13.6	6.3	3.6	12.6	2.2
f	Inf	Inf	Inf	1.4	2.2	Inf

(19)

	a	b	c	d	e	f
a	16.4	8.2	9.1	12.8	15.4	14.2
b	8.2	9.2	7.3	4.6	8.2	6
c	9.1	7.3	11	5.5	6.3	6.9
d	12.8	4.6	5.5	9.2	3.6	1.4
e	15.4	13.6	6.3	3.6	12.6	2.2
f	Inf	Inf	Inf	1.4	2.2	Inf

(21)

(20)

	a	b	c	d	e	f
a	16.4	8.2	9.1	12.8	15.4	14.2
b	8.2	9.2	7.3	4.6	8.2	6
c	9.1	7.3	11	5.5	6.3	6.9
d	12.8	4.6	5.5	9.2	3.6	1.4
e	15.4	8.2	6.3	3.6	12.6	2.2
f	Inf	Inf	Inf	1.4	2.2	Inf

(22)

	a	b	c	d	e	f
a	16.4	8.2	9.1	12.8	15.4	14.2
b	8.2	9.2	7.3	4.6	8.2	6
c	9.1	7.3	11	5.5	6.3	6.9
d	12.8	4.6	5.5	9.2	3.6	1.4
e	15.4	8.2	6.3	3.6	7.2	2.2
f	Inf	Inf	Inf	1.4	2.2	Inf

	a	b	c	d	e	f
a	16.4	8.2	9.1	12.8	15.4	14.2
b	8.2	9.2	7.3	4.6	8.2	6
c	9.1	7.3	11	5.5	6.3	6.9
d	12.8	4.6	5.5	9.2	3.6	1.4
e	15.4	8.2	6.3	3.6	7.2	2.2
f	14.2	Inf	Inf	1.4	2.2	Inf

(23)

	a	b	c	d	e	f
a	16.4	8.2	9.1	12.8	15.4	14.2
b	8.2	9.2	7.3	4.6	8.2	6
c	9.1	7.3	11	5.5	6.3	6.9
d	12.8	4.6	5.5	9.2	3.6	1.4
e	15.4	8.2	6.3	3.6	7.2	2.2
f	14.2	6	Inf	1.4	2.2	Inf

(24)

	a	b	c	d	e	f
a	16.4	8.2	9.1	12.8	15.4	14.2
b	8.2	9.2	7.3	4.6	8.2	6
c	9.1	7.3	11	5.5	6.3	6.9
d	12.8	4.6	5.5	9.2	3.6	1.4
e	15.4	8.2	6.3	3.6	7.2	2.2
f	14.2	6	6.9	1.4	2.2	Inf

(25)

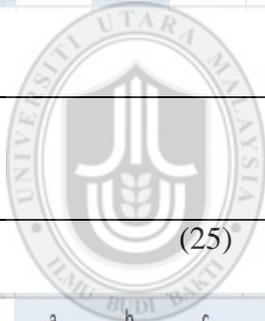
	a	b	c	d	e	f
a	16.4	8.2	9.1	12.8	15.4	14.2
b	8.2	9.2	7.3	4.6	8.2	6
c	9.1	7.3	11	5.5	6.3	6.9
d	12.8	4.6	5.5	9.2	3.6	1.4
e	15.4	8.2	6.3	3.6	7.2	2.2
f	14.2	6	6.9	1.4	2.2	2.8

(26)

	a	b	c	d	e	f
a	16.4	8.2	9.1	12.8	15.4	14.2
b	8.2	9.2	7.3	4.6	8.2	6
c	9.1	7.3	11	5.5	6.3	6.9
d	12.8	4.6	5.5	7.2	3.6	1.4
e	15.4	8.2	6.3	3.6	7.2	2.2
f	14.2	6	6.9	1.4	2.2	2.8

(27)

(28)



UUM  
Universiti Utara Malaysia

	a	b	c	d	e	f
a	16.4	8.2	9.1	12.8	15.4	14.2
b	8.2	9.2	7.3	4.6	8.2	6
c	9.1	7.3	11	5.5	6.3	6.9
d	12.8	4.6	5.5	2.8	3.6	1.4
e	15.4	8.2	6.3	3.6	7.2	2.2
f	14.2	6	6.9	1.4	2.2	2.8

	a	b	c	d	e	f
a	16.4	8.2	9.1	12.8	15.4	14.2
b	8.2	9.2	7.3	4.6	8.2	6
c	9.1	7.3	11	5.5	6.3	6.9
d	12.8	4.6	5.5	2.8	3.6	1.4
e	15.4	8.2	6.3	3.6	4.4	2.2
f	14.2	6	6.9	1.4	2.2	2.8



**UUM**  
 Universiti Utara Malaysia

## Appendix B

### Integration of BAODV, EAODV, SSP-AODV, and PGO-DLLA Protocols to NS-2 Environment

#### Integration of BAODV Protocols to NS-2 Environment

This section presents the implementation of the BAODV protocol in NS-2. The code of the BAODV protocol is installed in the directory of ns2.33/baodv which contains badov.cc, baodv.h, baodv.tcl, baodv\_queue.h, baodv\_queue.cc, and baodv\_packet.h. After that, the two files are changed: BAODV protocol agent and NS-2 makefile. The pseudo code below explains the changes for the BAODV protocol agent.

#### Black Hole Ad hoc On-demand Distance Vector (BAODV) Routing Protocol Agent in Network Simulator NS-2

```
1: Begin
2: BAODV{
   setragent [$self create-baodv-agent $node]
3: Simulator instproc create-baodv-agent{node } {
   setragent [new Agent/BAODV [$node node-addr]]
   $self at 0.0 "$ragent start" // start BEACON/HELLO Messages
   $node set ragent_ $ragent
   return $ragent
}
4: End
```

The BAODV protocol agent pseudo code

the changes for NS-2 makefile to add the BAODV protocol agent as show below.

#### **Adding the BAODV Agent to the MAKEFILE**

**1: Begin**

**2:** `baodv/baodv_logs.obaodv/baodv.o \`

**3:** `baodv/baodv_rtable.obaodv/baodv_rqueue.o \`

**4: End**

To add the behavior of black hole nodes to C++, some changes are needed in the packet reception routines. Pseudo code below, shows the if statement for accepting or dropping the packets.

#### **Adding the Black Hole Behavior**

**1:Begin**

**2:IF** `(u_int32_t)ih->saddr() == index)`

**3:forward**`((baodv_rt_entry*) 0, p, NO_DELAY);`

**4:Else**

**5:Drop**`(p, DROP_RTR_ROUTE_LOOP);`

**6:End**

If statement for accepting or dropping the packets pseudo code

#### **Integration of EAODV Protocol to NS-2 Environment**

This section provides the extended NS-2 to simulate the EAODV protocol. A new agent named eaodv has been implemented. Pseudo code below, shows the changes for the EAODV protocol agent.

## **Enhanced Ad hoc On-demand Distance Vector (EAODV) Routing Protocol Agent in Network Simulator NS-2**

### **1:Begin**

**2:EAODV** {*setragent* [*\$self create-eaodv-agent \$node*]}

**3:Simulator instproc create-eaodv-agent** { *node* } {

**4:***setragent* [*new Agent/EAODV* [*\$node node-addr*]]

**5:***\$self at 0.0 "\$ragent start" // start BEACON/HELLO Messages*

**6:***\$node set ragent\_ \$ragent*

**7:return** *\$ragent* }

### **8:End**

The EAODV protocol agent pseudo code

## **Configuration and Installation of EAODV Protocol to NS-2**

The EAODV routing protocol is implemented using high level language C++ to be compiled inside the NS-2 network simulator. Inside the directory of NS-2.33, a new folder eaodv is created to input the code to this folder which contains all the header files and C++ files such as eaodv.cc and eaodv.h. Some editing in the main files on NS-2 is needed to create a packet type and declare its contents. Pseudo code below, shows the changes for NS-2 make file to add the EAODV protocol agent.

### **Adding the EAODV Agent to the MAKEFILE**

#### **1:Begin**

**2:** *eaodv/baodv\_logs.o**eaodv/eaodv.o \*

**3:***eaodv/eaodv\_rtable.o**eaodv/eaodv\_rqueue.o \*

#### **4:End**

The EAODV protocol make file agent pseudo code

To add the behavior of black hole nodes to C++, some changes are needed in the packet reception routines. Pseudo code below, shows the if statement for accepting or dropping the packets.

### **Adding the Black Hole Behavior**

#### **1:Begin**

**2:IF**( *(u\_int32\_t)ih->saddr() == index*)

**3:Forward**((*eaodv\_rt\_entry\**) 0, *p*, *NO\_DELAY*);

#### **4:Else**

**5:Drop**(*p*, *DROP\_RTR\_ROUTE\_LOOP*);

#### **6:End**

### **Integration of SSP-AODV Protocol to NS-2 Environment**

This section presents the implementation of the SSP-AODV protocol in NS-2. The code of the SSP-AODV protocol is installed in the directory of ns2.33/sspaodv which contains sspaodv.cc, sspaodv.h, sspaodv.tcl, sspaodv\_queue.h, sspaodv\_queue.cc and

sspaodv\_packet.h. After that two files are changed: SSP-AODV protocol agent and NS-2 makefile. Pseudo code below, shows the changes for the SSP-AODV protocol agent.

### Shortest Secure Path for Ad hoc On-demand Distance Vector (SSP-AODV) Routing Protocol Agent in Network Simulator NS-2

**1:Begin**

**2:SSP-AODV**

```
{  
    setragent [$self create-sspaodv-agent $node]      }  
3:Simulator instproc create-sspaodv-agent{node } {  
    setragent [new Agent/SSPAODV [$node node-addr]]  
    $self at 0.0 "$ragent start" // start BEACON/HELLO Messages  
    $node set ragent_ $ragent  
    return $ragent  
}
```

**4: End**

The SSP-AODV protocol agent Pseudo code

Pseudo code below, shows the changes for NS-2 makefile to add the SSP-AODV protocol agent.

### **Adding the SSP-AODV Agent to the MAKEFILE**

#### **1:Begin**

**2:***sspaodv/sspaodv\_logs.o* *sspaodv/sspaodv.o* \

**3:***sspaodv/sspaodv\_rtable.o* *sspaodv/sspaodv\_rqueue.o* \

#### **4:End**

The SSP-AODV protocol make file agent pseudo code

To add the behavior of black hole nodes to C++, some changes are needed in the packet reception routines. Pseudo code below, shows the if statement for accepting or dropping the packets.

### **Adding the Black Hole Behavior to SSP-AODV Protocol**

#### **1:Begin**

**2:IF**( *(u\_int32\_t)ih->saddr() == index* )

**3:Forward**(*(sspaodv\_rt\_entry\*) 0, p, NO\_DELAY*);

#### **4:Else**

**5:Drop**(*p, DROP\_RTR\_ROUTE\_LOOP*);

#### **6:End**

The SSP-AODV protocol if statement for accepting or dropping the packets pseudo code

### **Integration of PGO-DLLA Protocol to NS-2 Environment**

This section presents the implementation of the PGO-DLLA protocol in NS-2. The code of the PGO-DLLA protocol is installed in the directory of ns2.33/pgo-dlla which contains *pgodlla.cc*, *pgodlla.h*, *pgodlla.tcl*, *pgodlla\_queue.h*, *pgodlla\_queue.cc* and *pgodllav\_packet.h*. After that two files are changed: PGO-DLLA protocol agent and

NS-2 makefile. Pseudo code below, shows the change for the PGO-DLLA protocol agent.

**Parallel Grid Optimization based on Virtual Daddy Long-legs Algorithm Routing Protocol Agent in Network Simulator NS-2**

**1:Begin**

**2:PGO-DLLA{**

```
    setragent [$self create-pgodlla-agent $node]      }
```

**3:Simulator instproc create-pgodlla-agent{node } {**

```
    setragent [new Agent/PGODLLA [$node node-addr]]  
    $self at 0.0 "$ragent start" // start BEACON/HELLO Messages  
    $node set ragent_ $ragent  
    return $ragent
```

**}**

**4:End**

The PGO-DLLA agent pseudo code

**Adding the PGO-AODV Agent to the MAKEFILE**

**1:Begin**

**2:** *pgodlla/pgodlla\_logs.opgodlla/pgodlla.o \*

**3:** *pgodlla/pgodlla\_rtable.opgodlla/pgodlla\_rqueue.o \*

**4:End**

Pseudo code below, shows the changes for NS-2 make file to add the PGO-DLLA protocol agent.

To add the behavior of black hole nodes to C++, some changes are needed in the packet reception routines. Pseudo code below, shows the if statement for accepting or dropping the packets.

#### **Adding the Black Hole Behavior to PGO-DLLA Protocol**

**1:Begin**

**2:IF** (*u\_int32\_t*)*ih->saddr() == index*)

**3:Forward**((*pgodlla\_rt\_entry\**) 0, *p*, *NO\_DELAY*);

**4:Else**

**5:Drop**(*p*, *DROP\_RTR\_ROUTE\_LOOP*);

**6:End**

The PGO-DLLA if statement for accepting or dropping the packets pseudo code

## Appendix C

### Some Important Modification in NS-2 Files

There are some important NS-2 files that need to be modified on its directory to integrate the new protocols, such as packet header in the common folder as shown in pseudo code below.

**The File Name: common/packet.h**

**1:Begin**

*//Edit packet.h by adding a packet type PT\_NANE as following//*

*// insert new packet types here//*

**2:** *static constpacket\_t PT\_BAODV=62;*

**3:** *static constpacket\_t PT\_EAODV=63;*

**4:** *static constpacket\_t PT\_SSPAODV=64;*

**5:** *static constpacket\_t PT\_PGOAODV=65;*

**6:End**

Adding the packet header in common folder pseudo code

Pseudo code below, show the changes in the trace folder to add a new function for the new protocol to define the trace format and the case statement in cmu-trace.cc file.

**The File Name: trace/cmu-trace.h**

**1:Begin**

```
// Edit cmu-trace.h by adding a new function to define the trace format
for the new protocol as following //
```

**2:** *void format\_baadv(Packet \*p, int offset);*

**3:** *void format\_eaadv(Packet \*p, int offset);*

**4:** *void format\_sspaadv(Packet \*p, int offset);*

**5:** *void format\_pgoadv(Packet \*p, int offset);*

**6:End**

Adding new function to define the trace format pseudo code

**The File Name: trace/cmu-trace.cc**

**1:Begin**

**2:***case PT\_BAODV:format\_baadv(p, offset)*

*break;*

**3:***case PT\_EAODV:format\_eaadv(p, offset);*

*break;*

**4:***case PT\_SSPAODV:format\_sspaadv(p, offset);*

*break;*

**5:***case PT\_PGODLLA:format\_pgodlla(p, offset);*

*break;*

**6: End**

Adding the new protocol in trace folder pseudo code

## Appendix D

### Generated Input and Output files

Network simulator NS-2 is one of many simulations which are used to simulate ad hoc network scenarios. NS-2 is an open source simulator and it is an event driven strategy. It has two main languages: the object oriented TCL language and C++ language. The OTCL language is for writing a simulation script and it has an interpreter for translation and runs step by step.

C++ language has the ability to update an existing protocol or develop a new one. The two languages are fully compatible. Writing a script and setting up the simulation variables are done by OTCL. Before the NS-2 starts a compilation, two files should be input. The first input file is a random traffic connection of CBR.

It is available under `~home/indep_utils/cmu_sen_gen/cbrgen`. Pseudo code below, shows an example to create a CBR command under Linux Ubuntu 10.04 operating system.

```
#ns cbrgen.tcl -type -cbr -nn 50 -seed 1 -mc 10 -rate 2.0> cbr-50-10-2
```

- *Generate 50 CBR connection randomly*
- *Number of nodes 50*
- *Packet size 512 Byte*
- *At rate of 8 kbps*
- *Maximum connection of 10*

The generation of random traffic for NS-2 scenario pseudo code

The second file is the movement file or scenario file that is used to generate node movement for wireless scenario. It is available on `~home /indep_utils /cmu_sen_gen /setdest`.

Pseudo code below, shows an example to create a scenario file under Linux Ubuntu 10.04 operating system.

```
#ns setdest -v2 -n20 -s1 -m1 -M 10.0 -t500 -p10 -p2.0 -x 750 -y 750 > sen1-20
```

- *Number of nodes 20*
- *Minimum speed as 1*
- *Maximum speed as 10.0*
- *Maximum time 500*
- *Pause time 10*

The generated of node movements for NS-2 scenario

Then, the NS-2 will output two files: Nam file and trace file. These are used to visualize the NS-2 simulations and register all the node movements during the simulation running.

At least one trace file and one animation file are produced with one run of a scenario. The trace file contains all the events that occurred during the simulation running, such as sending or receiving packets by a node or a type of package that is sent or any other nodes that has been dropped.

In order to obtain the results of the trace file, AWK commands are used to cut and move some of these results to another file to compare them with previous results or convert them into graphs.

Pseudo code below, shows an example to run a graph file under Linux Ubuntu 10.04 operating system.

### **Some Commands for Trace File Analysis**

#### **1: AWK File**

```
#ns awk -f Mymetric.awk outtrace-sen1.tr
```

#### **2: XGRAPH File**

```
#ns xgraph outtrace-sen1.tr -geometry 800x400 -t "Average Delay" -x "Pause Time" -y  
"Delay" -bg white
```

The AWK and xgraph for trace file analysis pseudo code

## Appendix E

### Example of TCL, AWK script and graph file to obtain the result of the Packet Delivery Ratio (PDR) for PGO-DLLA, BAODV, and AODV protocols (see Figure 5.19, pp: 138)

```
set val(chan) Channel/WirelessChannel
set val(prop) Propagation/TwoRayGround
set val(netif) Phy/WirelessPhy
set val(mac) Mac/802_11
set val(ifq) Queue/DropTail/PriQueue
set val(ant) Antenna/OmniAntenna
set val(ll) LL
set val(ifqlen) 150
set val(nn) 50
set val(nnaodv) 49
set val(rp) AODV
set val(brp) blackholeAODV
set val(X) 800
set val(Y) 800
set val(cstop) 450
set val(stop) 1000
set val(cp) "blackhole50"
#set val(cc) "scenarios/cbr-blackhole50"
# Initialize Global Variables
set ns_ [new Simulator]

set tracefd [open Blackhole50.tr w]
$ns_ trace-all $tracefd
set namtrace [open Blackhole50.nam w]
$ns_ namtrace-all-wireless $namtrace $val(X) $val(Y)
#set up top
set topo [new Topography]
$topo load_flatgrid $val(X) $val(Y)
# Create God
set god_ [create-god $val(nn)]
#create channel
set chan_1_ [new $val(chan)]
set chan_2_ [new $val(chan)]
#configure nodes
$ns_ node-config -adhocRouting $val(rp) \
                 -llType $val(ll) \
```

```

        -macType      $val(mac) \
        -ifqType      $val(ifq) \
        -ifqLen       $val(ifqlen) \
        -antType      $val(ant) \
        -propType     $val(prop) \
        -phyType      $val(netif) \
        -topoInstance $topo \
        -agentTrace   ON \
        -routerTrace  ON \
        -macTrace     ON \
        -movementTrace ON \
        -channel      $chan_1_

# create nodes
for {set i 0} {$i $val(nnaodv)} {incr i} {
    set node_($i) [$ns_ node]
    $node_($i) random-motion 0;
}

$ns_ node-config -adhocRouting $val(brp)
for {set i $val(nnaodv)} {$i<$val(nn)} {incr i} {
    set node_($i) [$ns_ node]
    $node_($i) random-motion 0;
    $ns_ at 0.01 "$node_($i) label \"Blackhole Node\""
}

puts "loading random connection pattern..."
source $val(cp)

set j 0
for {set i 0} {$i<18} {incr i} {
    set udp_($j) [new Agent/UDP]
    $ns_ attach-agent $node_($i) $udp_($j)
    set null_($j) [new Agent/Null]
    $ns_ attach-agent $node_([expr $i+1]) $null_($j)

    set cbr_($j) [new Application/Traffic/CBR]
    puts "cbr_($j) has been created over udp_($j)"
    $cbr_($j) set packet_size_ 512
    $cbr_($j) set interval_ 1
    $cbr_($j) set rate_ 10kb
    $cbr_($j) set ransom_ flase
    $cbr_($j) attach-agent $udp_($j)
    $ns_ connect $udp_($j) $null_($j)
    puts "$udp_($j) and $null_($j) agents has been
connected each other"
    $ns_ at 1.0 "$cbr_($j) start"

    set j [expr $j+1]
    set i [expr $i+1]
}

```

```

#Define initial node position
for { set i 0} {$i<$val(nn)} {incr i} {
    $ns_ initial_node_pos $node_($i) 30
}
for {set i 0} {$i<9} {incr i} {
    $ns_ at $val(cstop) "$cbr_($i) stop"
}
for {set i 0} {$i<$val(nn)} {incr i} {
    $ns_ at $val(stop).000000001 "$node_($i) reset";
}

$ns_ at $val(stop) "finish"
$ns_ at $val(stop).0 "$ns_ trace-annotate \"Simulation has
ended\""
$ns_ at $val(stop).00000001 "puts \"NS exiting...\""; $ns_
halt"
#*****
proc PGO{} {
    bestpath();
    setragent [$self create-pgodlla-agent $node]
}
Simulator instproc create-pgodlla-agent{node } {
setragent [new Agent/PGODLLA [$node node-addr]]
$self at 0.0 "$ragent start" // start BEACON/HELLO Messages
$node set ragent_ $ragent
return $ragent
}

num_of_neighbors(sol);
sol=random(1,d);
body(x,y);
f(j)=fnn(s);
fmin(i)=min(f)
FILE *fp = fopen(NB_TRACE_FILE, "a+");

struct gpsr_neighbor *temp = head_;
fprintf(fp, "%d:\t", my_id_);
listmin=node
while(node){
    select(path)
    node = node->next_;
    break; {

if( (u_int32_t)ih->saddr() == index)
forward((pgodlla_rt_entry*) 0, p, NO_DELAY);
else
drop(p, DROP_RTR_ROUTE_LOOP);
    dele(path);
}
fprintf(fp, "\n");
fclose(fp);

```

```

    pgo=bestpath()
}

# gather neighbour information
proc num_of_neighbors {struct gpsr_neighbor *nblast} {
    struct gpsr_neighbor *temp = nblast;
    int counter = 0;
    while(temp){
        counter++;
        temp = temp->next_;
    }
    return counter;
}

#multichannel process
proc MASP_list {aadv_rt_entry *rt} {
    int num_non_zero = 0, i;
    double total_latency = 0.0;

    if (!rt)
        return ((double) node_distance_move );

    for (i=0; i < MAX_HISTORY; i++) {
        if (rt->rt_disc_latency[i] > 0.0) {
            num_non_zero++;
            total_latency += rt->rt_disc_latency[i];
        }
    }
    if (num_non_zero > 0)
        return(total_latency / (double) num_non_zero);
    else
        return((double) node_distance_move );
}

}

proc finish {} {
    global ns_tracefd namtrace
    $ns_flush-trace
    close $tracefd
    close $namtrace
    # exec nam bLACKhole50.nam &
    exit 0
}

puts "Starting simulation..."
$ns_run

#*****
#**** AWK script *****
#*****
BEGIN {
    sends=5;

```

```

        recvs=5;
        routing_packets=0.0;
        droppedBytes=0;
        droppedPackets=0;
        highest_packet_id =0;
        sum=-1;
        recvnum=0;
    }

    {
        time = $3;
        packet_id = $41;

        if ( ( $1 == "s" ) && ( $35 == "cbr" ) && ( $19=="AGT" ) ) {
            sends++; }

        if ( ( $1 == "r" ) && ( $35 == "cbr" ) && ( $19=="AGT" ) ) {
            recvs++; }

        if ( start_time[packet_id] == 0 ) start_time[packet_id] =
            time;
        if ( ( $1 == "r" ) && ( $35 == "cbr" ) && ( $19=="AGT" ) ) {
            end_time[packet_id] = time; }
        else { end_time[packet_id] = -1; }

        if ( ( $1 == "s" || $1 == "f" ) && $19 == "RTR" && $35
            == "AODV" ) routing_packets++;

        if ( ( $1 == "d" ) && ( $35 == "cbr" ) && ( $3 > 0 ) )
        {
            droppedBytes=droppedBytes+$37;
            droppedPackets=droppedPackets+1;
        }

        #find the number of packets in the simulation
        if ( packet_id > highest_packet_id )
            highest_packet_id = packet_id;
    }

END {

for ( i in end_time )
{
start = start_time[i];
end = end_time[i];
packet_duration = end - start;
if ( packet_duration > 0 )
{
    sum += packet_duration;
    recvnum++;
}
}
}

```

```

        delay=sum/recvnum;
        PDR = (recvs/sends)*100; #packet delivery
ratio[fraction]
        printf("Simulation Report\n");
        printf("*****\n");
        printf("send = %.2f\n",sends);
        printf("recv = %.2f\n",recvs);
        printf("PDF = %.2f\n",PDF+.30);
        printf("*****\n");
    }

```

```

#-----
# TitleText: "Packet Loss Results of PGO-DLLA, AODV and BAODV"
#-----

```

```

Device: Postscript
BoundingBox: true
Ticks: true
NoLines: true
Markers: true
XUnitText: Time
YUnitText: PL

```

```

"PGO-DLLA"
10. 000000000000000018 99.1
20.000000000000000022 100
40. 000000000000000010 99.0
60. 000000000000000016 99.5
80. 000000000000000018 100
100. 000000000000000001 99.4

```

```

"BAODV"
10. 000000000000000018 95.5
20. 000000000000000022 91.5
40. 000000000000000010 96
60. 000000000000000016 93
80. 000000000000000018 95
100. 000000000000000001 94

```

```

"AODV"
10. 000000000000000018 100
20. 000000000000000022 101
40. 000000000000000010 95.5
60. 000000000000000016 95
80. 000000000000000018 94.8
100. 000000000000000001 94.8

```