
An Adaptive Approach for Optimized Opportunistic Routing Over Delay Tolerant Mobile Ad Hoc Networks

Thesis

By: Xiaogeng Zhao



Submitted in fulfilment of
the requirements for the degree of
Doctor of Philosophy at Rhodes University

Supervisor: Professor Peter Clayton
Co-supervisor: Professor George Wells

Computer Science Department,
December 2007



RHODES UNIVERSITY
Where leaders learn

Abstract

This thesis presents a framework for investigating opportunistic routing in Delay Tolerant Mobile Ad hoc Networks (DTMANETs), and introduces the concept of an Opportunistic Confidence Index (OCI). The OCI enables multiple opportunistic routing protocols to be applied as an adaptive group to improve DTMANET routing reliability, performance, and efficiency.

The DTMANET is a recently acknowledged network architecture, which is designed to address the challenging and marginal environments created by adaptive, mobile, and unreliable network node presence. Because of its ad hoc and autonomic nature, routing in a DTMANET is a very challenging problem. The design of routing protocols in such environments, which ensure a high percentage delivery rate (reliability), achieve a reasonable delivery time (performance), and at the same time maintain an acceptable communication overhead (efficiency), is of fundamental consequence to the usefulness of DTMANETs.

In recent years, a number of investigations into DTMANET routing have been conducted, resulting in the emergence of a class of routing known as opportunistic routing protocols. Current research into opportunistic routing has exposed opportunities for positive impacts on DTMANET routing. To date, most investigations have concentrated upon one or other of the quality metrics of reliability, performance, or efficiency, while some approaches have pursued a balance of these metrics through assumptions of a high level of global knowledge and/or uniform mobile device behaviours. No prior research that we are aware of has studied the connection between multiple opportunistic elements and their influences upon one another, and none has demonstrated the possibility of modelling and using multiple different opportunistic elements as an adaptive group to aid the routing process in a DTMANET.

This thesis investigates OCI opportunities and their viability through the design of an extensible simulation environment, which makes use of methods and techniques such as abstract modelling, opportunistic element simplification and isolation, random attribute generation and assignment, localized knowledge sharing, automated scenario generation, intelligent weight assignment and/or opportunistic element permutation. These methods and techniques are incorporated at both data acquisition and analysis phases.

Our results show a significant improvement in all three metric categories. In one of the most applicable scenarios tested, OCI yielded a 31.05% message delivery increase (reliability improvement), 22.18% message delivery time reduction (performance improvement), and 73.64% routing depth decrement (efficiency improvement). We are able to conclude that the OCI approach is feasible across a range of scenarios, and that the use of multiple opportunistic elements to aid decision-making processes in DTMANET environments has value.

Acknowledgements

To my supervisor Professor Peter Clayton and co-supervisor Professor George Wells for their wise guidance and invaluable advices throughout the course of this work, for always keeping their doors open to me, willing to listen and help in any way possible. Above all for having faith in me for all these years. To my parents, without whom I could not possibly have gone this far. It is your love that gives me strength, a clear mind and determination that help me to overcome all difficulties. Although we are ten thousand miles apart, my heart has never left you. To the treasure of my life, my dear wife Johanna, for all the love and support she gives me. It is simply a blessing to be with you! To my good friends Fong, Kevin, James, Tim, Melekam, Jo, Lorenzo and Ellen for all the good time we spent together. To all the proofreaders, Ingrid, Uil, Kevin, Melekam, and Madeleine, thank you for the help! To Cheryl, Billy and Caro, for your support!

Funding

This work was undertaken in the Distributed Multimedia Centre of Excellence (CoE) at Rhodes University, with financial support from Telkom, Business Connexion, Comverse, Tellabs, StoreTech, Dimension Data, and THRIP. We also acknowledge the bursary support of the Andrew Mellon Foundation, National Research Foundation (NRF) and the Chinese Government.

Author's Declaration

I declare that the work in this dissertation was carried out in accordance with the regulations of Rhodes University. The work is original except where indicated by special reference in the text and no part of the dissertation has been submitted for any other degree. Any views expressed in the dissertation are those of the author and in no way represent those of Rhodes University. The dissertation has not been presented to any other university for examination either in the South Africa or overseas.

SIGNED: _____

DATE: December 14, 2007

"What you get by achieving your goals is not as important as what you become by achieving your goals."

-Zig Ziglar

Table of Contents

Chapter I: Introduction	1
1.1 General Introduction	2
1.1.1 Background.....	2
1.1.2 Field of Relevance	3
1.2 Project introduction	5
1.2.1 Motivation.....	5
1.2.2 Problem Domain and Field Scope	5
1.2.3 Research Objectives and Expected Contributions	7
1.2.4 Research Procedures and Experimental Approach	7
1.3 Thesis Organization	9
Chapter II: Literature Survey	10
2.1 DTMANET – A General Introduction	10
2.1.1 A Brief History of Message Oriented Networks	11
2.1.2 The Application of DTMANETs.....	13
2.1.3 “Wizzy Digital Courier” – A Current Example.....	15
2.2 Research on Delay Tolerant Networks	16
2.2.1 Characteristics of Delay Tolerant Networks.....	16
2.2.2 DTMANET- Delay Tolerant Mobile Ad Hoc Networks.....	16
2.2.3 DTMANET – The Diversity.....	17
2.2.4 Challenges in DTMANET Research.....	18
2.3 DTMANET Routing Mechanisms	19
2.3.1 DTMANET Routing Design Consideration	19
2.3.2 DTMANET Routing vs. MANET Routing	20
2.3.3 DTMANET Foundation Routing - Stateless Routing	22

2.3.3.1 Stateless Routing Algorithm	22
2.3.3.2 “Epidemic Routing”	23
2.3.4 Probabilistic Routing – Global Knowledge Based	24
2.3.4.1 Probabilistic Routing	24
2.3.4.2 Probabilistic Routing Classification	24
2.3.5 Opportunistic Routing – Local-Knowledge Based	26
2.4 Related Work and Open Issues	27
2.4.1 Related Work	27
2.4.2 Open Issues and the Research Focus of OCI	28
2.5 Network Routing Simulation	29
2.5.1 Existing Simulators	30
2.5.2 OCI-SIM	31
2.6 Chapter Review	32
Chapter III: Research Approach	33
3.1 The Concept of Opportunistic Confidence Index	33
3.2 The Network Simulation Approach	34
3.3 Assumptions	35
3.4 OCI Network Evaluation Methods	37
3.5 Design of the OCI Simulation Environment	39
3.5.1 Design Rationale	39
3.5.1.1 The perspective of Time in OCI	39
3.5.1.2 The Spatial Perspective in OCI	41
3.5.1.3 Timer-Controlled Spatial Alteration Method	42
3.5.1.4 Simplified Simulation Environment	42
3.5.2 OCI System Model Overview	42

3.5.2.1 The OCI Simulation Environment:	43
3.5.2.2 Movement Models in OCI	44
3.5.2.3 DTMANET Routing Models	47
3.5.3 DTMANET Scenario Dynamic Element Generation	52
3.5.3.1 Dynamic Scenario Recreation Methods	52
3.5.3.2 PRNG Used in OCI	53
3.5.3.3 Randomly Generated Dynamic Attributes in OCI	54
3.6 OCI Confidence Index Generating Method	56
3.6.1 Opportunistic Element Selection	56
3.6.2 One-to-one Opportunistic Filter Comparison	57
3.6.3 Opportunistic Filter Combination and Permutation	58
3.6.3.1 Filter Combination	58
3.6.3.2 Filter Permutation	58
3.6.4 Weight Assignment & Voting Procedure	60
3.6.4.1 Non-intelligent Weight Assignment and Voting methods	60
3.6.4.2 Intelligent Weight Assignment and Voting methods	61
3.6.4.3 Permutation Number Limitations	66
3.7 OCI Simulation Validity Control Mechanisms	67
3.7.1 Routing Loop Control Method	67
3.7.2 Statistic control	68
3.7.3 Benchmarking control	69
3.7.3.1 Theoretical value based benchmarking control	69
3.7.3.2 Routing Protocol based benchmarking control	70
3.7.4 Optimization control	70
3.8 Chapter Review	71
Chapter IV: The OCI Simulation Tool Set	72

4.1 DTMANET SIMULATION	72
4.1.1 Dynamic Heterogeneity.....	72
4.1.2 DTMANET Simulation Tools.....	72
4.1.3 OCI-SIM – an Alternative Approach	73
4.2 OCI-SIM Design	75
4.2.1 OCI-AUTO	75
4.2.1.1 OCI-AUTO Design Approach	75
4.2.1.2 OCI-AUTO sub-components.....	76
1) OCI-AUTO Scenario Designer.....	76
4.2.1.3 OCI-AUTO Execution Mode.....	81
4.2.2 OCI-CUSTOM.....	81
4.2.2.1 OCI-CUSTOM Design philosophy.....	81
4.2.2.2 OCI-CUSTOM sub-components.....	82
4.2.3 OCI-VISUAL.....	84
4.2.3.1 OCI-VISUAL Design philosophy.....	85
4.2.3.2 OCI-VISUAL sub-components.....	85
4.2.4 OCI-GRAPH.....	87
4.2.4.1 OCI-GRAPH Design philosophy.....	87
4.2.4.2 OCI-GRAPH sub-components.....	87
4.3 A Taxonomy of OCI-SIM Simulation Modes	88
4.3.1 Taxonomy by user intervention level	88
4.3.1.1 Fully automatic simulation mode.....	88
4.3.1.2 Semi-automatic simulation mode.....	89
4.3.1.3 Single simulation mode	89
4.3.1.4 Interactive single simulation mode	89
4.3.2 Taxonomy by simulation randomness level.....	90
4.3.2.1 Theoretical static simulation.....	90
4.3.2.2 Random element isolation simulation.....	90
4.3.2.3 Dynamic diversity full randomness simulation.....	91

4.3.3 Taxonomy by filter organization.....	91
4.3.3.1 Isolated opportunistic filter simulation	91
4.3.3.2 Combined opportunistic filter simulation.....	92
4.3.3.3 Permuted opportunistic filter simulation.....	92
4.3.3.4 Intelligent weighted opportunistic filter simulation.....	92
4.4 Overall Evaluation of OCI-SIM Design.....	92
4.5 Chapter Review	94
Chapter V: Data Classification & Validity Control.....	95
5.1 Data Composition and Classification	95
5.1.1 Initial input data classification.....	95
5.1.1.1 Peer data	96
5.1.1.2 Message data	99
5.1.1.3 Global statistical Data.....	102
5.1.2 Control Data Classification.....	103
5.1.2.1 Movement Model data.....	104
5.1.2.2 Filter data	105
5.1.3 Output Data and Data Evaluation Classification	107
5.2 OCI Simulation Scenario Design and Construction.....	108
5.2.1 Simulation Scenario – Representation of DTMANET.....	109
5.2.2 DTMANET Scenarios – A Coarse-grained Taxonomy.....	109
5.2.2.1 Peer Conformity Based Taxonomy	110
5.2.2.2 Priority Preference Based Taxonomy.....	111
5.2.2.3 Spatial-constraint Based Taxonomy.....	112
5.2.3 Quantitative Definition of Relative Terms	113
5.2.3.1 The Density Measurement Unit	113
5.2.3.2 The minimum threshold – Absolute “Low” density	114
5.2.3.3 The maximum threshold – Absolute “High” density.....	114

5.2.3.4 Relative “Low” and Relative “High”	115
5.2.4 OCI Simulation Procedure	117
5.2.4.1 Random assignment of Opportunistic elements	117
5.2.4.2 OCI Auto Simulation Procedure	118
5.3 OCI Simulation Validity Control Procedure	120
5.3.1 PRNG Validity Control	121
5.3.2 Define Relative terms	122
5.3.3 Statistical Validity Control	125
5.3.4 TTL and HTL Validity Control	126
5.3.5 Base Routing Scenario Classification:	129
5.4 Chapter Review	131
Chapter VI: OCI Data Analysis & Evaluation	132
6.1 Opportunistic Elements and Their Impacts on DTMANETS	132
6.1.1 “Peer Velocity” and its effects on message delivery	133
6.1.1.1 Effects on message delivery time	133
6.1.1.2 Effects on Routing depth and Message duplication	135
6.1.2 “Peer movement territory” and its effects on message delivery	137
6.1.2.1 Effects on message delivery rate	137
6.1.2.2 Effects on message delivery time	139
6.1.2.3 Effects on message routing depth	140
6.1.2.4 Effects on message duplication	141
6.1.3 Other Elements and Their Effects on message delivery	142
6.2 Opportunistic Element Importance Index	142
6.3 Modelling Opportunistic Elements as Individual Filters	143
6.3.1 Filters and Their Effects on Message Delivery	143
6.3.1.1 Applying Opportunistic Filters to “Custody Transfer”	143

6.3.1.2 Applying Filters to “Custody Transfer Propagation”	146
6.3.1.3 Applying Filters to “Custody Retaining Propagation”	150
6.3.1.4 Applying Filters to “Simple Flood Propagation”	153
6.3.2 OCI Weight Metrics assignment	157
6.3.2.1 OCI Index Weight Assignment for “Custody Transfer” Base Protocol	157
6.3.2.2 OCI Index Weight Assignment for Other Base Protocol	162
6.4 Testing OCI indexes	162
6.4.1 Test OCI Index over “Custody Transfer” Base Routing	162
6.4.2 Test OCI over “Custody Transfer Propagation” Base Routing	167
6.4.3 Test OCI over “Custody Retaining Propagation” Base Routing	171
6.4.4 Test OCI over “Simple Flood Propagation” Base Routing	175
6.5 Data Analysis, Evaluation and Conclusion	179
6.6 Chapter Review	181
Chapter VII Conclusion and Future Work	182
7.1 Thesis and Findings Review	182
7.2 Contributions	184
7.2.1 Proposed and Proved a Concept	184
7.2.2 Establishment of a Framework	184
7.2.3 Development of a Proof of Concept System	185
7.3 Critical Analysis of the Investigation and Findings	185
7.3.1 Selection of the Test Environment	185
7.3.2 Choice of Behavioural Pattern Modelling	186
7.3.3 Representation of DTMANET	187
7.3.4 Choice of Opportunistic Elements	188
7.3.5 Choices of Non-Opportunistic Base Routings	188
7.3.6 Simplified Scenario Modelling	189

7.3.7 Output Data Evaluation Benchmarks	189
7.3.8 Computational Cost of OCI Determination and Routing.....	190
7.4 Future Research.....	191
7.4.1 Realistic Testing Environment	191
7.4.2 A Comprehensive Opportunistic Taxonomy.....	191
7.4.3 Multi-Layered Time-Divided Routing Visualization	191
7.4.4 Multiple Attribute Decision Making System	192
7.5 Concluding Comments	193
Bibliography	194
Appendix A: OCI Companion DVD	209
Appendix B: OCI-SIM Structure Organization.....	210
Appendix C: Weight Assignment.....	211
Appendix D: Element Selection	216
Appendix E: Element Importance Index	232
Appendix F: Extreme High Density Scenario.....	238
Appendix G: Hardware and Software Environment.....	241

List of Tables

TABLE 3-1: OCI FILTER COMBINATION POSSIBILITIES	66
TABLE 3-2: OCI FILTER PERMUTATION POSSIBILITIES	66
TABLE 3-3: VALIDITY CONTROL (AVERAGE DELIVERY DUPLICATION MESSAGE)	69
TABLE 5-1: EXAMPLE SCENARIO CONFIGURATION	121
TABLE 5-2: AVERAGE DELIVERY STATUS (1 TO 400 SIMULATION REPEATS).....	125
TABLE 5-3: TTL / HTL OPTIMIZATION	129
TABLE 6-1: “CUSTODY TRANSFER” SIMULATION RESULTS.....	158
TABLE 6-2: IMPROVEMENT / TRADEOFFS “CUSTODY TRANSFER”	158
TABLE 6-3: ABSOLUTE OCI INDEX - “CUSTODY TRANSFER”	159
TABLE 6-4: EMPHASIZED BALANCING OCI INDEX - “CUSTODY TRANSFER”	160
TABLE 6-5: BALANCED OCI INDEX - “CUSTODY TRANSFER”	161
TABLE 6-6: SIMULATION VALUE – “CUSTODY TRANSFER”	166
TABLE 6-7: IMPROVEMENT/TRADEOFFS – “CUSTODY TRANSFER”.....	166
TABLE 6-8: SIMULATION VALUES – “CUSTODY TRANSFER PROPAGATION”.....	170
TABLE 6-9: IMPROVEMENT/TRADEOFFS – “CUSTODY TRANSFER PROPAGATION”	170
TABLE 6-10: SIMULATION VALUES – “CUSTODY RETAINING PROPAGATION”.....	174
TABLE 6-11: IMPROVEMENT/TRADEOFFS – “CUSTODY RETAINING PROPAGATION”.....	174
TABLE 6-12: IMPROVEMENT/TRADEOFFS – “SIMPLE FLOOD PROPAGATION”.....	178
TABLE 6-13: IMPROVEMENT/TRADEOFFS – “SIMPLE FLOOD PROPAGATION”.....	178
TABLE C-1: CUSTODY TRANSFER PROPAGATION.....	211
TABLE C-2: IMPROVEMENT / TRADEOFF PERCENTAGE - “CUSTODY TRANSFER PROPAGATION” ...	211
TABLE C-3: ABSOLUTE INDEX - “CUSTODY TRANSFER PROPAGATION”	211
TABLE C-4: EMPHASIZED BALANCING INDEX- “CUSTODY TRANSFER PROPAGATION”.....	212
TABLE C-5: BALANCED INDEX- “CUSTODY TRANSFER PROPAGATION”.....	212
TABLE C-6: “CUSTODY RETAINING PROPAGATION”	212
TABLE C-7: IMPROVEMENT / TRADEOFFS - “CUSTODY RETAINING PROPAGATION”	213
TABLE C-8: EMPHASIZED BALANCING INDEX- “CUSTODY RETAINING PROPAGATION”.....	213
TABLE C-9: EMPHASIZED BALANCING INDEX- “CUSTODY RETAINING PROPAGATION”.....	213

TABLE C-10: BALANCED INDEX- “CUSTODY RETAINING PROPAGATION”	214
TABLE C-11: “SIMPLE FLOOD PROPAGATION”	214
TABLE C-12: IMPROVEMENT / TRADEOFFS PERCENTAGE - “SIMPLE FLOOD PROPAGATION”	214
TABLE C-15: BALANCED INDEX- “SIMPLE FLOOD PROPAGATION”	215
TABLE E-1: PER UNIT INCENSEMENT	232
TABLE E-2: PER UNIT INCENSEMENT BASED IMPORTANT LIST	233
TABLE E-3: OVERALL IMPROVEMENT	233
TABLE E-4: OVERALL IMPROVEMENT BASED IMPORTANCE LIST	233
TABLE E-5: PER UNIT INCENSEMENT	234
TABLE E-6: PER UNIT INCENSEMENT BASED IMPORTANT LIST	235
TABLE E-7: OVERALL IMPROVEMENT	235
TABLE E-8: OVERALL IMPROVEMENT BASED IMPORTANCE LIST	235
TABLE E-9: PER UNIT INCENSEMENT	236
TABLE E-10: PER UNIT INCENSEMENT BASED IMPORTANT LIST	237
TABLE E-11: OVERALL IMPROVEMENT	237
TABLE E-12: OVERALL IMPROVEMENT BASED IMPORTANCE LIST	237
TABLE F-1: PEER DENSITY AND ROUTING QUALITY	239

List of Figures

FIGURE 2-1: TRADITIONAL MOBILE AD HOC NETWORK ROUTING	21
FIGURE 2-2: DTMANET ROUTING PROCEDURE.....	21
FIGURE 2-3: SOLAR ROUTING.....	25
FIGURE 2-4: MOVEMENT BASED ROUTING ALGORITHMS.....	25
FIGURE 3-1: ROUTING LOOP CONTROL	67
FIGURE 3-2: VALIDITY CONTROL (AVERAGE DELIVERY TIME).....	69
FIGURE 4-1: PEER/MESSAGE, AUTO INCREASE OPTION AND SELF-ADAPT DESIGNER.....	77
FIGURE 4-2: COMMUNICATION PROTOCOL AND MOVEMENT MODEL SELECTOR.....	77
FIGURE 4-3: OCI FILTER ORGANIZER.....	78
FIGURE 4-4: OCI AUTO SCHEDULER	79
FIGURE 4-5: XML DATA FORMAT	80
FIGURE 4-6: TRADITIONAL RESEARCH APPROACH CYCLE.....	81
FIGURE 4-7: OCI DEVICE BUILDER AND DEVICE BUILDER VISUAL AID DESIGNER.....	82
FIGURE 4-8: OCI DEVICE BUILDER PROTOCOL EDITOR	82
FIGURE 4-9: OCI FILTER BUILDER AND OCI MOVEMENT BUILDER.....	84
FIGURE 4-10: OCI-VISUAL.....	84
FIGURE 4-11: GLOBAL INFO VIEWER AND DOCUMENT & DATA VIEWER	86
FIGURE 4-12: MESSAGE STATE VIEWER	86
FIGURE 4-13: OCI-GRAPH.....	88
FIGURE 5-1: ABSOLUTE LOW DENSITY.....	114
FIGURE 5-2: ABSOLUTE HIGH DENSITY	115
FIGURE 5-3: INITIAL RELATIVE DENSITY.....	115
FIGURE 5-4: COMPRESSED RELATIVE DENSITY	116
FIGURE 5-5: EXPANDED RELATIVE DENSITY	117
FIGURE 5-6: PRNG DISTRIBUTION TEST	121
FIGURE 5-7: RELATIVE PEER DENSITY	123
FIGURE 5-8: RELATIVE PEER VELOCITY	124

FIGURE 5-9: AVERAGE DELIVERY FLUCTUATION (1 TO 400 SIMULATION REPEATS)..... 125

FIGURE 5-10: AVERAGE DELIVERY RATE OVER TTL (CUSTODY TRANSFER)..... 126

FIGURE 5-11: AVERAGE DELIVERY RATE OVER TTL (CUSTODY TRANSFER PROPAGATION) 127

FIGURE 5-12: AVERAGE DELIVERY RATE OVER TTL (CUSTODY RETAINING PROPAGATION) 127

FIGURE 5-13: AVERAGE DELIVERY RATE OVER TTL (SIMPLE FLOOD)..... 128

FIGURE 5-14: ROUTING PERFORMANCE AND EFFICIENCY 129

FIGURE 6-1-A TO D: CUSTODY TRANSFER PROPAGATION 133

FIGURE 6-2: PEER VELOCITY EFFECTS ON DELIVERY TIME 134

FIGURE 6-3-A TO D: CUSTODY TRANSFER PROPAGATION 135

FIGURE 6-4: PEER VELOCITY EFFECTS ON DELIVERY ROUTING DEPTH 136

FIGURE 6-5: PEER VELOCITY EFFECTS ON DELIVERY MESSAGE DUPLICATION 136

FIGURE 6-6-A TO D: CUSTODY TRANSFER PROPAGATION 137

FIGURE 6-7: PEER TERRITORY EFFECTS ON MESSAGE DELIVERY RATE 138

FIGURE 6-8-A TO D: CUSTODY TRANSFER PROPAGATION 139

FIGURE 6-9: PEER TERRITORY EFFECTS ON MESSAGE DELIVERY TIME 140

FIGURE 6-10-A TO D: CUSTODY TRANSFER PROPAGATION 140

FIGURE 6-11: PEER TERRITORY EFFECT ON MESSAGE ROUTING DEPTH..... 141

FIGURE 6-12-A TO D: CUSTODY TRANSFER PROPAGATION 141

FIGURE 6-13: CUSTODY RETAINING PROPAGATION 142

FIGURE 6-14: DELIVERY RATE OVER “CUSTODY TRANSFER” 144

FIGURE 6-15: DELIVERY TIME OVER “CUSTODY TRANSFER” 145

FIGURE 6-16: DELIVERY ROUTING DEPTH OVER “CUSTODY TRANSFER” 145

FIGURE 6-17: DELIVERY RATE OVER “CUSTODY TRANSFER PROPAGATION” 147

FIGURE 6-18: DELIVERY TIME OVER “CUSTODY TRANSFER PROPAGATION” 147

FIGURE 6-19: DELIVERY ROUTING DEPTH OVER “CUSTODY TRANSFER PROPAGATION” 148

FIGURE 6-20: MESSAGE DUPLICATION OVER “CUSTODY TRANSFER PROPAGATION” 149

FIGURE 6-21: MESSAGE DELIVERY RATE OVER “CUSTODY RETAINING PROPAGATION” 151

FIGURE 6-22: MESSAGE DELIVERY TIME OVER “CUSTODY RETAINING PROPAGATION” 151

FIGURE 6-23: DELIVERY ROUTING DEPTH OVER “CUSTODY RETAINING PROPAGATION” 152

FIGURE 6-24: MESSAGE DUPLICATION OVER “CUSTODY RETAINING PROPAGATION” 153

FIGURE 6-25: MESSAGE DELIVERY RATE OVER “SIMPLE FLOOD PROPAGATION” 154

FIGURE 6-26: MESSAGE DELIVERY TIME OVER “SIMPLE FLOOD PROPAGATION” 154

FIGURE 6-27: MESSAGE DELIVERY ROUTING DEPTH OVER “SIMPLE FLOOD PROPAGATION” 155

FIGURE 6-28: MESSAGE DUPLICATION OVER “SIMPLE FLOOD PROPAGATION” 156

FIGURE 6-29: MESSAGE DELIVERY RATE OVER “CUSTODY TRANSFER” 163

FIGURE 6-30: MESSAGE DELIVERY TIME OVER “CUSTODY TRANSFER” 164

FIGURE 6-31: MESSAGE DELIVERY ROUTING DEPTH OVER “CUSTODY TRANSFER” 165

FIGURE 6-32: MESSAGE DELIVERY RATE OVER “CUSTODY TRANSFER PROPAGATION” 167

FIGURE 6-33: MESSAGE DELIVERY TIME OVER “CUSTODY TRANSFER PROPAGATION” 168

FIGURE 6-34: MESSAGE ROUTING DEPTH OVER “CUSTODY TRANSFER PROPAGATION” 169

FIGURE 6-35: MESSAGE DUPLICATION OVER “CUSTODY TRANSFER PROPAGATION” 169

FIGURE 6-36: MESSAGE DELIVERY RATE OVER “CUSTODY RETAINING PROPAGATION” 171

FIGURE 6-37: MESSAGE DELIVERY TIME OVER “CUSTODY RETAINING PROPAGATION” 172

FIGURE 6-38: MESSAGE ROUTING DEPTH OVER “CUSTODY RETAINING PROPAGATION” 173

FIGURE 6-39: MESSAGE DUPLICATION OVER “CUSTODY RETAINING PROPAGATION” 173

FIGURE 6-40: MESSAGE DELIVERY RATE OVER “SIMPLE FLOOD PROPAGATION” 175

FIGURE 6-41: MESSAGE DELIVERY TIME OVER “SIMPLE FLOOD PROPAGATION” 176

FIGURE 6-42: MESSAGE ROUTING DEPTH OVER “SIMPLE FLOOD PROPAGATION” 177

FIGURE 6-43: MESSAGE DUPLICATION OVER “SIMPLE FLOOD PROPAGATION” 177

FIGURE B-1: OCI-SIM STRUCTURE ORGANIZATION MAP 210

FIGURE D-1 TO 4: CUSTODY TRANSFER PROPAGATION 216

FIGURE D-5: PEER ROUTING REPUTATION AND ITS EFFECTS ON MESSAGE DELIVERY RATE 216

FIGURE D-6 TO 9: CUSTODY TRANSFER PROPAGATION 217

FIGURE D-10: PEER ROUTING REPUTATION AND ITS EFFECTS ON MESSAGE DELIVERY TIME 217

FIGURE D-11 TO 14: CUSTODY TRANSFER PROPAGATION 218

FIGURE D-15: REPUTATION AND ITS EFFECTS ON ROUTING DEPTH 218

FIGURE D-16 TO 19: CUSTODY TRANSFER PROPAGATION 219

FIGURE D-20: REPUTATION AND ITS EFFECTS ON DUPLICATION 219

FIGURE D-21 TO 24: CUSTODY TRANSFER PROPAGATION 220

FIGURE D-25: RANGE AND ITS EFFECTS ON DELIVERY RATE 220

FIGURE D-26 TO 29: CUSTODY TRANSFER PROPAGATION 221

FIGURE D-30: REPUTATION AND ITS EFFECTS ON DELIVERY RATE 221

FIGURE D-31 TO 34: CUSTODY TRANSFER PROPAGATION 222

FIGURE D-35: RANGE AND ITS EFFECTS ON ROUTING DEPTH 222

FIGURE D-36 TO 39: CUSTODY TRANSFER PROPAGATION 223

FIGURE D-40: RANGE AND ITS EFFECTS ON MESSAGE DUPLICATION 223

FIGURE D-41 TO 44: CUSTODY TRANSFER PROPAGATION 224

FIGURE D-45: MULTI-ADAPTOR AND ITS EFFECTS ON DELIVERY RATE 224

FIGURE D-46 TO 49: CUSTODY TRANSFER PROPAGATION 225

FIGURE D-50: MULTI-ADAPTOR AND ITS EFFECTS ON DELIVERY TIME 225

FIGURE D-51 TO 54: CUSTODY TRANSFER PROPAGATION 226

FIGURE D-55: MULTI-ADAPTOR AND ITS EFFECTS ON ROUTING DEPTH 226

FIGURE D-56 TO 59: CUSTODY TRANSFER PROPAGATION 227

FIGURE D-60: MULTI-ADAPTOR AND ITS EFFECTS ON MESSAGE DUPLICATION 227

FIGURE D-61 TO 64: CUSTODY TRANSFER PROPAGATION 228

FIGURE D-65: CONSISTENCY AND ITS EFFECTS ON DELIVERY RATE 228

FIGURE D-66 TO 69: CUSTODY TRANSFER PROPAGATION 229

FIGURE D-70: CONSISTENCY AND ITS EFFECTS ON DELIVERY TIME 229

FIGURE D-71 TO 74: CUSTODY TRANSFER PROPAGATION 230

FIGURE D-75: CONSISTENCY AND ITS EFFECTS ON ROUTING DEPTH 230

FIGURE D-76 TO 79: CUSTODY TRANSFER PROPAGATION 231

FIGURE D-80: CONSISTENCY AND ITS EFFECTS ON MESSAGE DUPLICATION 231

FIGURE F-1: DELIVERY RATE OVER TRANSMISSION RANGE 238

FIGURE F-2: DELIVERY RATE OVER TRANSMISSION RANGE 238

FIGURE F-3: QUALITY MATRIX OVER TRANSMISSION RANGE 239

FIGURE F-4: QUALITY MATRIX OVER TRANSMISSION RANGE 240

List of Codes

CODE 5-1: AN EXAMPLE OF PEER DATA IN XML FORMAT	96
CODE 5-2: AN EXAMPLE OF MESSAGE DATA IN XML FORMAT	102
CODE 5-3: AN EXAMPLE OF MOVEMENT MODEL SOURCE CODE.....	105
CODE 5-4: AN EXAMPLE OF FILTER MODULE SOURCE CODE	107

Chapter I: Introduction

“Everything should be made as simple as possible, but not simpler.”

– Albert Einstein

This thesis presents a framework for investigating opportunistic routing in Delay Tolerant Mobile Ad hoc Networks (DTMANETs), and introduces the concept of the Opportunistic Confidence Index (OCI). The OCI enables multiple opportunistic routing protocols to be applied as an adaptive group to improve DTMANET routing reliability, performance, and efficiency. DTMANET is a newly emerged network architecture, which enables mobile wireless communication under extreme environments. DTMANETs often face many challenging conditions, such as the dynamics and the heterogeneities of mobile devices within the network. Currently, there are a number of research attempts to leverage the dynamics, such as device movement patterns, to aid in routing decision-making. This approach is known as opportunistic routing. The results of existing research on opportunistic routing have made a positive impact on DTMANETs. However, current proposals on opportunistic routing focus on single elements of mobile device dynamics and ignore others that may provide additional contributions and improvements to the DTMANET routing process.

In this work, we propose an approach called OCI. OCI addresses this issue and presents the concept of using multiple opportunistic routing protocols to provide additional improvements for DTMANET routing for all three quality metrics: routing reliability, performance and efficiency. To achieve this, OCI uses the techniques of intelligent weighting and adaptive multiple-attribute decision-making, in contrast to single opportunistic routing.

This chapter first provides an overview of the background to the technologies that are fields of interest to this project. Secondly, it analyses unresolved issues within current research trends. Thirdly, it states the motivations and inspirations upon which this

work is based. Next, it outlines the research scope, project focus, goals to be achieved and the novel contributions to be expected. Finally, it details a layout of the subsequent chapters.

1.1 General Introduction

1.1.1 Background

It has been over two decades since the birth of the world's first commercial cell phone, the Motorola DynaTAC 8000X, in 1983 [Motorola, 2007]. Today, following rapid technological developments, mobile wireless communication technologies have evolved to an unprecedented level. Mobility has penetrated every aspect of people's lives and wireless connectivity has covered every corner of modern society [ITU, 2007]. While people are enjoying the convenience of communication brought by seemingly ubiquitous wireless connectivity such as cellular and WIFI, there are circumstances when network communications face extreme conditions where no existing infrastructure is available. These environments include disaster relief networks, ad hoc information discovery and distribution networks and the ad hoc interplanetary communication network [Burleigh *et al*, 2003]. In such environments, current mobile wireless communication models do not perform well. This is because current models assume the availability of wireless communication infrastructures and the reliability of connectivity, and are not designed to target extreme networking environments.

On the other hand, while some areas enjoy the benefits of pervasive information access through mobile wireless technologies, other areas are left behind, unexplored and underdeveloped, such as developing regions in Africa. In these cases, the current Internet model of communication is unaffordable because of the cost of infrastructure deployment and the running cost required by telecommunication providers. Alternative methods of connectivity are thus required. New technologies under development that specifically target these issues include the MANET (Mobile Ad hoc

Network) and the DTN (Delay Tolerant Network). In addition, while not born of necessity, MANET and DTN architectures may be utilized by communities who wish to build their own networks for purposes such as toll bypass, entertainment, and community-building applications.

1.1.2 Field of Relevance

1) Mobile and Wireless Communication

Mobile wireless communication is a rather generic term, which describes the communication ability among devices that are not physically interconnected and are often in remote or roaming environments. To achieve this ability to communicate, mobile devices need to be connected wirelessly. Such a connection ties the mobile device to centralized or distributed information and services, often through the use of devices that are portable, battery powered, and have wireless communication capability such as cell phones, PDAs and laptop computers. A wide variety of mobile and wireless technologies are available today such as Bluetooth (802.15), WIFI (802.11), WIMAX (802.16), GPRS, EDGE, HSDPA, and Direct Satellite Communication. These technologies comprise a number of different protocols and specifications, and each of them is designed to suit a specific communication requirement. Some address the issue of the connection throughput, some emphasise long distance communication and others offer solutions to personal connectivity. Meanwhile, current research on mobile wireless communication emerges from a large number of different fields, such as pervasive computing, digital sensor networks, agent networks, mobile ad hoc networks. In this project, the research focuses on Delay Tolerant Mobile Ad Hoc Networks.

2) Wireless Mobile Ad hoc Communication

A wireless MANET is a self-configuring network of mobile routers and associated hosts connected by wireless links. Its connectivity is created dynamically and in an ad

hoc fashion. Unlike traditional fixed or static wireless networks, a MANET has no fixed hierarchy, and it forms an arbitrary topology [Royer *et al.*, 1999]. The routers organize themselves randomly. Therefore, the topology of the network may change frequently and unpredictably. Commonly, a MANET operates in a standalone fashion, but it may also be connected to other networks such as the Internet [Mauve *et al.*, 2001] [Maihöfer, 2004]. “The subject matter of ad hoc networks clearly represents a new paradigm in wireless communications and networking and it’s now clear that in the next decade there will be a myriad of applications of this new paradigm, ranging from military ad hoc wireless networks, environmental sensor networks, car-based ad hoc networks, to biomedical sensor networks, etc... As a new area in mobile wireless communications, mobile ad hoc communication is going to prevail in the next few decades. Understanding the full potential of this technology will lead to new applications benefiting both civilian and military usage” [Ozan *et al.*, 2006].

3) Delay Tolerant Wireless Mobile Ad hoc Communication

DTMANET is a solution for communication networks that aim to address issues in mobile wireless communication in extreme environments such as those that lack pre-existing infrastructure and continuous connectivity. In a DTMANET, communication is conducted asynchronously, via message oriented routing mechanisms, in a store-and-forward manner between source peers, message-relaying intermediate peers, and the destination peer. DTMANET can bridge communication over varied network transport layers (both IP and non-IP based transport protocols). Thus, the DTMANET architecture operates as an overlay network, providing a new architecture and coarse-grained communication services. The DTMANET architecture is the focus of this work.

1.2 Project introduction

1.2.1 Motivation

DTMANET is a new, challenging and exciting field, and our proposed hypothesis has not been investigated by any prior research, which provides the primary motivation for this work.

Secondly, DTMANET creates opportunities for new types of applications covering a wide range of fields. We feel that despite the application level variations of DTMANETs, routing quality is essential to the usefulness of all DTMANET applications. Thus, through this work, a contribution can be made to DTMANETs at a fundamental level.

Thirdly, the characteristics of DTMANET offer the potential to facilitate alternative communication in challenging environments where there is lack of network coverage. Our research is being undertaken in the Republic of South Africa, which has vast rural areas and a large number of rural communities¹. There are many schools in those communities that have no network access since there is no existing network infrastructure. It is hoped that through this research into the technology for Delay Tolerant Mobile Ad hoc Network, a contribution can be made to help the future development of alternative communication infrastructures for those schools and help to bridge the “digital divide” in under-developed rural communities in South Africa and other developing countries.

1.2.2 Problem Domain and Field Scope

Since the late 1990s, mobile ad hoc networks have become a popular research topic in both academia and the military as many wireless network standards have become widespread. However, at present, applications for mobile ad hoc networks, especially

¹ 42.5% of South Africa’s population live in rural areas, according to Stats SA census in 2001.

for DTMANET, are limited to a handful of military and other experimental systems. This is because, firstly, unlike a traditional wireless network, which is centralized, structured and has a static topology, the dynamic, autonomic and unpredictable nature of a mobile ad hoc network means that routing in such an environment is challenging and problematic. Routing issues such as reliability, performance, efficiency [Theoleyre *et al*, 2005] [Ma *et al*, 2003], trust [Pirzada *et al*, 2004], security [Yang *et al*, 2004] and the fair usage of networks, also become major problems when transforming mobile ad hoc networks into real-life applications.

A wide range of routing protocols for MANET has been proposed. These include: the Ad hoc On Demand Distance Vector (AODV) routing algorithm [Perkins *et al*, 1999]; the Dynamic Source Routing Protocol (DSR) [Johnson *et al*, 2004]; the Temporally-Ordered Routing Algorithm (TORA) [Park *et al*, 1997]; the Optimized Link State Routing Protocol (OLSR) [Clausen *et al*, 2001]; the Delay Tolerant Network (DTN) [Fall, 2003]; opportunistic routing [LeBrun, 2005]; epidemic routing [Vahdat *et al*, 2000]; probabilistic routing [Lindgren *et al*, 2003]; MobySpace routing [Leguay *et al*, 2005]; and hot potato routing (deflection routing) [Teixeira *et al*, 2004]. Meanwhile, research has been conducted to improve the usability of mobile ad hoc networks. For example, there are reputation based systems [Sankhla, 2004], game theoretic systems [Turocy, 2001] [Fang *et al*, 2004], and Peer-to-Peer MANET overlay systems [Hu *et al*, 2004].

Among the research mentioned above, opportunistic routing is the focus of this research. Current proposals of opportunistic routing select a single element of the mobile device dynamics, while other elements are ignored. The hypothesis of this work is that by using multiple opportunistic routing elements collaboratively and intelligently, extra improvement can be provided over DTMANET routing in all three networking measurement aspects, namely, reliability, performance and efficiency. OCI is designed to fill this gap in current opportunistic routing research. At this point, it is necessary to clarify that the purpose of this work is neither to create another opportunistic routing protocol nor to exhaustively evaluate and compare existing

individual opportunistic routing protocols for DTMANETs. Instead, it proposes and evaluates a new way of leveraging opportunistic elements to aid DTMANET routing.

1.2.3 Research Objectives and Expected Contributions

The primary objectives of this work are: propose and prove a concept; design and establish a framework; develop and make use of a software network simulator tool-set as the proof of concept environment.

The first goal of this project is to propose and to evaluate the concept of using multiple heterogeneous opportunistic routing protocols to aid routing decision-making processes in DTMANETs, through investigating its viability.

The second goal is to establish an extensible general-purpose framework, the Opportunistic Confidence Index. Using OCI, different opportunistic routing protocols can be applied in an organized fashion to aid the routing decision-making process and therefore to increase the probability of successful message delivery and improve the overall communication performance and efficiency in a DTMANET.

The third goal is to test the concept of OCI. A software tool set (OCI-SIM) is developed to aid the research through simulation, analyzing and visualizing the framework design. It is hoped the tools will be useful for future studies in this and related areas.

1.2.4 Research Procedures and Experimental Approach

This work has taken an experimental system building and simulation approach of computer science, which made use of methods and techniques such as abstract modelling, opportunistic element simplification and isolation, random attribute generation and assignment, pair-wise knowledge sharing, intelligent weight assignment and/or opportunistic element permutation, and automated scenario generation incorporating both data acquisition and analysis phases.

To test the concept of using multiple opportunistic routing protocols to aid DTMANET routing decision-making and to establish the framework of OCI, firstly, a group of opportunistic elements were selected and modelled abstractly. Each of these elements was tested for its impact on DTMANET routing quality in a controlled environment, through variable isolation and targeted element value linear increment, thus establishing an index of importance and relevance for these selected opportunistic elements.

Secondly, rules of routing protocols for these opportunistic elements were modelled as OCI opportunistic routing filters. Each of these filters was tested individually as the principle opportunistic routing protocol on top of a group of selected non-opportunistic base routing protocols in a randomly generated environment. The results of this procedure were used to construct OCI indexes and served as benchmarks for tests of OCI.

Thirdly, based on previous results, the opportunistic filters were assigned weights and ordered according to their impact on given DTMANET scenarios as OCI indexes. Rules of using OCI indexes as opportunistic routing protocols were constructed.

Fourthly, OCI was tested as the principle opportunistic routing protocol in a series of randomly generated environments on top of a group of selected non-opportunistic base protocols. The results of this procedure were analysed and used as the primary evidences to prove the concept of OCI.

1.3 Thesis Organization

Chapter I	is a general introduction to this project, which outlines the research focus within the context of the field of study, the objectives of the research, and the contributions made.
Chapter II	presents a literature survey of the field of study, and discusses a number of related research efforts into delay tolerant networks and mobile ad hoc networks.
Chapter III	is an overview of the algorithms, theories, methodology, and the conceptual system design that underpin this work.
Chapter IV	introduces the design philosophy, as well as the functionality, of the simulation tool set (OCI-SIM) developed to support this investigation.
Chapter V	presents and classifies results of specific scenarios using the simulation environment. It also discusses the measures taken in terms of the validity control for this research.
Chapter VI	analyzes the data produced by the OCI framework simulation procedures, and evaluates the extent to which the OCI proposal is feasible.
Chapter VII	draws conclusions from the study, evaluating and discussing the findings, the contributions, and the limitations of this investigation. It also highlights possible areas of future work.
Appendix A	is a companion DVD which contains an electronic version of this thesis, application and source code, data and graphs, and videos of the proof-of-concept system in action.
Appendix B to G	provides additional data and analysis graphs to support the findings of this thesis.

Chapter II: Literature Survey

"Solved problems aren't news. Tell the press a story in two halves - the problem first and the solution later. Then they get a disaster story one day and triumph story the next."

Sir Humphrey – "Yes Minister"

This chapter is a literature survey of the field of study in two parts. The first part is an introduction to the concept of delay-tolerant, mobile ad hoc networks, with a comparison to conventional mobile ad hoc networks, which includes a definition, a brief history, and potential applications (sections 2.1-2.2). The second part is a discussion of routing protocols for supporting DTMANET, which covers routing algorithm classification, current research on DTMANET routing, a survey of opportunistic routing protocols, and an overview of network simulation (sections 2.3-2.5).

2.1 DTMANET – A General Introduction

Today's Internet is highly successful and it has achieved worldwide adoption. Existing network architectures, communication models and network protocols are sufficient and efficient for the use of the Internet under normal circumstances. However, there are regions and situations where no fixed network infrastructure is available, the networks are frequently partitioned and there is no guarantee of continuous, stable connectivity. Furthermore, networked devices in such environments are often constrained by their transmission range, processing power, storage space, and power supply. In such environments, conventional networking protocols perform poorly, and are often not suitable. To address this issue, a number of research initiatives have been undertaken in recent years, primarily under the umbrella of the Delay Tolerant Network Research Group [DTNRG, 2005], and a general architecture class called the DTN (Delay Tolerant Network) has been proposed. The

DTN makes limited assumptions about end-to-end connectivity and peer resources. It utilizes network device mobility, message caching, and relaying techniques to achieve an asynchronous, connected message-oriented communication network.

2.1.1 A Brief History of Message Oriented Networks

The concept of message-oriented systems has been understood for several decades. As early as 1978, message-oriented networks were being used as the basis for formal work in multiprocessor architectures [Hoare, 1978]. By request of the US Army, an electronic message-based mail system, the MMDF (Multi-channel Memorandum Distribution Facility) system [MMDF 2005] with its underlying supporting protocols, was created. MMDF provided routing of ARPANET-like [Hauben, 1994] mail messages over regular telephone lines. At about the same time, the SendMail [Vixie *et al.*, 2001] transfer agent was also developed. While MMDF was focused upon Internet-style electronic mail, SendMail offered multiple mail format relaying. Consequently, MMDF lost its popularity to SendMail, because it relied on Internet-like addressing and mail routing protocols, and was incapable of handling multiple addressing and multiple formats. By contrast, SendMail could relay mail messages from formats as different as, Internet, BitNet, and UUCP.

SendMail, MMDF, and later systems, such as MMDF2 [Farber, 2000] and Qmail [Qmail, 2007], employed message store-and-forward operations. However, the routing decisions of these systems are made through underlying protocols, such as a static routing table, which commonly lack intelligent routing mechanisms. Thus, these systems did not take advantage of different forms of connectivity to deliver a single message, but instead attempted to multiplex and de-multiplex messages onto their implied transports, while supporting store-and-forward operations in a single process [Demmer *et al.*, 2004].

One of the earliest and widely used messaging networks among research communities was BitNet [Fuchs *et al.*, 2007]. BitNet emerged in the early 1980s. It used a

store-and-forward mechanism to offer services such as email, mailing lists, interactive chat, and remote file access. BitNet is derived from the IBM VNet [VNET, 2007] email system. IBM VNet uses the remote spooling (RSCS) and Network Job Execution protocols (NJE/NJI). It employs a tree structure for routing, and depends on underlying protocols for reliability control. BitNet runs on top of rented telephone lines, which offer relatively low message-delivery latency. However, for many BitNet operators, such infrastructure was costly, since, at that time, not everyone could afford to keep a BitNet Site continuously running over a rented telephone line, and this became the bottleneck for any further adoption of BitNet.

To address this problem, around 1981, UseNet [Usenet, 2001] [Moraes, 1999] emerged as a newsgroup-like system, which provided similar functionality to BitNet. UseNet sacrificed some of the real-time communication ability, but provided extended disconnection tolerance. Usenet supported various underlying transport protocols, historically the most important one being UUCP [Ziegast, 1997]. UUCP stands for Unix-to-Unix Copy Program which provides the underlying capability for remote asynchronous execution. The limitation of UUCP is that it is source routed, which became too cumbersome to use. To solve this problem, an automated approach was developed. This approach used locally-stored UUCP links and periodically updated link-cost information to calculate shortest paths from the local node to the destination.

Delay Tolerant Networks were first proposed as a new concept in 2002 by V. Cerf and Kevin Fall [Cerf, 2002] [Fall, 2002]. In the same year, the DTNRG [DTNRG, 2005] (Delay Tolerant Network Research Group) was formed under the sponsorship of the Internet Research Task Force (IRTF) [IETF, 2007]. In the last few years, DTN has received increasing interest from the research community. DTN addresses and generalizes problems encountered on several fronts in early message-oriented systems. These problems include issues of dynamic routing, network fragmentation, multiple routing support, time-varying routing possibilities, and an explicit method to encode the names of other regions without “rewriting” rules [Demmer *et al*, 2004].

2.1.2 Potential Applications of DTMANETs

DTNs naturally arise in a number of interesting areas. Arguably, applications that require real-time communication and fixed bit-rate transmission are beyond the capability of current DTMANET implementations, since a reliable route between the source and the destination is required for the use of such applications. However, applications with different requirements may very well benefit from the advantages of DTMANET where constant packet transmission and timely delivery are not essential. DTMANET is especially helpful where there is no fixed network infrastructure or where network partitioning occurs frequently. Typical applications include ad hoc information discovery and distribution networks, disaster recovery and relief communication networks, mobile sensor networks, interplanetary networks, military battlefield tactical communication networks, and smart dust.

1) Ad hoc information discovery and distribution

Social locations where people are likely to gather in large numbers, such as a school, shopping mall or city square, offer an ideal environment for DTMANET applications to bridge communication and provide ad hoc services such as advertisements, entertainment and social networking. Where such social locations may be disconnected from each other, communication between locations is possible through the mobility of DTMANETs (see *section 2.1.3*).

2) Emergency rescue / Disaster recovery and relief

For emergency rescue and disaster relief missions, it is likely that there will be no available communication infrastructure. However, rescue workers can rely on DTMANETs to communicate vital information with each other and on the control post to coordinate rescue and recovery procedures. For example, consider the following scenario: a group of cave climbers are trapped in a deep and winding cave, where conventional communication methods such as satellite- and cell

phones will not help. However, communication can be established through mobile communication devices carried by rescue workers and multiple, strategically-placed, relay points.

3) Mobile sensor networks

Mobile sensor networks are networks of wireless sensor devices with communication capabilities that are distributed over a geographic area [Estri *et al*, 1999] [Heinzelman, 1999]. These sensors may be used to monitor and record motion, chemicals, temperature, or their surrounding environment. For example, a recent project [Markham *et al*, 2007] proposes the use of a DTMANET to monitor animal social behaviours. It uses animal movements to gather data and relays it to a data collection post.

4) Interplanetary Network:

An Interplanetary Network might sound like science fiction, but it is becoming science fact. The aim of an Interplanetary Network is to form a backbone connecting a series of hubs on or around planets, space ships, and at other points in space. These hubs would provide high-capacity, high-availability Internet traffic over distances that could stretch up to hundreds of millions of miles. Currently, Internet addresses have already been assigned to all the planets, satellites, and spacecraft in our solar system [Cerf *et al*, 2002]. The Interplanetary Network is classified as a DTMANET, where information is stored and forwarded to any hub on the system. The DTMANET is ideal to provide an always-on connection between planets, spacecraft, and the terrestrial Internet. For example, if a DTMANET were to be setup on Mars, hubs could be installed on a series of satellites circling the planet. Messages could then be sent from the far side of Mars, and those messages could be relayed to the nearest hub for routing back to Earth. The "store-and-forward" protocol of the Interplanetary Network would help minimize problems that might arise due to the vast distances involved, problems

such as high error and latency rates that might often be minutes or even hours long.

[Gray, 2003] [Lesh, 2001]

5) Military Battlefield Communication:

On the battlefield, military personnel and combat vehicles are equipped with sensors and mobile communication gear. Using DTMANET, information can be gathered and communicated from a territory when conventional communication methods are destroyed or under an electronic jamming attack. Thus, command, control and communication (C3) can still be carried out using DTMANETs

[Halvardsson *et al*, 2004].

6) Smart Dust:

A Smart dust network is comprised of micro-electrical mechanical sensors (MEMS) [Kahn, 1999]. Compared with traditional sensor networks, smart dust is much smaller in size, consumes much less power and can be deployed into a large area more easily [Vahdat, 2000].

2.1.3 “Wizzy Digital Courier” – A Current Example

“Wizzy Digital Courier” is a project started in early 2003 to bring "low cost Internet access" to schools in South Africa. It distributes useful data to places with no Internet connection; primarily used for e-mail, it also carries web content (stored locally in a web cache).

An early description of the project explained: “Data normally carried by the dial-up telephone link is instead physically carried by a mobile computer between the end user's location and a high bandwidth data drop to the Internet.” [Wizzy, 2007]

Delivery mechanisms are by overnight dial-up, taking advantage of special calling rates outside business hours, or USB memory stick. The USB stick uses the UUCP

protocol, carrying information to and from a better-connected location - perhaps a school or local business, which acts as the drop-off for email, and fetches web content by proxy. The email and web content is re-packaged as a UUCP transaction, and ferried back on the USB stick [Wiki, 2007].

2.2 Research on Delay Tolerant Networks

2.2.1 Characteristics of Delay Tolerant Networks

There are quite a few definitions of Delay Tolerant Networks available from the existing literature [Wick, 2007]. From the definition given by the DTNRG [DTNRG, 2005] in [Cerf *et al.*, 2007], the characteristics of a Delay Tolerant Network can be summarized as:

- 1) A network that has a high level of heterogeneity
- 2) A network that suffers from frequent network partitioning
- 3) A network that endures recurrent interruption and failures
- 4) A network that has asymmetric, long and variable data rates
- 5) Devices in a network suffer from energy, bandwidth, storage/memory and cost constraints.

2.2.2 DTMANET- Delay Tolerant Mobile Ad Hoc Networks

A Delay Tolerant Mobile Ad Hoc Network is a special case of MANET, [Borrel *et al.*, 2007] within which each communication is bundled into a single payload (message) and forwarded along a route. The latency from the source to the router, and then to the destination is greater than any conventional network measurement such as TCP RTT. In a DTMANET, it is not necessary that a fully connected end-to-end path from the source to the destination exist at any specific time, but it is necessary that there be a probabilistic expectation that a route will have existed before some predefined time interval has elapsed. It is a norm instead of an exception for a DTMANET to be

fragmented and to have a temporary unreachable route to the destination. However, the mobility in such a network provides the possibility of establishing a route for the delivery [Demmer *et al*, 2004]. DTMANETs, by their nature, are networks of caching and forwarding. Unlike other caching and forwarding networks, where a message is stored for a very short period of time, often measured in milliseconds or seconds, in a DTMANET a message must be stored until it is reliably forwarded to its destination. The delay can be hours, days or in extreme situations, such as in a Interplanetary Network, even years [Burleigh *et al*, 2003].

The following is an example of a typical DTMANET communication scenario: a mobile “Peer A” wants to communication with “Peer B”. However, neither “Peer A” nor “Peer B” is in range of any “base station”, nor does “Peer A” possess the knowledge either of the current location of “Peer B”, or of the route to “Peer B”. Meanwhile “Peer B” is also roaming. While roaming, “Peer A” periodically encounters other peers in the DTN, when communication can occur and a message can be relayed. This process repeats until one peer with the relayed message meets “Peer B” and completes the delivery.

2.2.3 DTMANET – The Diversity

DTMANET is a broad definition. Although DTMANETs share a number of similar characteristics, different DTMANETs vary from one another in many aspects. Some of the diversities are: protocol usage variation; topology and architecture variation; mobility variation; density variation; timing sensitivity variation; and reliability variation. These variations are not mutually exclusive. On the contrary, most real-life DTMANETs experience multiple variations simultaneously. For instance, a battlefield communication DTMANET will be both timing critical and reliability critical. However, within each category, a DTMANET cannot belong to different subcategories; a network cannot be both homogenous and heterogeneous at the same time, which is a logical contradiction [Fall, 2003].

2.2.4 Challenges in DTMANET Research

DTMANETs deal with challenging network environments that are drastically different from traditional fixed networks. Therefore, researches of DTMANETs face a series of challenges:

1) Routing challenge

Routing in a DTMANET is very different from a fixed network. Conventional routing approaches and techniques such as routing Hash tables and shortest end-to-end route computations cannot be applied to a DTMANET because of its constant network partitioning and changing topology [Tsai *et al.*, 1995].

2) Data Transfer challenges

Due to the lack of stable and durable routes between sources and destinations in a DTMANET, data cannot be streamed in real-time as in a fixed network [Patra *et al.*, 2003].

3) Naming and Addressing challenges

When bridging through partitioned networks from cluster to cluster, naming and addressing are challenging issues [Chuah *et al.*, 2006].

4) Quality of Services challenges

It is very challenging for a DTMANET to ensure quality of service, as it is running over a highly dynamic and irregular network [Lindgren, 2006].

5) Security challenges

DTMANETs commonly have a high degree of distribution and heterogeneity. Devices in DTMANETs are often highly autonomous, and therefore, enforcing security over DTMANETs is very challenging [Burgess *et al.*, 2007].

6) Time Synchronization

DTMANETs use a cache-and-forward mechanism for data transfer. Data can be distributed to different peers at different times, and therefore, synchronizing data in a timely order is yet another challenge of DTMANETs [Fall 2003].

2.3 DTMANET Routing Mechanisms

Routing and message diffusion face enormous challenges in a DTMANET because of information and resource constraints. In a DTMANET, information diffusion approaches commonly adopt opportunistic, probabilistic functions or metrics to aid the routing decision-making process, making use of different types of knowledge sources.

2.3.1 DTMANET Routing Design Consideration

DTMANET routing protocols are designed to adopt and accommodate so-called “Connectivity Challenged” networks that commonly feature frequently altered network partition and topology, high loss rate, episodic or scheduled link connectivity and power limitation. Traditional routing protocols, over fixed or wireless links, such as TCP/IP can be insufficient in such a dynamic network environment.

In a traditional fixed network, routing protocols assume networks are constantly connected peers and aim at finding the shortest end-to-end route between the source peer and the destination peer within the connected graphs. The process of such routing protocols is to select the shortest of available routes, by a routing-hop metric

computation. The growth of interest in mobile ad hoc networks, has resulted in several newly emerged routing protocols such as AODV [Schumacher, 2004], DSR [Johnson *et al*, 2004], TORA [Bertsekas, 2002] and OLSR [OLSR, 2007], which address the issues of mobility and temporary loss of connectivity. However, these protocols still maintain and rely on a connectivity graph vector to acquire connectivity information and make routing decisions. As a result they are suitable for conventional MANET routing, but not for a DTMANET with a highly and constantly partitioned topology.

Fixed network routing protocols and some of the ad hoc routing protocols etc. AODV, DSR, rely on consistent network connectivity to provide routing information. Therefore, they are not suitable to solve the routing problems faced in a frequently interrupted and constantly partitioned network. On the contrary, message-oriented protocols such as DTN, are more suitable for such environments, since they do not rely on constant connectivity to establish communication.

2.3.2 DTMANET Routing vs. MANET Routing

Although DTMANET is considered a special case of MANET [Borrel *et al*, 2007], the two are very different in terms of architecture and routing approaches. The key feature that distinguishes a DTMANET from a MANET is that in a DTMANET there may never be a contemporaneous end-to-end path, but the union of network snapshots over time may present an end-to-end path [Brustoloni, 2006]. Conventional MANET routing protocols typically drop packets in such situations and therefore are insufficient.

As *Figure 2-1* illustrates, the traditional mobile ad hoc network communicates by establishing and maintaining an end-to-end route between “Source A” and “Destination G”, and packets are forwarded along the route. In the *Figure 2-2*, “Peer A” initializes the communication and delivery of its payload to “Peer G” through the routing that is composed by A-C-D-F-G.

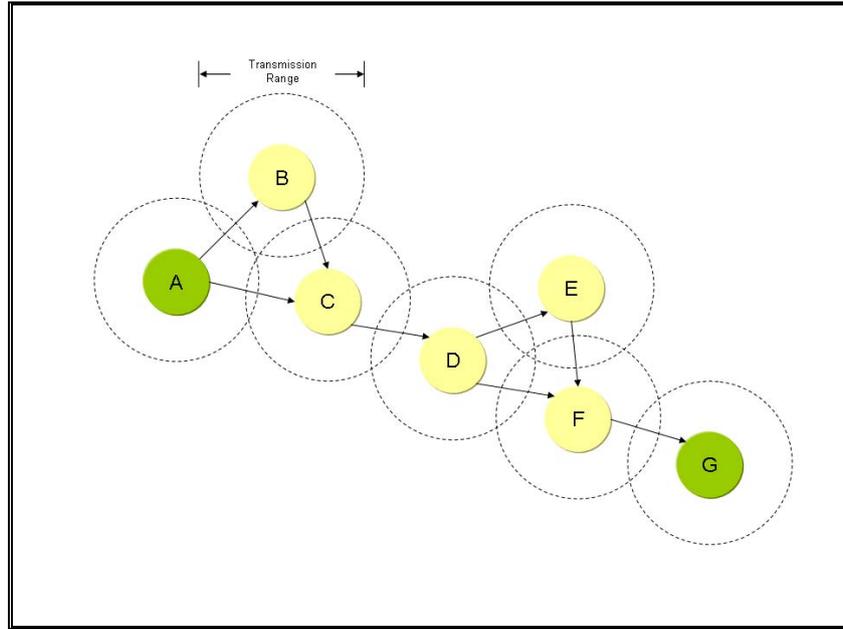


FIGURE 2-1: TRADITIONAL MOBILE AD HOC NETWORK ROUTING

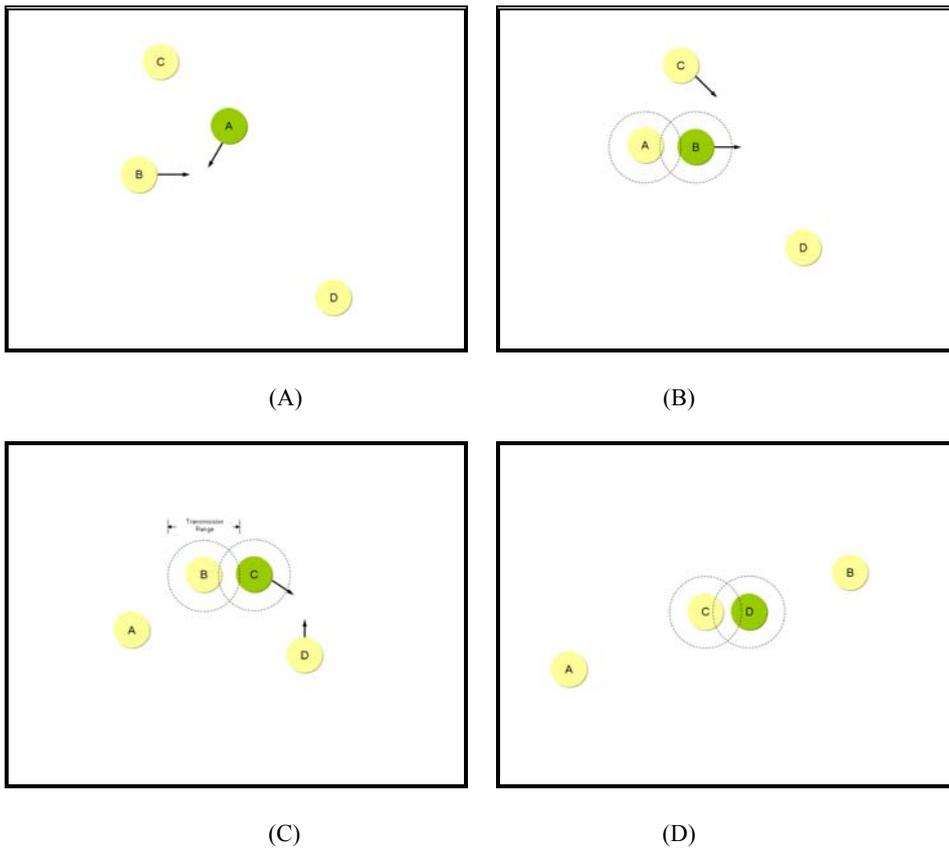


FIGURE 2-2: DTMANET ROUTING PROCEDURE

However, in a DTMANET it is common that no such route will ever exist between the source and the destination [Leguay *et al*, 2005]. Therefore, messages may have to be buffered for a time by intermediate peers, and the mobility of those nodes must be exploited to bring messages closer to their destination by exchanging messages between nodes as they meet. *Figure 2-2-A* to *Figure 2-2-C* is a simplified scenario that shows how the nodes leverage their mobility to successfully deliver a message to its destination by using a cache-and-forward mechanism in a DTMANET. In *Figure 2-2-A*, “Peer A” initializes the transmission of a message to be delivered to “Peer D”; since there is no direct route between peers “A” and “D”, the delivery of the message has to rely on the mobility of peers that allow the message first to be transferred to “Peer B”, then to “Peer C”, and finally, when “Peer C” moves within the transmission range of “Peer D”, to complete the delivery.

2.3.3 DTMANET Foundation Routing - Stateless Routing

DTMANET foundation routing refers to the algorithms that rely only on peer mobility and on nothing else to establish a communication route. Some of the examples are “Custody Transfer Routing” and “Epidemic Routing”.

2.3.3.1 Stateless Routing Algorithm

In a DTMANET, it is the norm that there is no detailed information such as contact and the previous and future available locations of peers in the network when communication is attempted [Widmer *et al*, 2005]. Therefore, there is no requirement for the phase of initial network topology information probing and gathering, which is commonly used in a number of non-“Delay Tolerant” MANET routing mechanisms. Stateless routing algorithms use pair-wise message duplication and/or removal techniques [Widmer *et al*, 2005]. Stateless routing is used as the foundation routing algorithm in OCI simulation.

2.3.3.2 “Epidemic Routing”

One of the existing examples of stateless routing is ‘Epidemic Routing’. Proposed by Vahdat and Becker, “Epidemic Routing” [Vahdat, 2000] is a protocol designed for an intermittently connected DTMANET. Vahdat and Becker claimed that the goals of epidemic routing are:

- 1) To maximize message delivery ratio
- 2) To minimize message end-to-end delivery time
- 3) To minimize the total message duplication during the message routing process.

An “Epidemic Routing Protocol” uses an epidemic algorithm [Demers *et al*, 1987] [Vogels *et al*, 2002], which provides eventual message delivery from source peer to any arbitrary destination peer. The routing process does not require any information or make any assumption regarding the underlying network topology and connectivity. Instead the “Epidemic Routing Protocol” relies on peer mobility to route a message, in which a pair of peers make contact and exchange messages. While in contact, peers exchange and cache messages. Peers have no actual route to the messages’ destination at the moment the messages are exchanged. However, after several contacts, with transitive caching and relaying, messages will eventually be delivered to their final destinations. Messages are globally uniquely identified in “Epidemic Routing”, and message exchanging records are kept on each peer to prevent a routing loop. When a message is initialized, it will spread like the epidemic of a virus, where the message exchange process is like a virus infection. Under “Epidemic Routing”, each message to be sent contains source and destination IDs. In addition it also has a field to hold the routing hop depth value, which is used to control the number of duplications when a message is relayed. For instance, if a message has its hop depth set to one, it will relay only to its final destination peer when it is in the transmission range and no other peers will participate in the routing process. In the simulation the authors have shown that through selecting appropriate Hop-to-Live (HTL) threshold values, message delivery ratios can still be reasonable while message duplication will be reduced [Cerf *et al*, 2002].

2.3.4 Probabilistic Routing – Global Knowledge Based

2.3.4.1 Probabilistic Routing

In most of the DTMANET research projects, software relies on random algorithms to simulate peer movement. However, in real life, mobility sometimes has predictable behaviours and patterns. For instance, if a peer has frequently travelled to point A over a constant period of time, it is most likely that this peer will travel to point A again in the near future. Such behaviour can be used to aid in the DTMANET routing decision-making processes by predicting message delivery in a probabilistic fashion. Such algorithms are called probabilistic routing algorithms. For example, [Lindgren *et al.*, 2003], a proposed probabilistic routing protocol called PROPHET, which works as follows:

After a message has been initialized from the source peer, it will rely on peer mobility for delivery to its destination. When the peer with the message encounters other peers in its transmission range, they exchange metrics that include delivery predictability information regarding a certain message delivery destination. This information helps message-carrying peers to choose a routing strategy. In the routing decision-making process, peer movement patterns and destination encounter probabilities are compared, and the message will be relayed to the peer with the highest probability value.

2.3.4.2 Probabilistic Routing Classification

With the different assumptions, objectives and sources of knowledge that exists, Probabilistic Routing approaches can be classified as:

1) Location based routing algorithms

In some DTMANETs, information about a peer's location is available, not only its previous and current locations, but also its future locations. In these cases, the location information can be used in a DTMANET for the purpose of message

relay. The Probabilistic Orbit Mobility Model [Ghosh *et al*, 2006] is one such example. *Figure 2-3* shows a three-level social orbital DTN, which uses a peer’s mobility pattern to route messages in different clusters in the network.

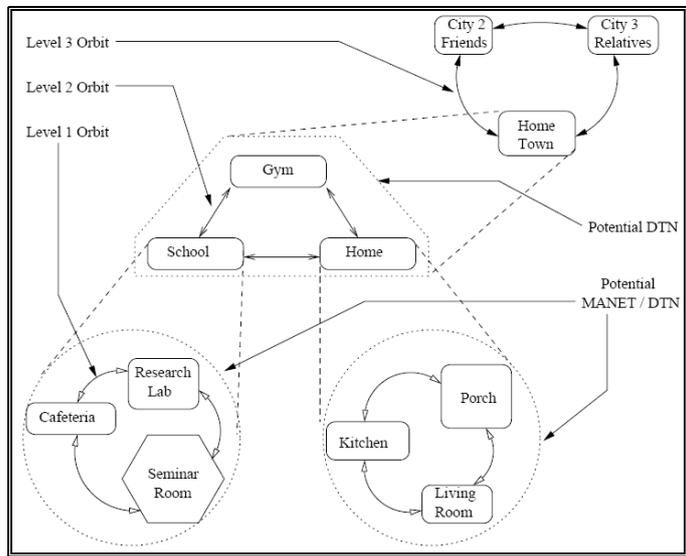


FIGURE 2-3: SOLAR ROUTING [Ghosh *et al*, 2006]

2) Movement based routing algorithms

When motion information, such as movement patterns and waypoints are available they can be used in a DTMANET to make movement-based routing decisions. *Figure 2-4* [LeBrun, 2005] is an example of “Knowledge-Based Opportunistic Forwarding MOVE”, which proposes a technique of calculating the predictable shortest distance to the destination peer.

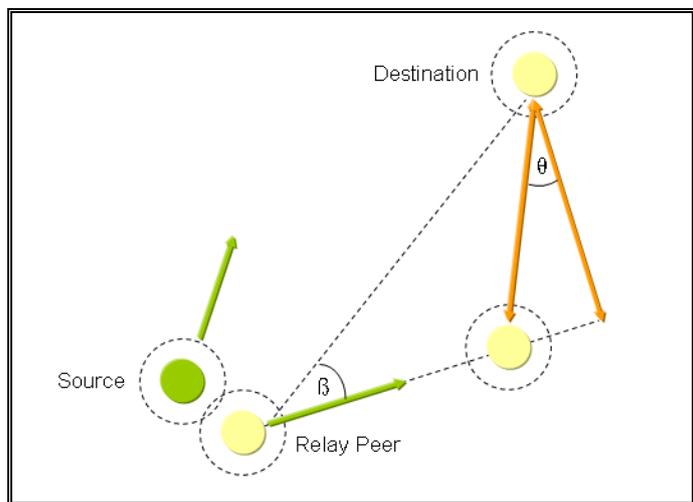


FIGURE 2-4: MOVEMENT BASED ROUTING ALGORITHMS

3) History-based routing algorithm

History-based algorithms leverage records of encounter history such as contact time, contact location, and contact frequency, to establish routing decision-making guidelines, as a result of the repetitive patterns of peer movement [Wang *et al*, 2005]. Unlike a stateless algorithm, this algorithm needs to go through the phase of network initialization and peer information acquisition.

4) Scheduling based routing algorithms

When the location of a mobile region and its timely movement patterns are known, messages in a DTMANET can be routed according to routing schedules.

Scheduling based routing algorithms include:

- a) On-Demand Scheduling
- b) Periodic Scheduling
- c) Storage-Based Scheduling

One of the examples of scheduling-based DTMANET routing protocols is the Inter-Regional Messenger Scheduling protocol [Khaled *et al*, 2006] and another example was the previously discussed Wizzy Digital Courier.

2.3.5 Opportunistic Routing – Local-Knowledge Based

Opportunistic routing is a newly emerged mechanism [Wang *et al*, 2005] [LeBrun, 2005] in DTMANET. Unlike probabilistic routing protocols, opportunistic routing protocols do not rely on movement history and movement probability prediction. Instead opportunistic routing protocols leverage the diversity of the physical differences of mobile peers, such as peer movement territory, peer velocity, peer message caching capacity and so on to make intelligent opportunistic routing decisions. The research of this thesis into the OCI concept is opportunistic based, so a wide range of routing protocols are used and will be discussed in detail in the subsequent chapters.

2.4 Related Work and Open Issues

2.4.1 Related Work

Currently, there are a few other DTMANET research projects that adopt approaches that are related to this work and they are:

1) Disconnected Transitive Communication Protocol (DTC)

This protocol, developed by Chen [Chen *et al*, 2001], proposed an approach that used an application-level function to make routing-forward decisions based on selective mechanisms among currently connected peer clusters. After every message relay, the peer that is currently in possession of the message is considered to be closer to the destination.

2) Ad hoc communication Localization

This research [Grossglauser *et al*, 2002] identified the problem of communication interference within mobile ad hoc networks. Grossglauser *et al* proposed a solution that recommended localizing of communications between juxtaposed peers within a peer cluster, while using peer mobility to carry messages from cluster to cluster. This proposed solution reduces ad hoc communication interference.

3) Pollen network:

Pollen network uses people as a communication medium, and was proposed by Glance [Glance *et al*, 2001].

4) Trajectory-based ad hoc routing protocol

Li [Li *et al.*, 2003] proposed a mobility trajectory prediction and movement-coordination-based protocol, which targeted communication problems similar to those found in delay tolerant mobile ad hoc networks. Li's proposal uses the peer's mobility trajectory information exchange to predict a “connected” route for routing decision making. The limitation of this approach is that in most DTMANET scenarios, peers have autonomous mobility, which means that it is difficult to predict their movement trajectory and it is not likely that peers will coordinate their movement to facilitate communication. However, in a highly disciplined network, such as a military network, this approach might work.

5) Interrogation-based relay routing

For routing in ad hoc satellite space networks that have frequently changing topologies, Shen [Shen *et al.*, 2002] proposed a routing protocol called Interrogation-Based Relay Routing. In this protocol, peers (satellites) “interrogate” one another to acquire information regarding network topology in order to make routing decisions.

6) Smart-tag based data dissemination

This research, conducted by Beaufour [Beaufour *et al.*, 2002], set its focus on data dissemination in sensor networks. It uses an epidemic-like algorithm that leverages the mobility of smart-tagged mobile peers, to achieve data dissemination across widely spread and disconnected static peers.

2.4.2 Open Issues and the Research Focus of OCI

In the survey of current investigations into DTMANET opportunistic routing protocols, it was found that each of the protocols leverages one of the characteristics which affect the success of the opportunistic routing algorithm. The results of these

studies reflect varying degrees of positive impact on DTMANET routing. However, each of these research projects focuses on only a single aspect of the opportunistic element, and neglects the interactions between different opportunistic elements. In contrast to this, the OCI concept put forward in this thesis offers a framework in which each of the opportunistic elements of peer characteristic and behaviour can be modelled, simulated, analyzed, and weighted. In addition, the OCI approach investigates the effects of each opportunistic element on other elements when used together, and therefore provides an overall confidence index for opportunistic routing. OCI indexes serve as guidelines to enable and optimize multiple opportunistic routing protocol overlays, through which may be achieved a balanced improvement of routing reliability, performance, and efficiency (see *section 3.1*).

2.5 Network Routing Simulation

In order to investigation and validate our hypothesis of OCI, different DTMANET scenarios need to be tested using a simulation approach, so that message delivery ratio, delivery latency, message duplication overhead and other metrics can be evaluated quantitatively. In the following sections, four of the existing network simulators are introduced, and their suitability for supporting the OCI concept discussed, using the following criteria:

- 1) To which network architecture is the simulator targeted?
- 2) Which protocols does the simulator support?
- 3) Does the simulator satisfy the requirements of the OCI hypothesis?
- 4) Is the simulator extensible and open for modification to suit new needs of OCI?

In addition, the quality of documentation and ease of use of the systems reviewed also affected the choice of simulation environment.

2.5.1 Existing Simulators

During the course of this research, four of the existing network simulators were investigated for their suitability as testing environments to support the research of OCI. They are NS2 [Johnson *et al.*, 1999], GloMoSim [Nuevo, 2004], QualNet [Kurkowski *et al.*, 2004] and OPNET [Haq *et al.*, 2005]. From the investigations, the following observations were made.

These simulators are designed to simulate a fixed wireless network with limited extension to an ad hoc mobile network. The simulation focuses on low-level network performance such as packet throughput and drop rate, while the research of OCI requires a higher level benchmark such as average delivery rate, average delivery time, and overall message duplication number.

Among the four network simulators a range of routing protocols are supported, which include DSR [Johnson *et al.*, 2004], AODV [Schumacher, 2004], DSDV [He, 2002] and TORA [Bertsekas, 2002], AODV+ [Chin, 2003], AODV-UU [Wiberg, 2003], MAODV [Zhu *et al.*, 2004], ODMRP [Lee *et al.*, 2002], SEAD [SEAD, 2007], ADMR [Jorjeta *et al.*, 2001], ZRP [Ray, 1999], WRP [Royer *et al.*, 2005], Fisheye [Yang *et al.*, 2005] and LAR [Ko *et al.*, 2004]. Although the listed routing protocols are mobile and ad hoc in nature, they do not support delay tolerant communication and none of them supports opportunistic routing.

All four simulators support third-party extensions. However, it became evident through our investigations that extending an existing general-purpose network simulation tool was not an ideal approach. Support for the delay tolerant and opportunistic routing aspects of the study were not natural bedfellows with the base simulation engines designed with more conventional network interaction in mind, and the extension proposals soon took on unattractive levels of clumsiness. The general-purpose nature of the existing network simulators also made the hiding of lower level detail to achieve a level of abstraction, well matched to the DTMANETs, difficult.

2.5.2 OCI-SIM

From the discussions in *section 2.5.1*, it is clear that a simulator targeting DTMANET opportunistic routing simulation is needed. To date, there is no existing network simulator which fits the requirements. Therefore, we developed OCI-SIM. OCI-SIM is designed specifically for delay tolerant communication simulation. In OCI-SIM, communication payloads are defined as messages rather than packets as they are in existing simulators. The support for opportunistic routing protocols is integrated into the core design. The benchmark parameters are defined to suit the high-level system evaluation. In addition, OCI-SIM provides a wide range of useful functions such as simulation automation, opportunistic filtering, a mobility model and device customization, data analysis and routing visualization.

2.6 Chapter Review

In this chapter, firstly, the concept of delay tolerant networks was introduced, with a comparison of conventional mobile ad hoc networks, a definition of DTMANET, a brief history of message-oriented communication and potential applications of DTMANETs. Secondly, routing protocols for supporting DTMANETs were classified and current research on opportunistic DTMANET routing protocols along with the appropriateness of commonly used simulators was surveyed and discussed. In the next chapter, methods and algorithms used in designing an OCI will be introduced.

Chapter III: Research Approach

“When we try to pick out anything by itself, we find it hitched to everything else in the Universe.”

– John Muir

This chapter introduces the Opportunistic Confidence Index proposal and provides an overview of the research approach undertaken to evaluate its potential for routing messages within DTMANETs.

The chapter starts by characterizing the OCI concept. This is followed by a description of the research approach by means of a network simulation and a list of the assumptions made during the investigation of the OCI concept. Network evaluation methods adopted in OCI are then introduced, the OCI simulation system framework is presented, and movement models and routing algorithms used in OCI simulations are discussed. Finally, overall research techniques used within the OCI simulations such as simulation validity control, random model, and loop control are covered in this chapter.

3.1 The Concept of Opportunistic Confidence Index

In DTMANETs, the physical networking environments and peers (mobile devices) display a very high degree of characteristic and behavioural variation, such as peer velocity, movement pattern, communication protocol type, communication range, and routing participation willingness. Traditionally, diversity and variation in characteristics are considered to have only negative effects on network routing reliability, performance, and efficiency.

In recent studies [Lindgren *et al*, 2003] [Wang *et al*, 2005] [LeBrun, 2005], there are a number of proposals that leverage these characteristic variations of the network environment to aid the message routing process in DTMANETs. Such routing algorithms are known as opportunistic routing. The results of these studies reflect a number of highly

positive impacts on DTMANET routing reliability, performance and efficiency. However, each of these studies focuses on a single opportunistic element respectively, while the interaction and effects among different opportunistic elements are neglected. In this field of research, there is currently no investigation of the possibility of using multiple opportunistic elements collaboratively.

In this work, OCI, as a routing framework, not only investigates each individual opportunistic element and its effects on message routing over DTMANETs, but also, more importantly, conducts a detailed study of the interactions and effects of opportunistic elements on one another when used together and their overall influence on the routing process, which provides an overall confidence index for intelligently applying multiple opportunistic routing protocols to DTMANETs. In OCI, each modelled opportunistic element is called a routing filter. Routing filters can be applied to the network individually or as a group. OCI offers a general framework where each of the opportunistic elements of network and peer characteristics and behaviours can be modelled, simulated, analyzed, and weighted against one another. As a result, overall message routing reliability, performance, and efficiency are improved. OCI indexes serve as guidelines to enable multiple opportunistic routing protocols to be applied as an adaptive group to improve DTMANET routing quality.

3.2 The Network Simulation Approach

To facilitate an efficient study over a range of scenarios of the evolution of DTMANETs under the influence of the dynamic model that OCI represents, experiments are conducted in a simulation environment. Using the simulation tool-set described in the next chapter, the characteristics and behaviours of different entities in a DTMANET may be modelled and simulated. One of the significant advantages of the simulation approach is that various parameters of network entities can be modified in a controlled manner, to simulate network behaviour under different conditions. In addition, using the simulation approach gives an efficient and cost-effective way to

evaluate the concept of OCI, where large numbers of mobile devices, a variety of network scenarios, and a large number of repetitions for experiments are required. These would be unachievable with a real test-bed environment. However, the simulation approach has its limitations and these limitations are discussed in *section 7.3.1*.

3.3 Assumptions

In order to focus the research upon opportunistic routing protocol issues, to accommodate the limitations of a high level network simulation approach, and to facilitate different types of opportunistic elements, the following assumptions have been made while designing the OCI viability experiments.

1) Boundary assumption:

To enable OCI simulations with random walk, random waypoint, random direction and other peer movement model variations, peers in OCI simulations are assumed to move within a common fixed boundary area, unless an individual peer territory filter is specified.

2) Unique peer-identification assumption:

To simplify the peer-addressing and message-routing history tracing process in OCI, every peer is assumed to have a unique global ID.

3) Initial connection assumption:

Connections between OCI peers are assumed to be established instantaneously, as peers encounter one another, unless an initial connection time cost filter is applied.

4) Message relay assumption:

OCI peers are assumed to cache every incoming message and relay it to the next encountered peer, if that peer meets the requirement of selected opportunistic filters, unless peer reputation (the willingness of routing participation), and/or message priority opportunistic elements, and/or a device memory and a storage limit filter are specifically indicated.

5) Message physical size assumption:

OCI messages are assumed to be physically weightless, unless a message size filter and/or device memory and storage limit filter are specified.

6) Message transmission time assumption:

Unless a message transmission time cost filter is applied in an OCI simulation, the time used for message transmission from one peer to another is assumed to be instantaneous.

7) Information sharing assumption:

Each peer in an OCI simulation is assumed to be willing to exchange its opportunistic routing information, such as speed, encounter history, movement territory, and destination, with every other peer it encounters, unless a peer reputation filter is specified.

8) Power consumption assumption:

Peers in OCI simulations are assumed to have unlimited power unless a power consumption filter is specifically applied.

3.4 OCI Network Evaluation Methods

When conducting network-based research, a series of evaluation methods and techniques are commonly employed, such as network self-organization ability evaluation, peer network join-time evaluation, network self-recovering ability estimation, network scalability, real-time services support evaluation, bandwidth evaluation, QoS routing benchmarking, robustness testing, network density influence testing, mobility influence testing, network security evaluation, peer power consumption estimation, and asymmetry testing. For message-oriented networks such as a DTMANET, message delivery-delay-time testing, and message duplication overhead testing are also included. In our OCI simulations, the following methods are employed:

1) Network scalability evaluation:

Different sized DTMANETs, from a small network, which contains tens of peers to medium and large sized networks that are composed of hundreds and thousands of peers, are simulated in this research to evaluate the scalability of the OCI framework.

2) Peer density evaluation:

The impacts of peer density on DTMANETs are taken into consideration in OCI simulations. DTMANETs with different density are simulated with the aim of evaluating the impact that network density has on an OCI simulation.

3) Mobility testing:

Peers with different mobility patterns in the network are modelled as movement models and their influence on the QoS of DTMANETs is investigated in order to evaluate the applicability of OCI to different types of mobility.

4) Asymmetry testing:

Asymmetrical networks, in which peers with different communication protocols, capacities and capabilities exist, are simulated in OCI simulations in order to evaluate the usefulness of OCI to heterogeneous, delay-tolerant, mobile ad hoc networks.

5) Opportunistic routing Protocol QoS evaluation:

The theory of OCI is based on evaluating, weighting, and indexing opportunistic routing protocols and their combined effects on DTMANETs. To do so, the following aspects are evaluated.

a) Message delivery rate evaluation:

In OCI simulations, this evaluation tracks the message delivery probability of DTMANETs with a given combination of opportunistic routing protocols within a certain time threshold, and/or hops threshold. The purpose of this test is to provide a benchmark for evaluating OCI reliability.

b) Message delivery delay testing:

Message delivery delay testing monitors the average time delay for successfully delivered messages in each OCI simulation. The aim of this testing is to identify the performance of opportunistic routing protocols with OCI.

c) Message delivery overhead evaluation:

For certain types of simulations such as propagation-based routing testing, OCI simulations produce message delivery overhead ratings, such as routing depth and message delivery duplication. This enables a comparison of the efficiency of OCI in different simulation scenarios.

3.5 Design of the OCI Simulation Environment

In this section, the system models and the design rationale of the OCI simulation environment are discussed

3.5.1 Design Rationale

The design of the OCI simulation environment has taken into consideration the perspectives of time and space, as well as the need for a rationalized simplification.

3.5.1.1 The perspective of Time in OCI

The understanding of time and the perspectives of time is essential in OCI research. In philosophy, phenomenology, religion, and science, people commonly attribute different interpretations and expectations to the concept of time. Terms like day-time, life-time, and cosmic time, are some examples of the different perspectives adopted towards time. In our research into the concept of OCI, time is modelled in two ways, using linear and non-linear models, and the measurement of a unit of time is defined abstractly.

1) *Linear time model T*

$$T = \sum_{i=0}^{\infty} t_{\Delta} = +\infty$$

This time model is strictly linear, extending in opposite directions into a past and a future that never meet. The origin of time in an OCI simulation is a moment that has no “before,” and an end time that has no “after”. Analysis of our system under the linear model falls on a finite line segment contained within this infinite line. This model is employed for OCI scenario specific simulations such as OCI visualization (see *section 4.2.5*), where opportunistic routing protocols are investigated within a given fragment of time.

2) *Non-linear self-iterate recursive model T_δ*

$$T = \begin{cases} \sum_{i=1}^{\delta} t_{\Delta} = t_{\Delta} \times \delta & (0 < i < \delta) \\ 0 & (i = \delta) \cup (i = 0) \end{cases}$$

In this model, time is cyclic. Study of an OCI system under the non-linear model focuses on timely recursive patterns and their effects on DTMANET communication. The non-linear model is used to represent history-based opportunistic movement patterns and it is also used for constructing automated OCI simulations (see *section 4.2.3*).

3) *Time measurement unit t_{Δ}*

In OCI simulations, time measurement is not bound to common units such as millisecond, second, minute and hour. Instead, time unit, t_{Δ} , is defined as one complete cycle of the execution of movement mechanisms combined with the message communication attempts of all peers in the system. This implies that, when measured, t_{Δ} may show different readings on a conventional timing device given any t_{Δ} in different positions on the linear line or in the non-linear cycles. It also does not guarantee the same reading of t_{Δ} at the same position in different executions. However, given the same scenarios, the value measured with t_{Δ} will always be consistent. This strengthens the validity and accuracy of the theoretical simulations of OCI: the same input guarantees the same output in terms of the number of t_{Δ} , regardless of where, when and how the simulation runs.

These basic models provide an objective benchmark for OCI data analysis. At the same time, scenarios with a wide range of time and timing requirements can be

easily simulated in their theoretical conditions, with the substitution of t_{Δ} for any other time conventions desirable for the simulation. For instance, choosing the linear model, a disconnected inter-planet network can be simulated with the time-scale set to a year. Using the same system, the behaviour of a transient network in a battle field can also be studied where the time requirement is in seconds and minutes. Furthermore, DTMANET simulations can be conducted with recursive behaviour patterns using the non-linear model, such as the daily routing pattern of a public transport system network.

3.5.1.2 The Spatial Perspective in OCI

We live in a three dimensional world. However, due to the scale of most DTMANETs, network communications and peer mobility can be interpreted within two dimensions. In this research into the concept of OCI and in designing of the OCI-SIM, the concept of space is simplified into two dimensions.

1) The definition of space:

In OCI, space is defined as: an enclosed area measured by predefined boundaries. The distance between two points in the space is measured by a predefined special unit.

2) The measurement unit of space:

The measurement of space is not bound by conventional geographical distance such as meters or kilometres. Instead, it is defined as an abstract unit S_{Δ} .

3) The interpretation and representation of spatial simulation and visualization

In order to represent the simulation visually in OCI-SIM, the space is measured using the smallest unit that a monitor can display, one pixel. Since space is

defined in an abstract level, the visualization can be interpreted very differently, depending on the requirements of the simulation scenario. In this sense, the output of OCI-Visual can be viewed as a two dimensional map, much like a GPS system or as a geographical map or, a particle world, or even a collection of star-gate portals if one wishes.

3.5.1.3 Timer-Controlled Spatial Alteration Method

An OCI simulation is defined as a timer-controlled spatial alteration. The timer acts as a trigger generating an event. Every time the timer “ticks”, one time unit t_{Δ} elapses, and entities such as peers will alter their spatial parameters according to selected movement models. At the same time messages will be passed, based on routing algorithms. In an OCI simulation, with a continuous timer action, and correspondent entity reactions, a DTMANET scenario is thus simulated.

3.5.1.4 Simplified Simulation Environment

In *section 3.3*, the assumptions of this research were presented, implying that the OCI simulates a simplified DTMANET environment, or an abstraction of a physical DTMANET environment. This simplification hides the low-level aspects of DTMANETs which are not required to prove the concept, and keeps the focus on the opportunistic elements at the routing protocol level. Additionally, this simplification greatly reduced the simulation time-consumption for generating coarse-grained statistic pattern indexes, which require a large number of repetitions to be statistically sound and valid.

3.5.2 OCI System Model Overview

In this section, OCI system models are introduced which include the simulation environment model, the movement model and the routing protocol model.

3.5.2.1 The OCI Simulation Environment:

The OCI simulation environment can be described as follows:

- 1) A timer-controlled two-dimensional DTMANET System with n number of mobile peers;

$$n \in \{0,1, 2, 3, 4, \dots, 1000\}.$$
²

- 2) At any time t_{Δ} , peers may alter their position $(x(t), y(t))$ with a certain velocity $v(t)$ and direction $\varphi(t)$, based on a peer movement model and obeying a predefined boundary rule:

$$(x_{t_{\Delta}+1}, y_{t_{\Delta}+1}) = f(x_{t_{\Delta}}, y_{t_{\Delta}}, v_{t_{\Delta}}, \varphi_{t_{\Delta}})$$

When $x_{t_{\Delta}} \geq X_{\text{WIDTH_LIMIT}}$ and/or $y_{t_{\Delta}} \geq Y_{\text{HEIGHT_LIMIT}}$;

- 3) At any given time t_{Δ} , there are m number of messages in the system;

$$m \in \{0,1,2, 3, 4, \dots, 1000\}.$$
³

- 4) Message caching and relaying follows specified DTMANET routing algorithms, and uses a set of opportunistic routing filters.
- 5) An OCI System initializes a simulation by randomly distributing peers within the simulation area. When the timer is started, peers may move according to given rules, and wait for message input. After messages are added, they will be relayed from peer to peer based on specified opportunistic routing protocols. Delivered

² The number of peers in OCI is limited to 1000 due to the constraint of the computational power of the machine on which simulations are carried out.

³ The total number of message allowed in OCI is limited to 1000 due to the constraint of the computational power of the machine on which simulations are carried out.

messages will be removed from the system, as well as messages that expired due to TTL (Time to Live) and/or Hops to Live (HTL) threshold control parameters. When there are no messages in the system, it goes back to the initial stage and waits for a new message to be input. The OCI system repeats this cycle until interrupted by the tester's instruction or a programmatic STOP_SIMULATION command in the OCI-SIM⁴ simulation configuration script.

3.5.2.2 Movement Models in OCI

In simulation studies of DTMANET, it is commonly desirable to model the movement of peers realistically and accurately. For small simulations, it is possible to manually specify the movement model used by individual nodes. However, to test a general concept such as OCI, it is very difficult to collect realistic movement data for every single simulation scenario. Instead, among the DTMANET research community, tradeoffs have been made to utilise several random movement models. These models are the random walk movement model, the random waypoint movement model and the random direction model [Spyropoulos *et al.*, 2005].

1) Random Waypoint Model

The random waypoint model is one of the most commonly used movement models for DTMANET simulations. Its algorithm is described as follows:

- a) Peers in the simulation environment randomly select a destination point (random waypoint) in the defined area.
- b) Upon the start of a simulation, peers move with a constant pace ν ; ν is chosen within a defined minimum and maximum value pair from a uniform distribution or a Gaussian distribution.

$$\nu \in [\nu_{\min}, \nu_{\max}]$$

⁴ See Chapter IV for detailed description of OCI-SIM

- c) Peers travel in a straight line to the waypoint.
- d) Upon reaching the waypoint, a peer pauses for a certain time, and then chooses another random waypoint.
- e) The process repeats until the end of the simulation.
- f) In the random waypoint model, a peer is defined by its current parameters over the function of time t :
 - i. current peer space vector $(x(t), y(t))$.
 - ii. current waypoint space vector $(x_{wp}(t), y_{wp}(t))$.
 - iii. current speed vector $v(t)$.
- g) The relative speed RS between peer i and peer j in a given time t in a random waypoint model, can be described as:

$$RS(i, j, t) = |V_j(t) - V_i(t)|$$

And the movement metric (M) can be formally described as [Sen, 2006]

$$\bar{M} = \frac{1}{|i, j|} \sum_{i=1}^N \sum_{j=i+1}^N \frac{1}{T} \int_0^T RS(i, j, t) dt$$

2) Random Direction Model

Instead of randomly selecting a destination waypoint, a peer in a random direction model chooses the direction randomly, and its algorithm is described as:

- a) Peers in the simulation environment randomly select a direction φ taken from an evenly distributed interval on $[0, 2\pi]$.

- b) Peers travel with the direction φ and speed ν for a certain period of time. After that time has elapsed, a new direction is chosen.
- c) When a peer reaches a predefined boundary it obeys one of the following boundary rules:
 - i. Reflection rule: a peer will be reflected at an angle $\theta(t)$ or $\Pi - \theta(t)$.
 - ii. Remove and add rule: a peer will be removed from the boundary and placed at a new location within the defined area of the simulation.
 - iii. Wrapping rule: a peer will wrap around and re-enter the boundary on the opposite side.
- d) Peers repeat this process until the end of the simulation.
- e) Peer speed in the random direction model can be described as:

$$\{\nu(t).\cos \theta(t), \nu(t). \sin \theta(t)\}$$

3) Random Walk Movement Model

The random walk movement model can be considered to be a special case of the random waypoint model that does not pause for a certain period before changing to another waypoint.

4) OCI Implementation of the Random Movement Model

In OCI simulations, all the characteristics of these three movement models are combined and implemented. In addition, for special scenarios such as scheduled or uniform DTMANETs, movement models like the linear movement model and

the chaos movement⁵ model are also introduced and available for simulation. OCI simulations allow different movement models to be used in conjunction with each other in simulation scenarios, and therefore provides a more diverse simulation environment.

3.5.2.3 DTMANET Routing Models

Routing models in OCI can be classified into two major categories. They are opportunistic models (OCI-filters) and non-opportunistic routing models (base routing protocols).

1) The basic message transmission classification in OCI is as follows:

a) End-to-end, Unicast:

In OCI simulations, unicast transmission is adopted to evaluate end-to-end routing reliability and the routing performance of a given set of opportunistic routing protocols in a specified DTMANET scenario.

b) One-to-many, Multicast:

Multicast transmission protocols are employed in OCI simulations to evaluate the routing overhead and efficiency of a given set of opportunistic routing protocols.

c) Many-to-one (many-to-many):

In some OCI simulations, multi-entry multicast transmissions are used to test the routing performance.

⁵ Chaos movement model follow no movement rule, at any given time, the location of each peer is randomly assigned.

2) Non-Opportunistic Base Routing Protocols Chosen for OCI Simulation

To support an OCI simulation in a DTMANET environment, two categories of non-opportunistic routing protocols were selected as the foundation protocols. They are flood-based routing protocols such as epidemic routing and message-custody-transfer-based routing protocols. Choices are made based on the need to cover different DTMANET scenario requirements. For instance, a notification system over a DTMANET is time critical and prefers the fast spread of messages in a network, and therefore flood-based routing is suitable for this situation. On the other hand, a media content distribution system over a DTMANET has to consider the possibility of much higher traffic congestion and a need to control the transmission overhead; custody transfer is favoured in this scenario. Other scenarios are mostly variations on these two basic situations.

- a) Flood-based routing algorithm – Epidemic Routing [Vahdat et al, 2000] [Demers et al, 1987]

The design of Epidemic routing is to maximize the message delivery rate and minimize delivery delay. The Epidemic Model is described as:

$$\frac{dI(t)}{dt} = \beta I(t)[N - I(t)]/N$$

Where N is the total number of the susceptible peers in a given DTMANET population, $I(t)$ is the number of infected peers at time t , and β is the rate at which a given infected node probes the total, susceptible population of peers [Cole, 2004].

- b) Custody-transfer-based routing algorithm – End-to-end routing

The design focus of custody-transfer routing is to minimize message delivery overheads while balancing message delivery rates and delays. Using

DTMANET end-to-end routing, a message is relayed from one peer to another and at the same time the custody of the message is handed over. The transfer of custody occurs immediately when two suitable peers encounter each other.

c) OCI implementation of foundation routing protocols

Four variations of foundation routing protocols are implemented in the OCI simulation environment. These cover most of the commonly used non-opportunistic routing scenarios [Jain et al, 2005] [Leguay et al, 2005], and thus demonstrate the general applicability of the concept of OCI. These chosen routing protocols are:

i. OCI “Custody Transfer” Routing:

The OCI implementation of “Custody Transfer” Routing protocol is a one-to-one-based routing protocol in which messages are relayed through peer encounters. At each successful message relay, the ownership of the message is also relayed, and therefore the message is removed from the sender or previous owner. At any given time, only one copy of each message is allowed in the system.

ii. OCI “Custody Transfer Propagation” Routing:

The “Custody Transfer Propagation” Routing protocol is a combination of the custody transfer and the epidemic routing protocol. In a “Custody Transfer Propagation” model, a message is broadcast and duplicated to all peers that are in the transmission range of message-carrying peers. After each successful message relay, the ownership of the message is relayed to multiple peers. In this way, the message is removed from the message-carrying peers. At any given time, there are multiple copies of each distinct message in the system.

iii. OCI “Custody-Retaining-Propagation” Routing

The “Custody-Retaining-Propagation” routing protocol is also an epidemic-like propagation model. However, only the original message owner can broadcast and duplicate a message. Other peers will hand over the message only to the next encountered peer, and then delete the message from themselves. The message, however, still remains in the custody of the initial sender.

iv. OCI “Simple Flood Propagation” Routing

As the name suggests, in “Simple Flood Propagation” routing, every peer carrying a message, sender or router, can broadcast and duplicate a message to other peers in-range.

3) Opportunistic-Routing Filters Used in OCI

In an OCI framework, opportunistic elements are modelled and referred to as opportunistic filters.

a) Mobility-aided opportunistic-routing protocols

Mobility-aided opportunistic-routing protocols leverage opportunistic elements extracted from peer movements to optimize routing performance. In OCI three mobile opportunistic elements are modelled and implemented. They are:

i. the peer velocity opportunistic filter

In OCI, peer velocity is used to aid the message-relaying decision-making process. In some scenarios, higher velocity peers have precedence, while in other situations peers with lower velocity (more stable) are selected.

ii. the peer movement territory opportunistic filter

Peers with wider territory have higher potential to meet more peers in a DTMANET.

iii. the peer movement consistency opportunistic filter

Peer movement consistency is another mobility attribute that is used to provide opportunistic reference for OCI routing decision making. The higher the value, the less chance there is for a peer to change its movement pattern. Movement consistency preferences are changed from scenario to scenario.

b) Device-Capability-Based Opportunistic Protocols

In device-capability-based opportunistic protocols, opportunistic elements are variations of the physical capability of mobile devices, which include:

i. Device transmission range opportunistic filter

In OCI simulations, by applying the device transmission range filter, a message will favour devices with a wider transmission range over peers with a narrower transmission range.

ii. Device communication versatility opportunistic filter

When this element is applied, a message will select a device with more versatile communication capabilities over peers which support fewer communication protocols.

c) Peer-social-behaviour-guided opportunistic protocols

In OCI, peer social behaviours are modelled and used as opportunistic filters.

i. Peer reputation opportunistic filter

Willingness to pass messages and a successful message-passing rate is referred to as peer reputation in OCI simulations. When applying this filter, in a competing situation, the peer with the highest reputation value gets the right of message custody.

ii. Peer-encounter-history opportunistic filter

A peer-encounter-history filter sets the peer encounter history as the opportunistic metric. During the decision-making process of message passing, peers with higher encounter rates have a higher precedence.

d) Other opportunistic filters

During the simulation process of OCI, new opportunistic routing protocols can be added through the interfaces of OCI-CUSTOM, which will be introduced in the next chapter.

3.5.3 DTMANET Scenario Dynamic Element Generation

A DTMANET environment contains a wide array of dynamic elements such as movement, communication capacity, and social behaviour. To simulate a DTMANET network, these elements must be modelled, and methods must be devised to reproduce and represent these dynamics.

3.5.3.1 Dynamic Scenario Recreation Methods

There are two commonly adopted methods within research communities that serve to recreate dynamic scenarios, and these are now itemized:

1) Live data-collection method:

Using this method, dynamic data is collected and recorded so that a simulation will represent the dynamics using real-life data. This is an important way to conduct realistic simulations. However, it has a number of significant drawbacks. Firstly, to collect and record real-life data is costly and time consuming. Secondly, because data collection is finite, simulations can run for only a certain length of time. Finally, recorded dynamic values are sets of constants, and therefore cannot fully represent the dynamics.

2) Random data-generation method:

A common alternative to the life data collection approach is to use random attribute generation methods. To generate random attributes, there are two principal methods. The first takes a physical phenomenon such as key stroke timing, atmospheric noise and radioactive decay, which is expected to be random, and then uses it to compensate possible biases in the random attribute generation process. The second applies mathematical algorithms, which produce sequences of attributes that appear to be random. However, the computational results are in fact completely determined by a shorter initial value, known as a seed or key. This computational approach is commonly referred to as pseudo random number generation (PRNG) [Luby, 1996] [Knuth, 1997].

3.5.3.2 PRNG Used in OCI

In OCI simulations, PRNG is selected to generate dynamic scenarios, because PRNG can be ideally incorporated into the tool sets developed to support the research of OCI and it provides enough accuracy since OCI dynamic scenarios have a relatively small numeric range requirement. The PRNG algorithm used in the OCI simulations is SHA1-PRNG. It is part of the Sun Java API [JavaAPI, 2006] [NIST, 2007] as: [*java.util.Random*]

3.5.3.3 Randomly Generated Dynamic Attributes in OCI

Scenarios modelled and simulated in OCI rely on PRNG number generations to represent dynamic elements. These elements are:

1) Basic Dynamic Attributes

a) Random peer distribution:

In an OCI simulation, a given number of peers are randomly distributed over a predefined simulation area.

b) Random peer physical attributes:

i. Peer communication range stack

A peer's communication range is randomly generated within the range of zero to the theoretical protocol communication range limit value.

ii. Peer movement velocity

A peer's movement velocity is randomly generated within the range of zero (stationary) to the maximum theoretical device movement speed limit.

iii. Peer movement territory

A peer's movement territory is randomly generated within the simulation area boundary.

iv. Peer movement direction

A peer's movement direction is randomly generated from either the random direction movement model or the random walk movement model employed in the OCI simulation.

c) Random peer social behaviour attributes:

i. Peer movement consistency

A peer's movement consistency is randomly generated between the range of a given constant pair ($Consistency_{min}$, $Consistency_{max}$). The higher the number is, the less the possibility of a peer to change its movement pattern.

ii. Peer message relay consistency

A peer's message relay consistency is randomly generated between the range of a given constant pair ($Consistency_{min}$, $Consistency_{max}$). The larger the number is, the higher the likelihood of a successful message passing.

d) Random message allocation

In an OCI simulation scenario, a message's origin and destination, (*sender*, *receiver*) pair, are randomly selected among available peers.

2) Derived Random elements

From the interaction among basic random elements, more random elements can be derived, such as peer encounter rate, network topology and so on. With the help of PRNG, OCI offers a full dynamic simulation environment.

3.6 OCI Confidence Index Generating Method

The methods employed for generating OCI Index in this research follows four basic steps. They are opportunistic element selection, one-to-one opportunistic filter comparison, opportunistic filter combination and permutation, weight assignment and voting procedure and finally OCI index generation.

3.6.1 Opportunistic Element Selection

In any given DTMANET scenario, there are a large number of opportunistic elements. The first step is to select those that have significant impact on one or more aspects of routing reliability, performance, and efficiency.

In an OCI simulation, the selection process is conducted in an automated fashion. The OCI simulator (OCI-SIM: *see chapter IV*) takes parameters specifying the range of value variations of the opportunistic element, and runs the simulation with certain increase step a number of times from the lowest threshold to the highest range limit. An example is the peer velocity opportunistic element:

$$\{\text{PeerVelocity_Filter}\}$$

The range of velocity variation is:

$$v \in [v_{\min}, v_{\max}]$$

Given that $v_{\min} = 0$, $v_{\max} = 20$, $v_{\text{step}} = 5$, with a result of five test sets and five result sets.

Test Sets:

[$v = 0$]

[$v = 5$]

[$v = 10$]

[$v = 15$]

[$v = 20$]

And each test has its own corresponding result set:

{Delivery_Rate, Delivery_Time, Routing_Depth, Message_Duplication}

The results of each simulation with the specific value are compared to reflect the level of impact of selected opportunistic element. The greater the difference between two thresholds, the greater the effect of the element.

3.6.2 One-to-one Opportunistic Filter Comparison

Using this method, each of the opportunistic filters subjected to OCI simulation is isolated and tested individually under a given scenario. For instance, if there are n ($n=5$) candidate opportunistic filters:

{Filter_A, Filter_B, Filter_C, Filter_D, Filter_E}

There will be a total of n ($n=5$) tests.

[Filter_A]

[Filter_B]

[Filter_C]

[Filter_D]

[Filter_E]

And each test has its own corresponding result set:

{Delivery_Rate, Delivery_Time, Delivery_RoutingDepth, Message_Duplication}

With this method, a message will be passed when the selected opportunistic filter condition is met. The result will provide a single opportunistic filter based “importance index” and “improvement value set” which are used to assist the weight allocation process described in *section 3.6.4*.

3.6.3 Opportunistic Filter Combination and Permutation

3.6.3.1 Filter Combination

With the combination method, opportunistic filters can be applied to a scenario as groups. The method of filter combination is defined as follow:

$$C = \sum_{i=1}^n C_n^i = \sum_{i=1}^n \frac{P_n^i}{i!} = \sum_{i=1}^n \frac{n!}{i! \times (n-i)!}$$

Where C is the number of possible combinations, n is the number of selected filters.

For example, if the selected filter group is:

$$\{\text{Filter_A}, \text{Filter_B}, \text{Filter_C}\}$$

Its absolute combination set is:

[Filter_A]

[Filter_B]

[Filter_C]

[Filter_A, Filter_B]

[Filter_A, Filter_C]

[Filter_B, Filter_C]

[Filter_A, Filter_B, Filter_C]

This combination method provides indexes of opportunistic filters of different grouping configuration. Within each configuration, unless intelligent voting methods are applied, filters are considered to be equally weighted.

3.6.3.2 Filter Permutation

In OCI simulations, the algorithm used to generate permutations starts from an initial combination of opportunistic filters. It computes subsequent permutations in lexicographic order. The algorithm will generate all permutations for a given combination of filters if the initial combination is the first in lexicographic order.

$$P = \sum_{i=1}^n P_n^i = \sum_{i=1}^n \frac{n!}{(n-i)!}$$

Where P is the number of possible permutations, while n is the number of selected opportunistic filters. For instance, if the selected filters are:

{Filter_A, Filter_B, Filter_C}

Its absolute permutation set is:

[Filter_A]
 [Filter_B]
 [Filter_C]
 [Filter_A, Filter_B]
 [Filter_B, Filter_A]
 [Filter_A, Filter_C]
 [Filter_C, Filter_A]
 [Filter_C, Filter_B]
 [Filter_B, Filter_C]
 [Filter_A, Filter_B, Filter_C]
 [Filter_A, Filter_C, Filter_B]
 [Filter_B, Filter_A, Filter_C]
 [Filter_B, Filter_C, Filter_A]
 [Filter_C, Filter_B, Filter_A]
 [Filter_C, Filter_A, Filter_B]

The permutation method provides indexes of opportunistic filters of different grouping configuration in an orderly fashion. Within each configuration, filters are weighted according to their index in the configuration.

3.6.4 Weight Assignment & Voting Procedure

In our work, the term “weight” is defined as the value assigned to opportunistic filter representing the importance of the filter. Through the use filter weight, routing decisions can thus be made in OCI simulations. There are two ways to assign a weight value for voting procedure for opportunistic filters.

3.6.4.1 Non-intelligent Weight Assignment and Voting methods

1) “Democratic” voting mode – for weightless filter combination

a) Weight assignment

In this mode, filters are considered to be equally weighted.

b) Voting Procedure

This mode is a pure democratic voting procedure used by the routing decision making process for configurations that are generated through weightless filter combination methods. There are a total of n votes if n opportunistic filters are used. Each opportunistic filter has one vote which can be either **Yea**, if the opportunistic condition of this filter is satisfied, or **Nay**, when the opportunistic condition is not met. A message will be relayed only if the total number of **Yea** votes is greater than the total number of **Nay** votes.

2) “Republican” voting mode – for index based filter Permutation

a) Weight assignment

When using the permutation technique to generate a confidence index for opportunistic filters, a weight is assigned to each filter according to its permutation indexing position.

For example: Given an opportunistic filter Set **F**:

$$\{\text{Filter_A}, \text{Filter_B}, \text{Filter_C}\}$$

The weight metrics are

$$\begin{aligned} \bar{M}_1 &= \begin{pmatrix} \text{Filter_A}, & 1 \\ \text{Filter_B}, & 2 \\ \text{Filter_C}, & 3 \end{pmatrix} & \bar{M}_2 &= \begin{pmatrix} \text{Filter_A}, & 1 \\ \text{Filter_B}, & 3 \\ \text{Filter_C}, & 2 \end{pmatrix} & \bar{M}_3 &= \begin{pmatrix} \text{Filter_A}, & 2 \\ \text{Filter_B}, & 1 \\ \text{Filter_C}, & 3 \end{pmatrix} \\ \bar{M}_4 &= \begin{pmatrix} \text{Filter_A}, & 2 \\ \text{Filter_B}, & 3 \\ \text{Filter_C}, & 1 \end{pmatrix} & \bar{M}_5 &= \begin{pmatrix} \text{Filter_A}, & 3 \\ \text{Filter_B}, & 1 \\ \text{Filter_C}, & 2 \end{pmatrix} & \bar{M}_6 &= \begin{pmatrix} \text{Filter_A}, & 3 \\ \text{Filter_B}, & 2 \\ \text{Filter_C}, & 1 \end{pmatrix} \end{aligned}$$

b) Voting Procedure

For each of the filters, their weight value will be positive if the peer meets the filter's condition, negative if not. A message will be relayed when the final sum of the total weight value is positive.

3.6.4.2 Intelligent Weight Assignment and Voting methods

Instead of relying on filter permutations to assign weights, or assume an equal weight in a weightless filter combination, a more intelligent weighting and voting method can be used in OCI simulations.

1) "Filter importance order based" weight assignment and voting mode

a) Weight assignment

As discussed in section 3.6.2, for each given set of opportunistic filters under a specified scenario, there is a "filter importance index" generated through a one-to-one filter comparison according to the level of improvements/tradeoffs. Using this index, weights are assigned to filters.

For example:

Given an opportunistic filter Set **F**:

{Filter_A, Filter_B, Filter_C, Filter_D, Filter_E}

There is a result set **R**:

{a, b, c, d, e}

In a time critical simulation scenario, *a* is dedicated to Filter_A's message delivery time value, *b* is Filter_B's message delivery time value and *c*, *d*, *e* are Filter_C, Filter_D, Filter_E's delivery time values respectively.

If $a > b > c > d > e$, then the filter weight set **W** is: {1, 2, 3, 4, 5}

In the term of delivery time, the smaller the value the better, therefore the filter with the smallest delivery time gets the highest weight. And because the value 0 has no effect over weight evaluation, 1 is chosen as the minimum weight value.

The filter weight metric $\bar{\mathbf{M}}$ is the value of a filter weight set **W** over filter set **F**:

$$\bar{\mathbf{M}} = \begin{pmatrix} \text{Filter_A,} & 1 \\ \text{Filter_B,} & 2 \\ \text{Filter_C,} & 3 \\ \text{Filter_D,} & 4 \\ \text{Filter_E,} & 5 \end{pmatrix}$$

b) Voting Procedure

Like republican voting procedures, according to the weight metrics, every filter represents a certain weight that was previously elected through the weighting process. These weight values are then used in a new round of voting.

The total number of votes for each of the filters equals to its weight value, and all votes of the same filter are the same. For instance, in the previous example, Filter_C has a weight value of 3, therefore, Filter_C has 3 votes. When Filter_C's opportunistic condition is satisfied, it will submit all three votes of **Yea**, otherwise all three votes of **Nay**. Under no circumstance, will Filter_C have votes such as 1 vote **Yea**, 2 votes **Nay** or 2 votes **Yea** and 1 vote **Nay**, which contradict each other.

Apparently, given n opportunistic filters, the total number of votes \mathbf{N} in a given scenario is:

$$\mathbf{N} = n!$$

The final decision D_{pass} will yield **YEA** and the message will be voted to pass if the sum of **Yea** votes across all filters is greater than the total number of **Nay** votes. The voting decision making process can be formally described as:

$$D_{\text{pass}} = \begin{cases} \text{YEA} & (\mathbf{N}_w > 0) \\ \text{NAE} & (\mathbf{N}_w \leq 0) \end{cases}$$

Where: \mathbf{N}_w is the sum of votes

$$\mathbf{N}_w = \sum_{i=1}^n i_w \times \beta$$

Given n is the number of employed opportunistic filters. Where

$$\beta = \begin{cases} 1 & (\text{Filter Condition} = \text{true}) \\ -1 & (\text{Filter Condition} = \text{false}) \end{cases}$$

2) “Filter improvement value based” weight assignment and voting mode

a) Weight assignment

As discussed in section 3.6.2, for each given set of opportunistic filters under a specified scenario, there is an “improvement value set”, which represents the increased percentage regarding to the control scenario (the result when no opportunistic filter is applied), generated through the one-to-one filter comparison. Using this data set, weights are assigned to the opportunistic filters.

For example:

Given an opportunistic filter Set **F**:

{Filter_A, Filter_B, Filter_C, Filter_D, Filter_E}

There is a result set **R**:

{5.23%, 74.34%, 35.59%, -6.01%, 118.10%}

In a time critical simulation scenario, the improvement percentages for the filters are calculated from the filters’ message delivery time values against the control value respectively.

The filter weight metric $\bar{\mathbf{M}}$ is the value of a result set **R** over filter set **F**:

$$\bar{\mathbf{M}} = \begin{pmatrix} \text{Filter_A,} & 5.23 \% \\ \text{Filter_B,} & 74.34 \% \\ \text{Filter_C,} & 35.59 \% \\ \text{Filter_D,} & -6.01 \% \\ \text{Filter_E,} & 118.10 \% \end{pmatrix}$$

b) Voting procedure

The final decision D_{pass} will yield **YEA** and the message will be voted to pass, if the sum of the weight values of all the filters is greater than the value of the best single improvement value, in this example, the value is 118.10%. The voting decision making process can be described as:

$$D_{\text{pass}} = \begin{cases} \text{YEA} & (N_W > \text{Weight}_{\text{high}}) \\ \text{NAE} & (N_W \leq \text{Weight}_{\text{high}}) \end{cases}$$

Where: N_W is the sum of votes, and $\text{Weight}_{\text{high}}$

$$N_W = \sum_{i=1}^n i_w \times \beta$$

Given n is the number of employed opportunistic filters. Where

$$\beta = \begin{cases} 1 & (\text{Filter Condition} = \text{true}) \\ -1 & (\text{Filter Condition} = \text{false}) \end{cases}$$

3) OCI choice of weight assignment and voting methods

OCI indexes can be generated through filter combination and permutation or through intelligent weight assignment. While the filter combination approach is mainly used for a quick test of the filter effects on one another, for simple scenarios where only a very limited number of opportunistic filters are used, the filter permutation method is the simplest and the most convenient way (one step operation using OCI-AUTO). However, while a large number of opportunistic filters are employed, the number of permutations will grow astronomically, and will therefore be unfit for a simulation. In such cases, a ‘‘Filter importance order’’ or ‘‘Filter improvement value’’ based, approach can be adopted to generate OCI indexes.

3.6.4.3 Permutation Number Limitations

From the discussion of the OCI generating techniques, we can calculate the possible numbers of combinations and permutations for each given opportunistic filter group.

OCI FILTER COMBINATION POSSIBILITIES		
Filter Num	Combination Num	Time
1 out of 13 filter:	13	13 secs
2 out of 13 filter:	78	1.3 mins
3 out of 13 filter:	286	4.46 mins
4 out of 13 filter:	715	11.91 mins
5 out of 13 filter:	1287	21.45 mins
6 out of 13 filter:	1616	26.93 mins
7 out of 13 filter:	1616	26.93 mins
8 out of 13 filter:	1287	21.45 mins
9 out of 13 filter:	715	11.91 mins
10 out of 13 filter:	286	4.46 mins
11 out of 13 filter:	78	1.3 mins
12 out of 13 filter:	13	13 secs
13 out of 13 filter:	1	1 sec
Total:	7991	2.22 hours

TABLE 3-1: OCI FILTER COMBINATION POSSIBILITIES

OCI FILTER PERMUTATION POSSIBILITIES		
Filter Num	Permutation Num	Time
1 filter:	1	1 sec
2 filters:	2	2 sec
3 filters:	6	6 sec
4 filters:	24	24 sec
5 filters:	120	2 min
6 filters:	720	12 min
7 filters:	5040	1.4 hour
8 filters:	40320	11.2 hour
9 filters:	362880	4.2 days
10 filters:	3628800	42 days
11 filters:	39916800	1.26 years
12 filters:	479001600	15.18 years
13 filters:	6227020800	197.45 years
Total:	6749977117	214.04 years

TABLE 3-2: OCI FILTER PERMUTATION POSSIBILITIES⁶

⁶ Assumption: Each repeat take same amount of time – one second

As shown in the *Table 3-1* and *Table 3-2*, an absolute permutation of a relatively small number of filters may take a very long time to complete, while in comparison, an absolute combination set of the same number of filters uses only a fragment of its time. For this exact reason, in this research OCI limits the automatic permutation within 6 filters at a time.

3.7 OCI Simulation Validity Control Mechanisms

The following control mechanisms/methods were employed to ensure the validity of this simulation based investigation. They are routing loop control, conformity control, statistic control, benchmarking control, and optimization control.

3.7.1 Routing Loop Control Method

Loop control in networks is a classic network technique to prevent package looping. It is essential for OCI simulations, consider the following scenario:

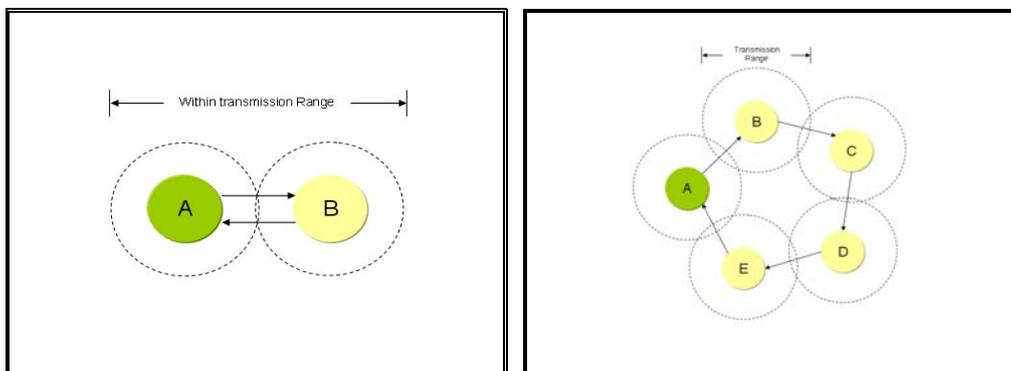


FIGURE 3-1: ROUTING LOOP CONTROL

In scenario one, a message will keep bouncing back and forth between Peer A and Peer B as long as they are in each other's transmission range and if a loop control is not employed. In scenario two, a message will eventually be passed back to its origin if there is no loop control mechanism.

In traditional fixed networks or conventional MANETs, algorithms such as the Distributed Bellman-Ford (DBF) [Walden, 2003] and Link State Algorithm [Jacquet *et al*, 2001] [Gray, 2005] efficiently avoid package looping by keeping a static topology structure. However, such approaches perform poorly when applied to networks with high-level dynamics and frequently changing topologies such as a DTMANET.

In an OCI simulation, it is impractical to track and keep a static topology record to avoid message looping, since the entire network changes constantly. Even if it is possible to acquire the topology information in a given time frame, such information will be rendered outdated and useless immediately after every topology change. Therefore, OCI simulations abandon the global approach, instead, each peer keeps identification records of passed messages, and thus, messages will not be looped back to peers that already relayed them.

3.7.2 Statistic control

In every OCI-SIM simulation, a statistic validity control is implemented to compute the optimized number of distinctive repetitions, so as to achieve a statistically sound simulation result for any given input. In experimental research, the method of auto repetition number increment is used. Here is an example:

In this example, we use the value of average delivery time and average delivery message duplication number as the scale. From *Table 3-3* we can see that for this specific input, in order to obtain truthful simulation we need at least around 100 repetitions. If it is less than this, the results will swing from simulation to simulation, as shown in *Figure 3.2*. To obtain accurate results we need to repeat the simulation around 500 times, however, any further increase of the number of simulation repetitions will not give significant accuracy gain, and it will introduce very heavy time overhead. During the course of the control simulation, OCI-AUTO adjusts the value of repetition until it reaches a balance point, which indicates that no more increases in repetition will be necessary, and it will use the smallest edge value within

the balanced range as the validity control repetition number.

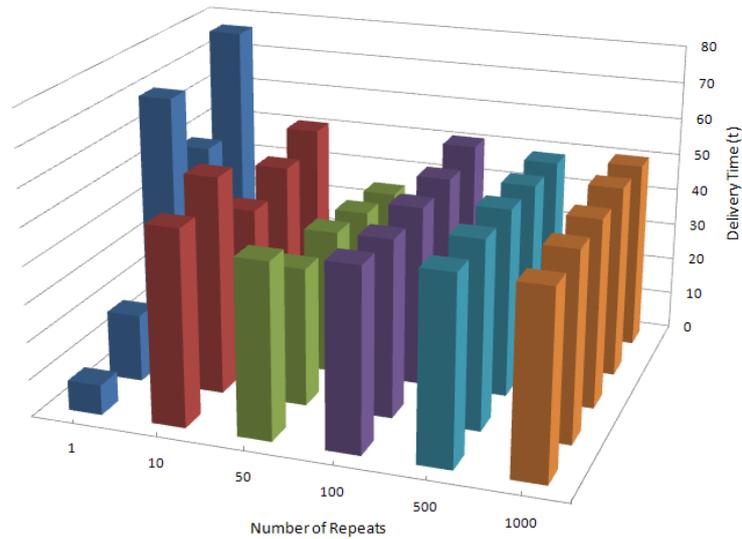


FIGURE 3-2: VALIDITY CONTROL (AVERAGE DELIVERY TIME)

REPETITION AND AVERAGE DUPLICATION VALUE						
repetition:	1	10	50	100	500	1000
A:	77	51	34.94	51.09	48.85	50.44
B:	50	46.82	36.72	48.69	49.40	51.31
C:	70	42	38.68	48.07	50.24	50.39
D:	18	58	36.72	47.28	50.12	50.45
E:	8	52.18	46.68	48.63	49.81	49.43

TABLE 3-3: VALIDITY CONTROL (AVERAGE DELIVERY DUPLICATION MESSAGE)

3.7.3 Benchmarking control

In OCI-SIM, validity controls such as the benchmarking control are used as benchmarking milestones to evaluate the subsequent simulations of a given input.

3.7.3.1 Theoretical value based benchmarking control

OCI-SIM has an option to perform simulations with the theoretical values of a given scenario. This feature is utilized to generate a benchmarking control for evaluating further and more realistic simulations for the purpose of OCI index generating, since it provides the best case scenarios for given configurations.

3.7.3.2 Routing Protocol based benchmarking control

One of these types of control scenarios is the “Simple Flood Propagation” mechanism, which results in the shortest time of delivery and at the same time the largest number of message duplication. Another control scenario uses message “Custody Transfer” based routing, which is on the complete opposite side to the flood mechanism. It keeps only one copy of the original message in the system at any given time, and hands over the message to a randomly selected device that is within its transmission range. This gives the longest delivery time and zero message duplication. The results of this control are mostly used in the data analysis state for the result of the main body of the simulation. For instance, any group with an average time value that is greater than “Custody Transfer” routing, or any duplication number that is greater than “Simple Flood Propagation” routing will be considered unsound, invalid data and will be excluded in the data analysis. The value of the benchmarking control simulation repetition is defined by the validity control simulation carried out in *section 3.7.3*.

3.7.4 Optimization control

OCI-SIM implements optimization control algorithms that use techniques such as auto Time-to-Live, Hops-to-Live and threshold value increasing. These results will provide optimized values of TTL and/or HTL as threshold values for subsequent simulations for optimized simulation performance (faster simulation speed) and efficiency (less unnecessary simulation repetition) (see *section 5.3.4*). The result generation process is similar to the benchmarking control; and it conforms to the statistic control.

3.8 Chapter Review

This chapter proposed the OCI concept, and described the investigation undertaken to establish its viability. It outlined the simulation approach, research assumptions, evaluation methods, OCI system model, movement model, opportunistic routing algorithms, simulation validity control, random model selection, and loop control techniques. The collection of algorithms, methods, and techniques described in this chapter represents the methodological foundation for the research into the concept of OCI. In chapter IV, OCI-SIM, the tool set created for supporting the OCI investigation will be introduced.

Chapter IV: The OCI Simulation Tool Set

“If the only tool you have is a hammer, you tend to see every problem as a nail.”

– Abraham Maslow

The purpose of this chapter is to describe the simulation tool set⁷ developed to support our investigation, to outline the facilities and to justify the design approach taken, and to validate its appropriateness as an investigative tool for the OCI concept.

4.1 DTMANET SIMULATION

4.1.1 Dynamic Heterogeneity

Two networks can simultaneously be classified as DTMANETs, while bearing a variety of highly variable and distinctive characteristics. For instance, the number of devices in a network can range from less than a dozen to thousands; they may communicate using the same communication protocol or a mixture of different protocols; they may represent a highly dynamic environment with devices autonomously and arbitrarily joining or leaving the network or the transmission signal of individual elements may increase or decrease. As a result of this, the content, coverage, topology, and partitioning of a DTMANET may change unpredictably. These factors influence the performance and the usability of the network, and make DTMANETs a complex environment to simulate.

4.1.2 DTMANET Simulation Tools

In the literature survey (see *section 2.5*), a decision was made to create a simulation software for supporting this research. The chosen approach uses an intentionally simplified network environment and focuses on the activities of the opportunistic

⁷ The attention of the reader is drawn to Appendix A on the companion DVD which includes a number of video recordings with voice-over commentary of the simulation environment in action on a number of OCI test scenarios.

routing layer at the conceptual level. It requires the support of multiple opportunistic routing protocol groupings. Its parameters of network entities need to be modified constantly; new device, new protocol, and new movement models need to be added and integrated into the system easily. Large numbers of automated generated batch processing scenarios and configuration are required when the filter permutation technique is used. Since none of the existing simulators fits all the requirements, a new simulator had to be developed which targets the specific needs of this research.

4.1.3 OCI-SIM – an Alternative Approach

OCI is a proposal that provides a wide range of applicability and a rapid prototyping mechanism to achieve coarse-grained, statistic pattern indexes of multiple opportunistic elements that influence DTMANET routings. This enables multiple opportunistic routing protocols to be applied as a group to adaptively improve the reliability, performance, and efficiency of routing processes in a DTMANET.

Our investigation of the effectiveness of OCI was undertaken through the established research method of problem isolation, abstract modelling, experimental system building, and simulation. By permuting filters, collecting, comparing, and analyzing a statistically valid number of simulation results, the outcome of the OCI viability tests can be used as a set of high-level guidelines for its application to specific network scenarios. To fulfil this objective, a tool-set was needed which would facilitate designing, deploying, testing, and evaluating DTMANET architectures. This architecture is consists of a large range of devices, protocols, and scenarios with reasonable efficiency, and credible accuracy.

OCI-SIM (incorporating OCI-AUTO, OCI-CUSTOM, OCI-GRAPH, and OCI-VISUAL) is the tool-set developed as part of this study to evaluate the viability of the OCI concept. One of the OCI-SIM design objectives was to generate opportunistic indexes with a high efficiency and a low human intervention

requirement. This is done to accommodate the requirements for multiple layers and multiple scenario simulations, OCI-SIM is built to be a highly customizable, extensible, and scalable system.

OCI-SIM is not intended to represent excessively detailed and realistic scenarios. In its simplest form, it requires specifications of only the essential characteristics that are common to all DTMANETs. It needs to know what constitutes the system (devices and their physical attributes), what communication needs to be accomplished (messages to be delivered), and what the performance concerns are (opportunistic filter models). This should be achieved without detailed consideration of how the targeted DTMANET is organized, where the devices are physically situated, and which devices are to be used for routing. These unspecified issues are left to the OCI-SIM tool-set to handle. OCI-SIM takes the input and feeds it into OCI-AUTO, which uses adaptive self-configurable permutation mechanisms to generate a substantial number of configurations. Because of the high level of abstraction, OCI-SIM is able to achieve a very high simulation speed, which makes massive simulations of large scale DTMANETs⁸ with large numbers of permutations possible. Consequently, it is easy to produce a high level of statistical validity in the data analysis phase. For instance, the following might be a typical OCI simulation configuration:

The OCI-SIM is informed that the system has a network of 1000 peers, 60% of them use Bluetooth, 30% of them are WIFI devices, and the remaining 10% are equipped with dual-adaptors, which support both Bluetooth and WIFI. The system is instructed to send 100 end-to-end messages between random peers. It is also provided with a list of opportunistic filters that need to be evaluated for performance. Finally, the number of distinctive repetitions is specified as 100.

⁸ 1000 distinctive repeats of over 1000 permutations of 1000-peer MANETs have been tested using OCI-SIM during the course of research.

4.2 OCI-SIM Design

OCI-SIM is a Java-based simulator (see *Appendix G*) composed of four major sub-components (see *Appendix B*) that are described in the subsequent sections.

4.2.1 OCI-AUTO

OCI-AUTO is the key component to experiencing and validating the hypothesis of OCI. The main purpose of OCI-AUTO is in testing and investigating opportunistic elements, which are modelled as OCI filters that influence the DTMANET's usability and performance in an automated fashion. It is designed to be applied across physical and opportunistic protocol layers of DTMANETs with differences in scale, type, and characteristics.

4.2.1.1 OCI-AUTO Design Approach

The design philosophy behind OCI-AUTO is to first observe a complex process in a dynamic environment and secondly, to break down and simplify the complex process into a number of essential components. For instance in OCI-AUTO, the process of DTMANET communication is broken down and simplified into three parts, namely: initial status (input), communication progress (sandbox) and final status (output). Thirdly, the input is further simplified to allow only essential variables to be considered as input. For example, in OCI simulations the number of initial peers, the number of messages in the system and the type of communication protocols are qualified as input, while peer initial location and message hosts are not. Another design consideration is to identify and isolate various facts in the dynamic environment that influence the output of the process. This is then followed by the abstraction of isolated facts into computational models in the OCI, called opportunistic filters. before the start of the simulation process, subsequent to the acquisition of the initial input and abstracted models, OCI-AUTO automatically generates a series of different testing configurations through listing, combining and

permuting all the abstracted models with identical input conditions. Because of the simplification and abstraction of inputs and models, OCI-AUTO can start and repeat an automated simulation process with high validity. This is as a result of reiterated simulations with identical input yielding different random elements every single time. The effects of result fluctuation of those random variables are eased through statistically sound repeating simulations. Finally, through comparing and analyzing the results of each of these configurations, OCI-AUTO computes and provides OCI indexes for multiple opportunistic filters based on their degree of capability that influence the output of the communication process.

4.2.1.2 OCI-AUTO sub-components

OCI-AUTO contains a number of sub-components to help build, test, and analyse OCI indexes.

1) OCI-AUTO Scenario Designer

The first Step in using OCI-AUTO is to compose one or more scenario configurations using the following designer tools.

a) OCI Peer/Message, Auto Increase Option and Self-Adapt Designer

This is a tool where the initial input of OCI-AUTO simulations are assigned (see *Figure 4-1*), it also provides options for a self-repeating incremental based simulation used for optimization of subsequent OCI index generating simulations.

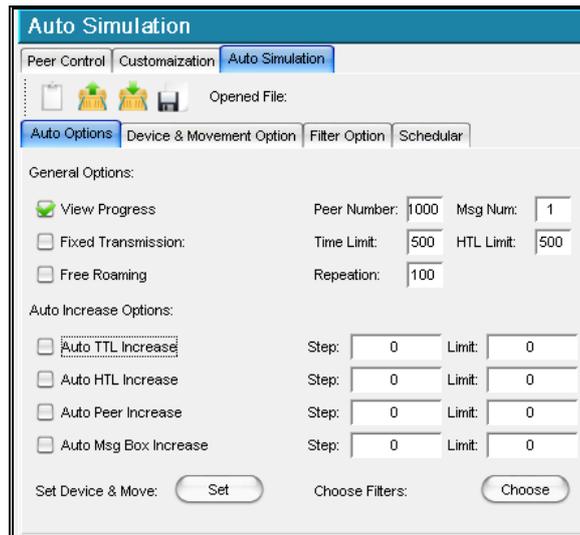


FIGURE 4-1: PEER/MESSAGE, AUTO INCREASE OPTION AND SELF-ADAPT DESIGNER

b) OCI Communication Protocol and Movement Model selector

The communication protocol selector gives OCI-AUTO options to simulate heterogeneous DTMANET communication with a specified percentage of each of the available communication protocols in the system.

As depicted in *Figure 4-2*, the movement model selector provides OCI-AUTO with the ability to simulate a DTMANET environment with high movement diversity. Furthermore, new movements can be dynamically loaded and integrated into the system.

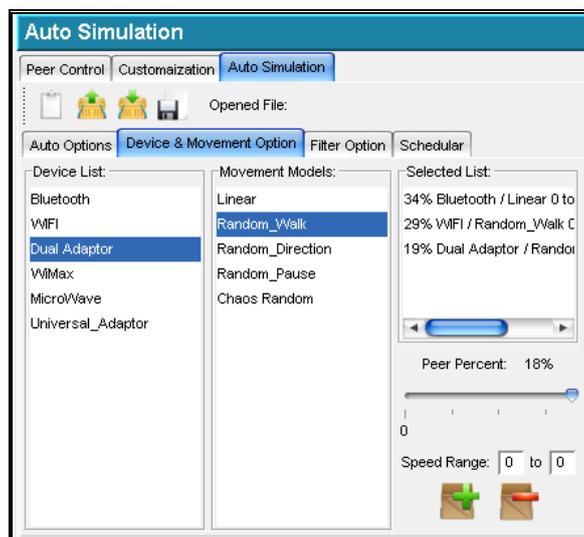


FIGURE 4-2: COMMUNICATION PROTOCOL AND MOVEMENT MODEL SELECTOR

c) OCI Filter Organizer

The OCI Filter Organizer is used to select and add filter models to a given scenario (see *Figure 4-3*), so that the OCI index can be generated for the filter models. New filters can be dynamic added into the system and used in the same way as other filters.

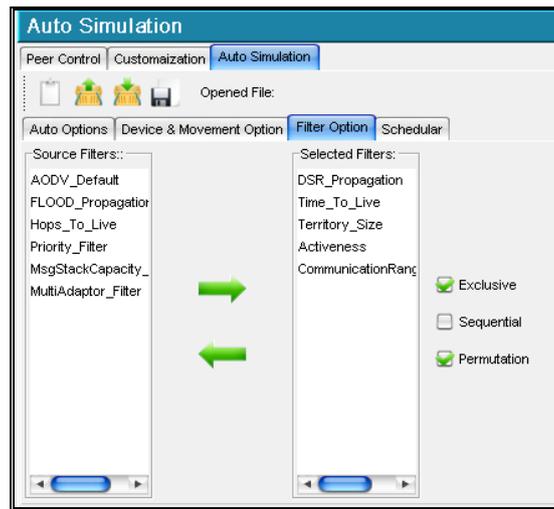


FIGURE 4-3: OCI FILTER ORGANIZER

i. OCI Filter Isolator

The filter isolator provides a way to compare the influence of each of the selected filters individually.

ii. OCI Filter Combiner

With the combiner, filters can be applied to scenarios as groups.

iii. OCI Filter Permutator

In OCI-AUTO the algorithm to generate permutations starts with an initial permutation order of filters and subsequent permutations get computed in lexicographic order. The algorithm will generate all permutations for a given combination of filters if the initial combination is the first in lexicographic order.

d) OCI Auto Generator

i. Auto Peer Generator

Auto peer generator is used to generate peers in a random fashion within a given scenario.

ii. Auto Message Generator

Auto message generator assigns messages to peers within a given scenario.

In OCI-SIM, only the essential constants are included as input. In the auto generator, a given number of messages are randomly generated within a given number of randomly generated peers, within a limited range. For any given scenario, in each different generation: peer location, transition range, territory, message originality, and destination are also randomly assigned, while the peer number, message number and communication protocol percentage remain unchanged.

e) OCI Auto Scheduler

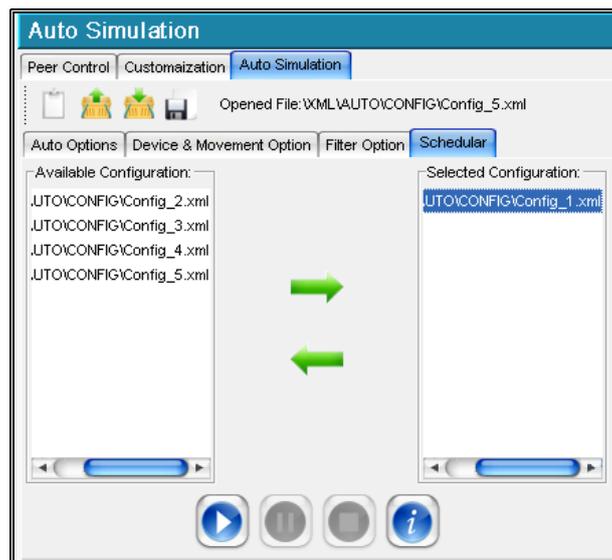


FIGURE 4-4: OCI AUTO SCHEDULER

Multiple configurations can be loaded and queued for sequential execution with the help of OCI Auto Scheduler (see *Figure 4-4*). This is useful in running simulations over a long period of time without any human intervention.

2) OCI-AUTO Data Recorder

In OCI-AUTO, device definitions, movement models, filter definitions, scenario configuration and data output are automatically recorded in XML format. For instance, new devices, filters and movement models created in one simulation can be reused in other simulations by reloading the recorded XML files (see *Figure 4-5*). A scenario can be restored should the simulation be interrupted. XML data records can be loaded for further analysis.

OCI-AUTO data recorder includes the following components:

- a) Device definitions, movement model, filter definition recorder
- b) Scenario configuration recorder
- c) Data output recorder

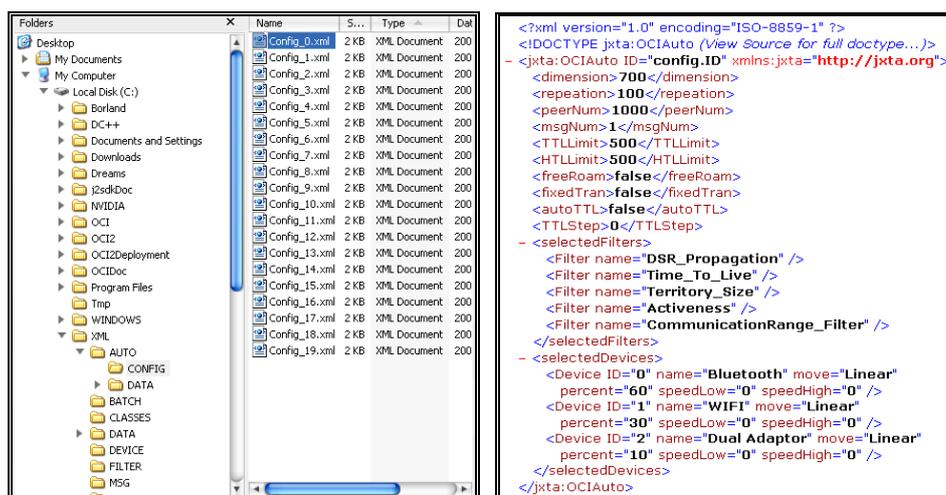


FIGURE 4-5: XML DATA FORMAT

4.2.1.3 OCI-AUTO Execution Mode

OCI-AUTO can be optionally set to three different execution modes: observation mode, console verbose mode, or silent mode. Observation mode gives a visual representation of the progress of the auto simulation. Console verbose mode prints vital information as the simulation progresses, and silent mode omits all visual and console output, writing the data to a log file when each configuration completes its execution. Without the overhead of runtime display, silent mode is the fastest way to run a simulation.

4.2.2 OCI-CUSTOM

OCI-CUSTOM is a central feature of OCI-SIM, which facilitates a quick and easy way to add new devices, opportunistic filters, and movement models. It provides a high level of extensibility and allows the concept of OCI to be tested across as many communication protocols and scenarios as can be envisaged by the user.

4.2.2.1 OCI-CUSTOM Design philosophy

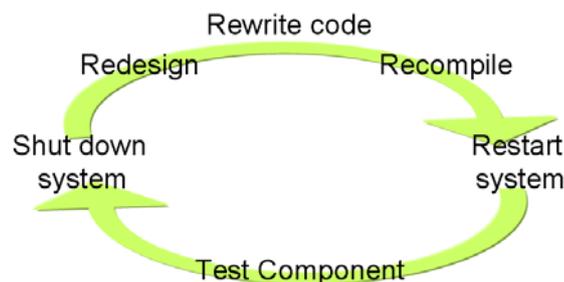


FIGURE 4-6: TRADITIONAL RESEARCH APPROACH CYCLE

The ability to simulate “any device”, “any environment” and the flexibility to apply changes at “any time” during the process of the simulation are the philosophies, which steered the design of OCI-CUSTOM. “Any device” and “any environment” imply that the research of OCI has extended values that enabled a wider range of applications and devices to be tested. Furthermore, opportunistic elements that are currently overlooked can be quickly modelled and added to the simulation system to compute

their OCI indexes. “Any time” indicates that in addition to the traditional research approach cycle (see *Figure 4-6*), the need for rapid prototyping and testing of OCI indexes requires dynamic “hot plug and play” components. Therefore, all sub-components of OCI-CUSTOM are designed to be operate-able at runtime. Consequently, new devices, opportunistic filters and movement models can be created, modified and loaded into a running simulation to see its immediate effects or saved as XML configuration records for later use.

4.2.2.2 OCI-CUSTOM sub-components

OCI-CUSTOM has three major sub-components; these are OCI-Device Builder, OCI-Filter Builder and OCI-Movement Model Builder.

1) OCI-Device Builder

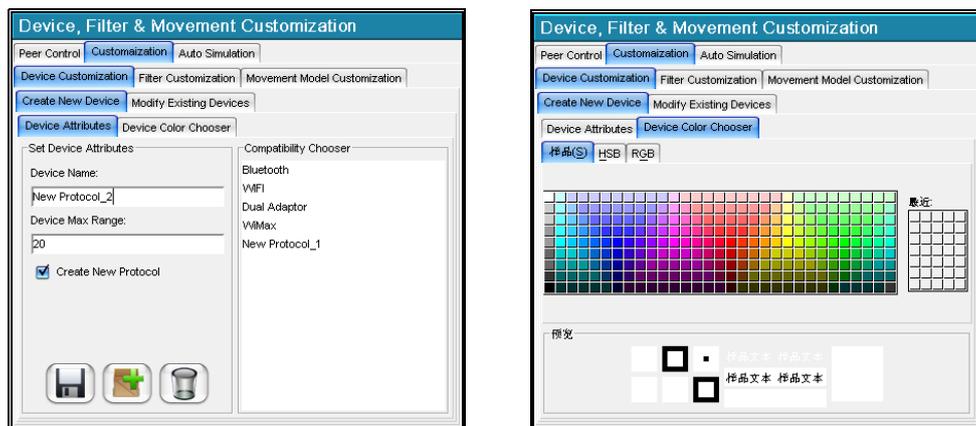


FIGURE 4-7: OCI DEVICE BUILDER AND DEVICE BUILDER VISUAL AID DESIGNER

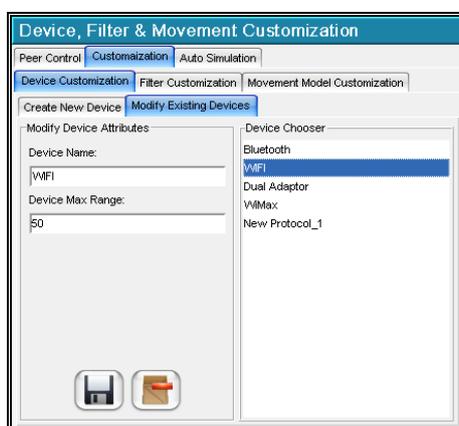


FIGURE 4-8: OCI DEVICE BUILDER PROTOCOL EDITOR

To prove the versatility of the theory of OCI, a variety of communication protocols and a wide range of device capability configurations needed to be constructed and tested. Hence, the OCI-Device Builder (see *Figure 4-7*, *Figure 4-8*) is designed and built so that it can create new communication protocols with new definitions and it can create or modify devices with multi-protocol compatibilities by choosing and combining existing protocols to simulate more sophisticated DTMANET situations.

2) OCI-Filter Builder and OCI- Movement Builder

Both OCI-Filter Builder and OCI-Movement Builder (see *Figure 4-9*) contain the following components:

- a) Programming IDE
- b) API Saver/Loader
- c) Dynamic Filter Compiler
- d) Dynamic Filter Loader.

Using the components above, newly created filters and/or movement models are ready for use after being dynamically compiled and loaded. The source can be saved as a Java source file for later use. This approach makes a fully dynamic, very flexible and runtime extensible testing environment for research into OCI. The technique employed to achieve the full dynamic is through:

- a) Java reflection API for dynamic source code compiling and class loading;
- b) Preserved empty data fields (empty Java Vectors) for new parameter assignment of peers and messages.

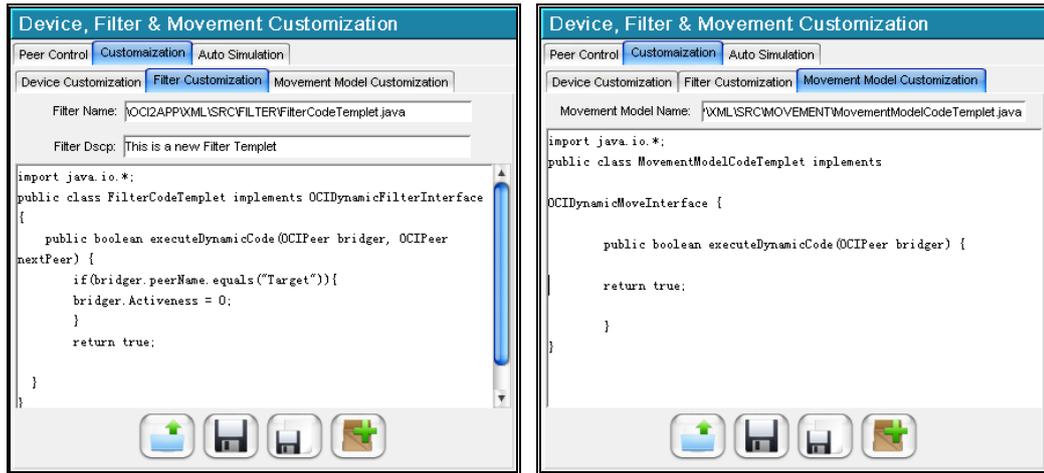


FIGURE 4-9: OCI FILTER BUILDER AND OCI MOVEMENT BUILDER

4.2.3 OCI-VISUAL

OCI-VISUAL is an add-on feature to aid OCI testing (see *Figure 4-10*). It offers a fully interactive visualization option to OCI-SIM, and is useful for progress and result monitoring in specific scenarios.

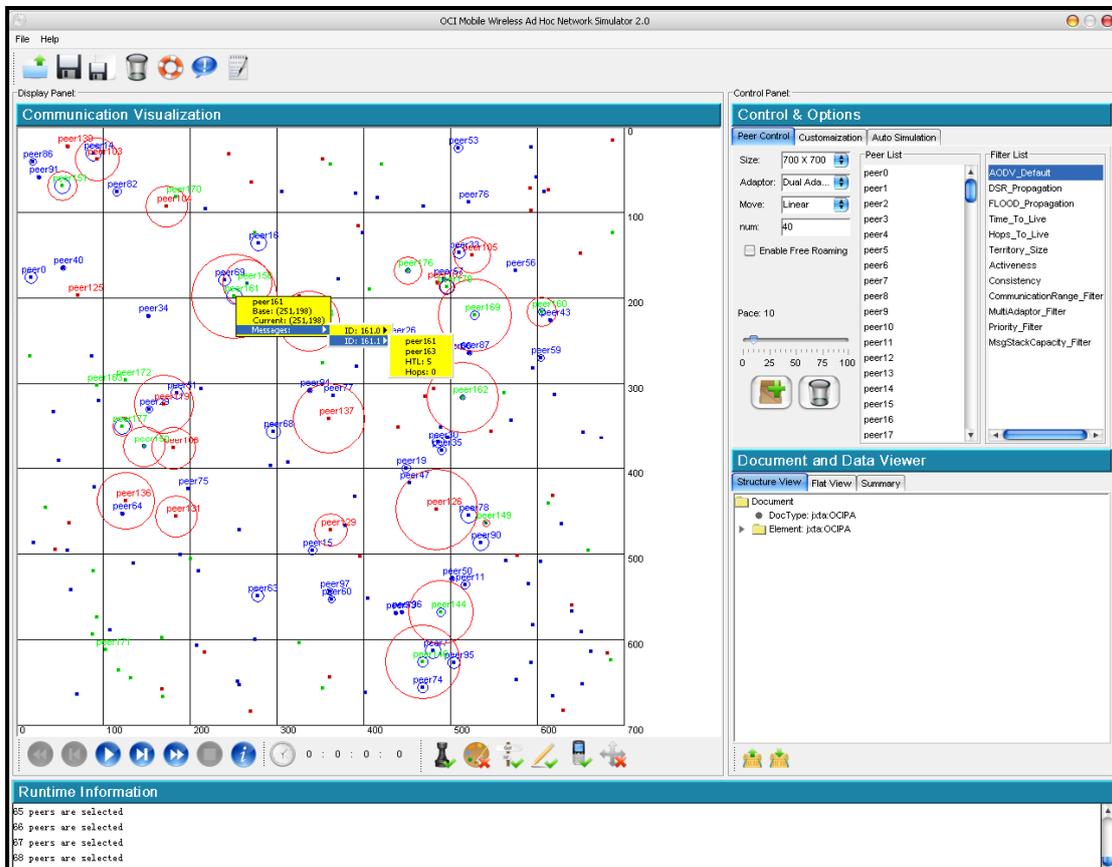


FIGURE 4-10: OCI-VISUAL

4.2.3.1 OCI-VISUAL Design philosophy

“A picture is worth a thousand words”. The cliché aptly characterizes the goals of visualization where large amounts of data must be contextualised or absorbed quickly by the viewer. In the DTMANET research community, visualization is a valuable and helpful technique to represent network topology, routing passage and communication processes. During the design process of OCI, a number of abstract ideas were introduced, and, through animated visualization, these ideas can be delivered more directly and intuitively. In addition, the interactive features of OCI-VISUAL provide a more convenient and straightforward method to creating specialized scenarios, and observing the communication process and results while testing the OCI concept.

4.2.3.2 OCI-VISUAL sub-components

OCI-VISUAL comprises: Scenario Controller, Dynamic Visualization Display, Interactive Controller, and Simulation Info-Viewer.

1) OCI-VISUAL Scenario Controller

Unlike the simplification and generalization in OCI-AUTO, OCI-VISUAL Scenario Controller is used to setup a DTMANET communication environment with specific restrictions and requirements. It states the number and types of communication protocols used; number of peers in the simulation, their location, network partition, active territory and filters to be used. It also offers facilities to record and restore specifically constructed scenarios through the use of XML data structures.

2) OCI-VISUAL Dynamic Display

Like the canvas to the nineteenth-century French impressionism painter Édouard Manet, OCI-VISUAL dynamic display is employed as the canvas for the OCI

DTMANET simulation; where the entire communication process can be visualized.

3) OCI-VISUAL Interactive Component

Unlike Manet’s canvas, OCI-VISUAL display is animated and interactive. Under interactive mode, OCI-VISUAL allows user level real-time actions such as adding more messages to selected peers, putting more peers in specified locations, regrouping peers, reassigning peer parameters with different values, and moving and replacing a peer for network partition bridging⁹, all while the simulation is in progress. When testing OCI indexes using this component, OCI simulation scenarios can be quickly setup and modified.

4) OCI-VISUAL Info-Viewer

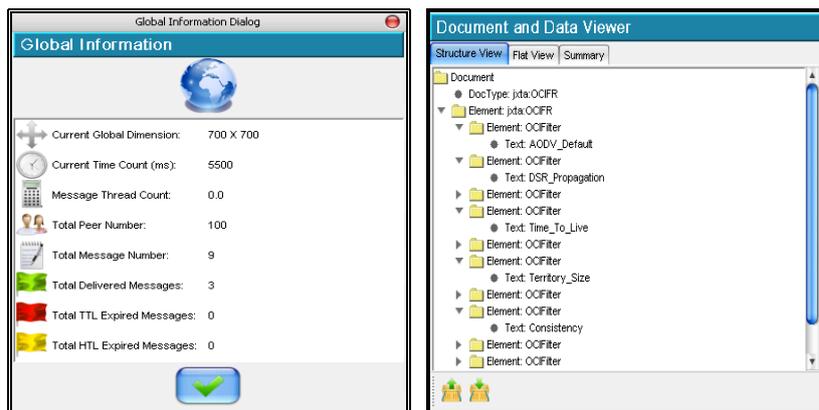


FIGURE 4-11: GLOBAL INFO VIEWER AND DOCUMENT & DATA VIEWER

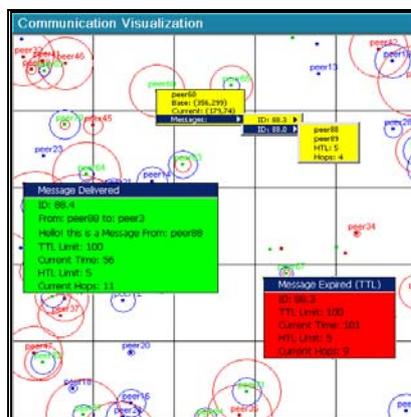


FIGURE 4-12: MESSAGE STATE VIEWER

⁹ Please refer to the accompanied video footage for visualized demonstration.

In any DTMANET visualization it is vital to track the status and information of a current simulation. Information of that nature in OCI-VISUAL includes individual peer parameters and the message delivery status through popup displays; global information through global info viewer; configuration and filter information through structured XML tree view (see *Figure 4-11*); and runtime information through text display (see *Figure 4-12*).

4.2.4 OCI-GRAPH

OCI-GRAPH is a useful tool to aid in data analysis. It loads data from XML files that were previously generated through OCI-AUTO and produce graphical representations of the data. OCI-GRAPH is created with a dynamic, x and y axis which can be assigned with different value sets dynamically and the correspondent graph will respond to the changes instantaneously.

4.2.4.1 OCI-GRAPH Design philosophy

Being an analytical data oriented project, namely testing the OCI theory, enormous amounts of data were generated for each of the simulations to achieve statistical soundness. Performing a detailed analysis of all the data would have greatly increased the length and the difficulty of research of this kind. Therefore, it is necessary to have a technique to generate a “quick preview” of the data; hence data could be selectively fed into a more advanced third party data analysis tool.

4.2.4.2 OCI-GRAPH sub-components

OCI-GRAPH contains data loader, XML data parser, data serializer, data analyzer, XML element/attribute chooser, graph generator, single graph display, multi-graph display, and graph recorder. Meanwhile, as shown in the *Figure 4-13*, OCI-GRAPH provides four types of graph rendering methods. These are line graph, dotted graph, bar graph, and filter name graph.



FIGURE 4-13: OCI-GRAPH

4.3 A Taxonomy of OCI-SIM Simulation Modes

OCI-SIM presents a range of techniques and operating modes for simulating DTMANET scenarios. They may be classified by user intervention level, the degree of randomness, and the organization of filters.

4.3.1 Taxonomy by user intervention level

OCI simulations can be conducted in the following modes, categorised by the degree of user intervention.

4.3.1.1 Fully automatic simulation mode

This mode relies on the automated simulation features of OCI-AUTO. All filters, routing protocol and movement models available in OCI-SIM will be automatically applied to given scenarios with different combinations, permutations, or intelligent weighting methods. The only human intervention is queuing up the XML scenario configuration files. After starting this simulation and before it is finished or

interrupted, no change and modification can be made to the simulation. The result of a fully automatic mode simulation will represent comprehensive OCI indexes for given scenarios.

4.3.1.2 Semi-automatic simulation mode

Under the semi-automatic mode, simulations can be supplied with selected filters and tested against specific movement models. Scenario and automation parameters such as the repetition number, auto increasing options, and realism level can be specified. After that, OCI-SIM automatically optimizes these parameters and computes the OCI indexes with the specified number of repetitions.

4.3.1.3 Single simulation mode

Unlike automatic and semi-automatic mode, single mode allows scenarios to be constructed in a much more protocol and scenario oriented fashion. Every single parameter and attribute can be individually and precisely supplied. The single simulation mode is useful for more refined tests and for individual validation of OCI indexes.

4.3.1.4 Interactive single simulation mode

Similar to single simulation mode, interactive single simulation mode is useful for testing results from OCI indexes individually in a visualized interactive simulation environment. In addition, message parameters, peer behaviour, peer parameters, network partitions and filter variations can dynamically be altered through interactive components provided by OCI-SIM while the simulation is in progress. OCI-VISUAL is required to facilitate the interaction. Using the interactive mode, a more intuitive and direct simulation feedback can be observed and analyzed.

4.3.2 Taxonomy by simulation randomness level

OCI simulations can also be classified into different categories based on the level of randomness of discrete elements.

4.3.2.1 Theoretical static simulation

Being a DTMANET simulation study, OCI-SIM simulations contain a large number of random elements that have to be taken into design consideration. However, in some situations, it is desirable to omit the randomness of one or more of the elements. Instead, theoretical values are assigned to these elements during the simulation, and OCI-SIM provides options to do so. In this mode:

- 1) A peer's movement retains its theoretical conformity;
- 2) Peers have no autonomy over the decision of moving or stop;
- 3) Peers share a common movement territory boundary;
- 4) Peers' communication ranges are set to their theoretical best;
- 5) Message relay and acceptance are compulsory;
- 6) Network partition situation consideration is omitted;

In this project, the theoretical static simulations are the most useful of the simulation validity control mechanisms, which will be discussed in detail in the next chapter.

4.3.2.2 Random element isolation simulation

It is a common scientific practice to identify and isolate certain opportunistic elements from other constituents to observe and analyze their effects on a targeted system. This was also adopted as one of the research techniques for this project, through using OCI-SIM. It was much easier to achieve random element isolation in OCI-SIM than in the real world or through other available MANET simulators. Under random element isolation simulation, only one random opportunistic element is granted with randomness when executing simulations.

4.3.2.3 Dynamic diversity full randomness simulation

In contrast to theoretical static simulations, this type of simulation takes all the modelled random elements in an OCI-SIM into consideration, it provides a more realistic and believable environment for a DTMANET simulation.

- 1) Random walk movement model is used to simulate peer movement;
- 2) A peer will randomly and with autonomy decide to move or stop ;
- 3) Each individual peer has its own movement territory boundary, and its boundary may or may not overlap with other peers;
- 4) Peers' communication ranges are randomly generated from none (offline), to their theoretical best (full power);
- 5) Message relay and acceptance are random;
- 6) A network may be partitioned due to peer movement and territory limitations.

Because of a higher degree of random level, results of a dynamic diversity full randomness simulation are the main evidence in establishing the OCI theory.

4.3.3 Taxonomy by filter organization

OCI-SIM simulation can be conducted in a range of modes, categorized by the organization and combination of filter types

4.3.3.1 Isolated opportunistic filter simulation

Under isolated filter simulation, only one opportunistic filter is selected for a particular simulation execution. The results of isolated opportunistic filter simulations are used in direct one to one comparisons among selected filters when computing and assigning initial weights for OCI indexes for targeted scenarios.

4.3.3.2 Combined opportunistic filter simulation

A combined opportunistic filter simulation joins a group of selected filters and tests the combination in OCI-SIM. In this simulation mode, filters are considered to be weightless, which indicates that the order of filters is not considered. Results of such simulations are valuable for the observation of effects of filter group variation, and they also provide a base for a permuted opportunistic filter simulation.

4.3.3.3 Permuted opportunistic filter simulation

During an execution of a permuted opportunistic filter simulation, filters are assigned with different weight values each time a new permutation is generated. Upon the exhaustion of permutation possibilities, an OCI index of selected opportunistic filters will be generated based on the analysis of the result of each permutation.

4.3.3.4 Intelligent weighted opportunistic filter simulation

The intelligent weighted opportunistic filter simulation uses the results of isolated filter simulations to sort opportunistic filters into order or assign weights to them. This technique generates a more refined OCI index, and it is used to generate data for discussion and demonstrations in the later chapters.

4.4 Overall Evaluation of OCI-SIM Design

1) Originality

OCI-SIM is designed specifically to support this research and to prove the OCI concept. It offers a range of new functions and features that are not in existing simulators (see *section 2.5*). Such features include delay tolerant communication modelling, opportunistic routing protocols, intelligent grouping and adaptive decision routing making, auto opportunistic filter permutations and auto scenario generation.

2) Validity

Simulation validity has been used as a guideline throughout the development cycle of OCI-SIM (see *section 3.7*, *section 5.3*). Through the control algorithms implemented in OCI-SIM, the validity of data analysis is ensured and therefore the result of the simulations are suitable for supporting the concept of OCI proposed in this thesis.

3) Applicability

The OCI-SIM design provides for the pertinent features required by the investigation, without unnecessary detail. The assumptions outlined in *section 3.3* are inherent, and the level of abstraction presented by the OCI-SIM delivers a close conceptual correspondence with the problem domain of modelling MANETs where delay tolerant communication and opportunistic routing are key features. The platform consequently supports both rapid prototyping and lucidity of purpose for scenarios.

4) Versatility and extensibility

With the help of OCI-CUSTOM (see *section 4.2*), new devices, new movement models and new routing filters can be added into the system to represent versatile and heterogeneous DTMANET simulation scenarios.

5) Flexibility

OCI-SIM is designed with runtime compiling and class loading mechanisms (see *section 4.2.2.1*), which provide flexibility for scenario configurations and modifications.

4.5 Chapter Review

In this chapter, an introduction and evaluation of the design concepts, philosophies and considerations of the experimental simulation environment, for this research, was made. The OCI-SIM environment and its sub-components, OCI-AUTO, OCI-CUSTOM, OCI-VISUAL and OCI-GRAPH were introduced and discussed. Taxonomies of simulations over OCI-SIM were provided. The next chapter will look at the different data on which OCI-SIM operates, and simulation validity control mechanisms used in OCI-SIM will be demonstrated.

Chapter V: Data Classification & Validity Control

“All credibility, all good conscience, all evidence of truth come only from the senses.”

– Friedrich Nietzsche

This chapter serves as the preparation stage for the data acquisition and analysis of the next chapter, by outlining the structure and method of data collection. It describes the data composition and classification used for evaluating the OCI concept, presents a taxonomy of scenarios used for data collection, and addresses the steps taken to ensure a measure of validity in the simulated results.

5.1 Data Composition and Classification

The following sections aim to clarify the types and properties of data used as inputs and the data produced as outputs by the OCI-SIM through the simulations. Such classification identifies the data upon which the OCI simulations operate and defines the scope within which evaluations are conducted.

5.1.1 Initial input data classification

To initialise a simulation, input data needed to be provided to OCI-SIM, this includes peer data, message data, and global statistic data. In OCI simulations, peer data constitutes each peer’s movement and opportunistic behaviour, which provides the foundation of opportunistic routing in DTMANET. Message data is essential for initialization, states tracking and termination of communications in a message-oriented system such as OCI. Global statistic data holds critical information regarding routing reliability, performance and efficiency, without which OCI indexes cannot be evaluated.

5.1.1.1 Peer data

In an OCI system, peer data includes a peer's physical attributes, its movement associated attributes and its social behaviour associated attributes. Some of these attributes are static and non-opportunistic, while others are dynamic and opportunistic and are used in decision-making processes for different filters in the simulation. In an OCI simulation, peer data can be assigned specifically or through a random generator. OCI-SIM implements peer data in XML format shown in Code 5-1.

<pre><?xml version="1.0" encoding="ISO-8859-1" ?> <!DOCTYPE jxta:OCIPA (View Source for full doctype...)> <jxta:OCIPA xmlns:jxta="http://jxta.org"> <PID>urn:jxta:uuid-0</PID> <GID>urn:jxta:jxta-NetGroup</GID> <Name>Peer0</Name> <Adaptor_List>0</Adaptor_List> <BaseLocation>x580y144</BaseLocation> <CurrentLocation>x580y144</CurrentLocation> <Territory>x103y137</Territory> <PaceRange>10</PaceRange> <Activeness>x3y-2</Activeness></pre> <p style="text-align: right;">Page 1 of 2</p>	<pre><MovementModel>0</MovementModel> <MsgStactSize>1</MsgStactSize> <compatibleDeviceList> <compatibleDevice>Bluetooth</compatibleDevic e> </compatibleDeviceList> <compatibleDeviceRangeList> <compatibleDeviceRange>10</compatibleDevic eRange> </compatibleDeviceRangeList> <OCIMessageList /> </jxta:OCIPA></pre> <p style="text-align: right;">Page 2 of 2</p>
---	--

CODE 5-1: AN EXAMPLE OF PEER DATA IN XML FORMAT

1) Peer physical attribute

Peer physical attributes are listed as the follows:

- a) Peer name: The name of a peer.
- b) Peer ID: A unique identifier a peer uses to identify and to be addressed within and/or across DTMANETs.
- c) Mobile communication protocol: The protocol a peer uses to communicate with other peers within and/or across DTMANETs.

- d) Compatible protocol list: A list of communication protocols that are compatible with the current protocol that a peer uses. For instance, in an OCI system, a mobile device might use 802.11g, its compatible protocol list contains 802.11g, 802.11b, since it can communicate with devices using the same protocol and it is backwards compatible with 801.11b devices. Or if a user defines a Bluetooth/WIFI dual adaptor mobile device, its compatible protocol list will contain those protocols.
 - e) Compatible protocol communication range list: This list contains a peer's communication ranges within which it communicates with other peers respectively under different protocols.
 - f) Message box size: message box size is defined to limit the number of messages that a peer will be able to store locally.
 - g) Peer colour: Peer colour is used in an OCI system when a visualization case study is needed to distinguish different communication protocols.
 - h) Peer diameter: Peer diameter is used to draw a physical representation of a peer in situations of visualization case studies.
- 2) Peer movement associated attributes
- a) Peer base location: Peer base location is the initial position (x_b, y_b) of a given peer in the system before its movement occurs.
 - b) Current location: Peer current location is the current position (x_c, y_c) in the system of a peer at a given time.
 - c) Peer territory: Peer territory sets the boundary within which peer movement may occur.

- d) Peer velocity: Peer velocity is a movement unit. In a two dimensional DTMANET simulation environment such as the OCI system, it represents the peer position changes along x , y axes in one given time unit in a DeCarl coordinate system.
 - e) Velocity pacer range: Velocity pacer range defines the maximum movement offset in a single time unit of a peer. It also differentiates movement potential of vehicles of different mobile devices. For example, a car based mobile device has a much higher activeness pacer range than a mobile device carried by a walking person. At the same time these devices will not move faster than its maximum speed namely its activeness pacer range.
 - f) Peer movement consistency level: The consistency level determines the frequency that a peer is likely to change its movement pattern (velocity, direction, etc).
 - g) Route history: Route history stores the track of a peer's movement.
- 3) Peer social behaviour associated attributes
- a) Peer encountering history: The peer encounter history attribute records IDs and names of other peers that a peer has communication within a given period.
 - b) Group: A group is a logical collection of peers that share common characteristics and/or interests.
 - c) Group ID: Group ID is the unique identifier of each group within and/or across MANETs.
 - d) Peer Reputation: Peer reputation refers to the level of willingness of a peer to accept/reject and/or hold/forward incoming and/or outgoing messages.

4) Customizable Peer attributes

As new mobile communication protocols and devices frequently emerge, there are apparent needs to update simulations. OCI-SIM provides OCI-CUSTOM by using the Java reflection API, and the technique of preserved empty data fields, new attributes can be assigned to peers dynamically when new filter restriction and routing mechanisms are introduced.

5.1.1.2 Message data

In the OCI simulations, “Message Data” refers to a message’s physical attributes, its routing decision making attributes and attributes that are used to control message behaviour.

1) Message physical attribute

- a) Message ID: The message ID is a unique identifier of each message within and/or across MANETs.
- b) Message body: Message body is the actual content of a message.

2) Message routing and location associated attributes

- a) Sender ID: Sender ID is a unique identifier which refers to the sender of a message.
- b) Receiver ID: Receiver ID is a unique identifier which refers to the receiver of a message.
- c) Send Location: is the position (x, y) from which a message is sent.
- d) Base Location: Base location is the base position (x, y) of the message sender peer.

- e) Destination: Destination is the base position (x, y) of the receiver peer of a message.

3) Message control attributes

- a) Message priority: Message priority is a value that indicates and sorts the importance of a message. There are ten levels of priority in an OCI simulation design. The higher the number the more important a message is.

$$p \in \{0,1,2,3,4,5,6,7,8,9\}$$

- b) Start time: Start time is the value of time units that a message is sent for the first time in the system.
- c) Time to Live (TTL): TTL is the maximum time unit value that a message will exist in the network.
- d) Current time value: Current time value represents the current value of time units regarding to the TTL value of a message. Its value increases by one when one time unit elapses.
- e) Hops to Live (HTL): HTL is the maximum hops value a message will travel in the network.
- f) Current hop value: Current hop value represents the current hop count of a message regarding to its HTL value. Its number increases by one every time the message is passed from current peer to the next peer.
- g) Is delivered flag: Is delivered flag indicates the current delivery status of a message, it is a binary value, and a message can only be in one of the two statuses, delivered or not delivered.

$$d \in \{0,1\}$$

After a message has been delivered, it will no longer be forwarded to any other peers and it will be considered as an inactive message.

- h) Is TTL expired flag: “Is TTL expired flag” indicates the current aliveness status of a message, it is a binary value, and a message can only be in one of the two statuses, expired due to TTL limit or not expired due to TTL limit.

$$f(t) = \begin{cases} 0 & (t < \text{TTL}) \\ 1 & (t = \text{TTL}) \end{cases}$$

A TTL expired message will no longer be forwarded to any other peers and it will be considered as an inactive message.

- i) Is HTL expired flag; “Is HTL expired flag” indicates the current aliveness status of a message, it is a binary value, and a message can only be in one of the two statuses, expired due to HTL limit or not expired due to HTL limit

$$f(\hat{h}) = \begin{cases} 0 & (\hat{h} < \text{HTL}) \\ 1 & (\hat{h} = \text{HTL}) \end{cases}$$

A HTL expired message will no longer be forwarded to any other peers and it will be considered as an inactive message.

4) Customizable Message Attributes

Like peer data, attributes can also be dynamically modified, assigned and added to messages to aid the new filter restrictions and routing decision-making mechanisms other than traditional attributes such as TTL and HTL values.

<pre><?xml version="1.0" encoding="ISO-8859-1"?> <!DOCTYPE jxta:OCIMSG> <jxta:OCIMSG xmlns:jxta="http://jxta.org"><OCIMessageEnvelop msgID ="55.0"> <senderStr>Peer55</senderStr> <receiverStr>peer56</receiverStr> <sendLocationStr>x657y239</sendLocationStr> <baseLocationStr>x618y154</baseLocationStr> <destinationStr>x0y0</destinationStr> <priorityStr>0</priorityStr> <startTimeStr>8550</startTimeStr></pre> <p style="text-align: right;">Page 1 of 2</p>	<pre><TTLStr>5000</TTLStr> <HTLStr>5</HTLStr> <currentTimeStr>8550</currentTimeStr> <currentHopStr>0</currentHopStr> <IsDelivered>0</IsDelivered> <IsTTLExpired>0</IsExpired> <IsHTLExpired>0</IsExpired> <OCIMsgBodyStr> Hello! this is a Message From: Peer55 </OCIMsgBodyStr> </OCIMessageEnvelop> </jxta:OCIMSG></pre> <p style="text-align: right;">Page 2 of 2</p>
--	--

CODE 5-2: AN EXAMPLE OF MESSAGE DATA IN XML FORMAT

5.1.1.3 Global statistical Data

One of the advantages of using the simulation approach to study DTMANETs is that data can be gathered in a global perspective. Using the initial input data an OCI system can compute vital statistic data to aid the data analysis and the generation of OCI indexes.

1) Global dimension:

A global dimension is an artificial boundary that all DTMANET simulation communications occurred within.

2) Peer number count:

Peer number count is the total number of peers in a given DTMANET simulation scenario.

3) Message number count:

Message number count is the total number of Messages in a given DTMANET simulation scenario.

4) Peer density:

Peer density is the ratio of total peer numbers and the dimension.

5) Message density:

Message density is the ratio of total message numbers and the dimension.

6) Average Opportunistic values:

a) Average peer velocity: It is the average velocity value of all peers in a given DTMANET simulation scenario.

b) Average peer communication range: It is the average value of communication range of all peers in a given scenario.

c) Average peer territory: Average peer territory is the value of average territory of all peers in a given DTMANET simulation.

d) Average peer message box depth: Average peer message box depth is the average value of the number of messages that peers can hold of all peers in a given DTMANET scenario.

5.1.2 Control Data Classification

To control peer behaviour and to make message routing decisions, control mechanisms and data need to be fed to the environment before simulation starts. Such data includes movement model control data, routing control data and filter control data.

5.1.2.1 Movement Model data

1) Movement Models

Mobility to a mobile ad hoc network is like the music to a concert, if there is no mobility there will be no mobile ad hoc network. However, because of the diversity of DTMANETs, mobility can be drastically different from one scenario to another. In a simulation environment like OCI-SIM, mobile characteristics are abstracted and modelled into computational components for each of the unique types of DTMANETs that are under investigation. A movement model controls the behaviour of peers' movements in MANETs. By applying different movement models, the OCI system can simulate a wide range of mobile scenarios with highly varied mobility and other characteristics, such as a campus network, animal tracking network, ocean current monitoring network, and C3¹⁰ networks in Future Combat systems (Integrated network/JTRS¹¹, personnel network/WIN-T¹² and airborne network/UCAVN¹³).

2) Built-in and Customizable Movement Models

In OCI simulations, there are several built-in movement models, which include random walk model, random direction model, random pause model, random waypoint model, linear movement model and chaos movement model. For example, one of the built-in movement models is "Random Walk Model". It simulates the mobility model of DTMANETs, in which all peers have total autonomy of their movement. In the random walk model, peers may randomly decide when to move or stop, change velocity or directions. Being a highly customizable and extensible system, new models can be created with the "Movement Model Customization Module" of OCI-CUSTOM, compiled and integrated into the system even when the simulator is running, no system restarting is required. The following code is a simple movement model that randomly changes a peer's movement direction.

¹⁰ C3: Command, Control, Communication.

¹¹ JTRS: Joint Tactical Radio System

¹² WIN-T: War-fighter Information Network-Tactical

¹³ UCAVN: Unmanned Combat Aerial Vehicle Network

<pre> public static int[] randomDirectionModel(int[] activness) { int XDirection = 1, YDirection = 1; switch (seed.nextInt(6)) { case 0: XDirection = -1; YDirection = 1; break; case 1: XDirection = 1; YDirection = -1; break; case 2: XDirection = -1; YDirection = -1; } } </pre> <p style="text-align: right;">Page 1 of 2</p>	<pre> break; default: break; } activness [0] *= XDirection; activness [1] *= YDirection; return activness; } </pre> <p style="text-align: right;">Page 2 of 2</p>
---	---

CODE 5-3: AN EXAMPLE OF MOVEMENT MODEL SOURCE CODE

3) Movement Models Overlay

In a complex DTMANET simulation, mobility is not conformed from one peer to another. Thus, in the OCI, mobility model overlay is provided, whereby different movement models can be assigned to different peers in the same simulation. Therefore, complex scenarios with mobile devices that use different movement patterns can be constructed and simulated.

5.1.2.2 Filter data

In the course of this research on the OCI, routing protocols are implemented as filters. Like the conductor of an orchestra, OCI filters instruct the way that peers' communication with one another and how the routing decisions should be made within a DTMANET.

1) Non-Opportunistic Routing Protocol Filters

The term “non-Opportunistic routing protocol” refers to those routing protocols that do not rely on opportunistic information to route messages in DTMANETs, and they are considered fundamental routing mechanisms. In OCI simulations, four

non-opportunistic DTMANET routing protocols are implemented and they are “Custody Transfer” routing, “Custody Transfer Propagation” routing, “Custody Retaining Propagation” routing and “Simple Flood Propagation” routing.

2) Opportunistic Routing Protocol Filters

The idea of OCI is to provide Opportunistic Confidence Indexes that will enable multiple opportunistic routing protocols to be used as an adaptive group, and will produce significant and balanced improvement for the routing process of any given DTMANET scenario.

Therefore, filters that represent opportunistic routing protocols are the most essential components for this research. In this research, there are a number of opportunistic routing protocol designs implemented. In these designs, a variety of factors are involved, these factors are defined across layers of DTMANET, and cover a large number of totally different elements such as time, hop count, transmission range, speed, territory, reputation and so on. In OCI, such elements are simplified as filters and their functions are also simplified as binary values, to determine the “to route OR not to route” policy for messages in the simulation.

Although this research works with a number of opportunistic routing protocols, the purpose of this research is neither to create yet another opportunistic routing protocol nor an exhaustive evaluation and comparison of the efficiency and performance for each existing individual opportunistic routing protocol. Instead, it proposes a new framework to look at all opportunistic routing protocols as a single entity, and to create a new methodology that may coordinate existing and emerging opportunistic routing protocols if suitable, which work together to increase the routing reliability, efficiency and performance for given DTMANET scenarios.

3) Customized Routing Protocol Filters

In OCI simulations, new protocols can be dynamically introduced into the simulation for both opportunistic and non-opportunistic routing protocols with the OCI-CUSTOM “Filter Customization Module”. The following example defines a filter that removes messages from a system when their HTL limits are reached.

<pre>public boolean HTLFilter(OCIMsg msg, String msgID, int HTL, int currentHopCount) { if (currentHopCount < HTL) { return true; } else { for (int removeExpiredMsgIndex = 0; removeExpiredMsgIndex < bridger. peerMsgVec.size(); removeExpiredMsgIndex++) { if (((OCIMsg) (bridger.peerMsgVec.get(removeExpiredMsgI ndex))). msgID.equals(msg.msgID)) { bridger.peerMsgVec.remove(removeExpired MsgIndex); } } int msgNum = 0;</pre> <p style="text-align: right;">Page 1 of 2</p>	<pre>OCIPeer tmpPeer = (OCIPeer) peerVec.get(i); for (int j = 0; j < tmpPeer.peerMsgVec.size(); j++) { if (((OCIMsg) (tmpPeer.peerMsgVec.get(j))). msgID.equals(msg.msgID)) { msgNum++; } } if (msgNum == 0) { //Start: update HTLExpired status msg.isHTLExpired = 1; //End: update HTLExpired status ((OCIMainGUI) parentFram5). mainInfoTextArea.append("Message ID: " + msgID + " expired(HTL)" + "\n"); OCIMainGUI.HTLExpiredMsgCount++; return false; }</pre> <p style="text-align: right;">Page 2 of 2</p>
---	---

CODE 5-4: AN EXAMPLE OF FILTER MODULE SOURCE CODE

5.1.3 Output Data and Data Evaluation Classification

Supplying an OCI system with initial peer data, initial message data, movement data and opportunistic filter data, a series of output data need to be generated using either OCI-AUTO or OCI-VISUAL. Choosing the initial data will cause output data to vary, therefore through an analytical comparison (see *section 6.2* and *section 6.3*), OCI will calculate and suggest an index of factors that influence message routing communication in a given DTMANET scenario. The following are a list of such data.

1) Message delivery rate:

Message delivery rate is the number of successfully received messages over the total number of messages that have been sent. The message delivery rate is commonly used to evaluate the reliability of a given DTMANET scenario.

2) Average message delivery latency:

Average message delivery latency is defined as the average end-to-end time delay incurred by the delivered messages. It is measured through the number of time units t_{Δ} that is defined in section 3.5.1.1. In OCI, the average message delivery latency is used to evaluate the performance of a given scenario.

3) Message delivery routing depth:

Message delivery routing depth is the average number of hop counts for successfully delivered messages. It is used to evaluate the efficiency of given scenarios.

4) Data Duplication:

Data duplication is measured through the number of cached duplicates for each successfully delivered message. This is important for the efficiency evaluation of propagation based routing protocols such as “Simple Flood Propagation”.

5.2 OCI Simulation Scenario Design and Construction

As a meta-protocol, the concept of OCI can be universally applied to any DTMANET that uses opportunistic elements to aid the routing decision making process. However, a specific OCI index will only be meaningful, useful, and applicable for a given class of DTMANET scenarios. The reason for this is that there are a range of possible opportunistic elements available, and the same opportunistic element might be used in

a variety of ways in different scenarios. For instance, the OCI index generated for an inter-planet DTMANET is unlikely to be useful when applied to a DTMANET that serves as a combat infantry personnel-messaging system.

5.2.1 Simulation Scenario – Representation of DTMANET

As discussed in chapter three, an OCI simulation scenario contains three major components, namely a Movement Model, Opportunistic Filters and DTMANET fundamental routing protocols (non-opportunistic). Therefore, theoretically, the number of possible combinations of scenario constructions is infinite, since new protocols, new movement models and new filters can be added into an OCI system.

$$\mathcal{N} = (\mathcal{M}_{m_1} + \mathcal{M}_{m_2} + \dots + \mathcal{M}_{m_m}) \times (\mathcal{F}_{f_1} + \mathcal{F}_{f_2} + \dots + \mathcal{F}_{f_f}) \times (\mathcal{P}_{p_1} + \mathcal{P}_{p_2} + \dots + \mathcal{P}_{p_p})$$

(\mathcal{N} : the number of possible scenarios; Given \mathcal{M} : movement model permutation; m : number of movement models; \mathcal{F} : filter permutation; f : number of filters; \mathcal{P} : protocol permutation; p : number of protocols)

To exhaustively simulate all possible scenarios is neither practical nor necessary. Instead, in this research, DTMANET scenarios are categorised into a coarse-grained taxonomy. Simulations are conducted using scenarios that distinctively represent the feature of each of the different types of DTMANET. Through this approach, the viability of the OCI concept may be assessed across a broad range of applications, and within a reasonable time scale.

5.2.2 DTMANET Scenarios – A Coarse-grained Taxonomy

In order to conduct meaningful simulations that represent a wide range of DTMANET scenarios in an organized manner, in this research, DTMANETs are coarsely classified into three principal categories based on their peer conformity, priority preference and spatial-constraint. Each of these principal categories is further sub-categorized into a number of sub-categories.

5.2.2.1 Peer Conformity Based Taxonomy

1) DTMANET peer communication protocol conformity

a) Homogeneous-device DTMANET

In a homogeneous-device DTMANET, all peers in the network use the same wireless communication protocol. This means that all peers have the potential to communicate with each other when in communication range.

b) Heterogeneous-device DTMANET

Supported wireless communication protocols in a heterogeneous-device network vary from peer to peer, which implies that communication can only be directly established between peers with compatible protocols. However, peers which support multiple communication protocols can act as message bridges.

2) DTMANET peer movement conformity

a) DTMANET movement spatial conformity

i. Uniform peer movement DTMANET

In such networks, peers maintain a uniform movement trend.

ii. Autonomous peer movement DTMANET

Peers in this network move in a non-uniform manner; each peer moves with autonomously decided velocity, territory and direction.

b) DTMANET movement schedule conformity

i. Scheduled movement DTMANET

Mobile entities such as buses and trains follow schedules, therefore movement in such networks is predictable.

ii. Randomly triggered movement DTMANET

Movement in such a network is decided autonomously and randomly by each peer, which does not follow any fixed pattern, therefore, movement is not predictable.

3) DTMANET peer routing participation conformity

a) Disciplined routing DTMANET

In a disciplined routing DTMANET, message caching and relaying are mandatory.

b) Autonomous routing DTMANET

Peers in an autonomous routing DTMANET decide their own participation in caching and relaying of specific messages. If its decision is negative, it will override all other opportunistic routing decision preferences.

5.2.2.2 Priority Preference Based Taxonomy

1) Performance priority DTMANET

In some situations, such as in battle field communication, a DTMANET will be designed with optimum time performance, due to the need for rapid information updates. Therefore, the OCI index for scenarios like this prefers rapid message delivery (performance) over message duplication (efficiency).

2) Efficiency priority DTMANET

An example to illustrate this type of DTMANET would be a sharing and distribution system for large-sized files and data, such as ultrahigh-resolution satellite surveillance photo or High quality Digital Video that can be a few GB in size. Understandably, file relay efficiency for such a network has to be the priority. Therefore, OCI uses file duplication as an indexing mechanism in this type of DTMANET.

3) Reliability priority DTMANET

A DTMANET used for long term monitoring purposes such as an ocean current monitoring network, does not require rapid update nor does it require huge transmission overhead. However, successful data relay is essential. In this case, an OCI index is generated based on the success of message delivery rates.

4) Balanced DTMANET

In a balanced DTMANET, all three elements of routing quality, reliability, performance and efficiency, have to be taken into consideration. However, because of the nature of a DTMANET, tradeoffs have to be made in order to achieve a balanced routing quality metric.

5.2.2.3 Spatial-constraint Based Taxonomy

1) DTMANET peer density

- a) High density network
- b) Low density network

2) DTMANET peer velocity

- a) High velocity network
- b) Low velocity network

- 3) DTMANET peer territory
 - a) Peers with big average territory
 - b) Peers with small average territory

5.2.3 Quantitative Definition of Relative Terms

Fundamentally, it is a philosophical question to quantitatively define the term high and low. How many peers in a DTMANET will make it a high density network? How fast do peers have to move to make a high velocity DTMANET? To avoid Sorites paradox¹⁴, the OCI defines the term high and low using the following system, using density as an example:

5.2.3.1 The Density Measurement Unit

When describing peer density in a DTMANET, intuitively, as most other research suggested, the density value D is a function of network boundary enclosed area over the peer number n :

$$D = f(n, a) = \frac{a}{n}$$

However, through simulations and observations, the value D does not provide conclusive and useful results through data analysis. This is because there is another essential element to the density evolution, and it is the peer transmission range. Therefore, in OCI peer density is defined as:

$$D_{\text{OCI}} = f(n, r, a) = \frac{a}{(n \times r^2)}$$

Where D_{OCI} is the density of a given simulation area, a contains n peers with average transmission range r .

¹⁴ Sorites Paradox is a classic paradox that mixes a quantitative concept with a cognitive concept [Franceschi, 2003]

5.2.3.2 The minimum threshold – Absolute “Low” density

Before defining the relative term “low”, it needs to be defined in its absolute condition, and the minimum threshold for density is:

$$D_{\text{Absolute Low}} = \frac{a}{(n \times r^2)}$$

$$\text{When: } n_{\text{Absolute Low}} = 2$$

If there are less than 2 peers in a simulation area, it cannot be called a network.

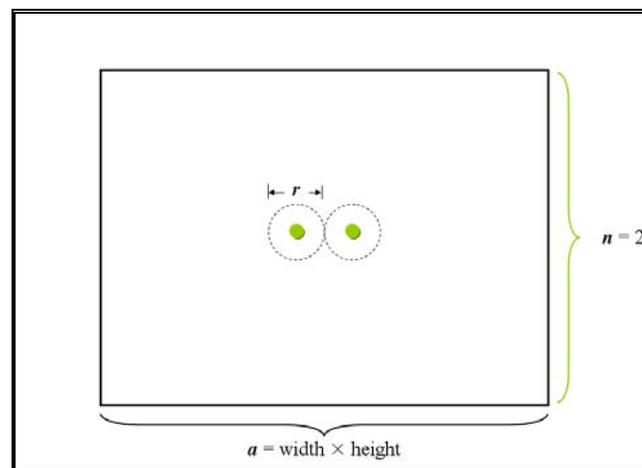


FIGURE 5-1: ABSOLUTE LOW DENSITY

5.2.3.3 The maximum threshold – Absolute “High” density

Before the term “high” can be defined in a relative term, its absolute value needs to be defined. In OCI the absolute “High” is defined as

$$D_{\text{Absolute High}} = \frac{a}{(n \times r^2)} = 1 = 100\%$$

$$\text{Where: } n_{\text{Absolute High}} = \frac{a}{r^2}$$

As shown in *Figure 5-2*, when n peers with average transmission range r are evenly distributed, the entire simulation area a is covered. Any $n > n_{\text{Absolute High}}$ is high.

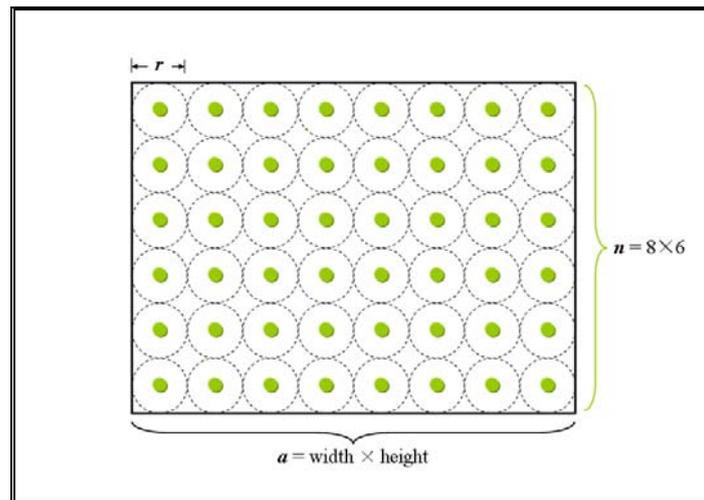


FIGURE 5-2: ABSOLUTE HIGH DENSITY

5.2.3.4 Relative “Low” and Relative “High”

1) Initial relative “Low”

To start, the initial relative “Low” is set to:

$$n_{\text{Initial Relative Low}} = n_{\text{Average}} - n_{\text{Average}} / 2^1;$$

$$\text{(Where: } n_{\text{Average}} = (n_{\text{Absolute High}} + n_{\text{Absolute Low}}) / 2;)$$

2) Initial relative “High”

The initial relative “High” is set to

$$n_{\text{Initial Relative High}} = n_{\text{Average}} + n_{\text{Average}} / 2^1;$$

$$\text{(Where: } n_{\text{Average}} = (n_{\text{Absolute High}} - n_{\text{Absolute Low}}) / 2;)$$

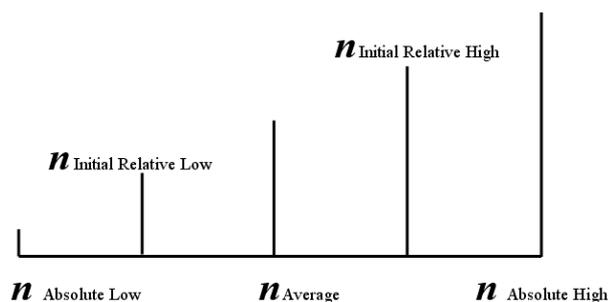


FIGURE 5-3: INITIAL RELATIVE DENSITY

3) “Compressed” Relative Low and High

To investigate the limit and boundary effect, through sub-sectioning, the new relative low and high value can be compressed to an average value:

$$n_{\text{Compressed Relative Low}} = n_{\text{Average}} - n_{\text{Average}} / 2^i;$$

And the Compressed relative “High” is set to:

$$n_{\text{Compressed Relative High}} = n_{\text{Average}} + n_{\text{Average}} / 2^i;$$

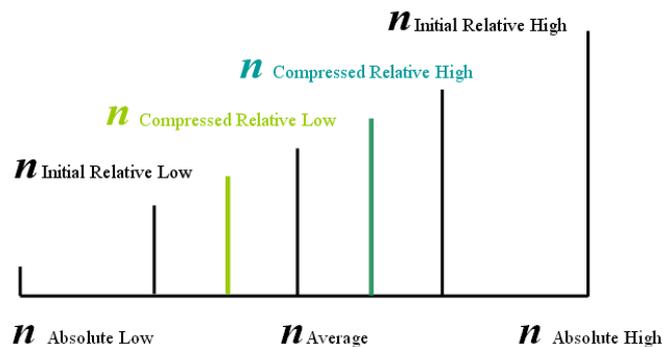


FIGURE 5-4: COMPRESSED RELATIVE DENSITY

4) “Expanded” Relative Low and High

At the same time, through the technique of sub-sectioning, new relative low and high values can be expanded from the average value:

$$n_{\text{Expanded Relative Low}} = n_{\text{Absolute Low}} + n_{\text{Average}} / 2^i;$$

And the expanded relative “High” is set to:

$$n_{\text{Expanded Relative High}} = n_{\text{Absolute High}} - n_{\text{Average}} / 2^i;$$

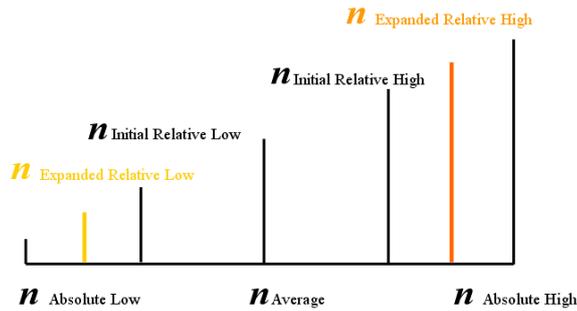


FIGURE 5-5: EXPANDED RELATIVE DENSITY

5.2.4 OCI Simulation Procedure

From the taxonomy discussion in the previous section, theoretically complete opportunistic DTMANET scenarios can now be constructed. They can be constructed from representations offered to them in OCI-SIM, and they will share the features outlined in the following sub-sections.

5.2.4.1 Random assignment of opportunistic elements

- 1) Communication protocol

Heterogeneous wireless communication protocols are randomly selected from an available OCI communication protocol reservoir.

- 2) Start up location

Peer initial location is randomly assigned within the limit of $[0, A_X]$, $[0, A_Y]$ where A is the boundary limit value of a given simulation area.

- 3) Transmission range

Peer transmission range is randomly generated within the boundary of $[0, R]$ where R is the maximum transmission range of a selected communication protocol.

4) Peer territory

Peer movement territory is randomly generated within the boundary of $[0, A_X]$ and $[0, A_Y]$ where A is the boundary limit value of a given simulation area.

5) Peer velocity

Peer velocity is randomly generated within the boundary of $[0, V]$ where V is the maximum velocity of a selected peer.

6) Random movement model

The peer movement is dictated through the Random Waypoint or Random Walk algorithms. A peer can change its velocity and direction at any given moment. In addition, peer movement is spontaneous and does not follow a set schedule. Hence, there are no fixed movement patterns.

7) Routing participation

Message caching and relaying are autonomous. A peer may reject message passing at any given moment.

5.2.4.2 OCI Auto Simulation Procedure

1) Control (Best-case and worst-case scenario) simulation

Control points have to be set before any further evaluation of DTMANET routing efficiency can continue. For this reason, several simple extreme case scenarios are constructed and tested. The results of such simulations are used as control points to benchmark subsequent experiments.

2) Variable time frame message delivery scenario simulation

In this scenario we first set a relatively small time value. If within this time frame

no delivery has been made, the time value will be increased automatically and the process repeated until the message is delivered to its destination, or the time frame value reaches its absolute threshold depending on the scenario the OCI system is simulating. By doing so, an ideal TTL can be calculated for a given scenario.

3) Variable routing depth message delivery scenario simulation

This scenario starts with a relatively small value of HTL. If no delivery has been made within this hop limit, the HTL value will be increased automatically and the process repeated until the message is delivered to its destination, or the HTL value reaches its absolute threshold, depending on the scenario the OCI system is simulating. From this, an ideal HTL can be calculated for a given scenario.

4) Variable peer density scenario simulation

This simulation scenario starts with a relatively small number of peers in a fixed network scope. If within a fixed time threshold no delivery has been made, more peers will be added into the system automatically and the process will be repeated until the message is delivered to its destination. This can test the theoretical minimum density of a DTMANET with certain types of communication protocols, in order to function and be useful. At the same time, this test can also calculate the optimised density for certain types of protocols at any given targeting delivery rate.

5) Other variable peer opportunistic attributes scenario simulation

Using the same technique as described in the previous two scenarios, users can construct many other scenarios to compute the optimized value of peer attributes of their choice.

6) Opportunistic filter isolation scenario simulation

In this simulation, only one opportunistic protocol at a time is used to aid the routing decision making process. The result of this will produce an “importance index” and “improvement value set” for head-to-head comparison of individual opportunistic filters. Alternatively, the results can later be used as initial weight assignments for an intelligent simulation.

7) Filter permutation multiple scenario simulation

In this simulation, we use fixed numbers of peers and assign them with optimized attributes such as TTL and HTL which are provided through previously mentioned scenarios. Through permuting selected filters, the data of each delivered message along with the associated filter permutation sequence are stored for further analysis.

$$\mathcal{N} = \mathcal{F}_{f_1} + \mathcal{F}_{f_2} + \dots + \mathcal{F}_{f_f}$$

(\mathcal{N} : the number of permutations; Given \mathcal{F} : filter permutation; f : number of selected filters)

8) Intelligently weighted opportunistic filter scenario simulation

In this simulation, the weights produced through opportunistic filter isolation are assigned to selected opportunistic filters. The result of this simulation will be the main evidence of the value and usefulness of certain OCI index combinations.

5.3 OCI Simulation Validity Control Procedure

One needs some basis for believing the output of a simulator, either by simulating scenarios of known behaviour and comparing the simulated results with the known behaviour, or by ensuring that the basis for the generation of the simulated results has been tested for accuracy in isolation. In the subsequent sections, an example scenario is chosen to demonstrate the procedures of OCI simulation validity control.

EXAMPLE SCENARIO CONFIGURATION	
(Spatial unit s_{Δ} Time unit t_{Δ})	
Peer Number:	100
Transmission Type:	Single Protocol (Bluetooth)
Transmission Range:	0 to 50 s_{Δ}
Simulation Area :	500 $s_{\Delta} \times 500 s_{\Delta}$
Peer Velocity:	0 to 25 s_{Δ} / t_{Δ}
Message Number:	10
PRNG:	java.util.Random

TABLE 5-1: EXAMPLE SCENARIO CONFIGURATION

5.3.1 PRNG Validity Control

To ensure the even distribution of random generated opportunistic attributes and other DTMANET simulation attributes, the PRNG (Pseudo Random Number Generator) chosen as the random generation mechanism is tested. Tests are set up and executed 100, 1000, 10 000 and 100 000 times respectively.

```
C:\OCIDoc>java RandomTest 100
*****
num within 0 to 9: 19
num within 10 to 19: 11
num within 20 to 29: 8
num within 30 to 39: 7
num within 40 to 49: 5
num within 50 to 59: 10
num within 60 to 69: 10
num within 70 to 79: 9
num within 80 to 89: 9
num within 90 to 99: 12
*****
num = 0: 2
num = 99: 1
```

(A) 100 REPEATS

```
C:\OCIDoc>java RandomTest 1000
*****
num within 0 to 9: 83
num within 10 to 19: 106
num within 20 to 29: 106
num within 30 to 39: 101
num within 40 to 49: 87
num within 50 to 59: 122
num within 60 to 69: 95
num within 70 to 79: 89
num within 80 to 89: 108
num within 90 to 99: 103
*****
num = 0: 13
num = 99: 11
```

(B) 1000 REPEATS

```
C:\OCIDoc>java RandomTest 10000
*****
num within 0 to 9: 955
num within 10 to 19: 1004
num within 20 to 29: 1025
num within 30 to 39: 1008
num within 40 to 49: 1048
num within 50 to 59: 984
num within 60 to 69: 1031
num within 70 to 79: 997
num within 80 to 89: 959
num within 90 to 99: 989
*****
num = 0: 90
num = 99: 96
```

(C) 10,000 REPEATS

```
C:\OCIDoc>java RandomTest 100000
*****
num within 0 to 9: 9909
num within 10 to 19: 9903
num within 20 to 29: 10027
num within 30 to 39: 10007
num within 40 to 49: 10200
num within 50 to 59: 9960
num within 60 to 69: 10088
num within 70 to 79: 10051
num within 80 to 89: 9875
num within 90 to 99: 9980
*****
num = 0: 1003
num = 99: 988
```

(D) 100,000 REPEATS

FIGURE 5-6: PRNG DISTRIBUTION TEST

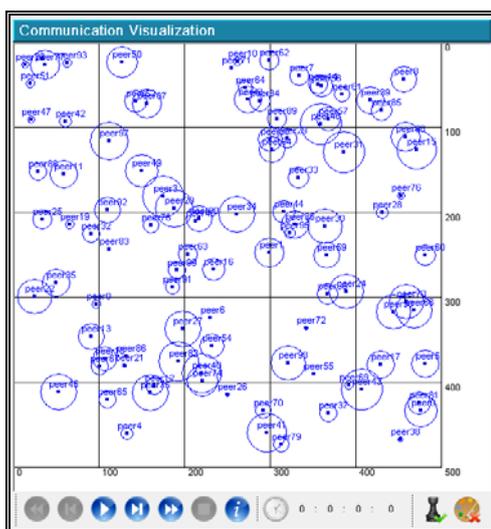
From the results presented in *Figure 5-6*, we can see that numbers are evenly distributed. PRNG theory has been extensively explored in the literature and shown to be suitable for use in simulations [Savrasovs, 2003], therefore the PRNG used is sufficient for OCI-SIM scenario generation.

5.3.2 Define relative terms

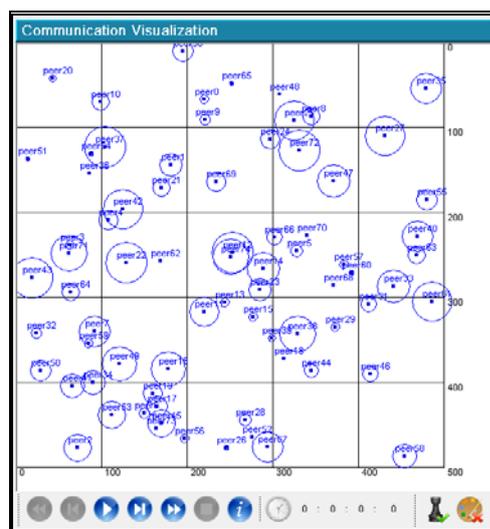
The usefulness of OCI indices are scenario sensitive. To give validity to an OCI index that is generated from a single simulation to a more general scope, relative terms are defined. Thus, for instance, an OCI index can be described as applicable to “high density” and “low velocity” scenarios. The technique of using a subsection to define relative terms such as density and velocity is described in *section 5.2.3*.

In the following scenario, the simulation area has a size of 500×500 spatial units s_{Δ} and mobile device transmission range is $50 s_{\Delta}$. The following relative terms are defined in relation to the constants of simulation area size and transmission range.

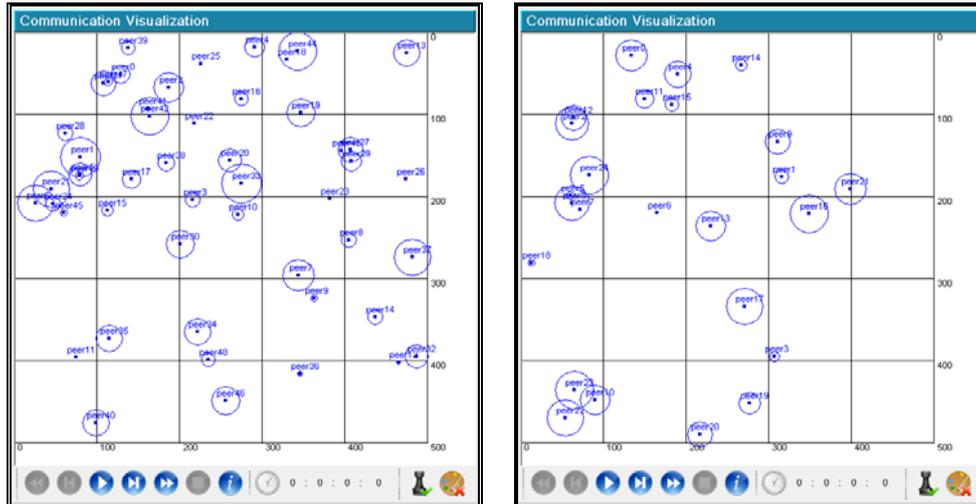
- 1) The term of density level is defined as:



(A) $N_{\text{ABSOLUTE HIGH}} = 100$



(B) $N_{\text{INITIAL RELATIVE HIGH}} = 75$



(C) $N_{\text{AVERAGE}} = 50$

(D) $N_{\text{INITIAL RELATIVE LOW}} = 25$

FIGURE 5-7: RELATIVE PEER DENSITY

2) The term of territory level is defined as:

a) Absolute big territory: $500 \times 500 s_{\Delta}$

Absolute big territory equals to the size of the simulation area.

b) Average size territory: $250 \times 250 s_{\Delta}$

Average size territory is 1/4 of the size of the simulation area.

c) Initial relative big territory: $375 \times 375 s_{\Delta}$

Initial relative big territory is defined as 9/16 of the size of the simulation area.

d) Initial relative small territory: $125 \times 125 s_{\Delta}$

Initial relative small territory is defined as 1/16 of the size of the simulation area.

3) The term of peer velocity level is defined as:

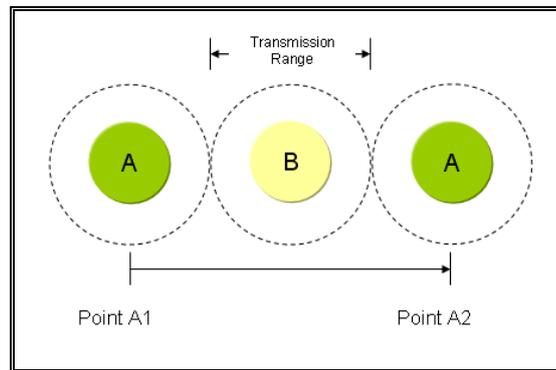


FIGURE 5-8: RELATIVE PEER VELOCITY

a) Absolute high velocity: $100 s_{\Delta} / t_{\Delta}$

Given peer B is stationary, absolute high velocity is defined as the distance for peer A to travel from point A1 to point A2 in a single time unit t_{Δ} , which means, if a peer's velocity is absolutely high, in one time unit t_{Δ} it will just pass through (skip) its neighbouring stationary peer.

b) Absolute low velocity: $0 s_{\Delta} / t_{\Delta}$

When a peer's velocity is absolute low, it is stationary.

c) Average velocity: $50 s_{\Delta} / t_{\Delta}$

Average velocity is 1/2 of the value of the difference between absolute low and absolute high velocity.

d) Initial relative high velocity: $75 s_{\Delta} / t_{\Delta}$

Initial relative high velocity is defined as 3/4 of the value of the difference between absolute low and absolute high velocity.

e) Initial relative low velocity: $25 s_{\Delta} / t_{\Delta}$

Initial relative low velocity is defined as 1/4 of the value of the difference between absolute low and absolute high velocity.

5.3.3 Statistical Validity Control

OCI-SIM performs repetition optimization before OCI index generation. *Table 5-2* and *Figure 5-9* show delivery rate, delivery time, and delivery hop values from 1 to 400 simulation repeats (1 simulation repeat contains 10 message transmissions).

AVERAGE DELIVERY STATUS			
Repeat:	Delivery Rate	Deliver Time	Deliver Hop
1 × 10 MSGs	70.0 %	81.00	27.00
5 × 10 MSGs	82.0 %	64.20	20.60
10 × 10 MSGs	67.0 %	60.30	17.80
20 × 10 MSGs	74.5 %	70.20	21.10
40 × 10 MSGs	74.3 %	62.15	19.50
60 × 10 MSGs	75.7 %	64.73	19.58
80 × 10 MSGs	77.0 %	70.29	20.53
100 × 10 MSGs	77.1 %	69.35	20.39
120 × 10 MSGs	73.6 %	66.39	19.74
140 × 10 MSGs	75.7 %	65.64	19.58
160 × 10 MSGs	76.4 %	66.96	20.09
180 × 10 MSGs	75.5 %	66.73	19.54
200 × 10 MSGs	74.8 %	64.68	19.47
400 × 10 MSGs	75.9 %	69.29	20.16

TABLE 5-2: AVERAGE DELIVERY STATUS (1 TO 400 SIMULATION REPEATS)

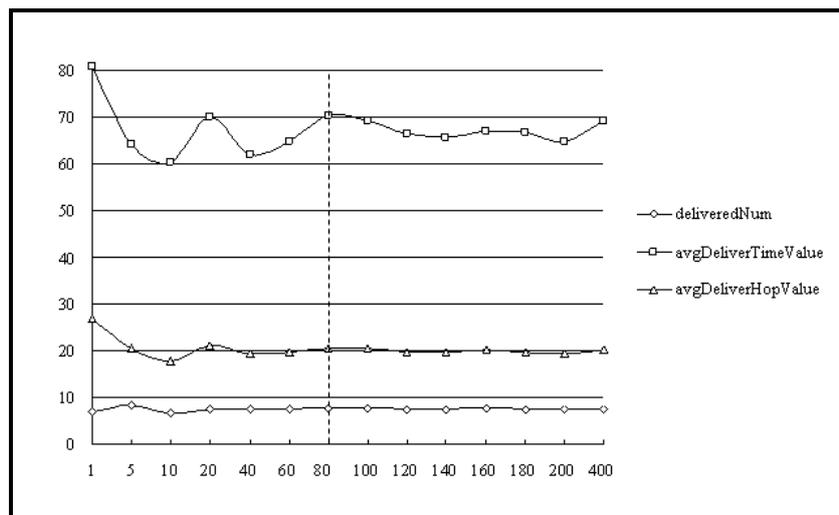


FIGURE 5-9: AVERAGE DELIVERY FLUCTUATION (1 TO 400 SIMULATION REPEATS)

From observation of the *Figure 5-9*, it is clear that fluctuation calms down after 80 simulation repeats. Thus, 80 simulations will be performed for OCI indexing generation of this particular scenario.

5.3.4 TTL and HTL Validity Control

TTL and HTL settings have substantial effects on the quality of messaging based routing protocols. If the values are too small, messages will be dropped prematurely, if the values are too large, the simulation will become very time consuming. In addition, different base routing protocols require different TTL (Time to Live) and HTL (Hops to Live) threshold settings. Four base routing protocols are implemented in OCI-SIM (see *section 3.5.2.3*), and their TTL and HTL values for the tested scenarios are listed as the following:

- 1) “Custody Transfer” Routing:

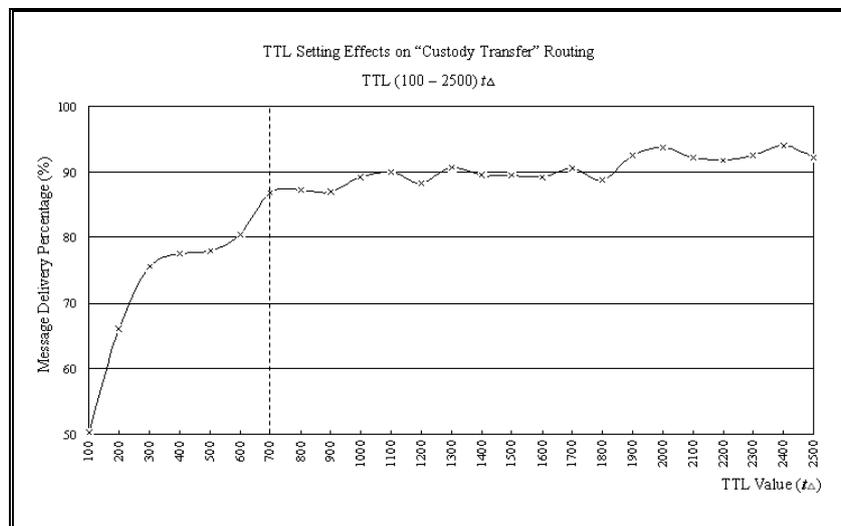


FIGURE 5-10: AVERAGE DELIVERY RATE OVER TTL (CUSTODY TRANSFER)

In this scenario, using “Custody Transfer” Routing, the optimized threshold is 700 $t\Delta$ where delivery percentage fluctuation eased.

2) “Custody Transfer Propagation” Routing:

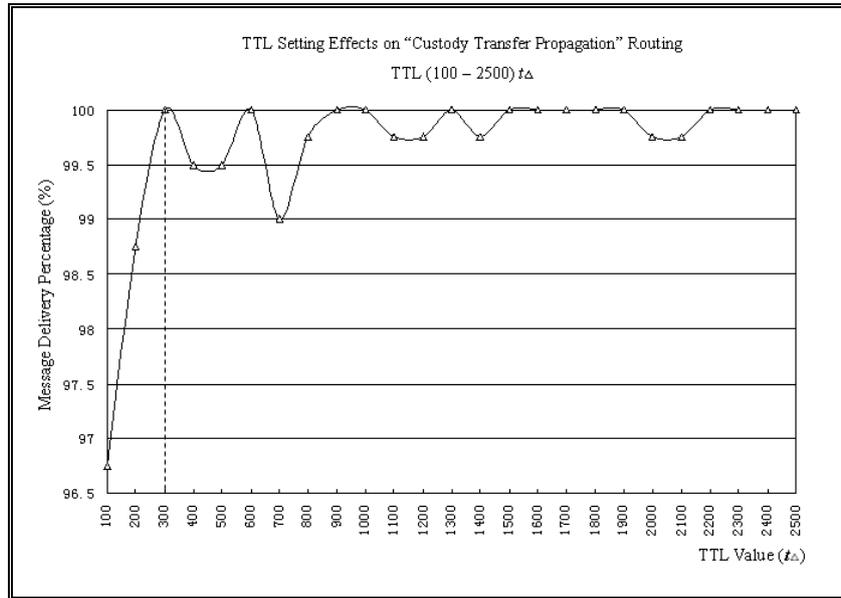


FIGURE 5-11: AVERAGE DELIVERY RATE OVER TTL (CUSTODY TRANSFER PROPAGATION)

With “Custody Transfer Propagation” Routing, after 300 t_{Δ} there is no significant improvement on delivery percentage.

3) “Custody Retaining Propagation” Routing:

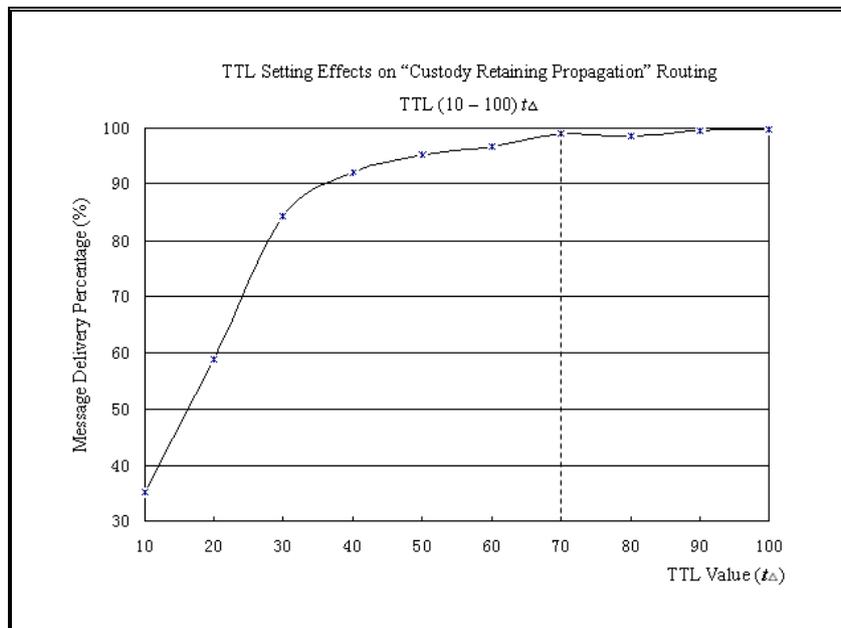


FIGURE 5-12: AVERAGE DELIVERY RATE OVER TTL (CUSTODY RETAINING PROPAGATION)

Under “Custody Retaining Propagation” Routing, it is obvious, from *Figure 5-12*, that when TTL is set to $70 t_{\Delta}$ the simulation system reaches its optimized peak delivery percentage.

4) “Simple Flood Propagation” Routing:

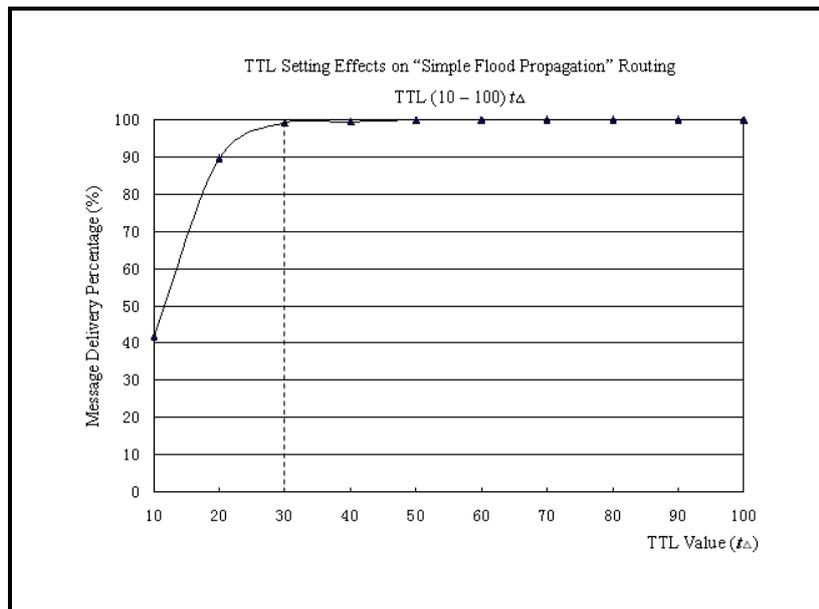


FIGURE 5-13: AVERAGE DELIVERY RATE OVER TTL (SIMPLE FLOOD)

While using the “Simple Flood Propagation” Routing, as illustrated in *Figure 5-13*, $30 t_{\Delta}$ TTL is enough to provide nearly 100% message delivery.

5) HTL Optimization

HTL optimized thresholds are set to the average delivery routing depth from the simulation that run under the optimized TTL settings.

6) Statistic Control

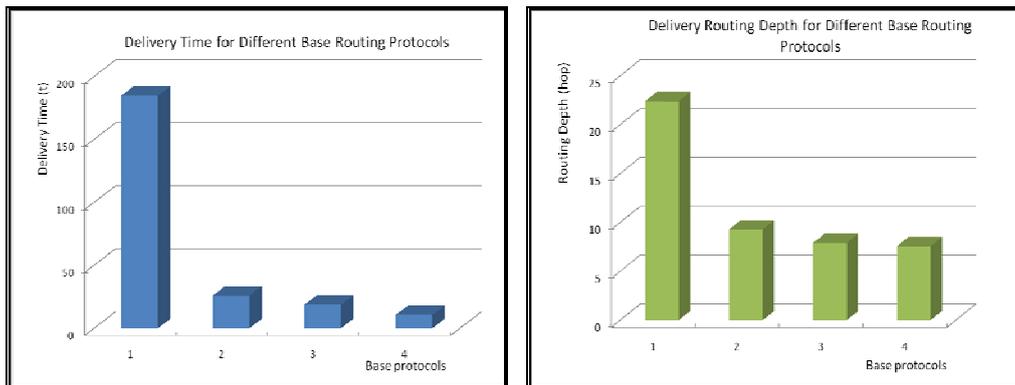
All the tests for generating optimized TTL and HTL values conform to the statistic control, which means, each value is calculated from 80 simulation repeats.

TTL / HTL Optimization		
Base Routing protocols	Optimized TTL	Optimized HTL
Custody Transfer	700 (t_{Δ})	23 (hops)
Custody Transfer Propagation	300 (t_{Δ})	10 (hops)
Custody Retaining Propagation	70 (t_{Δ})	8 (hops)
Simply Flood Propagation	30 (t_{Δ})	8 (hops)

TABLE 5-3: TTL / HTL OPTIMIZATION

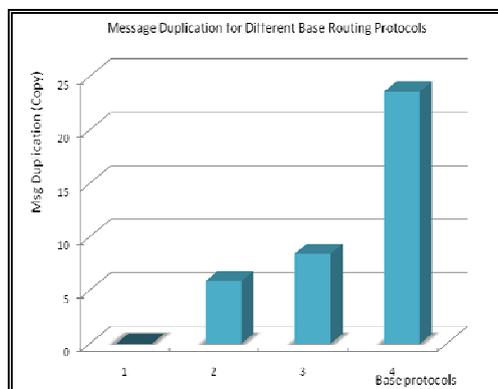
5.3.5 Base Routing Scenario Classification:

One important aspect that determines the usefulness of an OCI index is the non-opportunistic base routing protocol on which it operates. OCI indices generated from one base routing protocol are not comparable to scenarios that adopt different base routing protocols. Thus, to classify these non-opportunistic base routing protocols, and understand their strengths and weaknesses will help to target the evaluation of OCI indices credibly.



A. DELIVERY TIME FOR DIFFERENT BASE ROUTING

B. DELIVERY ROUTING DEPTH FOR DIFFERENT BASE ROUTING



C. MESSAGE DUPLICATION FOR DIFFERENT BASE ROUTING

FIGURE 5-14: ROUTING PERFORMANCE AND EFFICIENCY

The four columns in *Figure 5-14* represent the four base routing protocols respectively. From left to right they are:

1) “Custody Transfer” Routing

“Custody Transfer” Routing has zero communication overhead, however the time to deliver messages is 5 to 20 times slower than other base routing protocols, and it also takes deeper routes from the sender to the receiver. Therefore, it is suitable for the transfer of heavy weight message where time of delivery is flexible, such as multimedia content sharing or offline Internet content transfer.

2) “Custody Transfer Propagation” Routing

“Custody Transfer Propagation” Routing provides a much more balanced performance and efficiency. It provides a significant improvement on message delivery time and routing depth compared with “Custody Transfer” Routing and uses much less system resources than “Simple Flood”.

3) “Custody Retaining Propagation” Routing

Like “Custody Transfer Propagation”, the “Custody Retaining Propagation” Routing provides a very good balance between performance and efficiency. Meanwhile, through retaining the custody of a message, it is ideal for situations which require message distribution authentication.

4) “Simple Flood Propagation” Routing

“Simple Flood Propagation” Routing yields the best performance, but, it requires the highest communication overhead. The characteristics of the Simple Flood Propagation routing protocol make it suitable for emergency short message broadcasting and other scenarios that require fast low weight message delivery.

5.4 Chapter Review

This chapter covers the topic of data classification and validity control when conducting an OCI simulation. It describes the input and output data, as well as control data. It gives a coarse-grained taxonomy of OCI scenarios, and addresses the issue of validity control for the simulation approach adopted by this work. Furthermore, it outlines the steps taken to ensure that results would represent the test scenarios with reasonable accuracy. The next chapter will report the actual data analysis and the OCI performance findings of this work.

Chapter VI: OCI Data Analysis & Evaluation

“It is a capital mistake to theorize before one has data. Insensibly one begins to twist facts to suit theories, instead of theories to suit facts.”

–Sherlock Holmes

This chapter presents a series of simulation analyses over a selection of opportunistic elements, such as peer movement territory, velocity, movement consistency, reputation, and communication adaptor type. Opportunistic elements have various impacts on the reliability, performance, and efficiency of a DTMANET. The goal of this research is to assess the broad impact on opportunistic routing properties by setting up and testing specific network instances that are representative of the taxonomy of DTMANET scenarios detailed in the preceding chapter. The simulated test sequences apply OCI opportunistic filters to aid the routing decision making process, and consequently evaluate the viability of the OCI concept.

A large amount of data was collected during the course of this investigation. While the primary data to support the findings are included in the body of this chapter, additional data can be found in *Appendices C, D, E and F*.

6.1 Opportunistic Elements and Their Impacts on DTMANETs

This section shows the impact of opportunistic elements on DTMANET routing in OCI simulations. In each experiment, one parameter is varied at a time, which provides a controlled environment in which only the targeted element will influence the simulation results. The method of linear value variation is adopted, which alters the element's value with an even step or increment for each simulation. Therefore, the impact of each selected opportunistic element can be observed and analyzed qualitatively and quantitatively. The selected opportunistic elements are peer velocity, peer territory, communication range, communication protocol variation, movement consistency, and peer reputation. Each element is observed in simulations employing

four non-opportunistic base routing protocols (see *section 3.5.2.3*): “Custody Transfer” routing, “Custody Transfer Propagation” routing, “Custody Retaining Propagation” routing, and “Simple Flood Propagation” routing. Simulation environment configuration and validity control procedures of the following reports are described in *section 5.24*.

6.1.1 “Peer Velocity” and its effects on message delivery

In a DTMANET environment, peers in the network are most likely to travel at various speeds. This section includes simulation results of the impact of peer velocity on message delivery time, routing depth, and message duplication over the four base OCI routing protocols.

6.1.1.1 Effects on message delivery time

Figure 6-1-A to *Figure 6-1-D* display the resultant curves of delivery time with different velocity parameters over four different base non-opportunistic routing protocols “Custody Transfer”, “Custody Transfer Propagation”, “Custody Retaining Propagation” and “Simple Flood Propagation” respectively.

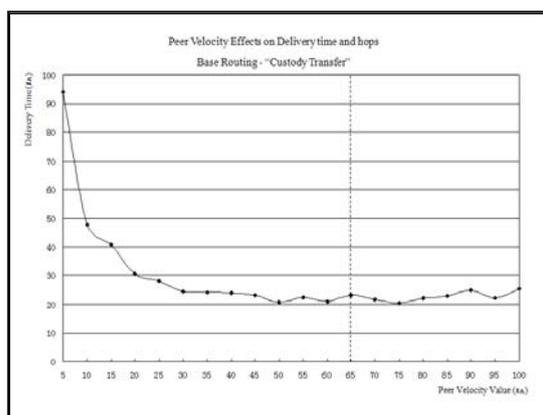


FIGURE 6-1-A: CUSTODY TRANSFER

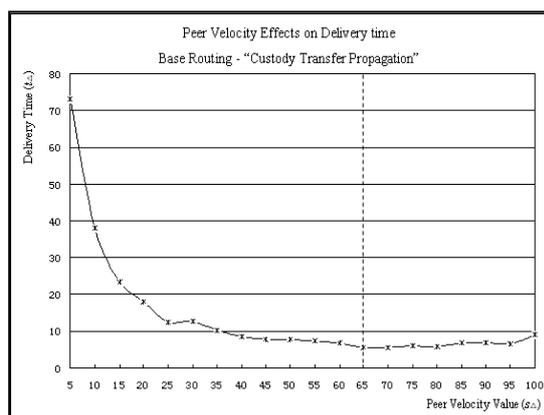


FIGURE 6-1-B: CUSTODY TRANSFER PROPAGATION

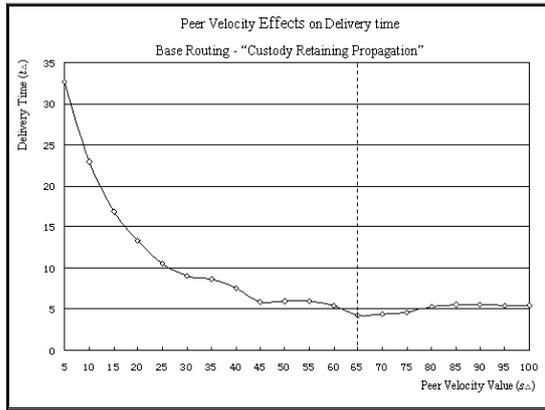


FIGURE 6-1-C: CUSTODY RETAINING PROPAGATION

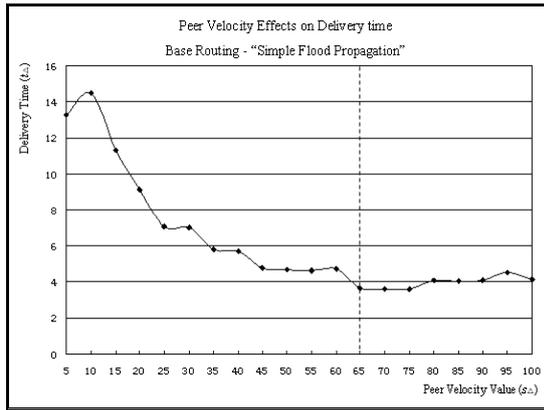


FIGURE 6-1-D: SIMPLE FLOOD PROPAGATION

These graphs show that peer velocity has a positive impact on the delivery time in this particular simulation configuration over all base protocols. When peer velocity ($v \in (0, 1, \dots, 64, 65)$), it reaches the peak performance at around $v = 65$, further increase of peer velocity does not improve the delivery time, instead, the Figures indicate a slightly negative effect.

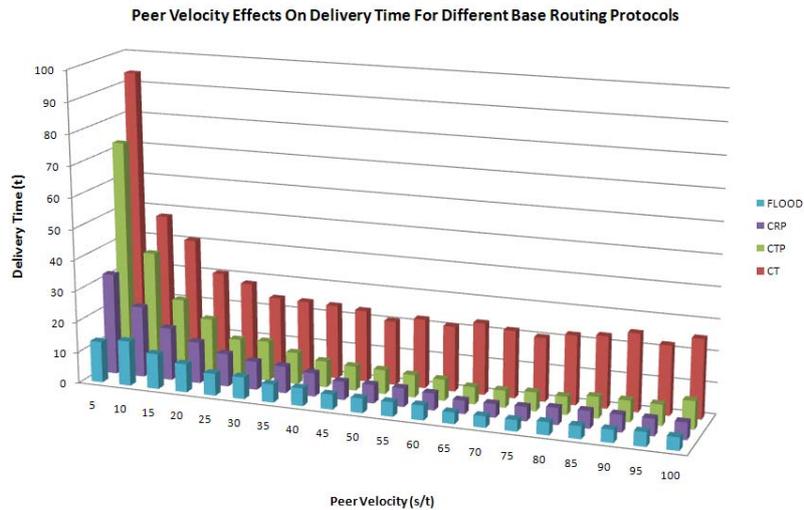


FIGURE 6-2: PEER VELOCITY EFFECTS ON DELIVERY TIME

Figure 6-2 shows that in all four base routing protocols, peer velocity has a considerable impact on message delivery time. However, it has the most significant effect on “Custody Transfer” routing and yields the least effect on “Simple Flood” routing.

6.1.1.2 Effects on Routing depth and Message duplication

Figure 6-3-A to Figure 6-3-D present observations of delivery routing depth and message duplication with different velocity settings over four different base non-opportunistic routing protocols.

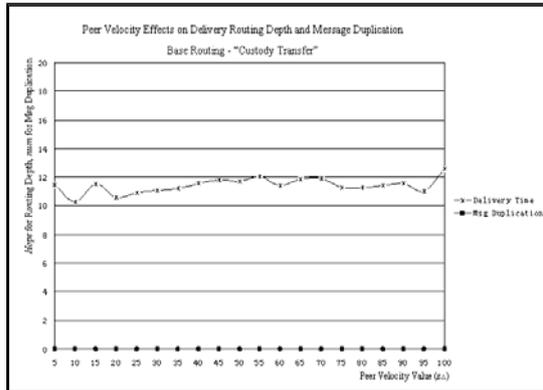


FIGURE 6-3-A: CUSTODY TRANSFER

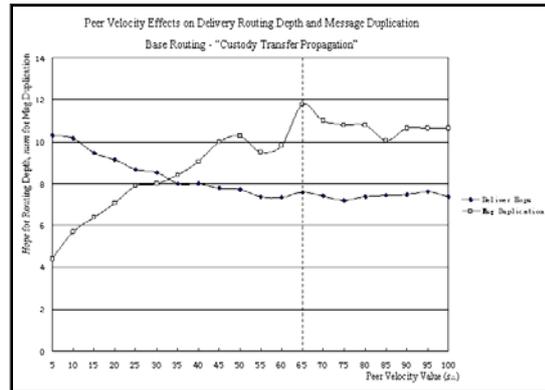


FIGURE 6-3-B: CUSTODY TRANSFER PROPAGATION

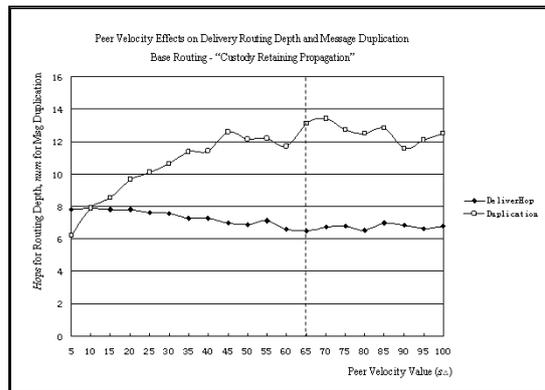


FIGURE 6-3-C: CUSTODY RETAINING PROPAGATION

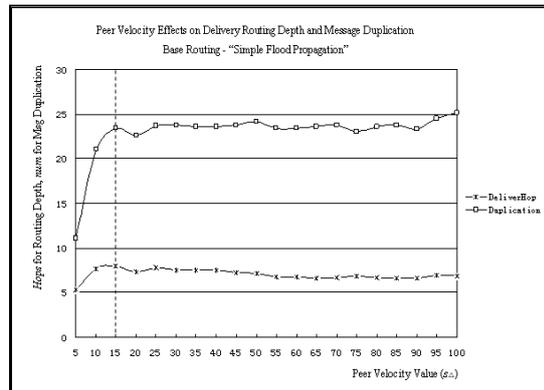


FIGURE 6-3-D: SIMPLE FLOOD PROPAGATION

The graphs above show that peer velocity has very different effects over different base protocols on both message delivery routing depth and message duplication time in the selected simulation configuration. For “Custody Transfer” routing (see Figure 6-3-A), the velocity opportunistic element has little impact on both routing depth and message duplication, due to its non-duplication nature. For “Custody Transfer Propagation” (see Figure 6-3-B) and “Custody Retaining Propagation” (see Figure 6-3-C) routing protocols, increasing peer velocity has a positive effect on routing depth but a negative effect on message duplication within the range $v = [5, 65]$. For “Simple

Flood” routing (see *Figure 6-3-D*), after velocity reaches $v = 15$, it has no significant impact on either routing depth or message duplication.

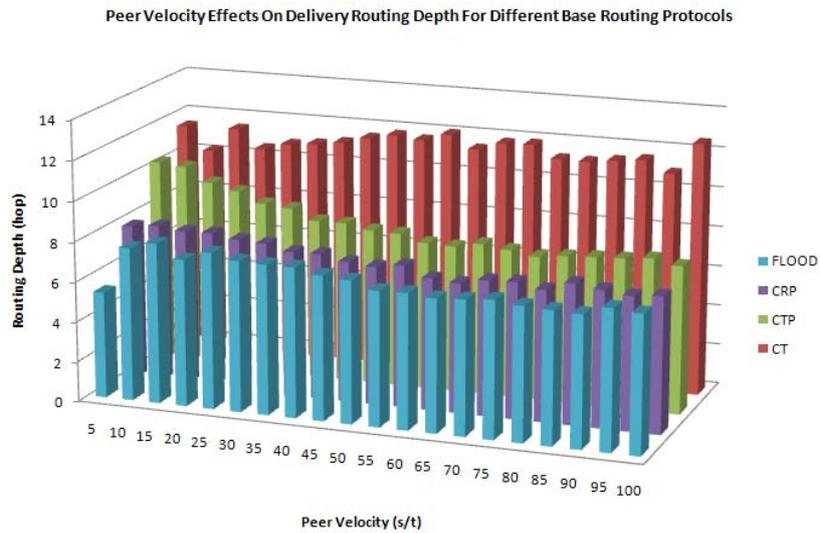


FIGURE 6-4: PEER VELOCITY EFFECTS ON DELIVERY ROUTING DEPTH

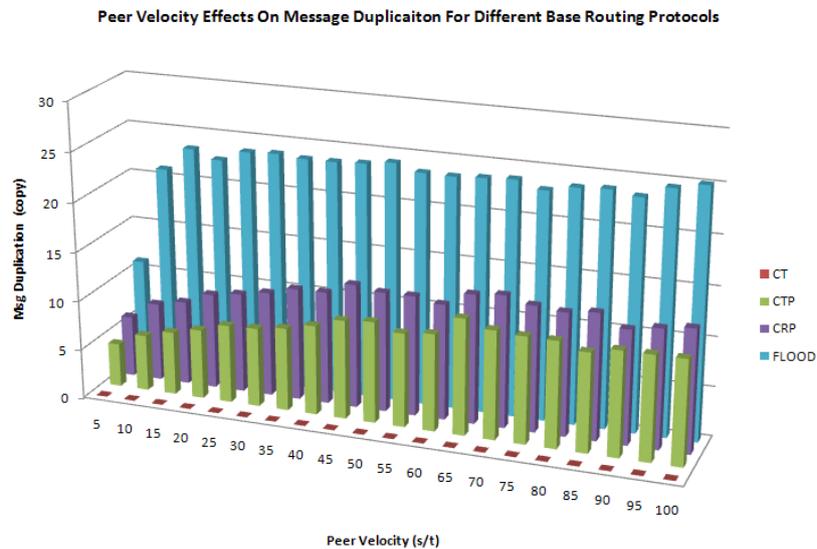


FIGURE 6-5: PEER VELOCITY EFFECTS ON DELIVERY MESSAGE DUPLICATION

From the observation of *Figure 6-2*, *Figure 6-4* and *Figure 6-5*, a conclusion can be made for this particular configuration that the velocity opportunistic element has a significant and uniformly positive impact on routing performance (delivery time). However, it has very little impact on routing efficiency (depth, duplication) for “Custody Transfer” and “Simple Flood” base routing. Therefore, the peer velocity

element is proven a suitable choice to compose a performance emphasized OCI index for this simulation scenario. This result is scenario specific and another simulation of a different scenario is tested which shows that in the scenario with extreme high peer density, the velocity element has absolutely no effect on peer delivery time (see *Appendix F*).

6.1.2 “Peer movement territory” and its effects on message delivery

In a DTMANET environment, peers tend to move within a limited area which may or may not overlap with other peers’ movement territories. This section includes simulation results of the impact of peers’ movement territory on message delivery rate, delivery time, routing depth, and message duplication. The results are over four base OCI routing protocols. The value range of this test is within [50, 500], where 500 is the simulation area boundary, and therefore the largest possible peer territory.

6.1.2.1 Effects on message delivery rate

Figure 6-6-A to *Figure 6-6-D* show the curvatures of delivery rate over different territory settings for four different base non-opportunistic routing protocols.

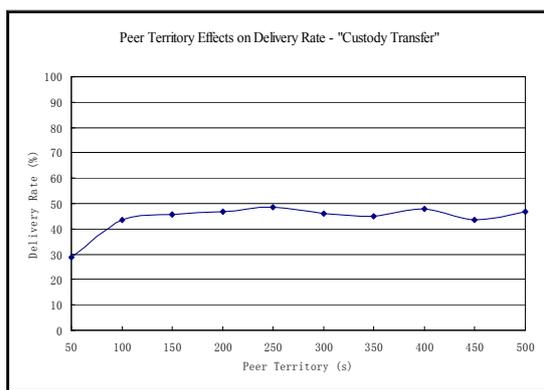


FIGURE 6-6-A: CUSTODY TRANSFER

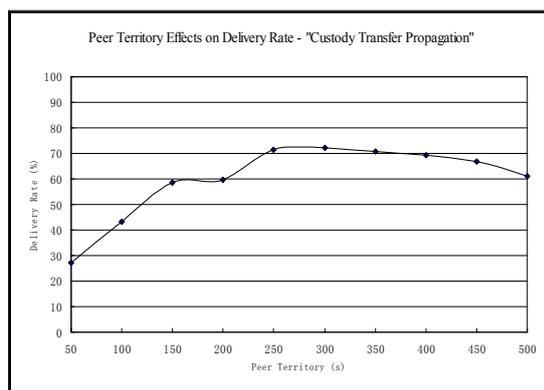


FIGURE 6-6-B: CUSTODY TRANSFER PROPAGATION

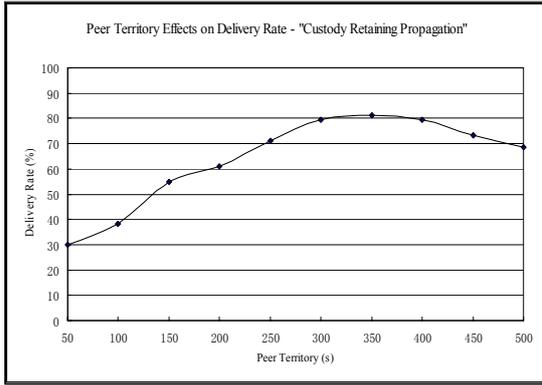


FIGURE 6-6-C: CUSTODY RETAINING PROPAGATION

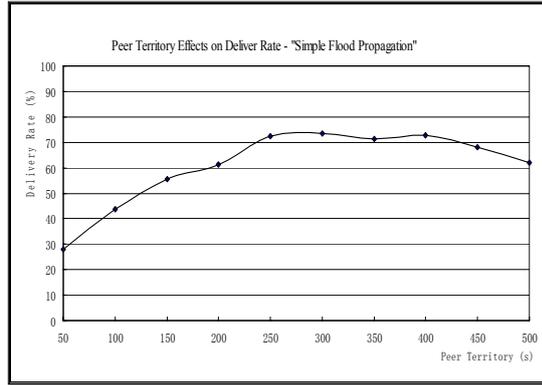


FIGURE 6-6-D: SIMPLE FLOOD PROPAGATION

The *Figure 6-6* shows that the increment of the peer movement territory element has significant positive impact on routing reliability. Its influence reaches a peak at the territory (**T**) value between $T = [300, 350]$ for the simulation configuration for all base protocols. Further territorial increment of peers does not improve the delivery rate. Instead, it is shown to have slightly negative effects.

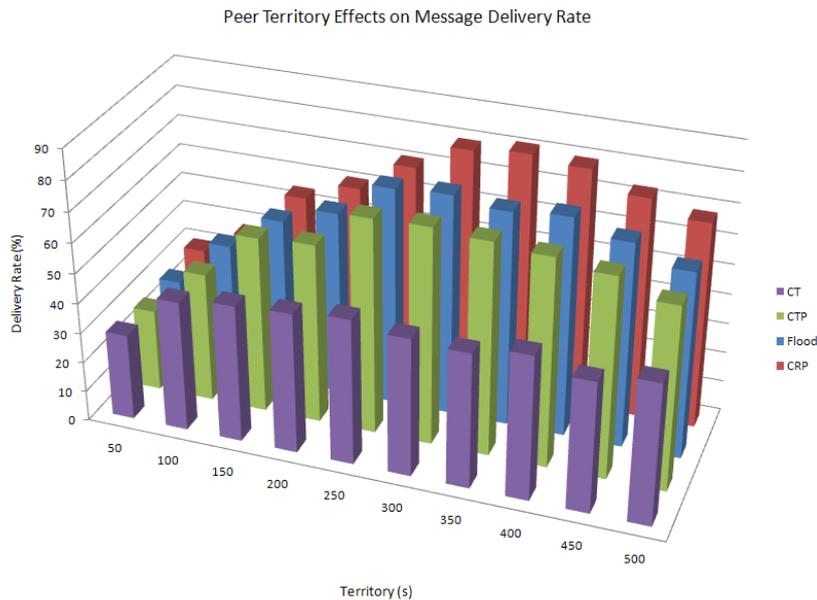


FIGURE 6-7: PEER TERRITORY EFFECTS ON MESSAGE DELIVERY RATE

Figure 6-7 shows a clear uniformly positive impact of peer movement territory over message delivery rate (routing reliability) under all four non-opportunistic base routing protocols. Thus, the peer territory element is proven to be an ideal candidate to compose reliability emphasized OCI index for this simulation scenario.

6.1.2.2 Effects on message delivery time

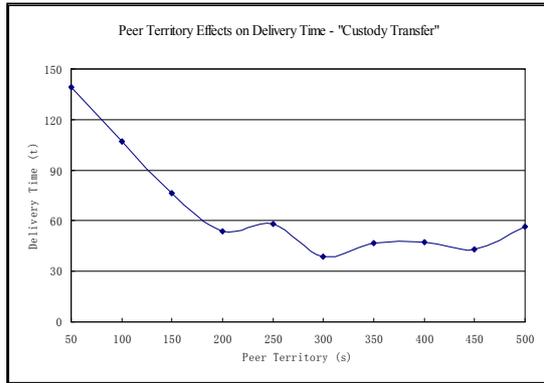


FIGURE 6-8-A: CUSTODY TRANSFER

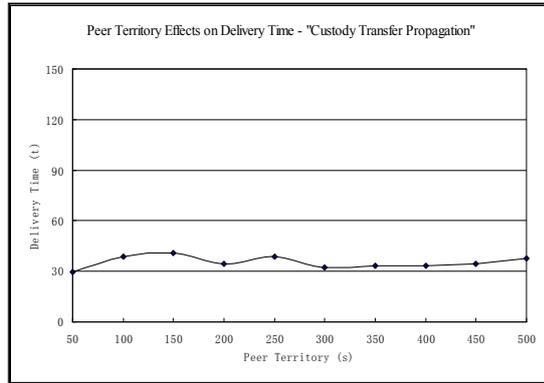


FIGURE 6-8-B: CUSTODY TRANSFER PROPAGATION

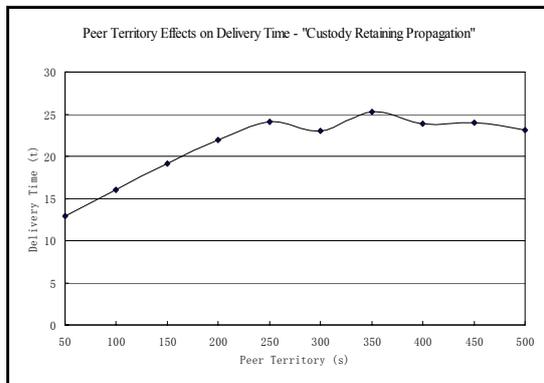


FIGURE 6-8-C: CUSTODY RETAINING PROPAGATION

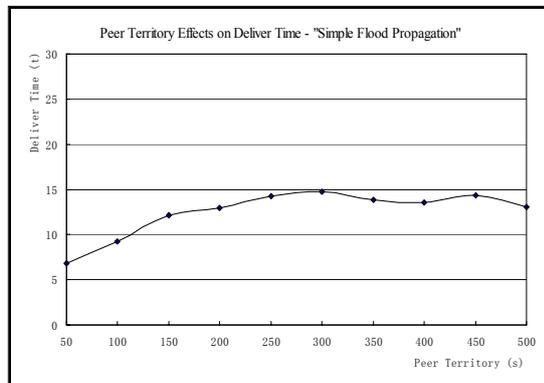


FIGURE 6-8-D: SIMPLE FLOOD PROPAGATION

From *Figure 6-8*, peer territory is shown to dramatically reduce delivery time for “Custody Transfer” routing (*Figure 6-8-A*), while having very little impact for “Custody Transfer Propagation” routing. For “Custody Retaining Propagation” and “Simple Flood Propagation” routing, territory increment increases message delivery time. It is evident that the peak value is in the interval $T = [300, 350]$.

Peer Territory Effects on Message Delivery Time

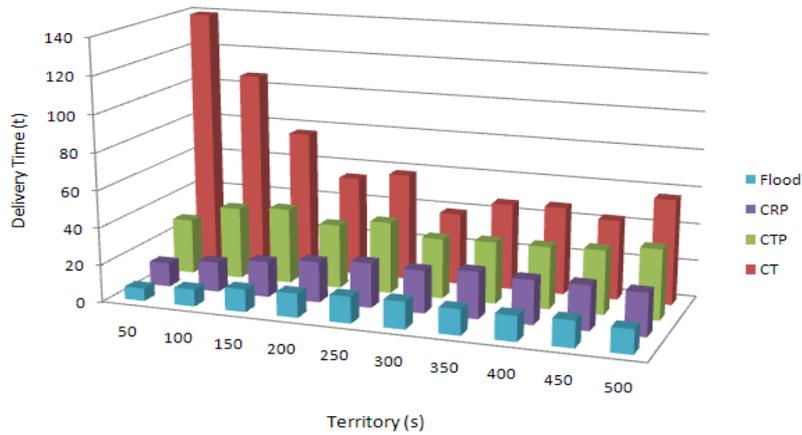


FIGURE 6-9: PEER TERRITORY EFFECTS ON MESSAGE DELIVERY TIME

From *Figure 6-9*, it is observed that the territory element yields very different effects over different base routings, and the most significant positive performance improvement is observed with “Custody Transfer” base routing.

6.1.2.3 Effects on message routing depth



FIGURE 6-10-A: CUSTODY TRANSFER

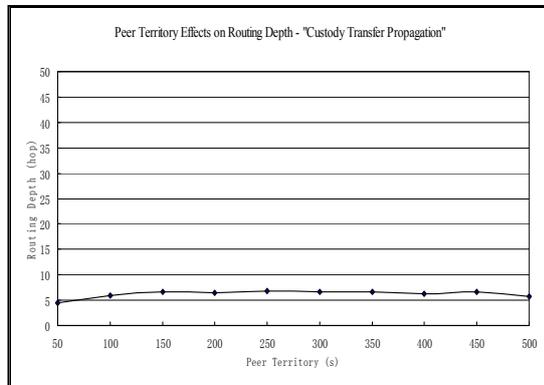


FIGURE 6-10-B: CUSTODY TRANSFER PROPAGATION

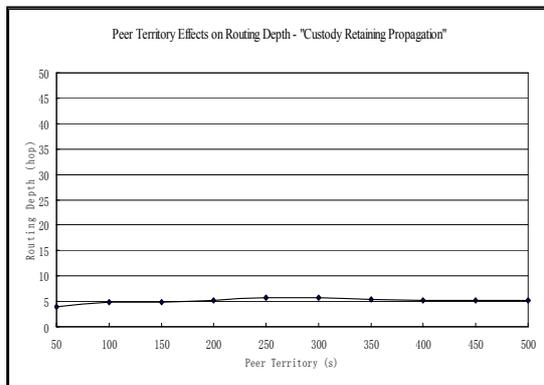


FIGURE 6-10-C: CUSTODY RETAINING PROPAGATION

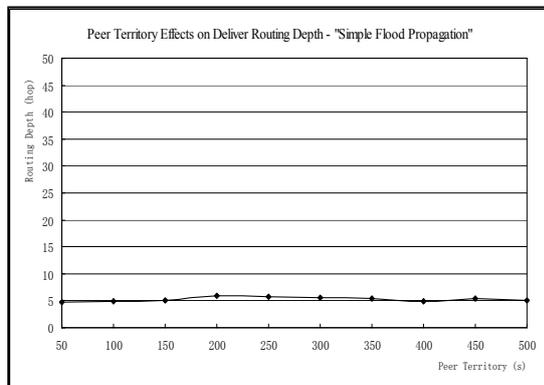


FIGURE 6-10-D: SIMPLE FLOOD PROPAGATION

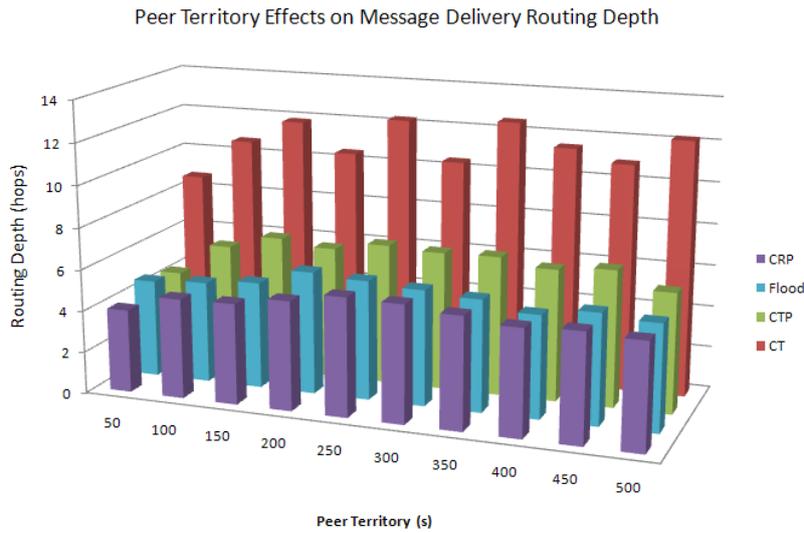


FIGURE 6-11: PEER TERRITORY EFFECT ON MESSAGE ROUTING DEPTH

Figure 6-10_A to Figure 6-10_D, and Figure 6-11 illustrate that peer territory elements has no significant effects over routing depth for all four base protocols.

6.1.2.4 Effects on message duplication

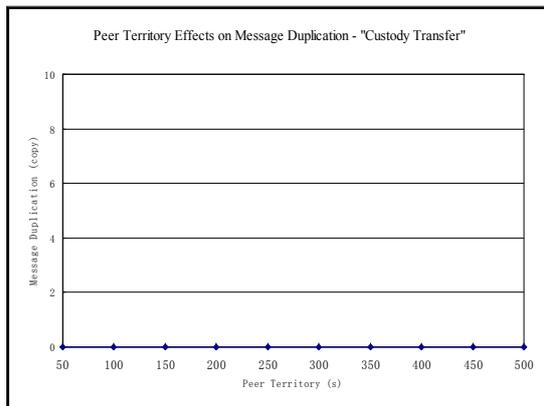


FIGURE 6-12-A: CUSTODY TRANSFER

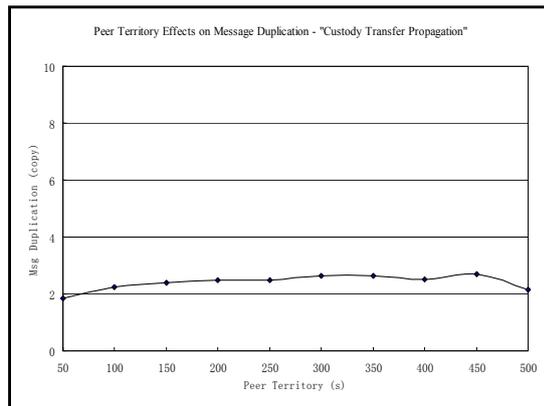


FIGURE 6-12-B: CUSTODY TRANSFER PROPAGATION

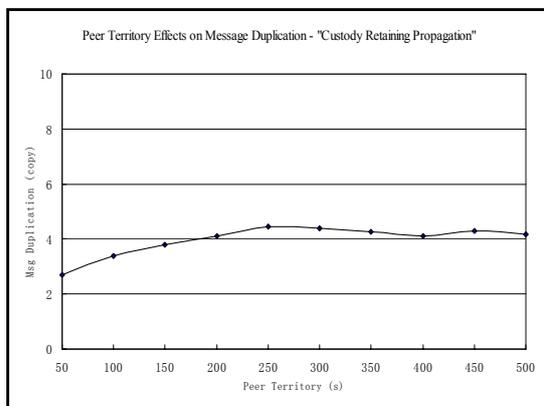


FIGURE 6-12-C: CUSTODY RETAINING PROPAGATION

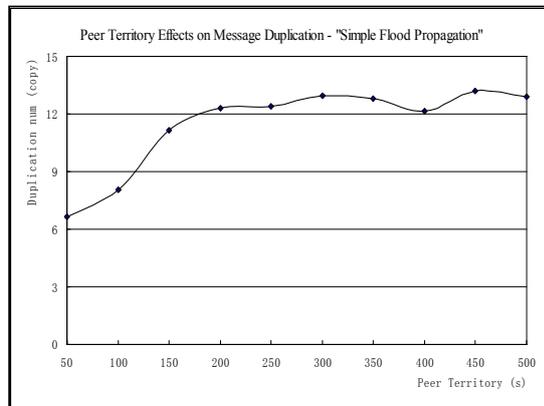


FIGURE 6-12-D: SIMPLE FLOOD PROPAGATION

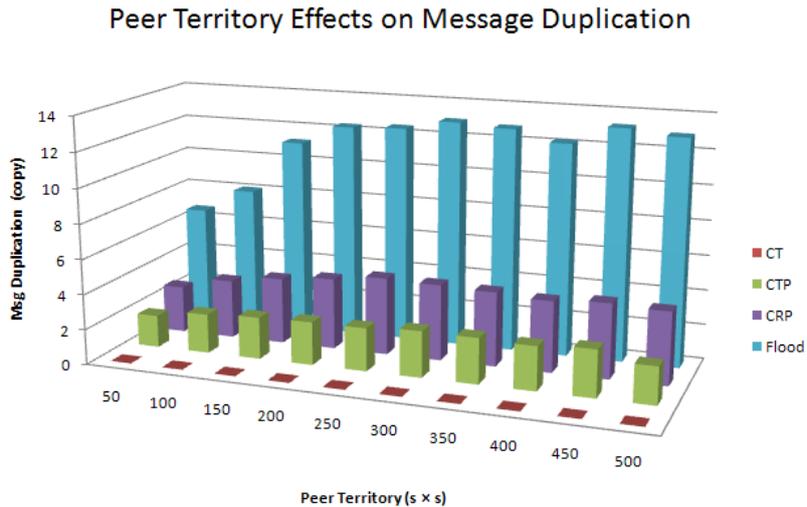


FIGURE 6-13: CUSTODY RETAINING PROPAGATION

Figure 6-12-A to Figure 6-12-D and Figure 6-13 show that in “Custody Transfer” routing, due to its nature, there is no message duplication allowed. In contrast to this, “Simple Flood Propagation” routing suffers from high efficiency loss with the increment of peer territory. The other two protocols have no significant efficiency loss from peer territory value variations.

6.1.3 Other Elements and Their Effects on message delivery

Other opportunistic elements used in OCI simulations namely, communication range, peer movement consistency, communication protocol support variety, and peer reputation are also simulated and analyzed through the same procedure as used in previous sections. See *Appendix D* for complete results and graphs of these elements.

6.2 Opportunistic Element Importance Index

The simulations conducted in *section 6.1* provide data that can be used to decide an opportunistic element’s usefulness if adopted as one of the OCI opportunistic filters. The greater an opportunistic element’s impact is on a DTMANET routing process in one or more aspects of reliability, performance, and efficiency, the more suitable it is to be used as an opportunistic routing protocol. The procedure is particularly useful

when a large number of opportunistic elements have to be tested and selected. See *Appendix E* for a complete set of lists of the importance indices.

6.3 Modelling Opportunistic Elements as Individual Filters

After selecting certain opportunistic elements that result in significant impact on the testing of a DTMANET scenario according to the results generated through the previous simulation steps (see *section 6.1* and *section 6.2*), selected opportunistic elements are individually modelled and tested as opportunistic routing filters. The result of this procedure provides a weight matrix that guides the construction of an OCI index, which enables the opportunistic routing filters to work collaboratively and effectively, with emphasis on balance. Like all the other simulations, this test is conducted using all four base non-opportunistic routing protocols.

6.3.1 Filters and Their Effects on Message Delivery

In this section, six opportunistic filters are tested and compared with each other and with a control configuration, where no opportunistic filter is applied.

6.3.1.1 Applying Opportunistic Filters to “Custody Transfer”

Firstly, the opportunistic filters are tested using “Custody Transfer” base routing, which is one-to-one based, and where the ownership of the message is relayed through peer encounter. Therefore, the message is removed from the sender or previous owner. At each given time, only one copy of each different message is allowed in the system.

- 1) Filter effects on message delivery rate

The impact of routing reliability (delivery rate at a given TTL) of each opportunistic filter is tested and shown as *Figure 6-14*.

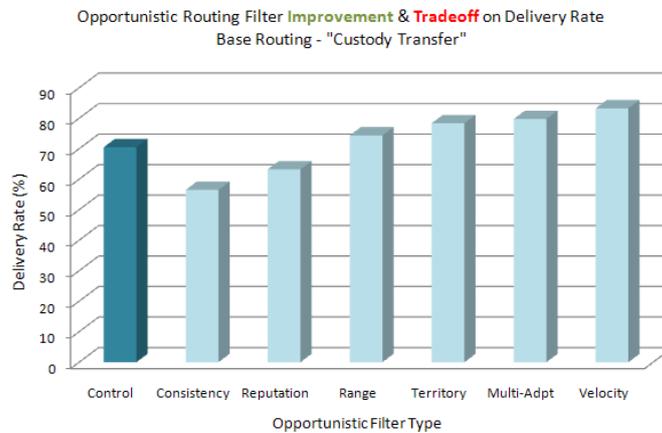


FIGURE 6-14: DELIVERY RATE OVER "CUSTODY TRANSFER"

Figure 6-14 shows that, by applying different opportunistic filters, within a given TTL value, there can be different results in terms of delivery rate. Both improvements and tradeoffs can be observed. The most significant improvement is demonstrated by the velocity filter, which improves upon the control configuration by 18.09%. The biggest tradeoff is brought about through the peer movement consistency filter, where the delivery rate decreased by 19.86%.

2) Filter effects on message delivery time

The impact of routing performance (average end-to-end time of each successfully delivered message) of each opportunistic filter is illustrated as follows:

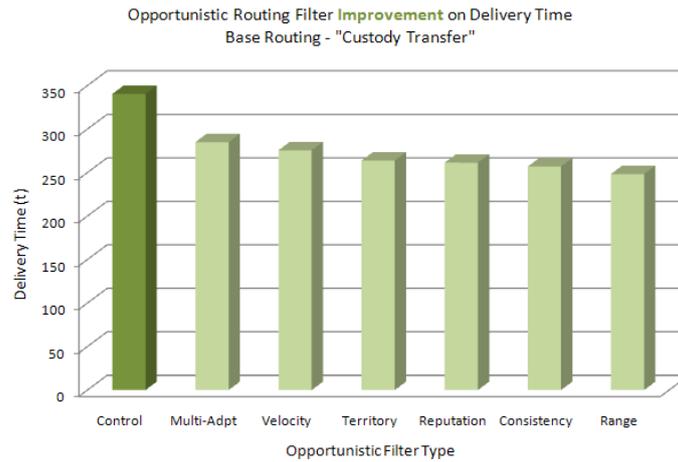


FIGURE 6-15: DELIVERY TIME OVER “CUSTODY TRANSFER”

Figure 6-15 clearly confirms that every opportunistic filter reduces message delivery time and therefore increases the performance of this simulation scenario. The most significant improvement is observed from the communication range opportunistic filter with an improvement of 27.05%, compared to the control result.

3) Filter effects on message delivery routing depth

Each filter and control configuration is compared over the value of message delivery routing depth (average hop count for each successfully delivered message). The routing depth in a DTMANET environment and in OCI simulations is one of the factors used to determine routing efficiency.

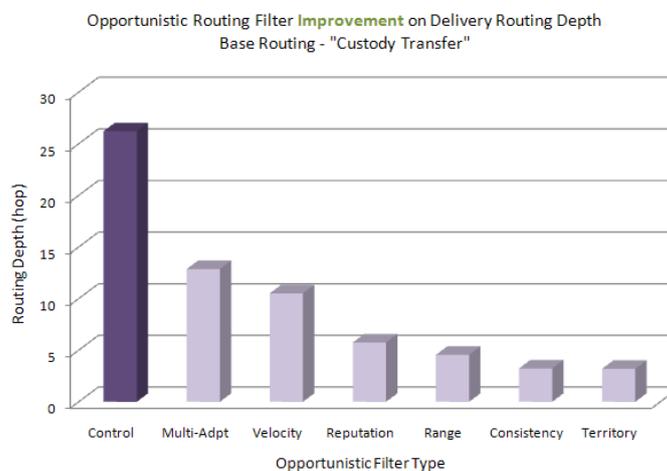


FIGURE 6-16: DELIVERY ROUTING DEPTH OVER “CUSTODY TRANSFER”

Figure 6-16 demonstrates significant improvements on routing depth, over “Custody Transfer” routing, for all tested opportunistic filters. The most improvement is observed from the peer movement territory opportunistic filter, which gives an 87.80% improvement over the control configuration.

4) Filter effects on message delivery duplication

Since the “Custody Transfer” routing protocol allows no message duplication, there can be no improvement to be gained or tradeoffs to be made through applying opportunistic filters.

5) Overall filter effects on “Custody Transfer”

From the demonstration in *Figure 6-14* to *Figure 6-16*, it is concluded that applying the opportunistic filters on “Custody Transfer” base protocol, for the testing scenario, results in a significant overall improvement on all the aspects of the routing quality metrics, reliability, performance, and efficiency.

6.3.1.2 Applying Filters to “Custody Transfer Propagation”

In this section, opportunistic filters are tested on “Custody Transfer Propagation” routing. The “Custody Transfer Propagation” routing protocol is a combination of the custody transfer and epidemic routing protocol. Under “Custody Transfer Propagation”, messages are broadcasted and duplicated to all peers that are in the transmission range of the message-carrying peers and messages are removed from the previous owner after each successful relay.

1) Filter effects on message delivery rate

The following graph illustrates the effect of each opportunistic filter and the control configuration regarding to the message delivery rate.

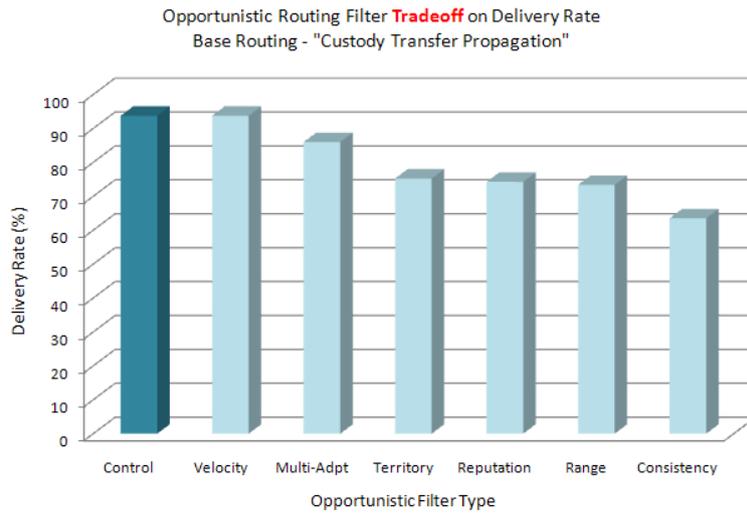


FIGURE 6-17: DELIVERY RATE OVER “CUSTODY TRANSFER PROPAGATION”

From *Figure 6-17* it is evident that under the “Custody Transfer Propagation” base routing protocol, routing reliability is reduced through all opportunistic filters except the velocity filter, which maintains the same delivery rate as the control configuration. The worst-case configuration is the consistency filter, where delivery rate decreased by 32.27%.

2) Filter effects on message delivery time

The following graph presents the comparison of each filter with respect to each configuration’s delivery time.

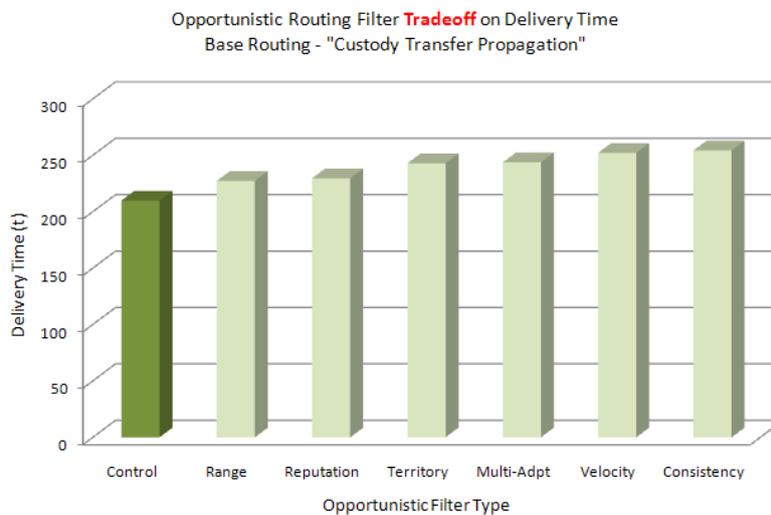


FIGURE 6-18: DELIVERY TIME OVER “CUSTODY TRANSFER PROPAGATION”

Figure 6-18 shows that, under “Custody Transfer Propagation”, routing performance is noticeably reduced for all opportunistic filters. The filter with the least loss in performance is the communication range filter, where the delivery time is increased by 8.32%. The consistency filter results in the highest performance loss, where delivery time is 21.17% longer than the control configuration.

3) Filter effects on message delivery routing depth

The following graph lists the overall comparison of each opportunistic routing filter and the control configuration in terms of message delivery routing depth, which is used to evaluate routing efficiency.

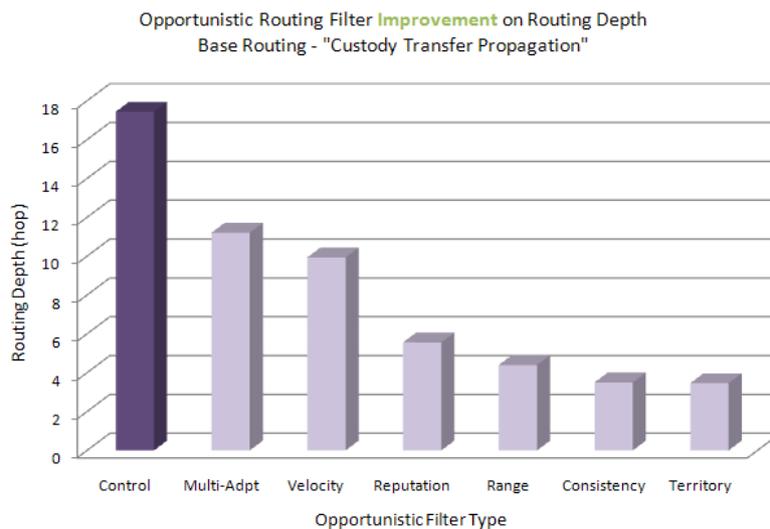


FIGURE 6-19: DELIVERY ROUTING DEPTH OVER “CUSTODY TRANSFER PROPAGATION”

Figure 6-19 presents a drastic improvement on message delivery routing depth through applying individual filters. The most gain comes from the peer territory opportunistic filter, which results in 80.14% reduction of routing depth compared with the control configuration.

4) Filter effects on message delivery duplication

The graph below is a direct comparison of delivery message duplication for the opportunistic routing filters and the control configuration, which is another important benchmark for the evaluation of routing efficiency.

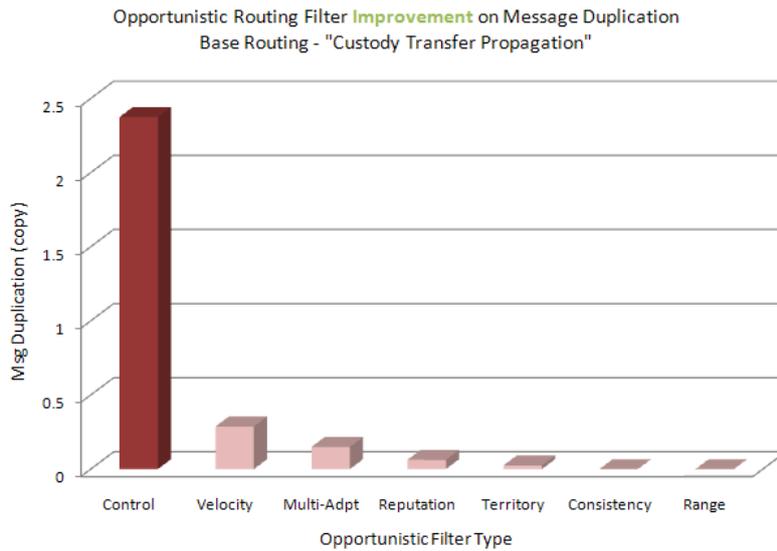


FIGURE 6-20: MESSAGE DUPLICATION OVER "CUSTODY TRANSFER PROPAGATION"

From *Figure 6-20* it is observed that each of the opportunistic filters results in a dramatic improvement in message duplication numbers. When the communication range filter or consistency filter is applied, they eliminate message duplication, which means 100% improvement over the control configuration for this simulation scenario.

One might argue that the control configuration only has an average of 2.4 message duplications, so even when there are filters that provide 100% improvement, it will only be 2.4 messages less, and thus it is hardly significant. However, consider the following scenario: a DTMANET of 100 peers, each peer sends a message to one other peer, which gives 100 messages in the system. In order to deliver the 100 messages, 240 extra messages have to be created as duplications. Therefore, there will be total of 340 messages in the system. Among the 340 messages, only 100 of them are actually payload, which means that over

70% of peers' storage space and over 70% of the network bandwidth are wasted, so, the network can only work in less than 30% efficiency. If the message duplication is reduced to 0, the network can theoretically work at 100% efficiency, and that is a significant improvement.

5) Overall filter effects on "Custody Transfer Propagation"

From the observations and explanations of *Figure 6-17* to *Figure 6-20*, a conclusion can therefore be drawn that by applying opportunistic filters to "Custody Transfer Propagation" base routing, significant improvements are to be made on routing efficiency, while slight tradeoffs are to be expected on routing reliability and performance.

6.3.1.3 Applying Filters to "Custody Retaining Propagation"

Another base routing protocol that the opportunistic filters are applied to is the "Custody Retaining Propagation" protocol. The "Custody Retaining Propagation" routing protocol is a propagation based model, however, only the original message owner can duplicate messages. Peers that are not message owners can only hand over the message to the next encountering peer, instead of duplicating the message. After the hand-over the peer removes the message from itself while the owner retains custody of the initial message.

1) Filter effects on message delivery rate

The impact of the routing reliability of each opportunistic filter is tested and compared through the values of message delivery rate, as shown in the following graph.

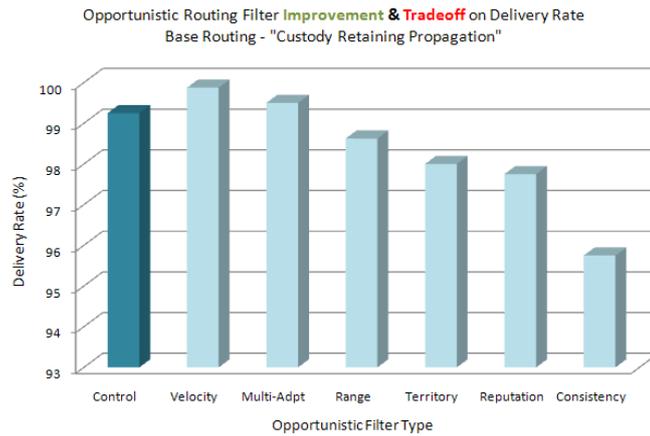


FIGURE 6-21: MESSAGE DELIVERY RATE OVER "CUSTODY RETAINING PROPAGATION"

Figure 6-21 shows that through applying opportunistic filters, slight gain and tradeoffs are possible, depending on the choice of filter. While the peer velocity filter gives 0.63% gain, the peer movement consistency filter loses 3.53%, when compared with the control configuration for this particular simulation.

2) Filter effects on message delivery time

In the following graph, the effect on routing performance of each opportunistic filter is tested and compared with respect to message delivery time.

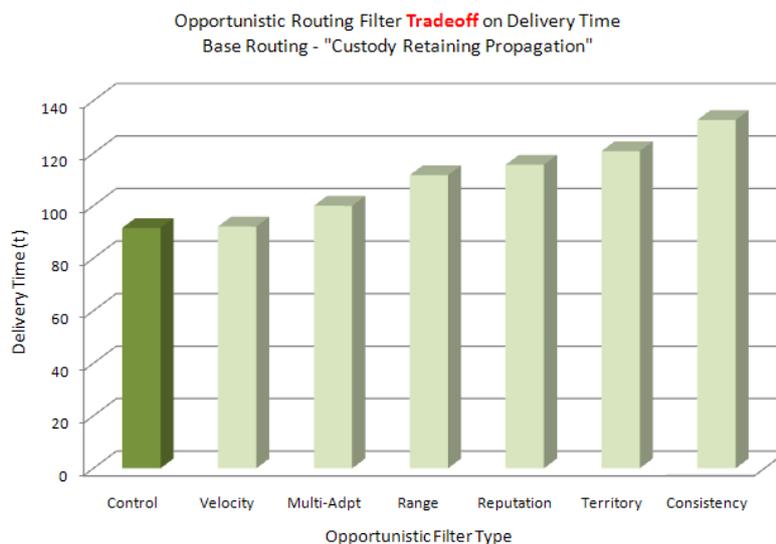


FIGURE 6-22: MESSAGE DELIVERY TIME OVER "CUSTODY RETAINING PROPAGATION"

From Figure 6-22 it is evident that there are noticeable increases in delivery times for each opportunistic filter. The smallest increase of 0.59% is observed from the

peer velocity filter, while the most significant increase is observed from the peer movement consistency filter, where average message delivery time is increased by 44.89%.

3) Filter effects on message delivery routing depth

The results of the effect of the opportunistic filters on the message routing depth compared to the control configuration are shown in the following graph to illustrate the routing efficiency.

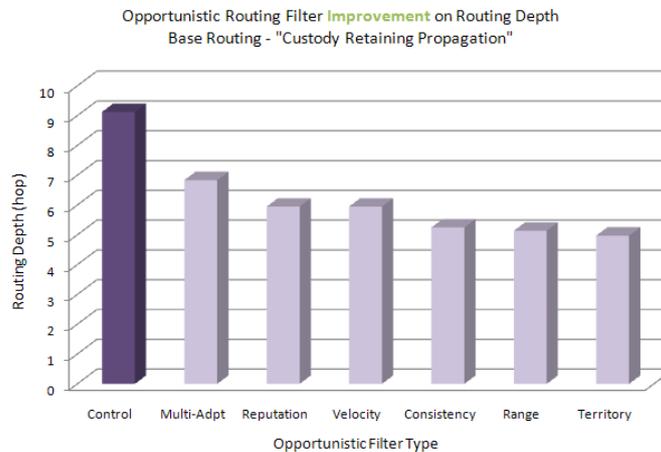


FIGURE 6-23: MESSAGE DELIVERY ROUTING DEPTH OVER “CUSTODY RETAINING PROPAGATION”

Figure 6-23 shows considerable routing depth reduction for all opportunistic filters compared with the control configuration. The multi-adaptor filter offers a 25.07% improvement, while the peer territory filter reduces the routing depth by 45.48%.

4) Filter effects on message delivery duplication

The following graph illustrates the message duplication values observed for each of the opportunistic filters and the control configuration. The message duplication value measures routing efficiency in propagation based routing.

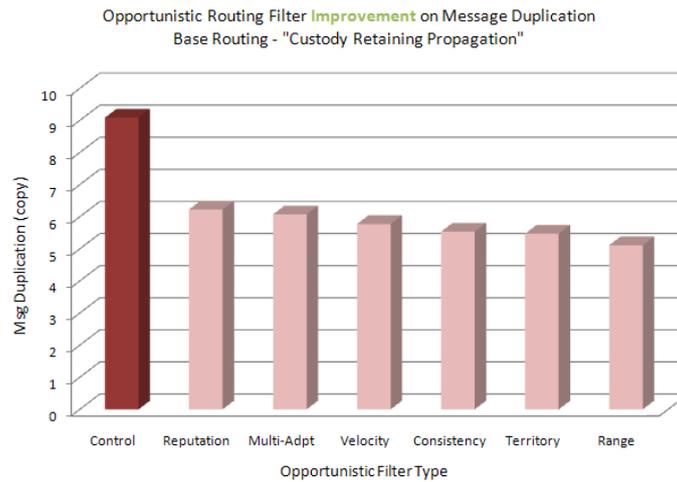


FIGURE 6-24: MESSAGE DUPLICATION OVER "CUSTODY RETAINING PROPAGATION"

Figure 6-24 shows a high level of improvement in routing efficiency for all opportunistic filters in comparison to the control configuration. The peer reputation filter reduces the message duplication by 31.50% while the peer communication range filter provides 43.74% improvement.

5) Overall filter effects under "Custody Retaining Propagation"

A conclusion can be drawn based on the results of Figure 6-21 to Figure 6-24. By applying opportunistic filters to "Custody Transfer Propagation" base routing, considerable gains can be obtained on routing reliability, significant improvements are to be made on routing efficiency, while minor reductions are observed on routing performance.

6.3.1.4 Applying Filters to "Simple Flood Propagation"

The last base routing tested here is the "Simple Flood Propagation" protocol. In the simple flood propagation routing, every peer carrying messages, sender or routers, can duplicate messages to all other in-range peers.

1) Filter effects on message delivery rate

The following graph compares the message delivery rate of each opportunistic filter and the control configuration. Message delivery rate is an indicator of the reliability of the routing process of a given scenario.

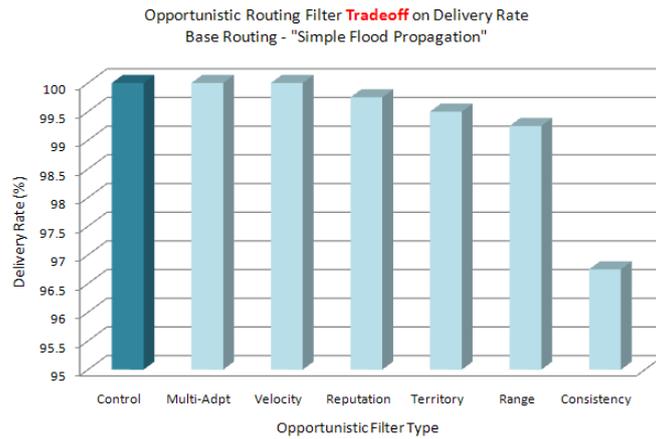


FIGURE 6-25: MESSAGE DELIVERY RATE OVER “SIMPLE FLOOD PROPAGATION”

From *Figure 6-25* it is evident that slight routing reliability reductions are experienced for most of the opportunistic filters. The closest delivery time in comparison to the control configuration is through the Multi-Adaptor opportunistic filter, where the reliability is reduced by 0.25%. The highest loss is observed from the consistency filter and amounts to only 3.53%.

2) Filter effects on message delivery time

The simulation results for message delivery time under “Simple Flood Propagation” base routing for each opportunistic filter and the control configuration are shown in the following graph.

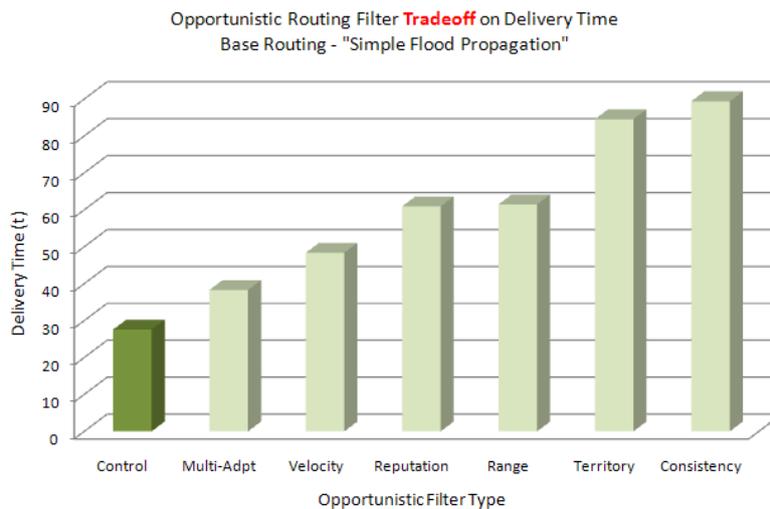


FIGURE 6-26: MESSAGE DELIVERY TIME OVER “SIMPLE FLOOD PROPAGATION”

From *Figure 6-26*, it is observed that there are significant performance compromises after applying some of the opportunistic filters. The least loss is observed by the multi-adaptor opportunistic filter at 38.83%, while the most severe loss of 224.06% is observed from the peer movement consistency filter.

3) Filter effects on message delivery routing depth

The impact on routing efficiency of each opportunistic filter and the control configuration are tested and compared through the values of message delivery routing depth, as shown in the following graph.

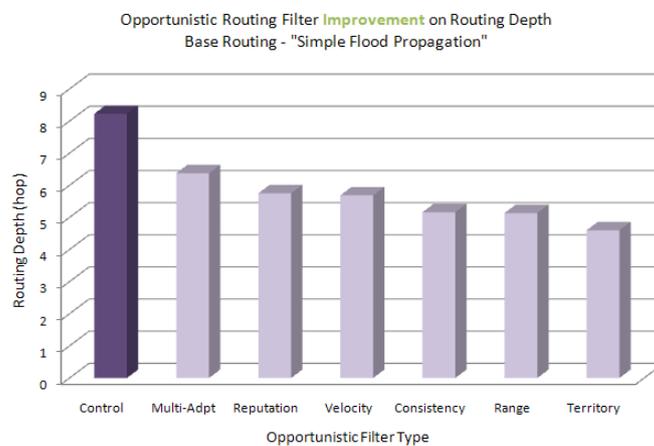


FIGURE 6-27: MESSAGE DELIVERY ROUTING DEPTH OVER “SIMPLE FLOOD PROPAGATION”

Figure 6-27 shows major decrease in routing depth, through each of the applications of the opportunistic filters. The most significant reduction results from the Peer Movement Territory Filter, where the routing depth is reduced by 44.01% and the least is from the Multi-Adaptor filter, with a reduction of 22.49%.

4) Filter effects on message delivery duplication

The following graph demonstrates the improvements made through the use of each opportunistic filter, in comparison to the control configuration, to routing efficiency, which is measured through the average rate of message duplication for each successfully delivered message.

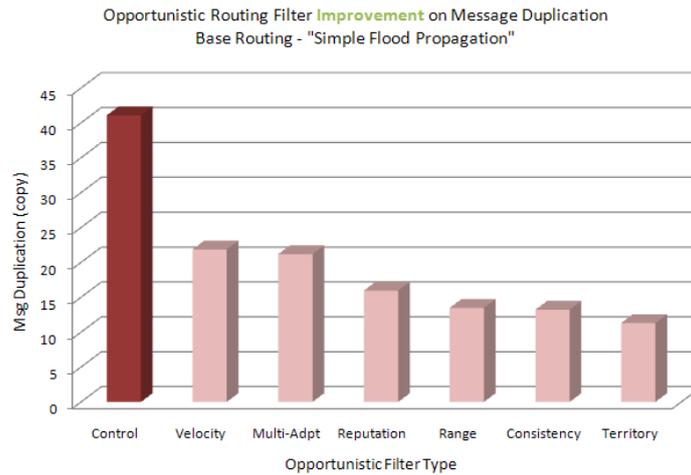


FIGURE 6-28: MESSAGE DUPLICATION OVER "SIMPLE FLOOD PROPAGATION"

Figure 6-28 shows significant routing efficiency improvement for all the opportunistic filters. The most significant improvement comes from Peer Movement Territory Filter, where the message duplication is reduced by 72.47% and the least is from Peer Velocity Filter, which results in a 46.79% message duplication reduction.

5) Overall filter effects under "Simple Flood Propagation"

From the result presented in *Figure 6-26*, one might argue that because the degree of performance loss is too great, to apply opportunistic filter over "Simple Flood Propagation" routing protocol is not practical. However, this can be justified as follows. Firstly, the "Simple Flood Propagation" is an extreme protocol that provides the best possible theoretical performance and at the same time the worst theoretical efficiency, therefore, there is no space for performance improvement. Secondly, even the worst-case delivery time, using the consistency filter, at an average of $89.4 t_{\Delta}$, is faster than any of the fastest delivery times simulated in the other three base protocols, which was $248.33 t_{\Delta}$ for "Custody Transfer", $209.86 t_{\Delta}$ for "Custody Transfer Propagation" and $91.54 t_{\Delta}$ for "Custody Retaining Propagation". Thirdly, the efficiency gain through opportunistic filters are equally great, as shown by the results presented in *Figure 6-27*, which shows that message routing depth dropped from 8.22 hops to 4.60 hops, and presented in

Figure 6-28, which shows that message duplication dropped from 41.09 copies per message to 11.32 copies per message. Therefore, the conclusion can be drawn that when applying opportunistic filters to the “Simple Flood Propagation” base routing model, though compromised performance results, significant routing efficiency improvement can be gained with consistent routing reliability.

6.3.2 OCI Weight Metrics assignment

In *section 3.6.4*, methods and techniques used for weight assignment and voting procedures are described, which include both non-intelligent methods and intelligent methods. In this section, only the intelligent weight assignment method is described due to the length restriction of this thesis.

The comparative evaluation results from the previous section show convincing evidence that for each base routing protocol of a given simulation scenario, behaviours and impacts of different opportunistic routing filters vary drastically. Therefore, it is asserted that only after having specified the context of the base routing protocol will an OCI index be meaningful and useful for the purpose of routing improvement.

6.3.2.1 OCI Index Weight Assignment for “Custody Transfer” Base Protocol

Table 6-1 lists the message delivery rate, message delivery time, message delivery routing depth and message duplication of the simulations of different opportunistic routing and the control configuration, under “Custody Transfer” base routing. Such data are gathered from individual opportunistic routing protocol tests.

"CUSTODY TRANSFER" SIMULATION RESULTS				
	Delivery Rate	Delivery Time	Routing Depth	Duplication
Control	70.50%	340.41	26.23	0.00
Consistency	56.50%	256.96	3.24	0.00
Multi-Adaptor	79.75%	285.01	12.86	0.00
Range	74.38%	248.33	4.55	0.00
Reputation	63.25%	261.43	5.75	0.00
Territory	78.38%	263.84	3.20	0.00
Velocity	83.25%	275.64	10.51	0.00

TABLE 6-1: "CUSTODY TRANSFER" SIMULATION RESULTS

Table 6-2 shows the percentage of improvement or tradeoffs, calculated from the results listed in Table 6-1, of message delivery rate, message delivery time, message delivery routing depth and message duplication, in comparison to the control configuration.

IMPROVEMENT / TRADEOFFS - "CUSTODY TRANSFER"				
	Delivery Rate	Delivery Time	Routing Depth	Duplication
Control	0.00%	0.00%	0.00%	0.00%
Consistency	-19.86%	24.51%	87.65%	0.00%
Multi-Adaptor	13.12%	16.27%	50.95%	0.00%
Range	5.50%	27.05%	82.65%	0.00%
Reputation	-10.28%	23.20%	78.07%	0.00%
Territory	11.17%	22.49%	87.80%	0.00%
Velocity	18.09%	19.03%	59.91%	0.00%

TABLE 6-2: IMPROVEMENT / TRADEOFFS "CUSTODY TRANSFER"

As shown in Table 6-3, absolute OCI indices are composed with sole consideration of only one aspect of either routing reliability, or routing performance or routing efficiency. For instance, a reliability oriented OCI index will only use the simulation result for delivery rate to determine the OCI index order and to assign weight to each participating opportunistic filter. Other data such as delivery time, routing hops and message duplication are not considered.

ABSOLUTE OCI INDEX - "CUSTODY TRANSFER"

Absolute Reliability		Absolute Performance		Absolute Efficiency	
Weight	OCI	Weight	OCI	Weight	OCI
18.09%	Velocity	27.05%	Range	87.80%	Territory
13.12%	Multi-Adaptor	24.51%	Consistency	87.65%	Consistency
11.17%	Territory	23.20%	Reputation	82.65%	Range
5.50%	Range	22.49%	Territory	78.07%	Reputation
-10.28%	Reputation	19.03%	Velocity	59.91%	Velocity
-19.86%	Consistency	16.27%	Multi-Adaptor	50.95%	Multi-Adaptor

TABLE 6-3: ABSOLUTE OCI INDEX - “CUSTODY TRANSFER”

The weight metrics (filter importance order based) of absolute orientations are listed as follows (see *section 3.6.4.1* for its voting procedure):

$$\bar{M}_{AR} = \begin{pmatrix} \text{Velocity} & 6 \\ \text{Multi-Adaptor} & 5 \\ \text{Territory} & 4 \\ \text{Range} & 3 \\ \text{Reputation} & 2 \\ \text{Consistency} & 1 \end{pmatrix} \quad \bar{M}_{AP} = \begin{pmatrix} \text{Range} & 6 \\ \text{Consistency} & 5 \\ \text{Reputation} & 4 \\ \text{Territory} & 3 \\ \text{Velocity} & 2 \\ \text{Multi-Adaptor} & 1 \end{pmatrix}$$

$$\bar{M}_{AE} = \begin{pmatrix} \text{Territory} & 6 \\ \text{Consistency} & 5 \\ \text{Range} & 4 \\ \text{Reputation} & 3 \\ \text{Velocity} & 2 \\ \text{Multi-Adaptor} & 1 \end{pmatrix}$$

The weight metrics (filter improvement value based) of absolute orientations are listed as follows (see *section 3.6.4.2* for its voting procedure):

$$\bar{M}_{AR} = \begin{pmatrix} \text{Velocity} & 18.09 \\ \text{Multi-Adaptor} & 13.12 \\ \text{Territory} & 11.17 \\ \text{Range} & 5.50 \\ \text{Reputation} & -10.28 \\ \text{Consistency} & -19.86 \end{pmatrix} \quad \bar{M}_{AP} = \begin{pmatrix} \text{Range} & 27.05 \\ \text{Consistency} & 24.51 \\ \text{Reputation} & 23.20 \\ \text{Territory} & 22.49 \\ \text{Velocity} & 19.03 \\ \text{Multi-Adaptor} & 16.27 \end{pmatrix}$$

$$\bar{M}_{AE} = \begin{pmatrix} \text{Territory} & 87.80 \\ \text{Consistency} & 87.65 \\ \text{Range} & 82.65 \\ \text{Reputation} & 78.07 \\ \text{Velocity} & 59.91 \\ \text{Multi-Adaptor} & 50.95 \end{pmatrix}$$

Emphasized balanced OCI indices are composed with emphasis on one or more of the three routing aspects of reliability, performance and efficiency. Other aspects are also considered in the routing decision making process. To achieve the balance, this simulation adopts parameters called “emphasis multipliers”, For instance, as shown in *Table 6-4*, a reliability emphasized balance OCI index is generated by assigning 2 as

the emphasis multiplier, and the rest with 1. Using the territory filter as an example, the reliability improvement weight is $11.17 \times 2 = 22.34$, the performance improvement weight is $22.49 \times 1 = 22.49$, the efficiency improvement weight is $87.80 \times 1 = 87.80$, and therefore the overall weight for the territory filter using reliability emphasized balancing weight assignment is $22.34 + 22.49 + 87.80 = 132.63$. The “emphasis multiplier” can be assigned different values to suit the needs of different requirements for different simulations where one or more routing quality metrics can be emphasised and take priority over others.

EMPHASIZED BALANCING INDEX - “CUSTODY TRANSFER”					
Reliability Emphasized Balancing		Performance Emphasized Balancing		Efficiency Emphasized Balancing	
Weight	OCI	Weight	OCI	Weight	OCI
132.63%	Territory	143.96%	Territory	209.26%	Territory
120.69%	Range	142.25%	Range	197.85%	Range
115.11%	Velocity	116.83%	Consistency	179.97%	Consistency
93.47%	Multi-Adaptor	116.06%	Velocity	169.07%	Reputation
80.71%	Reputation	114.20%	Reputation	156.94%	Velocity
72.45%	Consistency	96.62%	Multi-Adaptor	131.30%	Multi-Adaptor

TABLE 6-4: EMPHASIZED BALANCING OCI INDEX - “CUSTODY TRANSFER”

The weight metrics (filter importance order based) of emphasized balancing are listed as follows:

$$\bar{M}_{BR} = \begin{pmatrix} \text{Territory} & 6 \\ \text{Range} & 5 \\ \text{Velocity} & 4 \\ \text{Multi-Adaptor} & 3 \\ \text{Reputation} & 2 \\ \text{Consistency} & 1 \end{pmatrix} \quad \bar{M}_{BP} = \begin{pmatrix} \text{Territory} & 6 \\ \text{Range} & 5 \\ \text{Consistency} & 4 \\ \text{Velocity} & 3 \\ \text{Reputation} & 2 \\ \text{Multi-Adaptor} & 1 \end{pmatrix}$$

$$\bar{M}_{BE} = \begin{pmatrix} \text{Territory} & 6 \\ \text{Range} & 5 \\ \text{Consistency} & 4 \\ \text{Reputation} & 3 \\ \text{Velocity} & 2 \\ \text{Multi-Adaptor} & 1 \end{pmatrix}$$

The weight metrics (filter improvement value based) of emphasized balancing are

listed as follows:

$$\bar{M}_{BR} = \begin{pmatrix} \text{Territory} & 132.63 \\ \text{Range} & 120.69 \\ \text{Velocity} & 115.11 \\ \text{Multi-Adaptor} & 93.47 \\ \text{Reputation} & 80.71 \\ \text{Consistency} & 72.45 \end{pmatrix} \quad \bar{M}_{BP} = \begin{pmatrix} \text{Territory} & 143.96 \\ \text{Range} & 142.25 \\ \text{Consistency} & 116.83 \\ \text{Velocity} & 116.06 \\ \text{Reputation} & 114.20 \\ \text{Multi-Adaptor} & 96.62 \end{pmatrix}$$

$$\bar{M}_{BE} = \begin{pmatrix} \text{Territory} & 209.26 \\ \text{Range} & 197.85 \\ \text{Consistency} & 179.97 \\ \text{Reputation} & 169.07 \\ \text{Velocity} & 156.94 \\ \text{Multi-Adaptor} & 131.30 \end{pmatrix}$$

A balanced OCI index can be considered as an emphasized balancing OCI index with equal emphasis, therefore an equal “emphasis multiplier”, on all three aspects of routing reliability, routing performance and routing efficiency.

BALANCED INDEX - “CUSTODY TRANSFER”	
Weight	OCI
121.46%	Territory
115.20%	Range
97.03%	Velocity
92.31%	Consistency
90.99%	Reputation
80.35%	Multi-Adaptor

TABLE 6-5: BALANCED OCI INDEX - “CUSTODY TRANSFER”

The weight metrics (filter importance order based) and (filter improvement value based) of balancing are listed as follows:

$$\bar{M}_{\text{Balanced}} = \begin{pmatrix} \text{Territory} & 6 \\ \text{Range} & 5 \\ \text{Velocity} & 4 \\ \text{Consistency} & 3 \\ \text{Reputation} & 2 \\ \text{Multi-Adaptor} & 1 \end{pmatrix} \quad \bar{M}_{\text{Balanced}} = \begin{pmatrix} \text{Territory} & 121.46 \\ \text{Range} & 115.20 \\ \text{Velocity} & 97.03 \\ \text{Consistency} & 92.31 \\ \text{Reputation} & 90.99 \\ \text{Multi-Adaptor} & 80.35 \end{pmatrix}$$

6.3.2.2 OCI Index Weight Assignment for Other Base Protocols

Other base protocols, “Custody Transfer Propagation”, “Custody Retaining Propagation”, and “Simple Flood Propagation” are all assigned values through the same procedure as described above. Refer to *Appendix C* for a complete list.

6.4 Testing OCI indexes

In the previous section, all selected filters are tested individually as a single opportunistic routing protocol over all four non-opportunistic base routing protocols. Weights have been assigned to each of these opportunistic filters through absolute, emphasized balancing and overall-balanced intelligent weight assignment methods. Therefore, OCI indexes for the simulation scenario of each emphasis orientation have been established.

The rest of this section is dedicated to a series of tests of OCI indexes (balance assigned weights) through the comparison of the control configuration with the individual opportunistic filters. Each provides the best improvement or the least tradeoffs on one of the three measurements namely, the reliability, the performance and the efficiency. In addition, another simulation that uses multiple opportunistic filters in a non-OCI fashion (randomly choose filter order and weight), is also tested.

6.4.1 Test OCI Index over “Custody Transfer” Base Routing

In the same way as for all the previous tests, the tests for the OCI index are listed below following the order of the four non-opportunistic base protocols. It starts with the “Custody Transfer” routing.

1) OCI index’s influences on Reliability (message delivery rate)

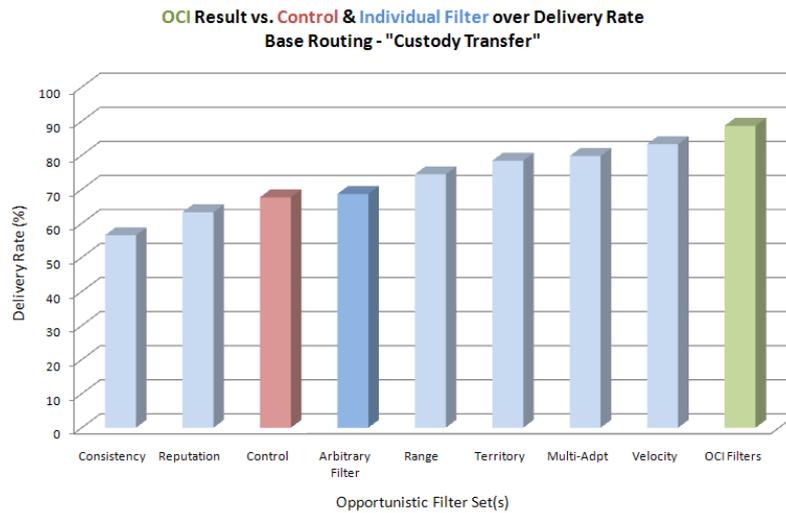


FIGURE 6-29: MESSAGE DELIVERY RATE FOR "CUSTODY TRANSFER"

Figure 6-29 shows the average message delivery rates over 800 message repetitions (see section 5.3.3) for each of the simulation configurations. The green bar specifies the delivery rate value of the simulation that is conducted using the OCI index as the opportunistic routing decision making algorithm. The red bar represents the delivery rate of the control configuration, the dark blue bar indicates the delivery rate value of a simulation that uses a non-OCI multi-filter, and the light blue bars are the delivery rate values of each individual opportunistic filter. Figure 6-29 clearly indicates the superiority of the OCI index over the control configuration and all other opportunistic routing methods in terms of routing reliability. Using OCI index, in comparison with the control configuration, the average message delivery rate is improved by 31.05%, where the best single filter improvement over control configuration is 23.11%.

2) OCI index’s influence on message delivery time (performance)

Figure 6-30 indicates the routing performance for each of the simulation configurations.

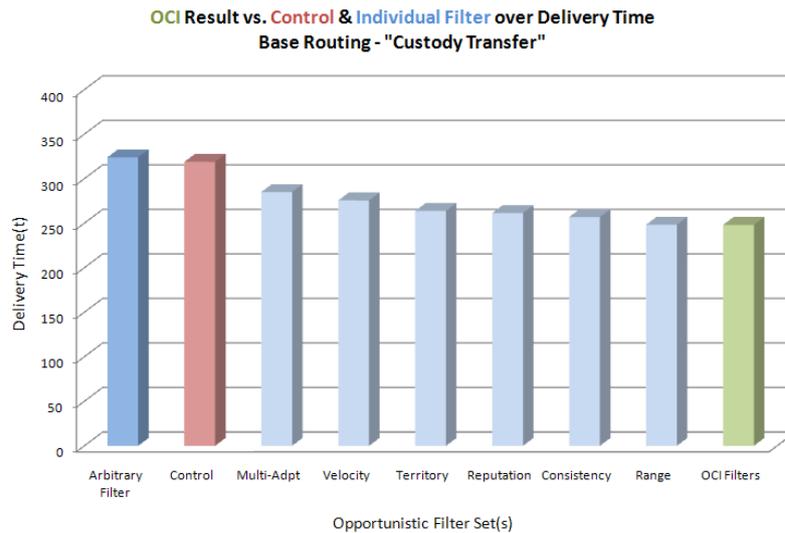


FIGURE 6-30: MESSAGE DELIVERY TIME OVER "CUSTODY TRANSFER"

In this graph, it is evident that the simulation result produced from the OCI index has the shortest delivery time and therefore the best performance. In comparison to the control configuration it reduces the delivery time by 22.18%, where the best single filter performance improvement, 22.14%, comes from the communication range filter.

3) OCI index’s influence on message delivery routing depth (efficiency)

The following graph shows the values of delivery routing depth, which are used to indicate the message routing efficiency for each of the given simulation configurations under "Custody Transfer" base routing.

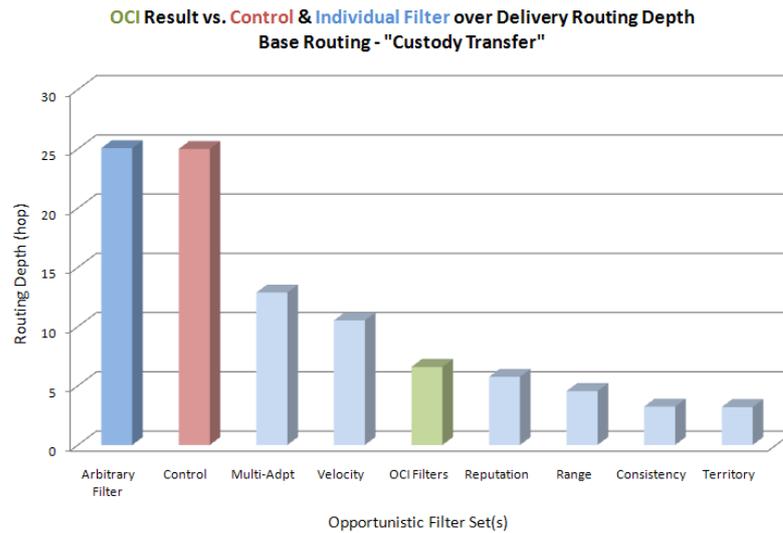


FIGURE 6-31: MESSAGE DELIVERY ROUTING DEPTH OVER "CUSTODY TRANSFER"

In *Figure 6-31*, it is observed that when using an OCI index for routing, compared with the other routing configurations, the routing depth is at an average level. Even so, OCI index gives 73.63% improvement over the control configuration. The highest routing depth improvement, 87.19%, is provided by the peer movement territory filter.

From *Figure 6-29* to *Figure 6-31*, one can see that the results of message delivery rate, message delivery time and message delivery routing depth, produced by the control configuration and the non-OCI multiple filter configuration are uniform and nearly identical. This is due to the fact that randomly generated filter orders using PRNG (see *section 3.5.3.2* and *section 5.3.1*) over 800 message repetitions are evenly distributed. Therefore, a conclusion can be drawn that a simple random grouping of individual opportunistic filter does not offer routing improvement, while OCI does.

4) OCI vs. non-opportunistic and best single opportunistic routing

Table 6-6 lists the simulation results, delivery rate, delivery time and delivery routing depth, of the control configuration, best single filter configuration and the OCI configuration.

Simulation Value – “Custody Transfer”			
Filter Category	Rate	Time	Depth
Control	67.63%	318.93	24.99
Best Single Delivery Rate (Velocity)	83.25%	275.64	10.51
Best Single Delivery Time (Range)	74.38%	248.33	4.55
Best Single Routing Depth (Territory)	78.38%	263.84	3.20
OCI Filters	88.63%	248.18	6.59

TABLE 6-6: SIMULATION VALUE – “CUSTODY TRANSFER”

Table 6-7 lists the percentage improvement of delivery rate, delivery time and delivery routing depth, for each of the best single filter configurations and the OCI configuration, in comparison to the control configuration.

Improvement / Tradeoffs – “Custody Transfer”			
Filter Category	Rate	Time	Depth
Control	0.00%	0.00%	0.00%
Best Single Delivery Rate (Velocity)	23.11%	13.57%	57.93%
Best Single Delivery Time (Range)	9.98%	22.14%	81.79%
Best Single Routing Depth (Territory)	15.90%	17.27%	87.19%
OCI Filters	31.05%	22.18%	73.64%

TABLE 6-7: IMPROVEMENT/TRADEOFFS – “CUSTODY TRANSFER”

The data listed above clearly shows that, in comparison to the control configuration, OCI indexing gives significant improvement to all aspects of the routing processes. When compared with the results of individual opportunistic filters, OCI is observed as the most reliable and the best performing opportunistic routing algorithm. For the aspect of routing efficiency, although OCI index loses to the Territory filter by 13.55%, it exceeds the Territory filter by 15.15% on the delivery rate and 4.91% on delivery time.

Thus a conclusion can be drawn, when applied to the “Custody Transfer” base routing protocol, OCI index offers significant and balanced improvement on all three aspects of routing reliability, routing performance and routing efficiency.

6.4.2 Test OCI over “Custody Transfer Propagation” Base Routing

In this section, an OCI index is tested against the control configuration, single opportunistic filter configuration and non-OCI multiple filter configurations, over the “Custody Transfer Propagation” base routing.

1) OCI index’s influence on message delivery rate (reliability)

The following graph lists the values of message delivery rate for each of the given simulation configurations under “Custody Transfer Propagation” base routing.

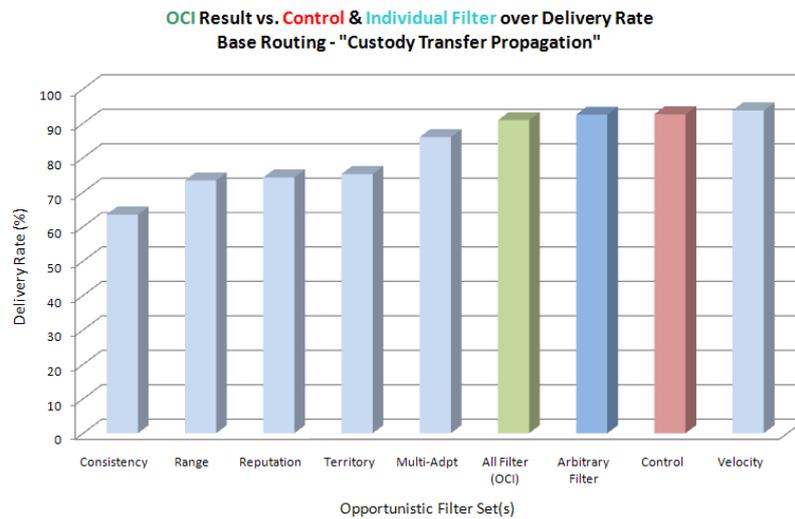


FIGURE 6-32: MESSAGE DELIVERY RATE OVER “CUSTODY TRANSFER PROPAGATION”

Figure 6-32 shows that while most of the individual filters suffer considerable reliability losses, OCI index only results in a very slight compromise, 1.89% delivery reduction. The only reliability improvement comes from the Peer Velocity Filter and it raises the delivery rate by 1.21%.

2) OCI index’s influence on message delivery time (performance)

The following graph shows the message delivery times for each of the given simulation configurations under the “Custody Transfer Propagation” base routing.

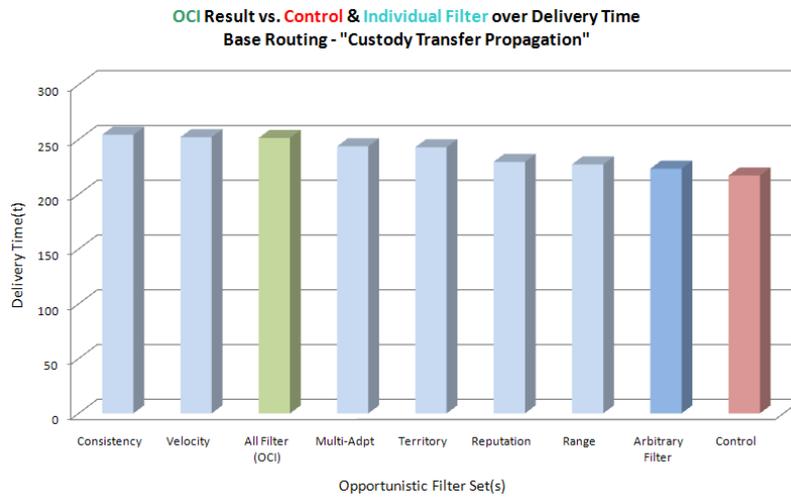


FIGURE 6-33: MESSAGE DELIVERY TIME OVER “CUSTODY TRANSFER PROPAGATION”

From *Figure 6-33* one can see that all opportunistic routing configurations result in routing performance decline. OCI index causes a noticeable performance compromise, and the delivery time is 15.84% more than the time of the control configuration. The Communication Range Filter gives the least compromise of 4.74%.

3) OCI index’s influence on message delivery routing depth (efficiency)

The following graph is a list of the values of message delivery routing depth for each of the given opportunistic simulation configurations and the control configuration under “Custody Transfer Propagation” base routing.

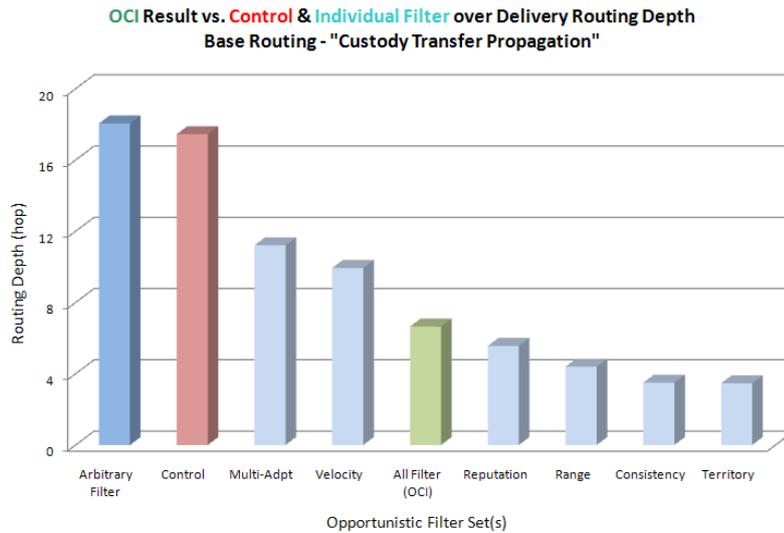


FIGURE 6-34: MESSAGE ROUTING DEPTH OVER "CUSTODY TRANSFER PROPAGATION"

As shown in *Figure 6-34*, OCI index gives a significant routing depth reduction of 61.89% over the control configuration. Compared with the single opportunistic filters, the improvement produced by OCI index is at an average level, where the most improvement, 80.16%, is from the Peer Movement Territory filter and the least, 35.70%, is produced by the Multi-Adaptor filter.

4) OCI index's influences on message delivery duplication (efficiency)

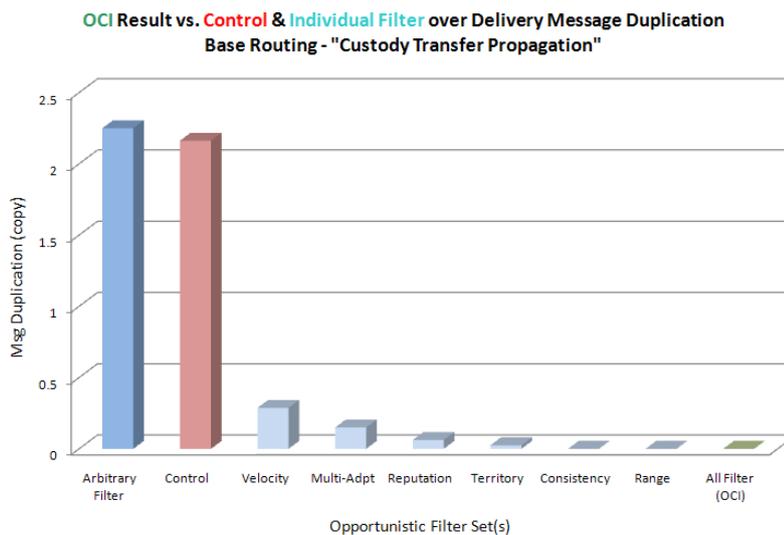


FIGURE 6-35: MESSAGE DUPLICATION OVER "CUSTODY TRANSFER PROPAGATION"

Figure 6-35 shows the message duplication values for each of the given opportunistic simulation configurations and the control configuration under “Custody Transfer Propagation” base routing. It illustrates an exciting result, when applying OCI, the message duplication has been eliminated, in other words, 100% improvement.

5) OCI vs. non-opportunistic & best single opportunistic routing

Table 6-8 is a list of simulation results that include message delivery rate, delivery time, delivery routing depth, and message duplication for the control configuration, best single filter configuration and the OCI index routing configuration.

Simulation Values – “Custody Transfer Propagation”				
Filter Category	Rate	Time	Depth	Duplication
Control	92.63	217.04	17.45	2.16
Best Single Delivery Rate (Velocity)	93.75	252.14	9.94	0.29
Best Single Time (Range)	73.38	227.33	4.40	0.00
Best Single Routing Depth (Territory)	75.25	242.94	3.46	0.03
Best Single Duplication (Consistency)	63.50	254.30	3.50	0.00
OCI Filters Group	90.88	251.43	6.65	0.00

TABLE 6-8: SIMULATION VALUES – “CUSTODY TRANSFER PROPAGATION”

Table 6-9 lists the percentage improvement for each of the best single filter configurations and the OCI configuration, in comparison to the control configuration.

Improvement / Tradeoffs – “Custody Transfer Propagation”				
Filter Category	Rate	Time	Depth	Duplication
Control	0.00%	0.00%	0.00%	0.00%
Best Single Delivery Rate (Velocity)	1.21%	-16.17%	43.05%	86.71%
Best Single Time (Range)	-20.78%	-4.74%	74.79%	100.00%
Best Single Routing Depth (Territory)	-18.76%	-11.93%	80.16%	98.84%
Best Single Duplication (Consistency)	-31.44%	-17.17%	79.94%	100.00%
OCI Filters Group	-1.89%	-15.84%	61.89%	100.00%

TABLE 6-9: IMPROVEMENT/TRADEOFFS – “CUSTODY TRANSFER PROPAGATION”

Observations can be made from the data listed above. Under the “Custody Transfer Propagation” base routing, OCI index provides significant gain on routing efficiency while maintaining very high routing reliability through limited sacrifice of routing performance. Compared with the results produced through single opportunistic filters, OCI offers a more balanced overall improvement on routing processes.

Therefore a conclusion can be drawn: OCI is applicable for “Custody Transfer Propagation” to gain significant improvement over routing efficiency.

6.4.3 Test OCI over “Custody Retaining Propagation” Base Routing

In this section OCI index is tested over the “Custody Retaining Propagation” base routing protocol and compared with the control configuration, single opportunistic filter configuration and non-OCI multiple filter configuration.

1) OCI index’s influence on message delivery rate (reliability)

The following graph lists the values of message delivery rate for each of the given simulation configurations under “Custody Retaining Propagation” base routing.

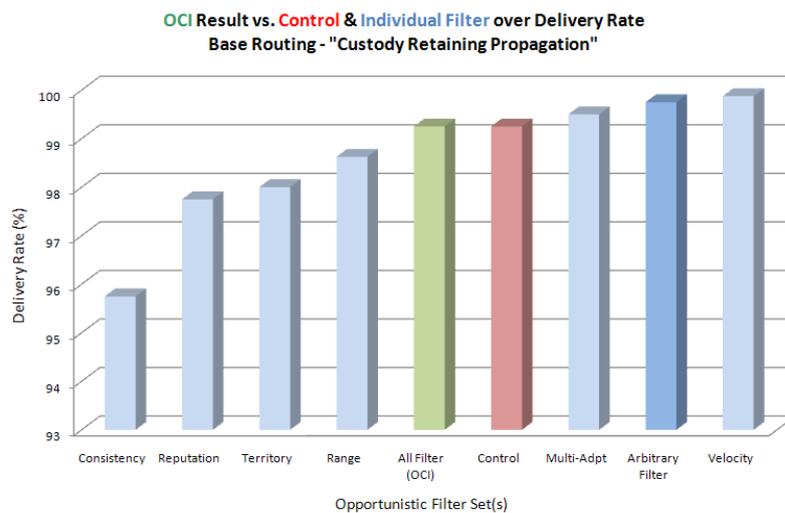


FIGURE 6-36: MESSAGE DELIVERY RATE OVER “CUSTODY RETAINING PROPAGATION”

Figure 6-36 shows that while most of the individual filters result in noticeable reliability losses, the OCI index maintains the same level as the control configuration with 0.00% improvement or tradeoff. The best improvement, 0.63%, is from the Peer Velocity filter.

2) OCI index’s influence on message delivery time (performance)

The following graph indicates the message routing performance for each of the given simulation configurations under “Custody Retaining Propagation” base routing.

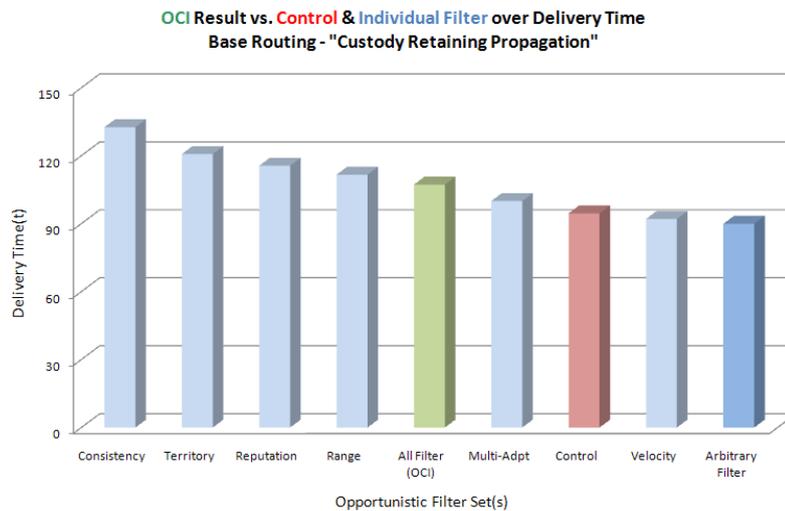


FIGURE 6-37: MESSAGE DELIVERY TIME OVER “CUSTODY RETAINING PROPAGATION”

From Figure 6-37, it is evident that most of the opportunistic routing configurations result in considerable performance decline. The compromise caused by OCI index, 13.40%, is below average level, where the only improvement, 2.64%, is brought about by the Peer Velocity filter.

3) OCI index’s influence on message delivery routing depth (efficiency)

In the following graph, the values of message delivery routing depth for each of the given routing configurations are listed to compare the routing efficiency under “Custody Retaining Propagation” base routing.

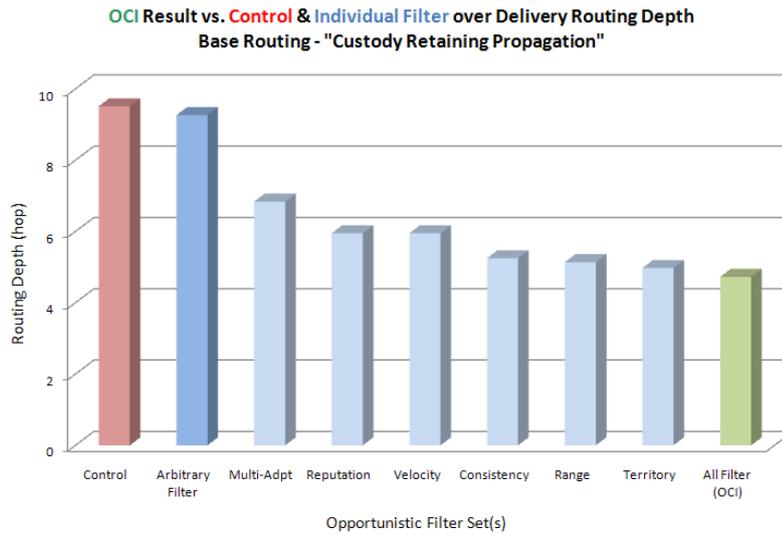


FIGURE 6-38: MESSAGE ROUTING DEPTH OVER "CUSTODY RETAINING PROPAGATION"

Figure 6-38 provides clear evidence that OCI index results in the lowest routing depth. In comparison to the control configuration, the OCI index reduces the message delivery routing depth by 50.33%.

4) OCI index's influence on message delivery duplication (efficiency)

The following graph shows the message duplication values for each of the given simulation configurations under "Custody Retaining Propagation" base routing.

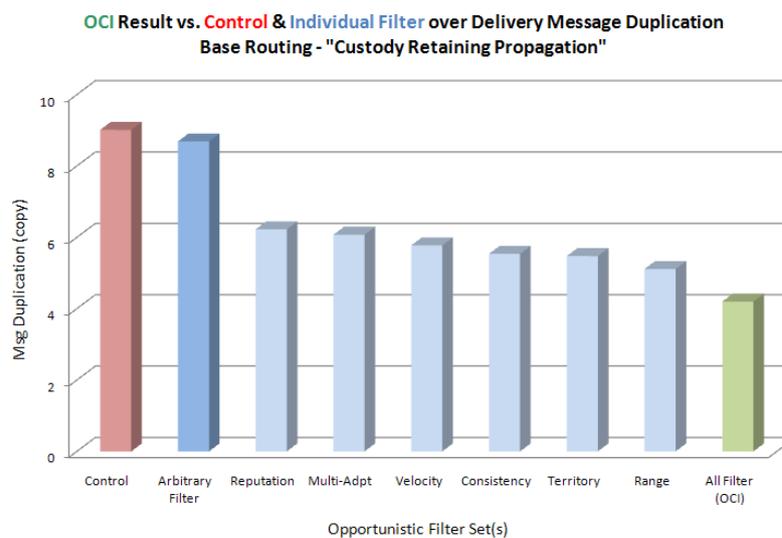


FIGURE 6-39: MESSAGE DUPLICATION OVER "CUSTODY RETAINING PROPAGATION"

From *Figure 6-39*, once again, OCI produces the best result in terms of routing efficiency improvement. It reduces the average message duplication value of the control configuration by 53.40%, while the best single filter improvement is at 43.27% by the Communication Range filter.

5) OCI vs. non-opportunistic & best single opportunistic routing

Table 6-10 lists the results that include delivery rate, delivery time, delivery routing depth, and message duplication for the control configuration, best single filter configuration and the OCI index routing configuration.

Simulation Values – “Custody Retaining Propagation”				
Filter Category	Rate	Time	Depth	Duplication
Control	99.25	94.58	9.51	9.01
Best Single Delivery Rate (Velocity)	99.88	92.08	5.95	5.78
Best Single Time (Velocity)	99.88	92.08	5.95	5.78
Best Single Routing Depth (Territory)	98.00	120.75	4.98	5.48
Best Single Duplication (Range)	98.63	111.64	5.14	5.11
OCI Filters	99.25	107.25	4.73	4.20

TABLE 6-10: SIMULATION VALUES – “CUSTODY RETAINING PROPAGATION”

Table 6-11 lists the percentage improvement or tradeoff for each of the best single filter configurations and the OCI configuration, in comparison to the control configuration.

Improvement / Trade-offs – “Custody Retaining Propagation”				
Filter Category	Rate	Time	Depth	Duplication
Control	0.00%	0.00%	0.00%	0.00%
Best Single Delivery Rate (Velocity)	0.63%	2.64%	37.45%	35.92%
Best Single Time (Velocity)	0.63%	2.64%	37.45%	35.92%
Best Single Routing Depth (Territory)	-1.26%	-27.68%	47.70%	39.25%
Best Single Duplication (Range)	-0.63%	-18.04%	45.99%	43.27%
OCI Filters	0.00%	-13.40%	50.33%	53.40%

TABLE 6-11: IMPROVEMENT/TRADE-OFFS – “CUSTODY RETAINING PROPAGATION”

From the data listed above, one can see that under the “Custody Retaining Propagation” base routing, OCI index provides a significant gain on routing

efficiency in both message delivery routing depth and the message duplication level. At the same time, OCI index keeps the same routing reliability level as the control configuration, and all these are achieved with very limited compromise of routing performance. Compared with the results produced through single opportunistic filters, OCI offers the best results on both message delivery routing depth and message duplication. Furthermore, it gives above average results for both message delivery rate and delivery time. Therefore, a conclusion can be made: OCI is highly applicable to the “Custody Retaining Propagation” to improve the routing efficiency.

6.4.4 Test OCI over “Simple Flood Propagation” Base Routing

The last base routing protocol to be used for testing the applicability of OCI is the “Simple Flood Propagation”.

1) OCI index’s influence on message delivery rate (reliability)

The following graph lists the values of message delivery rate for each of the given simulation configurations under “Simple Flood Propagation” base routing.

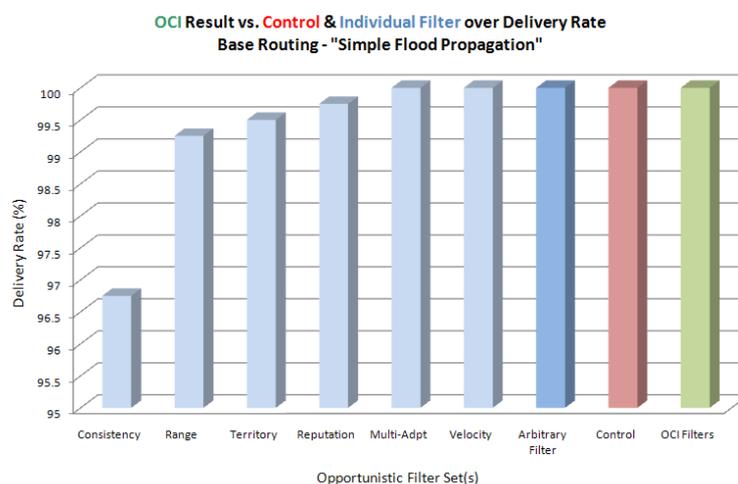


FIGURE 6-40: MESSAGE DELIVERY RATE OVER “SIMPLE FLOOD PROPAGATION”

Figure 6-40 illustrates that in this particular simulation scenario, applying OCI index keeps 100% delivery rate and has no negative impact on routing reliability,

while most single opportunistic filters result in noticeable declines in the message delivery rate.

2) OCI index’s influence on message delivery time (performance)

The following graph illustrates the routing performance for each of the given simulation configurations under “Simple Flood Propagation” base routing.

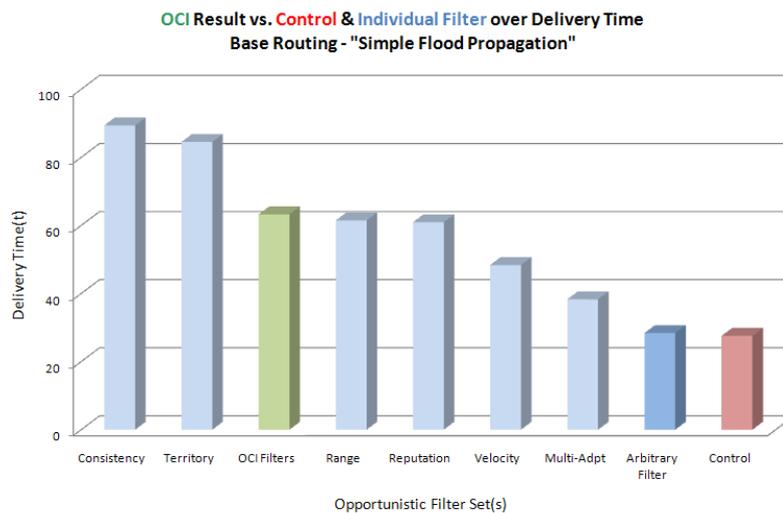


FIGURE 6-41: MESSAGE DELIVERY TIME OVER “SIMPLE FLOOD PROPAGATION”

Figure 6-41 shows that all opportunistic routing configurations have negative effects on the message delivery time. The OCI index has an average compromise, 129.42%, in comparison with the control scenario. And the least compromise, 38.89%, is produced through the Multi-Adaptor filter.

3) OCI index’s influence on message delivery routing depth (efficiency)

The graph below shows the value of message delivery routing depth for each of the given simulation configurations under “Simple Flood Propagation” base routing.

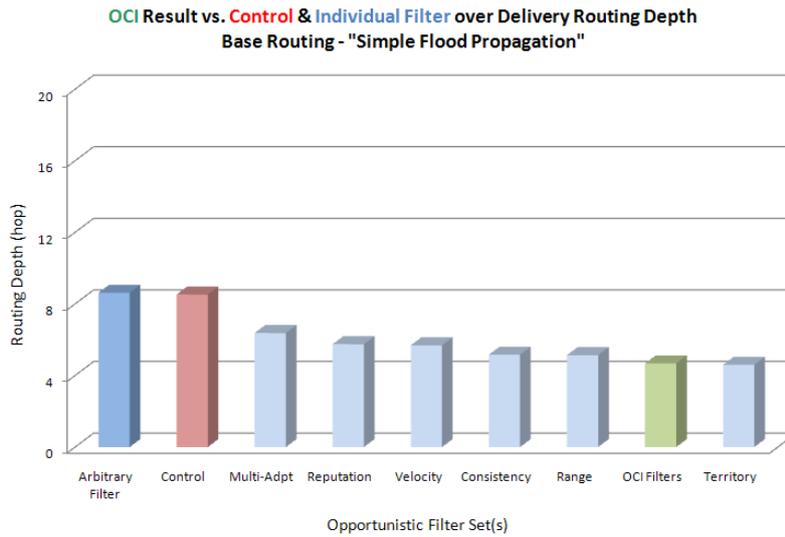


FIGURE 6-42: MESSAGE ROUTING DEPTH OVER "SIMPLE FLOOD PROPAGATION"

Figure 6-42 clearly shows that an OCI index provides a considerable gain over message delivery routing depth. It reduces the routing depth by 45.16%. At the same time, the most gain is from the Peer Movement Territory filter at 46.04%.

4) OCI index's influences on message delivery duplication (efficiency)

The following graph is a list of the values of message duplication for each of the given opportunistic simulation configurations and the control configuration under "Simple Flood Propagation" base routing.

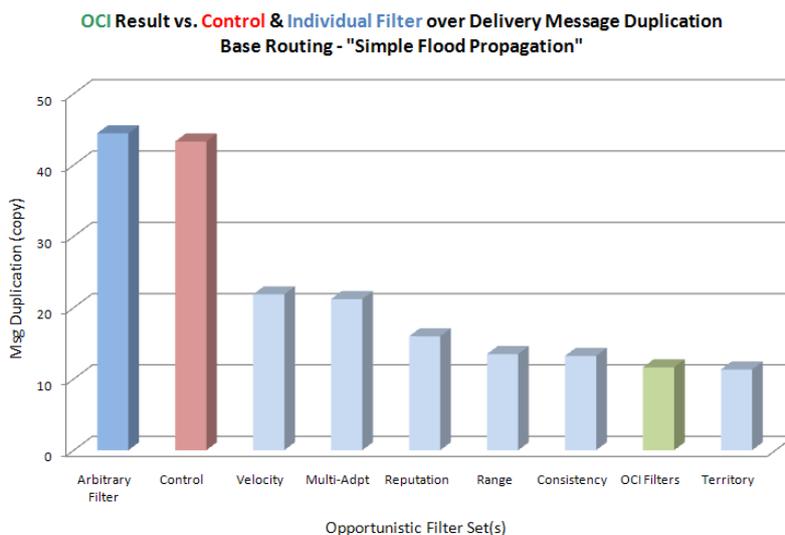


FIGURE 6-43: MESSAGE DUPLICATION OVER "SIMPLE FLOOD PROPAGATION"

Figure 6-43 shows that OCI index reduces the message duplication level significantly. In comparison to the control routing configuration, OCI reduces the message duplication by 73.15%, and it is almost as good as the Peer Movement Territory filter which reduces message duplication by 73.87%.

5) OCI vs. non-opportunistic & best single opportunistic routing

Table 6-12 lists the results that include delivery rate, delivery time, delivery routing depth, and message duplication for the control configuration, best single filter configuration and the OCI index routing configuration.

Simulation Values – “Simple Flood Propagation”				
Filter Category	Rate	Time	Depth	Duplication
Control	100.00	27.58	8.53	43.30
Best Single Rate (Multi-Adaptor)	100.00	38.30	6.38	21.19
Best Single Time (Velocity)	100.00	48.38	5.69	21.86
Best Single Routing Depth (Territory)	99.50	84.54	4.60	11.31
Best Single Duplication (Territory)	99.50	84.54	4.60	11.31
OCI Filters	100.00	63.26	4.68	11.63

TABLE 6-12: IMPROVEMENT/TRADE-OFFS – “SIMPLE FLOOD PROPAGATION”

Table 6-13 lists the percentage improvement or trade-off for each of the best single filter configurations and the OCI configuration, in comparison to the control configuration.

Improvement / Trade-offs – “Simple Flood Propagation”				
Filter Category	Rate	Time	Depth	Duplication
Control	0.00%	0.00%	0.00%	0.00%
Best Single Delivery Rate (Velocity)	0.00%	-75.43%	33.28%	49.51%
Best Single Time (Multi-Adaptor)	0.00%	-38.89%	25.22%	51.07%
Best Single Routing Depth (Territory)	-0.50%	-206.57%	46.04%	73.87%
Best Single Duplication (Territory)	-0.50%	-206.57%	46.04%	73.87%
OCI Filters	0.00%	-129.42%	45.16%	73.15%

TABLE 6-13: IMPROVEMENT/TRADE-OFFS – “SIMPLE FLOOD PROPAGATION”

From the data listed above, one can see that under “Simple Flood Propagation” base routing, OCI index provides a significant gain on routing efficiency in both the message delivery routing depth and the message duplication level, and at the same time, OCI index keeps 100% message delivery rate. However, the performance compromise is also high. Please refer to *section 6.3.1.4-E* for an argument and explanation regarding the performance compromise of “Simple Flood Propagation”.

From the observation of *Figure 6-40* to *Figure 6-43* and the tables above, one might notice that the Peer Movement Territory filter and the Multi-Adaptor filter provide very tough competition to the OCI index in terms of individual improvement. However, the Peer Movement Territory filter suffers greatly with respect to message delivery time, and the Multi-Adaptor does not provide the same level of efficiency gain as the OCI index. Since the OCI index tested here acquired its weight through a balanced approach (see *section 6.3.2*) this is an ideal result to reflect the positive applicability of the concept of OCI.

A conclusion can therefore be made: OCI is highly applicable for “Simple Flood Propagation” to significantly improve the routing efficiency, despite its high performance compromise.

6.5 Data Analysis, Evaluation and Conclusion

From the results presented in this chapter, derived from procedures of opportunistic element isolation and selection, individual opportunistic filter modelling and simulation, intelligent¹⁵ weight assignment, and OCI index testing, through evaluation and comparative analysis with control routing configuration, arbitrary combination of opportunistic routing, and specifically selected scenarios over all four base non-opportunistic routing protocols, and with statistically credible repetitions, it has been shown that the OCI concept has significant merit.

¹⁵ Intelligent weighted assignment is defined in section 3.6.4.2.

While the use of OCI routing provides broad improvements, it has to be acknowledged that in some scenarios certain trade-offs have to be made to certain aspects of the routing quality metrics in order to achieve an overall improvement. It is most noticeable when extreme base routing protocols are employed. For instance, OCI has considerable performance trade-offs to provide higher efficiency, for the “Simple Flood Propagation” protocol as discussed in *section 6.4.4*.

When multiple opportunistic filters are employed collaboratively and intelligently, OCI offers a novel routing approach to DTMANETs, which improves the reliability, performance, and efficiency of message interaction, which has been shown to be superior to the results that can be achieved through traditional single element opportunistic routing algorithms.

6.6 Chapter Review

This chapter reports on the findings of this project. Its goal is to provide evidence through data acquisition and analysis, which either supports or disproves the concept of OCI. Firstly, opportunistic elements are tested for their impact upon the DTMANET routing process. Secondly, each of the selected opportunistic elements are modelled as opportunistic filters, and put to the test. Thirdly, each filter is assigned weights intelligently, based on the individual test result to form OCI indexes. Finally, the balanced OCI index is chosen and tested against other routing configurations over four base routing protocols. In the next chapter, a critical evaluation and conclusion will be made for this research.

Chapter VII Conclusion and Future Work

“We remember our past, experience the present, and speculate about the future. While the future is full of possibility, the past is set in stone. And juxtaposed between the two is the directly experienced present that seems to move from the known past into the unknown future.”

Thomas J. McFarlane – “The Nature of Time”

Firstly in this chapter a review of the thesis is presented, which shows how the design, development, analysis, and evaluation of the concept of OCI was carried out and how the improvement of reliability, performance and efficiency was achieved using this approach. Secondly, the contributions made by this research are identified. Thirdly, a critical analysis of the results achieved by this investigation is presented, which includes challenges and limitations faced during the research. Fourthly, future extensions are discussed.

7.1 Thesis and Findings Review

In Chapter I, the hypothesis of this work was defined and the desired goals were stated.

In the literature survey presented in Chapter II, the scope for this work was outlined, the research direction was fixed and the originality of the work was presented. This was achieved by comparing the research scope of this work with other related research.

In Chapter III, the foundations for the framework of OCI were laid. Assumptions made in this work are listed in section 3.3, which helped to scope and focus the tests of OCI on opportunistic routing investigations. The philosophy of OCI design was presented and the OCI framework was established conceptually (see *section 3.5*) through detailed descriptions of adopted methods and procedures, such as OCI index

weight assignment methods and OCI voting procedures (see *section 3.6*). Descriptions made in this Chapter were applied as guidelines and blueprints for the design and construction of the OCI framework and were used as the implementation building blocks for the development of the proof-of-concept system.

By introducing the design philosophy and the implementation considerations (see *section 4.2*), Chapter IV provided evidence of the integrity, originality, rationality and versatility in designing and utilizing the experimental testing environment, the OCI-SIM, to aid in the simulation research approach of OCI.

Chapter V acted as a preparatory stage for the OCI simulations. Through a description of the testing of random number generation (see *section 5.3.1*), the definitions of the relative terms (see *section 5.3.2*), and the implementation of validity control procedures (see *section 5.3.3*), Chapter V offered a reliable and credible simulation environment for OCI data acquisition and analysis.

Chapter VI analyzed the data produced by OCI framework simulation procedures, and evaluated the extent to which the OCI proposal is feasible. From the simulation results provided in *section 6.1*, it was shown that opportunistic elements have a great impact on message routing in Delay Tolerant Mobile Ad Hoc Networks in general. In *section 6.3* it was shown that opportunistic elements can be modelled into routing protocols and can be used positively to assist routing decision-making process. *Section 6.4* gave details of how OCI indexes were subjected to a series of tests over a range of selected base routing protocols. The results of each test showed significant overall routing quality improvement and in some of the most applicable scenarios tested, for instance, when applied to “Custody Transfer” base protocol, OCI yielded a 31.05% message delivery increase (reliability improvement), 22.18% message delivery time reduction (performance improvement), and 73.64% decrease in routing depth (efficiency improvement).

Through the analytical reports presented in this chapter, conclusive evidence has been provided to validate the project hypothesis of using multiple opportunistic routing elements to aid message routing in Delay Tolerant Mobile Ad Hoc Networks. In addition, the data shows convincing proof that the Opportunistic Confidence Index as a general framework has made significant improvements on DTMANET routing within all aspects of the quality matrix of reliability, performance and efficiency.

7.2 Contributions

Through this work, we have proposed and proved a concept; designed and established a framework; and developed and made use of a tool-set as the proof-of-concept environment.

7.2.1 Proposed and Proved a Concept

The primary contribution made by this work is that it proposes and proves the concept of OCI, which uses multiple opportunistic routing protocols to make adaptive routing decisions through intelligent protocol selection, grouping and indexing. The results of OCI show significant improvements for DTMANET routing reliability, performance, and efficiency.

7.2.2 Establishment of a Framework

This research systematically established the framework and procedures to produce OCI indexes through opportunistic element selection, opportunistic filter modelling and analysis, intelligent filter weighting, and OCI index testing and evaluation. The procedure is supported by the methods and techniques of abstract modelling, opportunistic element simplification and isolation, random attribute generation and assignment, localized knowledge sharing, intelligent weight assignment and opportunistic element permutation, automated scenario generation, and statistically optimized simulation repeats.

7.2.3 Development of a Proof of Concept System

The implementation of a series of software tool sets, OCI-AUTO, OCI-VISUAL, OCI-CUSTOM and OCI-GRAPH serve as proof-of-concept systems. The OCI tool sets provide a working simulation environment that supports the tests and evaluations of the hypothesis of OCI. They are made available for further study and investigation for other researchers.

7.3 Critical Analysis of the Investigation and Findings

This section outlines the challenges faced and the decisions made during this research, discusses the algorithms, methods, and techniques adopted, and it identifies and addresses aspects that may be perceived as limitations.

7.3.1 Selection of the Test Environment

One important decision made during this research was to choose the empirical approach of software simulation rather than a physical test-bed. The software simulation approach is a widely used method to conduct network-oriented research. In a simulation environment, experiments can be carried out easily and in a controlled fashion. However, network simulation cannot fully represent real world applications as a physical test-bed does. Most simulation software represents a simplified and idealized network environment. A DTMANET often contains a large number of mobile devices. To prove the general applicability of OCI, experiments have to be conducted with a substantial number of mobile devices, and hundreds to thousands would have been necessary in this research. It was not practical to set up a test-bed to evaluate the concept of OCI on such a large scale in an academic research project like this.

In addition, in order to ensure the statistical soundness of the testing results, a high level of repetition was required. In this research, nearly a hundred scenarios were

repeated for each DTMANET and around a thousand were repeated for each message, as described in chapter V and VI. This would have been impractical to achieve on a real world test-bed environment. Therefore, for this work, simulation was the only practical option. Despite the limitations of a network simulation approach, the OCI simulation environment, OCI-SIM, provides a number of benefits besides larger numbers of mobile devices and improved statistical validity. A few of these benefits include automatic scenario configuration, opportunistic filter permutation, automatic simulation of multiple scenarios, and customized device simulation.

7.3.2 Choice of Behavioural Pattern Modelling

There are two ways to model behavioural patterns for network simulation studies. The first one is to use recorded real life data, the second is to generate data programmatically. For the OCI research the second approach was adopted. This is because, firstly, as a newly emerged architecture, DTMANET is still in its theoretical development stage and thus, there is no data publicly available from a real world recording. Secondly, using recorded data will limit both the length of the simulation and the dynamic behaviour of DTMANET. For this work, a programmatically generated behavioural pattern was a better choice in terms of higher simulation diversity and greater flexibility.

7.3.3 Representation of DTMANETS

One challenge faced during the course of this research was how to represent DTMANETS in a simulation environment. Initially, an attempt was made to use a manually constructed model. The manual model used specifically designed scenarios to determine and construct DTMANETS topologically. It worked well for case-specific simulations with limited simulation repeats. However, observation of the results generated from the manual model revealed its limitations in terms of the chaotic nature of a DTMANET. Firstly, a large number of simulation repeats was needed to reduce the result fluctuation. Secondly, a single fixed scenario could not fully represent the dynamics within a DTMANET, such as autonomic movement behaviour, and routing participation decision-making. Thirdly, because of the adoption of PRNG, the simulation length was limited and would repeat the simulation cycle. Fourthly, to prove the concept, a much more generally applicable approach was needed. Therefore, a new approach was adopted and a new model was created.

The new model uses a completely random scenario generation approach. In this model, a scenario is no longer specified through its topology. Instead, it is described by its contents, and the initial topology of a scenario is randomly generated according to the contents descriptions. With the new approach, the limitations listed were avoided. In addition, in a DTMANET, the more consistent the pattern for an opportunistic element, the easier it is to design a routing protocol using such a pattern. Therefore, a completely random scenario (see *section 4.3.2.3* and *section 5.2.4.1*), which follows no consistent pattern, represents the worst-case scenario. Simulation results presented in *section 6.4* were produced through completely random scenarios. Therefore, if OCI is proven applicable in such scenarios, it will be proven valuable in a more general sense.

7.3.4 Choice of Opportunistic Elements

Another challenge was how to choose the opportunistic elements to test the concept. The fact is acknowledged in this work that there are many elements that may have influences on the routing process in a given DTMANET. However, since the primary objective of this work is to confirm and validate a concept, it is not necessary to enumerate a large number of these elements. Instead, a selection of a limited but sufficient number of these elements will represent and convey the concept. The selection of opportunistic elements for this research uses the following principles. Firstly, elements have to be representative. Secondly, elements have to be intuitive and easy to perceive so that the use of multiple opportunistic elements can be presented clearly without losing focus to the complexity of any individual element. Thirdly, elements have to be suitable for simulation. Fourthly, elements have to make a substantial impact on DTMANET routing. For example, the velocity opportunistic element was selected to represent the temporal aspects (time), the territory opportunistic element was used for its spatial representation while the reputation opportunistic element was chosen because it represents social behaviour aspects.

7.3.5 Choices of Non-Opportunistic Base Routings

DTMANETs differ from one another drastically, and different applications of DTMANETs may have varied requirements. For instance, a DTMANET notification system emphasises rapid message delivery, while a content sharing DTMANET may need high efficiency, and other DTMANETs may require a balance between performance and efficiency. Several research initiatives have proposed a number of non-opportunistic routing protocols, each targeting specific features. The non-opportunistic base routing protocols chosen for supporting OCI might not be exhaustive, but they are representative. “Custody Transfer” routing represents DTMANETs with an extreme efficiency orientation, “Simple Flood Propagation” routing represents DTMANETs with performance priority and “Custody Transfer

Propagation” and “Custody Retaining Propagation” lean towards a less extreme orientation. Therefore, proving the value of OCI in all four of these base routing cases is to prove the versatility and general applicability of OCI.

7.3.6 Simplified Scenario Modelling

A number of assumptions were made (see *section 3.3*) during the modelling process of OCI. Such assumptions included the boundary assumption, the unique peer identification assumption, the initial connection assumption, the message relay assumption, the message physical size assumption, the transmission time assumption, the information sharing assumption and the peer power consumption assumption. Through these assumptions, the complexity of scenarios and opportunistic elements modelling were simplified and research into OCI could therefore focus on the opportunistic routing protocol level. Since the objective of this research was to prove and evaluate OCI at a conceptual level, this simplification was reasonable and justified. Nevertheless, it has to be acknowledged that OCI cannot be directly applied at a real world application level without considering the impact of these simplifications.

7.3.7 Output Data Evaluation Benchmarks

In traditional network studies, to evaluate a network routing protocol, parameters such as shortest route, bandwidth utilization, packets per second, round trip time (RTT), RTT variance, backbone packet loss, and circuit performance are commonly used [Sato *et al.*, 1994] [McRobb, 1997]. However, in this research, most of these common benchmarking parameters are not adopted. Instead, in order to evaluate the concept of OCI, output data such as message delivery rate, message delivery time, message routing hops and message duplication numbers are used as benchmarks to reflect evaluation categories such as routing reliability, routing performance, and routing efficiency. The reasons for these choices are, firstly, that the nature of message-oriented communication means that traditional benchmarking parameters

will not provide relevant measurements in the context of DTMANET. For example, “packets per second” make no logical sense as a measure for networks composed of peers that can cache and relay messages with intervals of up to hours, and where there is no concept of a “backbone”. At the same time, in OCI simulations the source peer and the destination peer of each end-to-end communication are randomly selected. Thus, there is no reason to implement an explicit round trip measurement, since each communication is equivalent to two single trips and in OCI message delivery is measured with one-way communication.

7.3.8 Computational Cost of OCI Determination and Routing

In this work, the determination of OCI and the routing decision-making processes are carried out programmatically in simulation-based investigations. During the course of this research, based on observations of experimental simulations, the computational overheads introduced into such processes appeared insignificant.

The OCI-SIM is designed to run on top of a single CPU as a single process, on which distributed parallelism cannot be implemented to compute peer movement patterns or OCI indices determination. Nor can it be used to make routing decisions. This implies that all movement patterns and routing decisions made at any given time in an OCI simulation have to be computed sequentially. With up to thousands of peers and hundreds of messages in a single simulation, the simulator performance did not noticeably deteriorate¹⁶. Thus, we uncovered no reasons why the real time computation overhead should be a significant impediment to its use in live DTMANETs.

¹⁶ However, a greater number of peers does slow down the simulation when routing processes are visualized. This is because the draw and paint methods used to display the peers are computationally intensive.

7.4 Future Research

The following possible future research directions represent extensions arising from this research that would be worth further investigation.

7.4.1 Realistic Testing Environment

As a concept, OCI has successfully demonstrated its usefulness. However, the results gathered through the research are from simulations, which have their limitations. Therefore, an immediate extension of this work is to set up a test-bed using real mobile devices so that the concept of OCI can be tested in environments that are more realistic.

7.4.2 A Comprehensive Opportunistic Taxonomy

The OCI has proved the applicability and the value of the concept of using the methods of adaptive, intelligent, grouping of multiple opportunistic routing protocols to improve DTMANET routing. However, only a few of the possible opportunistic routing protocols are used and tested as examples. Therefore, another possible extension of this research is firstly to establish a comprehensive taxonomy of opportunistic elements, which may have a high impact on DTMANET routing. Secondly, it would be possible to identify opportunistic elements' usefulness according to their commonly associated scenarios. For example, in a military application, message relay between peers is compulsory, so the reputational opportunistic element will not be useful, while in a file-sharing application the reputational element is critical. Thirdly, it would be possible to categorise them based on their opportunistic natures such as their spatial, temporal, or social behaviour.

7.4.3 Multi-Layered Time-Divided Routing Visualization

During the course of this research, a visualization tool (OCI-VISUAL) was developed

as a part of the OCI-SIM tool set. It provides an intuitive way for routing process observation and scenario construction. However, OCI-VISUAL gives a two dimensional view of DTMANET, and it is not capable of visualizing time-divided DTMANET routing processes. An extension of OCI-VISUAL might be to develop a multi-layered approach, in which each layer would represent a particular snapshot of the network topology at a given time, resulting in a three-dimensional visualization tool for the representation of such scenarios.

7.4.4 Multiple Attribute Decision Making System

This work demonstrates an efficient approach in a multiple attribute decision-making scenario. To make a routing decision, OCI considers the importance of each of the opportunistic elements according to its weight and the overall routing quality matrix preferences. These procedures can be extended and applied not only within the scope of mobile ad hoc networks to computer science, but also to other areas such as battlefield strategic decision-making and business planning, where decisions have to be made using multiple attributes with different impacts on a given scenario.

7.5 Concluding Comments

As a newly emerged technology, Delay Tolerant Mobile Ad Hoc Networks show significant potential, and promise to enable communication and information access in previously challenged or neglected network environments. Such environments may be as far away from us as outer space, and at the same time, they may be as close as a school in a neighbouring rural community. With the help of DTMANETs, communication in such environments may become possible without the high cost of traditional infrastructure deployment. However, we must acknowledge there are still many difficulties and challenges ahead to make such promises a reality, among which effective routing is one of the most fundamental challenges. We hope that improvements to Delay Tolerant Mobile Ad Hoc Networks such as those promised by the OCI approach to message routing will address and resolve these difficulties, and facilitate this type of readily constructible network, supporting current and future applications, and providing ubiquitous communication and information access.

Bibliography

"Love only grows by sharing. You can only have more for yourself by giving it away to others."

– Brian Tracy

[Adamson, 2007] Brian Adamson; "*Composite Computing and Mobile Ad-hoc Networks*"; NRL Code 5522, NAVAL Research Laboratory, Department of the NAVY; 2007.

[Barakat *et al*, 2000] C Barakat, E Altman, W Dabbous; "*On TCP performance in a heterogeneous network: a survey*"; IEEE Communications Magazine, page 40 to 45; 2000.

[Beaufour *et al*, 2002] Allan Beaufour, Martin Leopold, and Philippe Bonnet; "*Smart-tag based data dissemination*"; First ACM International Workshop on Wireless Sensor Networks and Applications (WSNA02); June 2002.

[Berkeley, 2007] UC Berkeley Robotics and Intelligent Machines Lab Home; Web image reference: <http://robotics.eecs.berkeley.edu>

[Bertsekas, 2002] Gafni-Bertsekas; "*Temporally-Ordered Routing Algorithm*"; Ad-Hoc Workshop Winter (Termine) Vainio; 2002

[Boardmantasker, 2007] boardmantasker.com website; Web image reference: <http://www.boardmantasker.com>

[Boeing, 2007] Boeing.com Defence Space website; Web image reference: http://www.boeing.com/defense-space/ic/fcs/bia/gallery/large/unmanned_ground2_lg.jpg

[Borrel *et al*, 2007] V Borrel, MH Ammar, EW Zegura; “*Understanding the Wireless and Mobile Network Space: A Routing-Centred Classification*”; Proceedings of the second workshop on Challenged networks; 2007.

[Brustoloni, 2006] Jose Brustoloni, Sherif Khattab, Christopher Santamaria, Brian Smyth and Daniel Mosse; “*Integrated Scheduling of Application- and Network-Layer Tasks in Delay-Tolerant MANETs*”; Dept. Computer Science, University of Pittsburgh; 2006

[Burgess *et al*, 2007] J Burgess, G Bissias, M Corner, BN Levine; “*Surviving Attacks on Disruption-Tolerant Networks without Authentication*”; Proceeding ACM Mobihoc; September 2007.

[Burleigh *et al*, 2003] S Burleigh, A Hooke, L Torgerson, K Fall, V Cerf; “*Delay-tolerant networking: an approach to interplanetary Internet*”; Communications Magazine, IEEE; 2003

[Cerf, 2002] Vinton G. Cerf; “*JCC(X) Data in Motion Workshop*”; presentation to ICIR; January 2002.

[Cerf *et al*, 2002] Vinton G. Cerf, Scott C. Burleigh, Adrian J. Hooke, Leigh Torgerson, Robert C. Durst, Keith L. Scott, Kevin Fall, Eric J. Travis, and Howard S. Weiss; “*Delay-tolerant network architecture: The evolving interplanetary internet*”; Internet Draft draft-irtf-ipnrg-arch-01.txt; April 2002.

[Cerf *et al*, 2007] Vinton G. Cerf *et al*; “*RFC 4838 Delay-Tolerant Networking Architecture*”; April 2007.

[Chen *et al*, 2001] Xiangchuan Chen and Amy L. Murphy; “*Enabling disconnected transitive communication in mobile ad hoc networks*”; in Proc. of Workshop on Principles of Mobile Computing, page 21–27 collocated with PODC’01; August 2001.

[Chin, 2003] Kwan-Wu Chin, John Judge, Aidan Williams and Roger Kermod; “*Implementation Experience with MANET Routing Protocols*”; Sydney Networks and Communications Lab Motorola Australia Research Centre; June 2003.

[Chuah *et al*, 2006] MC Chuah, P Yang, BD Davison, L Cheng; “*Store-and-Forward Performance in a DTN*”; Vehicular Technology Conference IEEE; 2006.

[Cole, 2004] Robert G. Cole; “*Initial Studies on Worm Propagation in MANETS for Future Army Combat Systems*”; JHU Applied Physics Laboratory; September 2004.

[Clausen *et al*, 2001] Thomas Heide Clausen, Philippe Jacquet, Laurent Viennot, Anis Laouiti, Amir Qayyum; “*Optimized Link State Routing Protocol*”; IEEE INMIC, in proceedings; 2001.

[Demers *et al*, 1987] Alan Demers, Dan Greene, Carl Hauser, Wes Irish, John Larson, Scott Shenker, Howard Sturgis, Dan Swinehart, and Doug Terry; “*Epidemic algorithms for replicated database maintenance*”; in Proceedings of the ACM Symposium on Principles of Distributed Computing, page 1–12; 1987.

[Demmer *et al*, 2004] Michael Demmer, Eric Brewer, Kevin Fall, Sushant, Jain, Melissa Ho, Robin Patra; “*Implementing Delay Tolerant Networking*”; Intel Research Berkeley Technical Report IRB-TR-04-020; December 2004.

[Demmer, 2005] Michael Demmer; “*The DTN Reference Implementation*”; IETF DTNRG Meeting; March 2005.

[Di Caro, 2003] Gianni A. Di Caro; “*Analysis of simulation environments for mobile ad hoc networks*”; Technical Report No. IDSIA-24-03,IDSIA / USI-SUPSI Dalle Molle Institute for Artificial Intelligence Galleria 2, 6928 Manno, Switzerland; December 2003.

[DTNRG, 2005] DTNRG Website; “*Delay Tolerant Network Research Group*”; Web reference: <http://www.dtnrg.org/wiki/About>; November 2005.

[Estri *et al*, 1999] Deborah Estrin, Ramesh Govindan, John Heidemann, and Satish Kumar; “*Next Century Challenges: Scalable Coordination in Sensor Networks*”; In Proceedings of the Fifth Annual ACM/IEEE International Conference on Mobile Computing and Networking, pages 263–270; 1999.

[Fall, 2002] Fall, Kevin; “*Delay-Tolerant Networking for Challenged Internets*”; presentation to ICIR; June 2002.

[Fall, 2003] Fall, Kevin; “*A delay-tolerant network architecture for challenged internets*”; Proceedings IRB-TR-03-003; February 2003.

[Fang *et al*, 2004] Zuyuan Fang and Brahim Bensaou; “*Fair Bandwidth Sharing Algorithms based on Game Theory Frameworks for Wireless Ad-hoc Networks*”; INFOCOM; 2004.

[Farber, 2000] David Farber; “*MMDF - Remote Mail Program*”; Web reference: http://www-mice.cs.ucl.ac.uk/multimedia/misc/tcp_ip/8813.mm.www/0045.html; March 2000.

[Franceschi, 2003] Paul Franceschi; “*On the Circularity in the Sorites Paradox*”; University of Corsica, France; November 2003.

[Fuchs *et al*, 2007] Ira Fuchs, Greydon Freeman; “*BitNet History*”; Web reference: http://www.livinginternet.com/u/ui_bitnet.htm; July 2007.

[Ghosh *et al*, 2006] Joy Ghosh, Chunming Qiao, Sumesh J. Philip, Hung Ngo, Seokhoon Yoon; “*Sociological Orbit aware Location Approximation and Routing (SOLAR) in DTN*”; Department of Computer Science and Engineering University at Buffalo; April 2006.

[Glance *et al*, 2001] Natalie Glance, Dave Snowdon, and Jean-Luc Meunier; “*Pollen: using people as a communication medium*”; Computer Networks, vol. 35, no. 4, page 429–442; March 2001.

[Gilleland, 2004] Michael Gilleland; “*Combination Generator*”; Merriam Park Software; Web reference: <http://www.merriampark.com/comb.htm>; July 2004

[Gilleland, 2004] Michael Gilleland; “*Permutation Generator*”; Merriam Park Software; Web reference: <http://www.merriampark.com/perm.htm>; July 2004

[Gray, 2003] Rich Gray; “*On the Edge: Interplanetary Internet*”; Web reference: http://www.space.com/business/technology/technology/ontheedge_0305.html; May 2003

[Gray, 2005] Doug Gray; “*A Taxonomy of loop free routing protocols*”; University of Colombo School of Computing; 2005.

[Grossglauser *et al*, 2002] Matthias Grossglauser and David Tse; “*Mobility increases the capacity of ad-hoc wireless networks*”; IEEE/ACM Transactions on Networking, vol. 10, no. 4; August 2002.

[Halvardsson *et al*, 2004] Mattias Halvardsson, Patrik Lindberg; “*Reliable group communication in a military Mobile Ad hoc Network*”; Reports from MSI, ISSN 1650-2647 , School of Mathematics and Systems Engineering, Växjö universitet; February 2004.

[Hauben, 1994] Michael Hauben; “*History of ARPANET Behind the Net - The "Open" History of the ARPANET/Internet*”; Web reference: <http://www.dei.isep.ipp.pt/~acc/docs/arpa.html> hauben@columbia.edu; December 1994.

[He, 2002] Guoyou He; “*Destination-Sequenced Distance Vector (DSDV) Protocol Networking*”; Laboratory Helsinki University of Technology; September 2002

[Heinzelman 1999] Wendy Rabiner Heinzelman, Joanna Kulik, and Hari Balakrishnan; “*Adaptive Protocols for Information Dissemination in Wireless Sensor Networks*”; In Proceedings of the Fifth Annual ACM/IEEE International Conference on Mobile Computing and Networking, pages 174–185; 1999.

[Haq *et al*, 2005] Furqan Haq and Thomas Kunz; “*Simulation vs. Emulation: Evaluating Mobile Ad Hoc Network Routing Protocols*”; Systems and Computer Engineering Carleton University; May 2005

[Hoare, 1978] Hoare, C.A.R.; “*Communicating Sequential Processes*”; Comm. ACM, 21(8), Page 666-677; 1978.

[Hu *et al*, 2004] Y. Charlie Hu, Saumitra M. Das, Himabindu Pucha; “*Peer-to-Peer Overlay, Abstractions in MANETs*”; Handbook on Theoretical and Algorithmic Aspects of Senso, Page 47, School of Electrical and Computer Engineering, Purdue University; 2004.

[IETF, 2007] IETF Website; “*Internet Research Task Force*”; Web reference: <http://www.irtf.org/>; June 2007.

[ITU, 2007] ITU Web Site; Web reference: <http://www.itu.int/osg/spu/ni/futuremobile/pervasive/index.html>; 2007.

[Jacquet *et al*, 2001] Jacquet, P.; Muhlethaler, P.; Clausen, T.; Laouiti, A.; Qayyum, A.; Viennot, L; “*Optimized Link State Routing Protocol for Ad Hoc Networks*”; Multi Topic Conference, 2001. IEEE INMIC 2001. Technology for the 21st Century. Proceedings. IEEE International Volume, Issue, 2001 Page(s): 62 - 68 2006.

[Jain *et al*, 2005] S Jain, K Fall, R Patra; "Routing in a Delay Tolerant Network"; University of Washington; ACM Portal – portal.acm.org; 2005

[JavaAPI, 2006] Java API document; "Secure Random"; Web reference: <http://java.sun.com/j2se/1.4.2/docs/api/java/security/SecureRandom.html>; 2006

[Jorjeta *et al*, 2001] Jorjeta G. Jetcheva David B. Johnson; "Adaptive Demand-Driven Multicast Routing in Multi-Hop Wireless Ad Hoc Networks"; Department of Computer Science Carnegie Mellon University Rice University Pittsburgh; 2001.

[Johnson *et al*, 1999] David B. Johnson, Josh Broch, Yih-Chun Hu, Jorjeta Jetcheva, David A. Maltz; "The CMU Monarch Project's Wireless and Mobility Extensions to ns"; The Monarch Project Carnegie Mellon University; 1999.

[Johnson *et al*, 2004] David B. Johnson, David A. Maltz, Josh Broch; "DSR: The Dynamic Source Routing Protocol for Multi-Hop Wireless Ad Hoc Networks"; Computer Science Department Carnegie Mellon University Pittsburgh; September 2004.

[Kahn *et al*, 1999] J. M. Kahn, R. H. Katz, and K. S. J. Pister; "Next Century Challenges: Mobile Networking for Smart Dust"; In Proceedings of the Fifth Annual ACM/IEEE International Conference on Mobile Computing and Networking, pages 271–278; 1999.

[Khaled, 2006] Khaled A. Harras and Kevin C.; "Inter-Regional Messenger Scheduling in Delay Tolerant Mobile Networks"; Almeroth Department of Computer Science University of California; 2006.

[Knuth, 1997] Donald Knuth; "The Art of Computer Programming, Volume 2: Seminumerical Algorithms, Third Edition"; page 1–193; Addison-Wesley, ISBN 0-201-89684-2; 1997.

[Ko *et al*, 2004] Young-Bae Ko and Nitin H. Vaidya; “*Location-Aided Routing (LAR) in mobile ad hoc networks*”; Department of Computer Science, Texas A&M University, College Station; August 2004.

[Kurkowski *et al*, 2004] Stuart Kurkowski, Tracy Camp, and Michael Colagrosso; “*MANET Simulation Studies: The Current State and New Simulation Tools*”; Department of Math, and Computer Sciences Colorado School of Mines; May 2004.

[L’Ecuyer, 1999] Pierre L’Ecuyer and Richard Simard; “*On the Performance of Birthday Spacings Tests with Certain Families of Random Number Generators*”; Universite de Montreal; September 1999.

[LeBrun, 2005] Jason LeBrun, Chen-Nee Chuah, Dipak Ghosal, Michael Zhang; “*Knowledge-Based Opportunistic Forwarding in Vehicular Wireless Ad Hoc Networks*”; IEEE VTC Spring; 2005.

[Lee *et al*, 2002] Sung-Ju Lee William Su Mario Gerla; “*ODMRP - On-Demand Multicast Routing Protocol in Multihop Wireless Mobile Networks*”; Computer Science Department, UCLA; ACM/Baltzer Mobile Networks and Applications, special issue on Multipoint Communication in Wireless Mobile Networks; 2002.

[Lee *et al*, 2006] Choonhwa Lee, Sungshick Yoon, Eensam Kim, and Sumi Helal; “*An Efficient Service Propagation Scheme for Large-Scale MANETs*”; College of Information and Communications, Hanyang University, South Korea; November 2006.

[Leguay *et al*, 2005] Jeremie Leguay, Timur Friedman, Vania Conan; “*Evaluating Mobility Pattern Space Routing for DTNs*”; Universite Pierre et Marie Curie, Laboratoire LiP6–CNRS, Thales Communications; 2005

[Lesh, 2001] James R. Lesh; “*Technologies for the Interplanetary Network*”; Core Technologies for Space Conference; November 2001.

[Li, 2006] Li Yang; “*Add dynamic Java code to your application*”; JavaWorld; Web reference: <http://www.javaworld.com/javaworld/jw-06-2006/jw-0612-dynamic.html>; June 2006.

[Li *et al*, 2003] Qun Li and Daniela Rus; “*Communication in disconnected ad-hoc networks using message relay*”; Journal of Parallel and Distributed Computing; 2003.

[Lindgren *et al*, 2003] Anders Lindgren, Avri Doria, and Olov Schel’én; “*Probabilistic routing in intermittently connected networks*”; In Proceedings of The Fourth ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc); June 2003.

[Lindgren, 2006] Anders Lindgren; “*Routing and Quality of Service in Wireless and Disruption Tolerant Networks*”; Luleå University of Technology, Department of Computer Science and Electrical Engineering, Division of Systems and Interaction; June 2006.

[Luby, 1996] Michael Luby; “*Pseudorandomness and Cryptographic Applications, A definitive source of techniques for provably random sequences*”; Princeton University Press; 1996.

[Ma *et al*, 2003] Rui Ma and Jacek Ilow; “*Reliable Multipath Routing with Fixed Delays in MANET Using Regenerating Nodes*”; LCN'03 Proceedings; 2003.

[Mackera *et al*, 2007] Joseph P. Mackera, M. Scott Corson; “*Mobile Ad Hoc Networking and the IETF*”; IT Division, Naval Research Laboratory, Washington, DC, USA, Mobile Ad Hoc Networking (manet) Working Group of the Internet Engineering Task Force (IETF); 2007.

[Maihöfer, 2004] Maihöfer, C. A; “*Survey on Geocast Routing Protocols*”; IEEE Communications Surveys and Tutorials 6 (2); 2004.

[Markham *et al*, 2007] Andrew C. Markham, Andrew J. Wilkinson; “*A bio-mimetic ranking system for mobile energy constrained wireless networks*”; 2007.

[Mauve *et al*, 2001] Mauve, M., Widmer, J., Hartenstein, H; “*A Survey on Position-Based Routing in Mobile Ad Hoc Networks*”; IEEE Network 1 (6): page:30-39; December 2001.

[McRobb, 1997] Daniel McRobb; “*Introduction to Network Performance Measurement*”; Web reference:
<http://www.academ.com/nanog/june1997/performance.html>; 1997.

[MIT, 2007] MIT 100\$ laptop project website; Web reference:
<http://laptop.media.mit.edu/>; 2007.

[MMDF, 2005] MMDF Users Group; web reference: <http://mmdf.sourceforge.net/>;
2005.

[Moraes, 1999] Mark Moraes; “*What is Usenet*”; Web reference:
<http://www.faqs.org/faqs/usenet/what-is/part1/>; December 1999.

[Motorola, 2007] Motorola Press; Web reference:
http://motoinfo.motorola.com/motoinfo/press_releases.asp?prid=10; 2007

[NIST, 2007] NIST website; “*Security Requirements for Cryptographic Modules*”;
Web reference: <http://csrc.nist.gov/cryptval/140-2.htm>; 2007.

[Nuevo, 2004] Jorge Nuevo; “*A Comprehensible GloMoSim Tutorial*”; INRS -
Universite du Quebec; March 2004.

[Oceanor, 2007] Oceanor.com website; Web image reference:
<http://www.oceanor.no/products/index.htm>

[OLSR, 2007] OLSR website; Web reference: <http://www.olsr.org/>; 2007.

[Ozan *et al*, 2006] Ozan, K. Tonguz, Gianluigi Ferrari; “*Ad Hoc Wireless Networks: A Communication- Theoretic Perspective*”; John Wiley & Sons; May 2006.

[Park *et al*, 1997] V. Park, M.S. Corson; “*A Highly Adaptive Distributed Routing Algorithm for Mobile Wireless Networks*”; Proc. IEEE INFOCOM '97; April 1997.

[Patra *et al*, 2003] R. Patra, S. Nedeveschi; “*DTNLite: A Reliable Data Transfer Architecture for Sensor Networks*”; Deeply Embedded Networks, 2003.

[Perkins *et al*, 1999] Charles E. Perkins and Elizabeth M. Royer; “*Ad hoc On-Demand Distance Vector Routing*”; Proceedings of the 2nd IEEE Workshop on Mobile Computing Systems and Applications, New Orleans, LA, pp. 90-100; February 1999.

[Pirzada *et al*, 2004] AA Pirzada, C McDonald; “*Establishing trust in pure ad-hoc networks*”; Proceedings of the 27th conference on Australasian Computer Science; 2004.

[Qmail, 2007] Qmail Website; “*Qmail: Second most popular MTA on the Internet*”; web reference: <http://www.qmail.org/top.html>; August 2007.

[Rau, 2006] A. Rau; “*Gamma-Ray Bursts*”; Web image reference: <http://www.astro.caltech.edu/~arne/grb.html>

[Ray, 1999] Suprio Ray; “*Realistic Mobility for MANET Simulation*”; Regional Engineering College, India; 1999.

[Rea *et al*, 2004] Susan Rea, Dirk Pesch; “*Source Routing Based on Destination Frequency Analysis for Wireless Mobile Ad Hoc Networks*”; Adaptive Wireless Systems Group, Department of Electronic Engineering, Cork Institute of Technology, Cork, Ireland; November 2004.

[Ross, 2007] Shane Ross; “*Interplanetary Network*”; Web image reference: <http://www.esm.vt.edu/~sdross/images/superhighway.jpg>

[Royer *et al*, 1999] Elizabeth M. Royer, Santa Barbara, Chai-Keong Toh; “*A Review of Current Routing Protocols for Ad Hoc Mobile Wireless Networks*”; University of California; April 1999.

[SAMSUNG, 2007] SAMSUNG.com website; Web image reference:

http://www.samsung.com/.../Products_Security.html

[Sankhla, 2004] Vishal Sankhla; “*SMART: A Small World based Reputation System for MANETs*”; Thesis; Faculty of the Graduate School, University of Southern California; December 2004.

[Sato *et al*, 1994] KI Sato, S Okamoto, H Hadama; “*Network performance and integrity enhancement with optical pathlayer technologies*”; Selected Areas in Communications, IEEE Journal; 1994.

[Savrasovs, 2003] M. Savrasovs; “*Research of Random-Number Generator Quality*”; 2003, Issue 4 "Proceedings of Student Research Works", Special Issue Prepared by Students of Transport and Telecommunication Institute (Riga, Latvia) and Negev Academic College of Engineering (Beer-Sheva, Israel); ISBN 9984-668-55-X

[Scheemaecker, 2001] Marc De Scheemaecker; “*NanoXML*”; Web reference:

<http://nanoxml.cyberelf.be/index.html>; 2001.

[Schumacher, 2004] Andre Schumacher; “*Ad-hoc On-Demand Distance Vector (AODV) Protocol Part 3 of 6*”; Department of Computer Science Faculty of Science, University of Helsinki; May 2004.

[SEAD, 2007] SEAD Website; “*How to use Ariadne and SEAD in ns-2.27-allinone on FC-2*”; Web reference:

<http://www1.ietf.org/mail-archive/web/manet/current/msg06513.html>; February 2006.

[Sen, 2006] Anandamoy Sen; “*Swarm Intelligence based optimization of MANET cluster formation*”; Thesis, The University of Arizona; 2006.

[Shen *et al*, 2002] Chien-Chung Shen, Girish Borkar, Sundaram Rajagopalan, and Chaiporn Jaikaeo; “*Interrogation-based relay routing for ad hoc satellite networks*”; in Proceedings of IEEE Globecom 2002, Taipei, Taiwan; November 2002.

[Spacetoday, 2007] Spacetoday website; Web image reference:

<http://www.spacetoday.org/images/SolSys/DeepSpaceNetwork/GoldstoneDSNantennaDay.jpg>

[SpaceRef, 2007] SpaceRef.com website; Web image reference:

<http://images.spaceref.com/news/02.28.00.marsnetwork.jpg>

[Spyropoulos *et al*, 2005] T Spyropoulos, K Psounis, CS Raghavendra; “*Spray and wait: an efficient routing scheme for intermittently connected mobile networks*”; Applications, Technologies, Architectures, and Protocols ACM, 2005.

[Stevenson *et al*, 2004] Sharlita Stevenson, Dr. Jennifer Welch; “*Analyzing Renaming Algorithms for MANET*”; Department of Computer Science; Wayne State University; 2004.

[Teixeira *et al*, 2004] Renata Teixeira, Aman Shaikh, Tim Griffin, Jennifer Rexford; “*Dynamics of HotPotato Routing in IP Networks*”; ACM SIGMETRICS Performance Evaluation Review; 2004.

[Theoleyre *et al*, 2005] Fabrice Theoleyre, Fabrice Valois; “*Robustness and reliability for virtual topologies*”; CITI-INRIA INSA Lyon 21, Avenue Jean Capelle, 69621 Villeurbanne, France; 2005.

[Tsai *et al*, 1995] K Tsai, RP Ma; “*DARTING: a cost-effective routing alternative for large space-based dynamic-topology networks*”; Military Communications Conference, 1995.

[Turocy, 2001] Theodore L. Turocy; “*Game Theory*”; Texas A&M University, Bernhard von Stengel, London School of Economics, CDAM Research Report LSE-CDAM-2001-09; October 2001.

[Typepad, 2007] Typepad.com website; Web image reference:
http://healthnex.typepad.com/photos/uncategorized/2007/06/08/aboutdiagram_03.gif.

[Usenet, 2001] Google Usenet Archive; “*20 Year Usenet Timeline*”; Web reference:
http://www.google.com/googlegroups/archive_announce_20.html; December 2001.

[Vahdat *et al*, 2000] Amin Vahdat and David Becker; “*Epidemic routing for partially connected ad hoc networks*”; Technical Report CS-200006, Duke University; April 2000.

[Viewimages, 2007] Viewimages.com website; Web image reference:
<http://www.viewimages.com>

[Vixie *et al*, 2001] Paul A. Vixie and Frederick M. Avolio; “*Sendmail: Theory and Practice 2nd Edition*”; Digital Press, ISBN 155558229X; December 2001.

[VNET, 2007] IBM VNET; Web reference: <http://en.wikipedia.org/wiki/VNET>;
November 2007.

[Vogels *et al*, 2002] Werner Vogels, Robbert van Renesse, and Ken Birman; “*The power of epidemics: Robust communication for large-scale distributed systems*”; In Proceedings of First Workshop on Hot Topics in Networks (HotNets-I); October 2002.

- [Walden, 2003] David Walden; “*The BELLMAN-FORD Algorithm and ‘Distributed BELLMAN-FORD’*”. Web reference:
<http://www.walden-family.com/public/bf-history.pdf>; 2003.
- [Wang *et al*, 2005] Y Wang, S Jain, M Martonosi, K Fall; “*Erasure-coding based routing for opportunistic networks*”; proceeding, ATAP, 2005
- [Wiberg, 2003] Björn Wiberg; “*Porting AODV-UU to ns-2 and Enabling Trace-based Simulation*”; UPPSALA University Master’s Thesis in Computer Science; May 2003.
- [Wick, 2007] Dave Wick; “*Delay Tolerant Networks in a Nutshell*”; RVS Group University of Berne; June 2007.
- [Wiki, 2007] Wiki Reference; “*Wizzy Digital Courier*”; Web reference:
http://en.wikipedia.org/wiki/Wizzy_Digital_Courier; 2007.
- [Wizzy, 2007] Wizzy website; “*Wizzy Digital Courier*”; Web reference:
<http://www.wizzy.org.za/article/articlestatic/19/1/2/>; 2007
- [Yang *et al*, 2004] H Yang, H Luo, F Ye, S Lu, L Zhang; “*Security in mobile ad hoc networks*”; Wireless Communications, IEEE; 2004.
- [Yang *et al*, 2005] Chun-Chuan Yang and Li-Pin Tseng; “*Fisheye Zone Routing Protocol for Mobile Ad Hoc Networks*”; Multimedia and Communications Laboratory Department of Computer Science and Information Engineering National Chi Nan University, Taiwan; January 2005.
- [Zhu *et al*, 2004] Yufang Zhu and Thomas Kunz; “*MAODV Implementation for NS-2.26 Systems and Computer Engineering*”; Carleton University, Technical Report SCE-04-01; January 2004.
- [Ziegast, 1997] Eric Ziegast; “*The UUCP Project*”; Web reference:
<http://www.uucp.org/info.shtml>; 1997.

Appendix A: OCI Companion DVD

A DVD was made to host material related to this research, which includes an electronic version of the thesis, video demonstrations of the OCI-SIM environment, a software release of OCI-SIM, source code, references and reading material, and data generated through this work.

Appendix B: OCI-SIM Structure Organization

The following chart shows the OCI-SIM modular design layout for each of the sub-components.

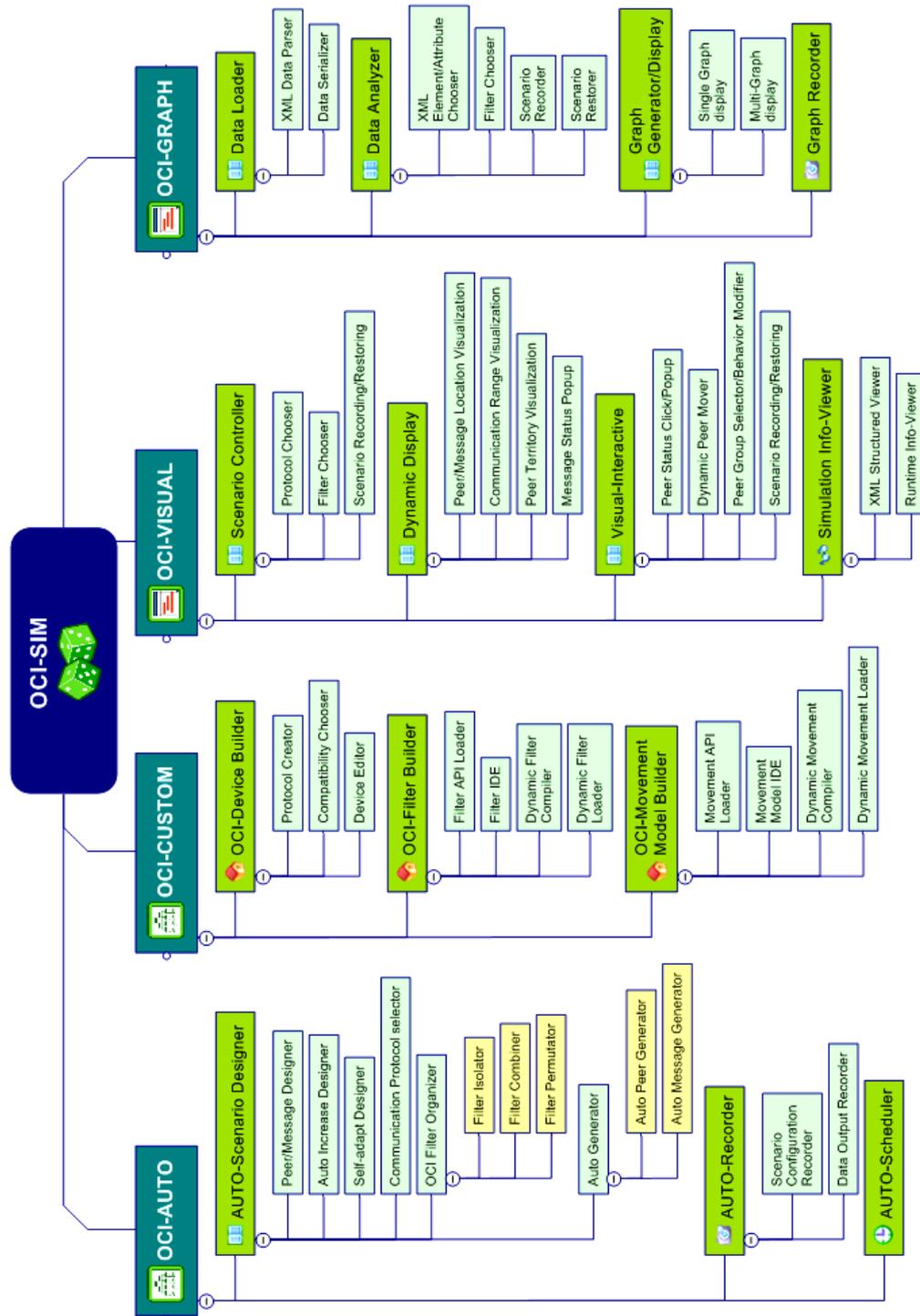


FIGURE B-1: OCI-SIM STRUCTURE ORGANIZATION MAP

Appendix C: Weight Assignment

C.1 OCI for “Custody Transfer Propagation”

“CUSTODY TRANSFER PROPAGATION”				
	Delivery Rate	Delivery Time	Routing Depth	Message Duplication
Control	93.75	209.86	17.44	2.38
Consistency	63.5	254.30	3.50	0.00
Multi-Adaptor	86	243.70	11.21	0.15
Range	73.375	227.33	4.40	0.00
Reputation	74.25	229.61	5.56	0.06
Territory	75.25	242.94	3.46	0.03
Velocity	93.75	252.1375	9.9375	0.2875

TABLE C-1: CUSTODY TRANSFER PROPAGATION

IMPROVEMENT / TRADE-OFFS PERCENTAGE - “CUSTODY TRANSFER PROPAGATION”				
	Delivery Rate	Delivery Time	Routing Depth	Duplication
Control	0.00%	0.00%	0.00%	0.00%
Consistency	-32.27%	-21.17%	79.93%	100.00%
Multi-Adaptor	-8.27%	-16.12%	35.70%	93.68%
Range	-21.73%	-8.32%	74.77%	100.00%
Reputation	-20.80%	-9.41%	68.10%	97.37%
Territory	-19.73%	-15.76%	80.14%	98.95%
Velocity	0.00%	-20.14%	43.01%	87.89%

TABLE C-2: IMPROVEMENT / TRADE-OFF PERCENTAGE - “CUSTODY TRANSFER PROPAGATION”

ABSOLUTE INDEX - “CUSTODY TRANSFER PROPAGATION”					
Absolute Reliability		Absolute Performance		Absolute Efficiency	
Weight	OCI	Weight	OCI	Weight	OCI
0.00%	Velocity	-8.32%	Range	179.93%	Consistency
-8.27%	Multi-Adaptor	-9.41%	Reputation	179.09%	Territory
-19.73%	Territory	-15.76%	Territory	174.77%	Range
-20.80%	Reputation	-16.12%	Multi-Adaptor	165.47%	Reputation
-21.73%	Range	-20.14%	Velocity	130.91%	Velocity
-32.27%	Consistency	-21.17%	Consistency	129.38%	Multi-Adaptor

TABLE C-3: ABSOLUTE INDEX - “CUSTODY TRANSFER PROPAGATION”

EMPHASIZED BALANCING INDEX- “CUSTODY TRANSFER PROPAGATION”					
Reliability Emphasized Balancing		Performance Emphasized Balancing		Efficiency Emphasized Balancing	
Weight	OCI	Weight	OCI	Weight	OCI
45.31%	Velocity	49.01%	Range	144.71%	Range
35.60%	Range	43.11%	Reputation	143.60%	Territory
34.32%	Territory	38.29%	Territory	135.26%	Reputation
32.03%	Multi-Adaptor	25.16%	Velocity	126.49%	Consistency
31.72%	Reputation	24.18%	Multi-Adaptor	110.76%	Velocity
4.26%	Consistency	15.35%	Consistency	104.99%	Multi-Adaptor

TABLE C-4: EMPHASIZED BALANCING INDEX- “CUSTODY TRANSFER PROPAGATION”

BALANCED INDEX- “CUSTODY TRANSFER PROPAGATION”	
Weight	OCI
57.33%	Range
54.05%	Territory
52.52%	Reputation
45.31%	Velocity
40.30%	Multi-Adaptor
36.52%	Consistency

TABLE C-5: BALANCED INDEX- “CUSTODY TRANSFER PROPAGATION”

C.2 OCI for “Custody Retaining Propagation”

“CUSTODY RETAINING PROPAGATION”				
	Delivery Rate	Delivery Time	Routing Depth	Message Duplication
Control	99.25	91.54	9.13	9.09
Consistency	95.75	132.62	5.25	5.54
Multi-Adaptor	99.5	99.99	6.84	6.08
Range	98.625	111.64	5.14	5.11
Reputation	97.75	115.59	5.95	6.23
Territory	98	120.75	4.98	5.48
Velocity	99.875	92.08	5.95	5.78

TABLE C-6: “CUSTODY RETAINING PROPAGATION”

IMPROVEMENT / TRADE-OFFS PERCENTAGE - "CUSTODY RETAINING PROPAGATION"				
	Delivery Rate	Delivery Time	Routing Depth	Duplication
Control	0.00%	0.00%	0.00%	0.00%
Consistency	-3.53%	-44.89%	42.47%	39.06%
Multi-Adaptor	0.25%	-9.23%	25.07%	33.15%
Range	-0.63%	-21.96%	43.70%	43.74%
Reputation	-1.51%	-26.27%	34.79%	31.50%
Territory	-1.26%	-31.91%	45.48%	39.75%
Velocity	0.63%	-0.59%	34.79%	36.45%

TABLE C-7: IMPROVEMENT / TRADE-OFFS PERCENTAGE - "CUSTODY RETAINING PROPAGATION"

ABSOLUTE INDEX - "CUSTODY RETAINING PROPAGATION"					
Absolute Reliability		Absolute Performance		Absolute Efficiency	
Weight	OCI	Weight	OCI	Weight	OCI
0.63%	Velocity	-0.59%	Velocity	87.44%	Range
0.25%	Multi-Adaptor	-9.23%	Multi-Adaptor	85.23%	Territory
-0.63%	Range	-21.96%	Range	81.53%	Consistency
-1.26%	Territory	-26.27%	Reputation	71.25%	Velocity
-1.51%	Reputation	-31.91%	Territory	66.29%	Reputation
-3.53%	Consistency	-44.89%	Consistency	58.22%	Multi-Adaptor

TABLE C-8: EMPHASIZED BALANCING INDEX- "CUSTODY RETAINING PROPAGATION"

EMPHASIZED BALANCING INDEX- "CUSTODY RETAINING PROPAGATION"					
Reliability		Performance		Efficiency	
Emphasized Balancing		Emphasized Balancing		Emphasized Balancing	
Weight	OCI	Weight	OCI	Weight	OCI
36.30%	Velocity	35.08%	Velocity	71.29%	Velocity
20.50%	Range	10.90%	Multi-Adaptor	64.85%	Range
20.38%	Multi-Adaptor	-0.83%	Range	52.06%	Territory
8.18%	Territory	-20.91%	Reputation	49.24%	Multi-Adaptor
3.85%	Reputation	-22.47%	Territory	38.51%	Reputation
-11.17%	Consistency	-52.53%	Consistency	33.12%	Consistency

TABLE C-9: EMPHASIZED BALANCING INDEX- "CUSTODY RETAINING PROPAGATION"

BALANCED INDEX- "CUSTODY RETAINING PROPAGATION"	
Weight	OCI
35.67%	Velocity
21.13%	Range
20.13%	Multi-Adaptor
9.44%	Territory
5.36%	Reputation
-7.65%	Consistency

TABLE C-10: BALANCED INDEX- "CUSTODY RETAINING PROPAGATION"

C.3 OCI for "Simple Flood Propagation"

"SIMPLE FLOOD PROPAGATION"				
	Delivery Rate	Delivery Time	Routing Depth	Message Duplication
Control	100	27.5875	8.225	41.0875
Consistency	96.75	89.4	5.1625	13.225
Multi-Adaptor	100	38.3	6.375	21.1875
Range	99.25	61.475	5.1375	13.4875
Reputation	99.75	60.95	5.75	15.975
Territory	99.5	84.5375	4.6	11.3125
Velocity	100	48.375	5.6875	21.8625

TABLE C-11: "SIMPLE FLOOD PROPAGATION"

IMPROVEMENT / TRADE-OFFS PERCENTAGE - "SIMPLE FLOOD PROPAGATION"				
	Delivery Rate	Delivery Time	Routing Depth	Message Duplication
Control	0.00%	0.00%	0.00%	0.00%
Consistency	-3.25%	-224.06%	37.23%	67.81%
Multi-Adaptor	0.00%	-38.83%	22.49%	48.43%
Range	-0.75%	-122.84%	37.54%	67.17%
Reputation	-0.25%	-120.93%	30.09%	61.12%
Territory	-0.50%	-206.43%	44.07%	72.47%
Velocity	0.00%	-75.35%	30.85%	46.79%

TABLE C-12: IMPROVEMENT / TRADE-OFF PERCENTAGE - "SIMPLE FLOOD PROPAGATION"

ABSOLUTE INDEX - "SIMPLE FLOOD PROPAGATION"					
Absolute Reliability		Absolute Performance		Absolute Efficiency	
Weight	OCI	Weight	OCI	Weight	OCI
0.00%	Multi-Adaptor	-38.83%	Multi-Adaptor	116.54%	Territory
0.00%	Velocity	-75.35%	Velocity	105.05%	Consistency
-0.25%	Reputation	-120.93%	Reputation	104.71%	Range
-0.50%	Territory	-122.84%	Range	91.21%	Reputation
-0.75%	Range	-206.43%	Territory	77.64%	Velocity
-3.25%	Consistency	-224.06%	Consistency	70.93%	Multi-Adaptor

TABLE C-13: ABSOLUTE INDEX - "SIMPLE FLOOD PROPAGATION"

EMPHASIZED BALANCING INDEX- "SIMPLE FLOOD PROPAGATION"					
Reliability Emphasized Balancing		Performance Emphasized Balancing		Efficiency Emphasized Balancing	
Weight	OCI	Weight	OCI	Weight	OCI
-3.37%	Multi-Adaptor	-42.20%	Multi-Adaptor	32.09%	Multi-Adaptor
-36.53%	Velocity	-111.88%	Velocity	2.29%	Velocity
-71.98%	Range	-194.07%	Range	-18.87%	Range
-75.83%	Reputation	-196.51%	Reputation	-29.97%	Reputation
-149.16%	Territory	-355.10%	Territory	-90.39%	Territory
-178.04%	Consistency	-398.85%	Consistency	-122.26%	Consistency

TABLE C-14: EMPHASIZED BALANCING INDEX- "SIMPLE FLOOD PROPAGATION"

BALANCED INDEX- "SIMPLE FLOOD PROPAGATION"	
Weight	OCI
-3.37%	Multi-Adaptor
-36.53%	Velocity
-71.23%	Range
-75.58%	Reputation
-148.66%	Territory
-174.79%	Consistency

TABLE C-15: BALANCED INDEX- "SIMPLE FLOOD PROPAGATION"

Appendix D: Element Selection

D.1 "Reputation Element" and its effects on message delivery

1) Peer routing reputation and its effects on message delivery rate

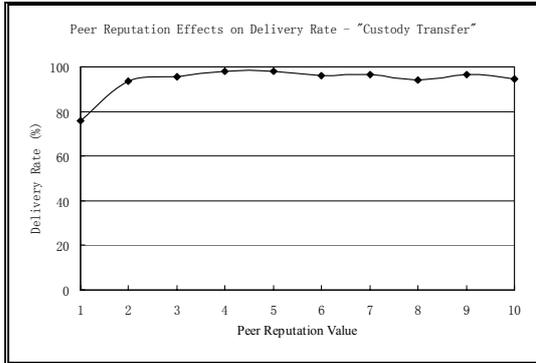


FIGURE D-1: CUSTODY TRANSFER

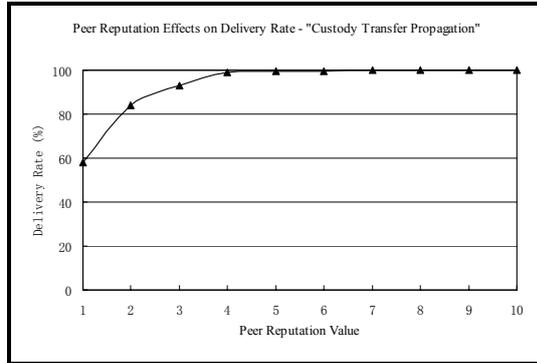


FIGURE D-2: CUSTODY TRANSFER PROPAGATION

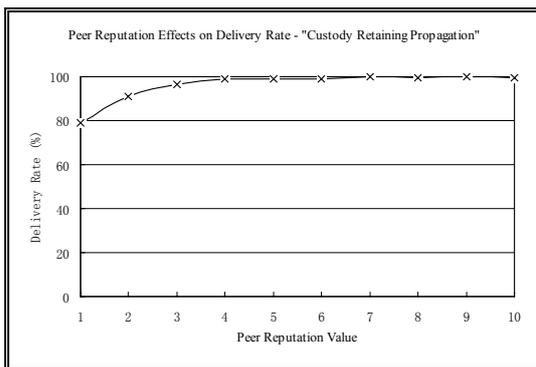


FIGURE D-3: CUSTODY RETAINING PROPAGATION

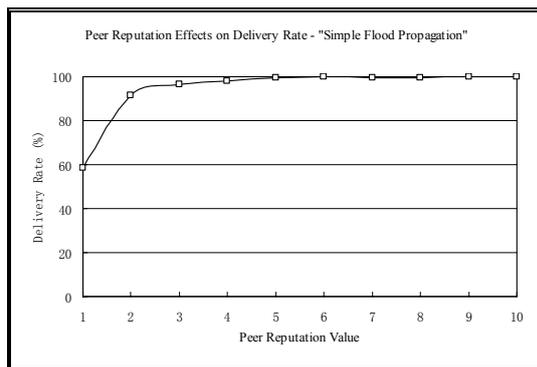


FIGURE D-4: SIMPLE FLOOD PROPAGATION

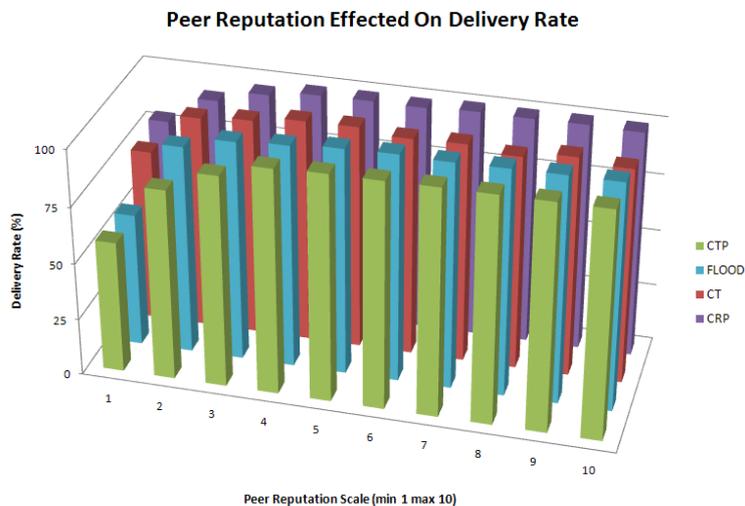


FIGURE D-5: PEER ROUTING REPUTATION AND ITS EFFECTS ON MESSAGE DELIVERY RATE

2) Peer routing reputation and its effects on message delivery time

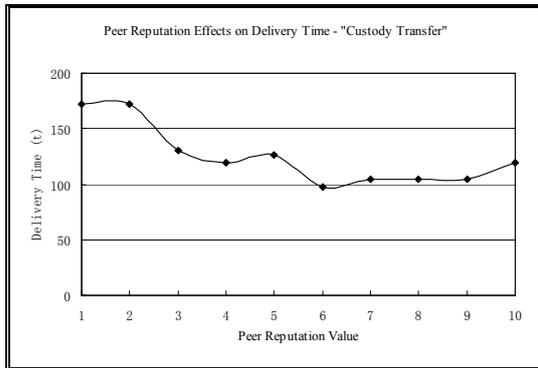


FIGURE D-6: CUSTODY TRANSFER

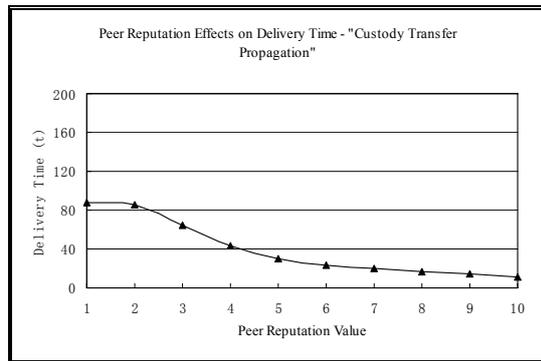


FIGURE D-7: CUSTODY TRANSFER PROPAGATION

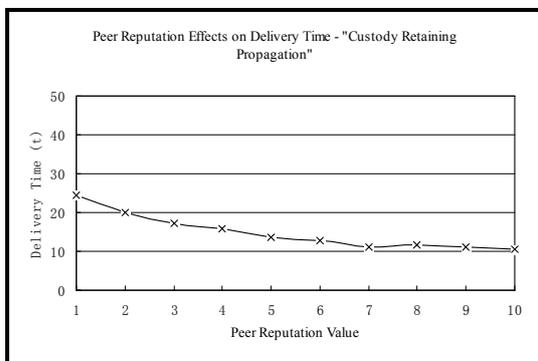


FIGURE D-8: CUSTODY RETAINING PROPAGATION

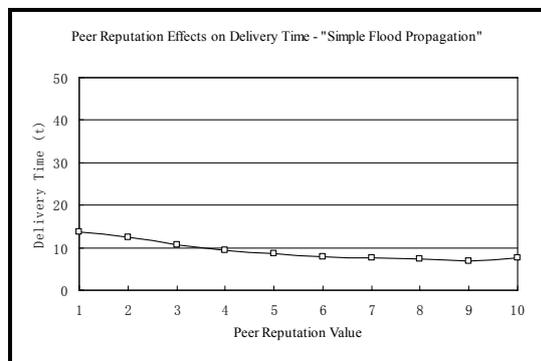


FIGURE D-9: SIMPLE FLOOD PROPAGATION

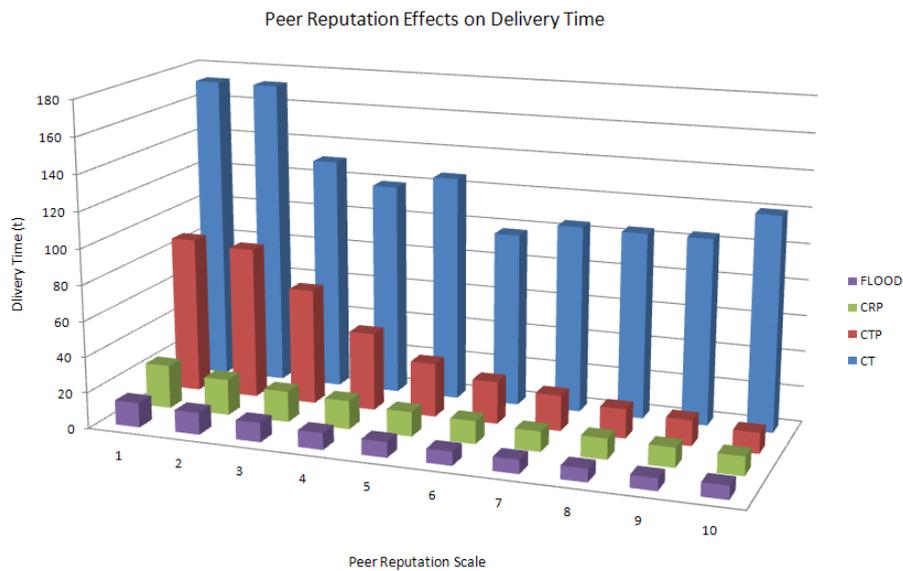


FIGURE D-10: PEER ROUTING REPUTATION AND ITS EFFECTS ON MESSAGE DELIVERY TIME

3) Peer routing reputation and its effects on message delivery routing depth

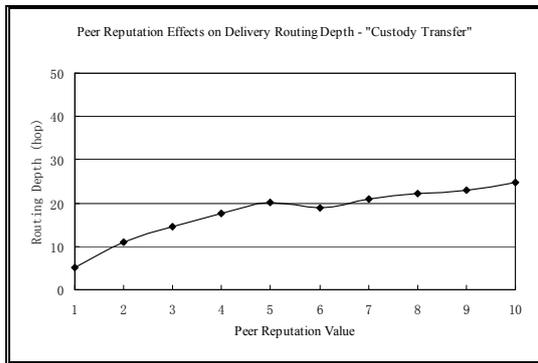


FIGURE D-11: CUSTODY TRANSFER

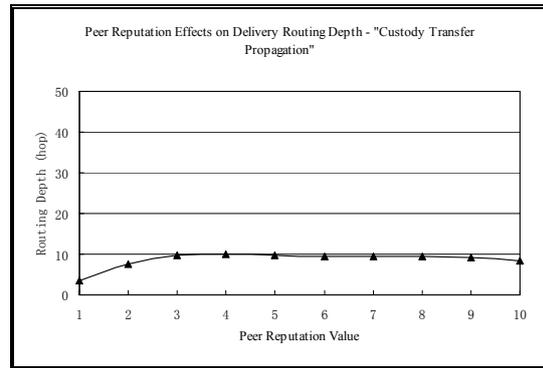


FIGURE D-12: CUSTODY TRANSFER PROPAGATION

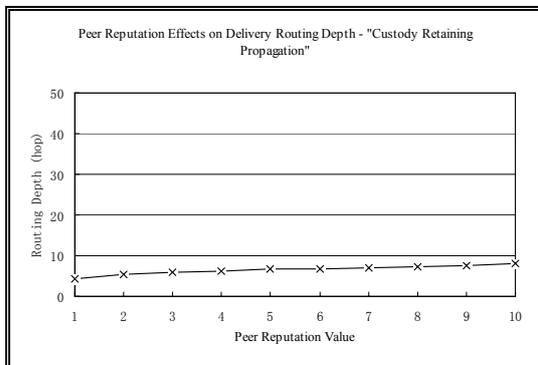


FIGURE D-13: CUSTODY RETAINING PROPAGATION

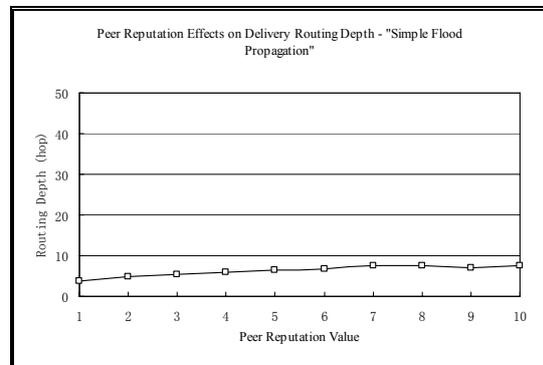


FIGURE D-14: SIMPLE FLOOD PROPAGATION

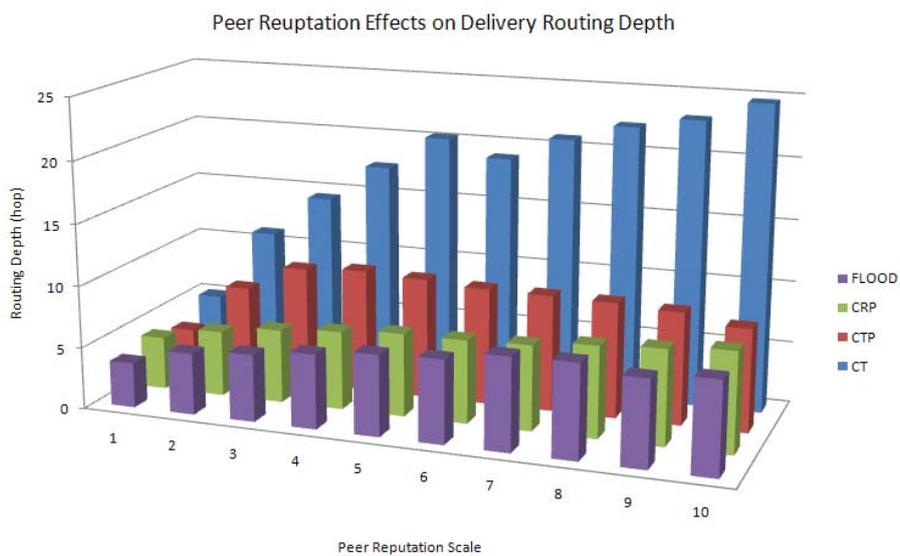


FIGURE D-15: REPUTATION AND ITS EFFECTS ON ROUTING DEPTH

4) Peer routing reputation and its effects on message duplication

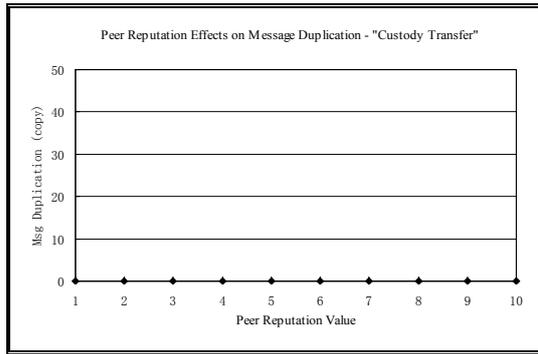


FIGURE D-16: CUSTODY TRANSFER

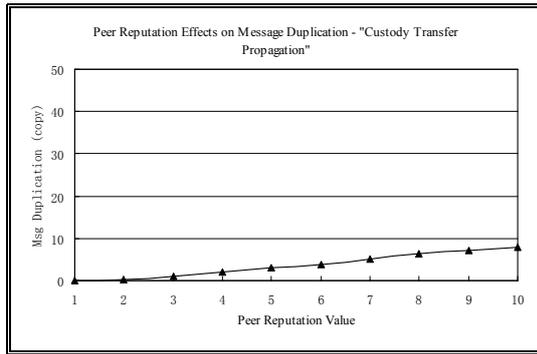


FIGURE D-17: CUSTODY TRANSFER PROPAGATION

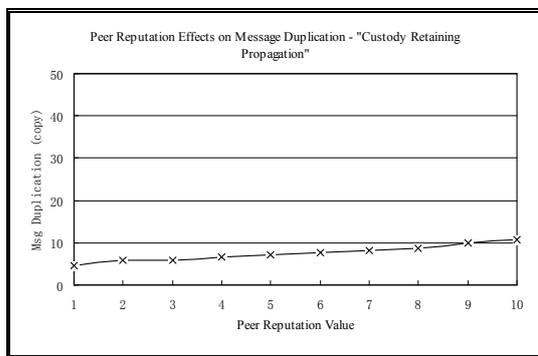


FIGURE D-18: CUSTODY RETAINING PROPAGATION

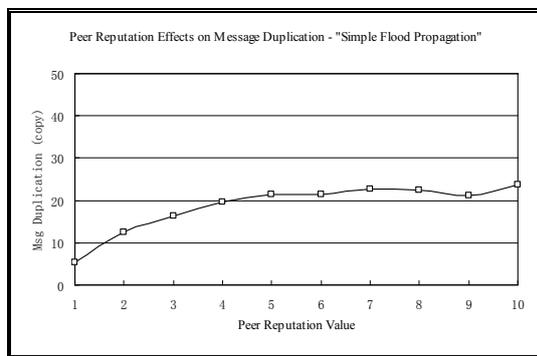


FIGURE D-19: SIMPLE FLOOD PROPAGATION

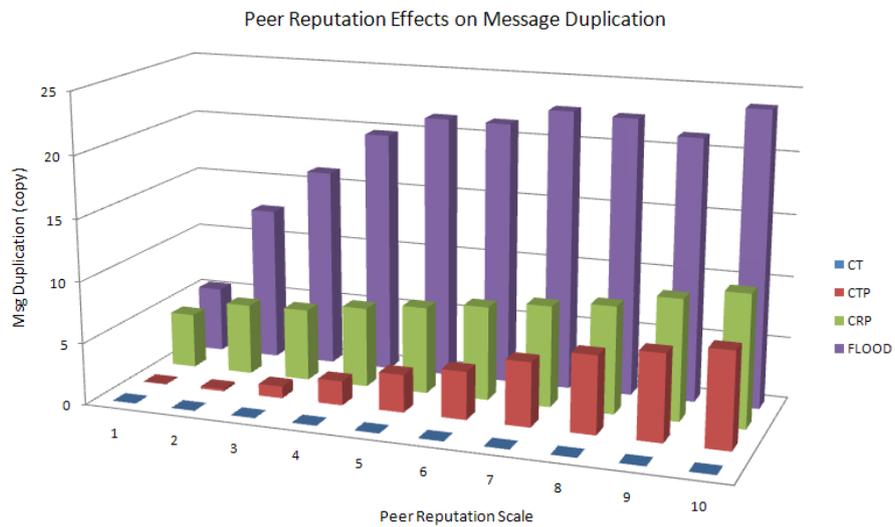


FIGURE D-20: REPUTATION AND ITS EFFECTS ON DUPLICATION

D.2 "Transmission Range Element" and its effects on delivery

1) Transmission range and its effects on message delivery rate



FIGURE D-21: CUSTODY TRANSFER

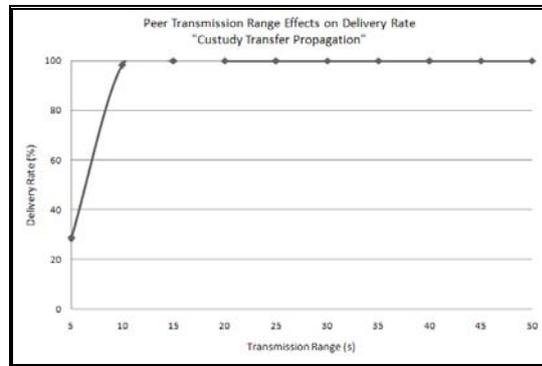


FIGURE D-22: CUSTODY TRANSFER PROPAGATION

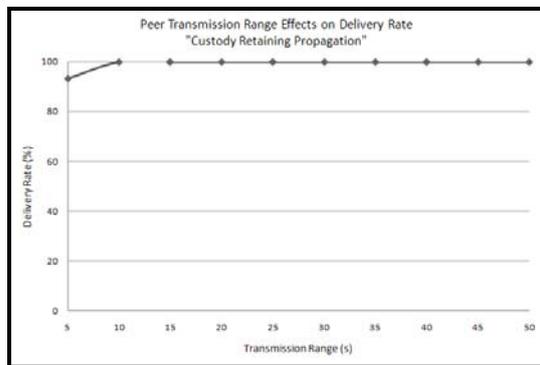


FIGURE D-23: CUSTODY RETAINING PROPAGATION

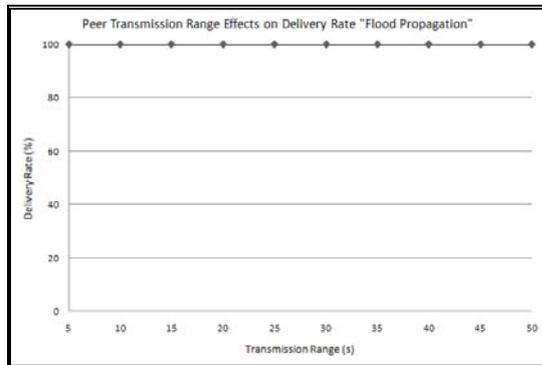


FIGURE D-24: SIMPLE FLOOD PROPAGATION

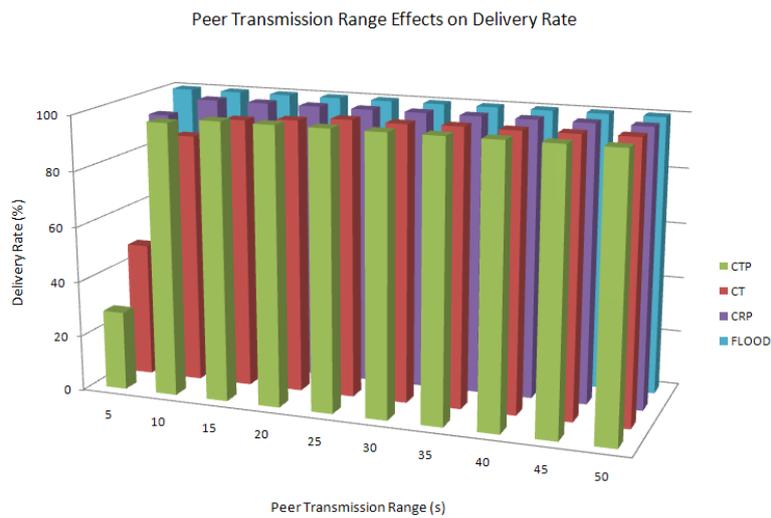


FIGURE D-25: RANGE AND ITS EFFECTS ON DELIVERY RATE

2) Transmission range and its effects on message delivery time

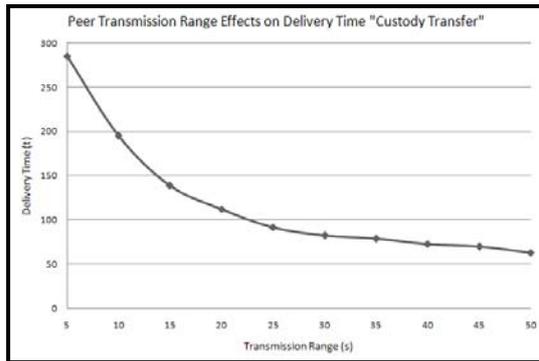


FIGURE D-26: CUSTODY TRANSFER

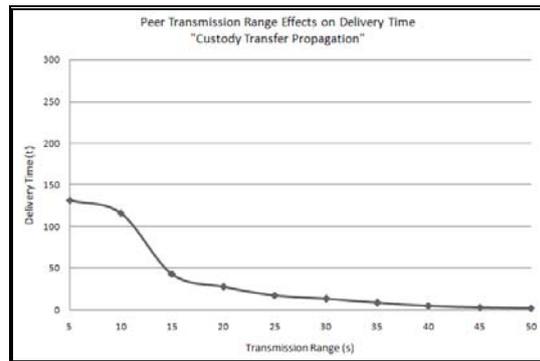


FIGURE D-27: CUSTODY TRANSFER PROPAGATION



FIGURE D-28: CUSTODY RETAINING PROPAGATION

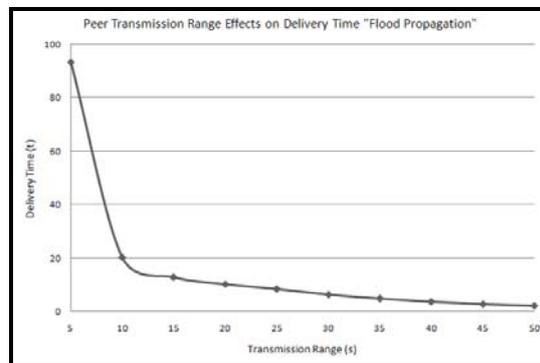


FIGURE D-29: SIMPLE FLOOD PROPAGATION

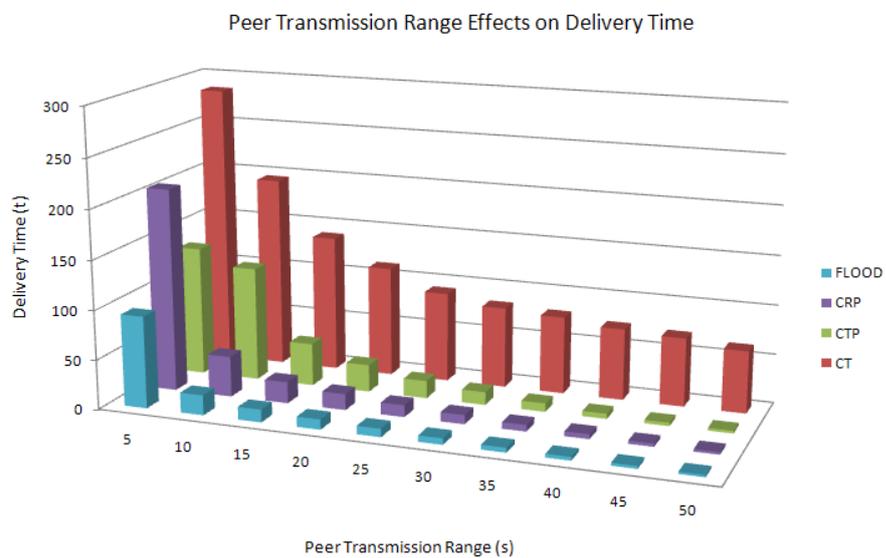


FIGURE D-30: REPUTATION AND ITS EFFECTS ON DELIVERY RATE

3) Transmission range and its effects on message delivery routing depth

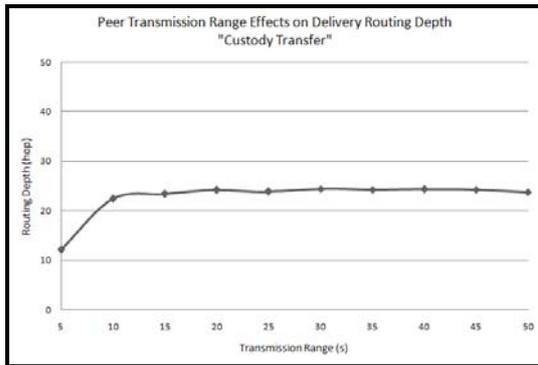


FIGURE D-31: CUSTODY TRANSFER

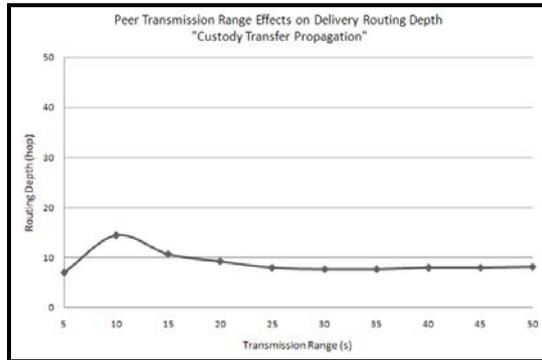


FIGURE D-32: CUSTODY TRANSFER PROPAGATION

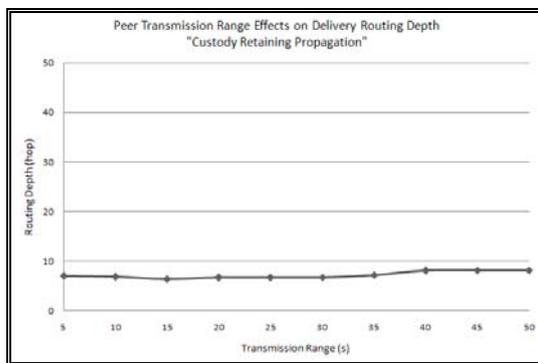


FIGURE D-33: CUSTODY RETAINING PROPAGATION

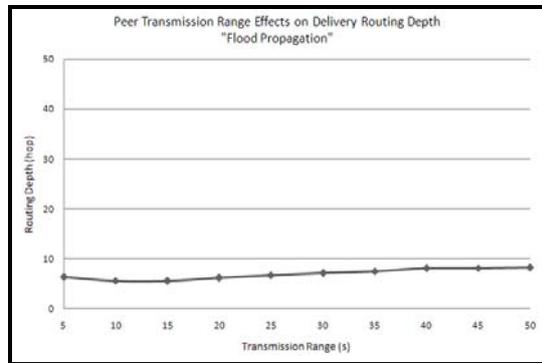


FIGURE D-34: SIMPLE FLOOD PROPAGATION

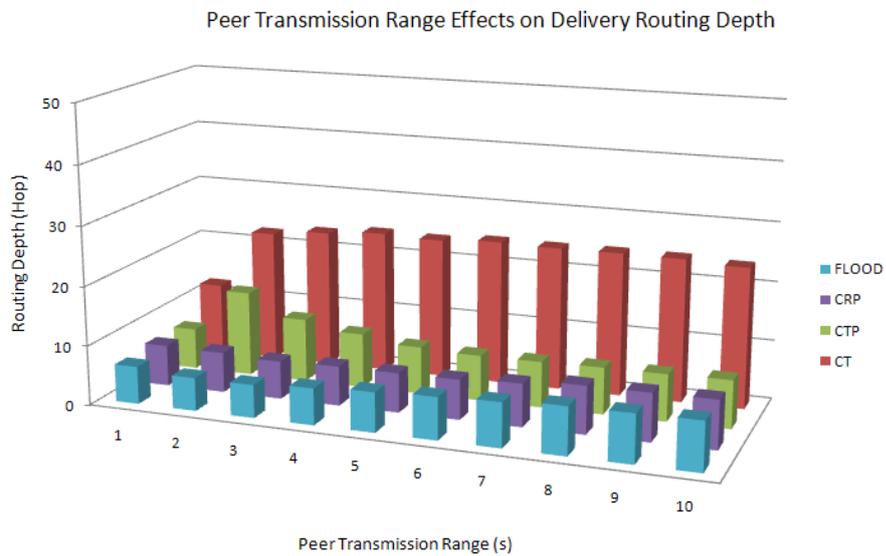


FIGURE D-35: RANGE AND ITS EFFECTS ON ROUTING DEPTH

4) Transmission range and its effects on message duplication

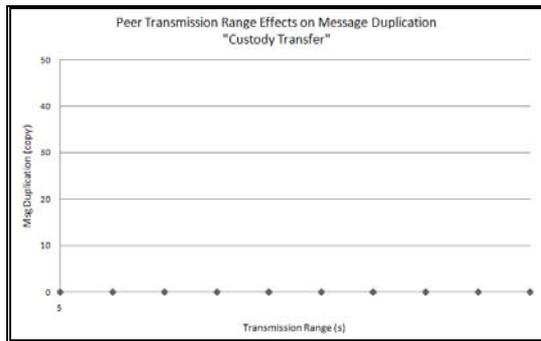


FIGURE D-36: CUSTODY TRANSFER

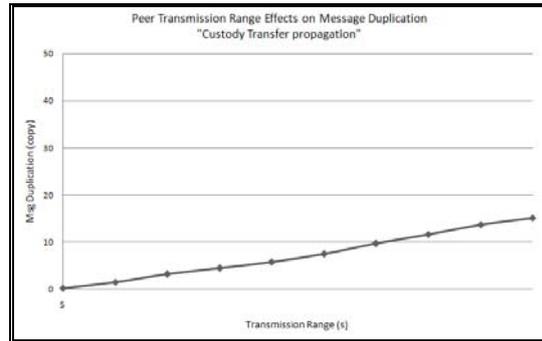


FIGURE D-37: CUSTODY TRANSFER PROPAGATION

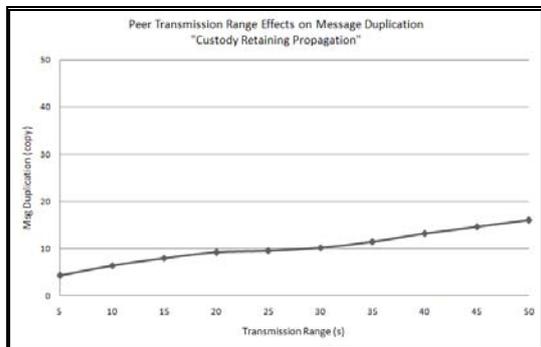


FIGURE D-38: CUSTODY RETAINING PROPAGATION

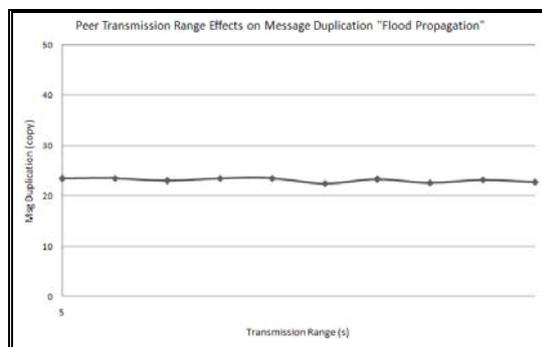


FIGURE D-39: SIMPLE FLOOD PROPAGATION

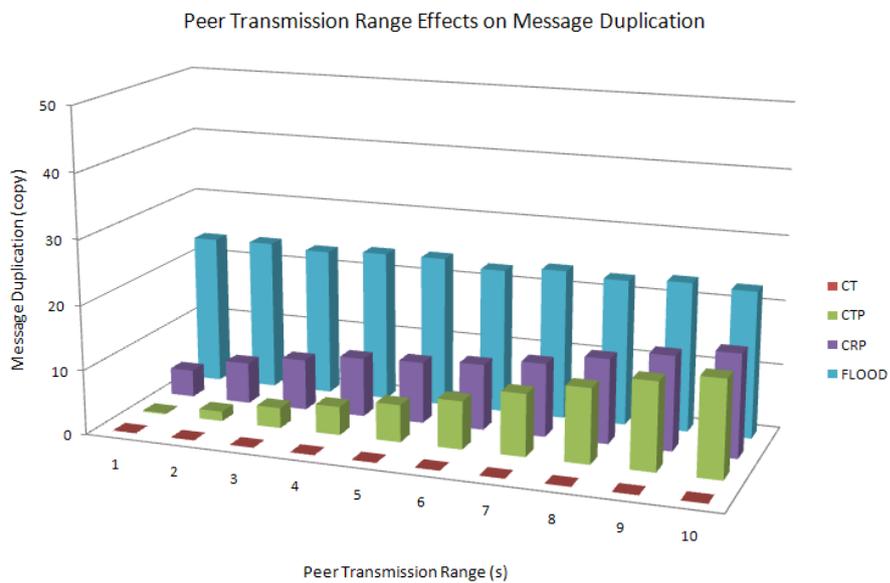


FIGURE D-40: RANGE AND ITS EFFECTS ON MESSAGE DUPLICATION

D.3 “Multi-adaptor Element” and its effects

1) Multi-adaptor and its effects on message delivery rate

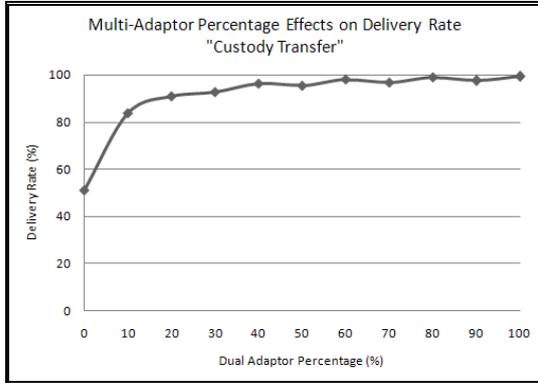


FIGURE D-41: CUSTODY TRANSFER

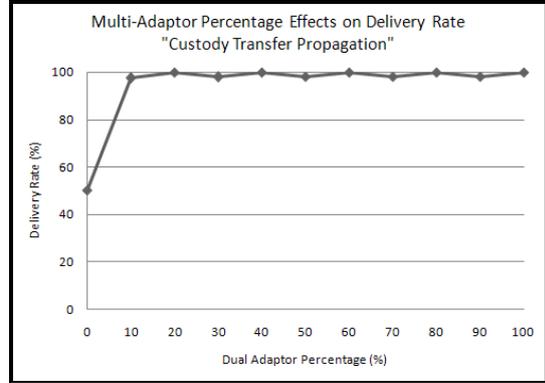


FIGURE D-42: CUSTODY TRANSFER PROPAGATION

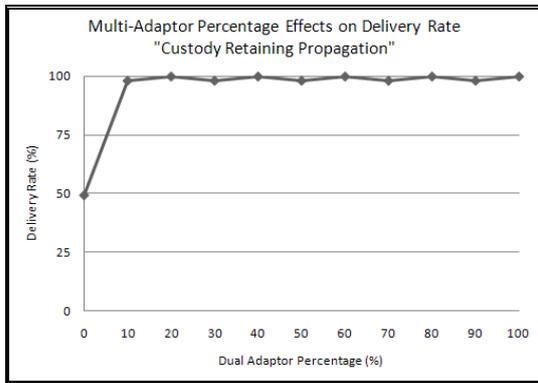


FIGURE D-43: CUSTODY RETAINING PROPAGATION

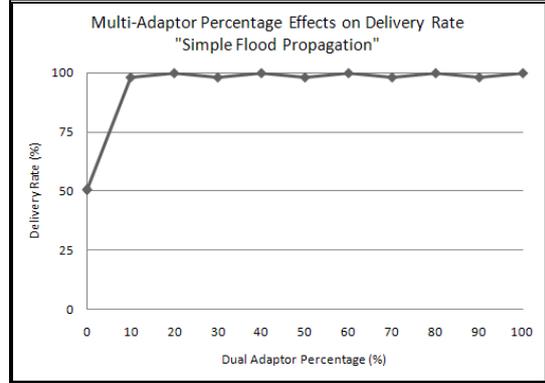


FIGURE D-44: SIMPLE FLOOD PROPAGATION

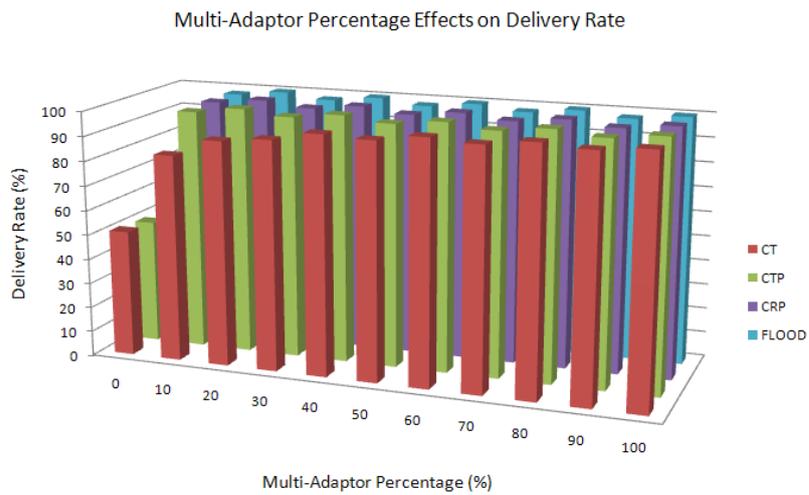


FIGURE D-45: MULTI-ADAPTOR AND ITS EFFECTS ON DELIVERY RATE

2) Multi-adaptor and its effects on message delivery time

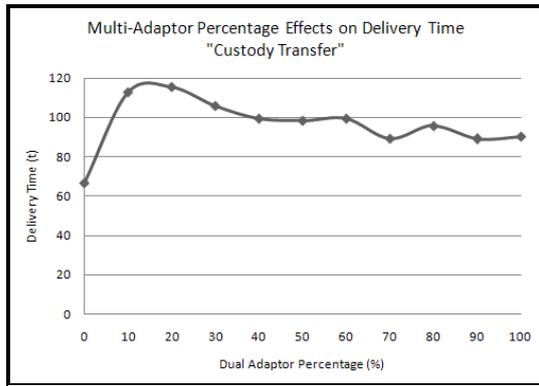


FIGURE D-46: CUSTODY TRANSFER

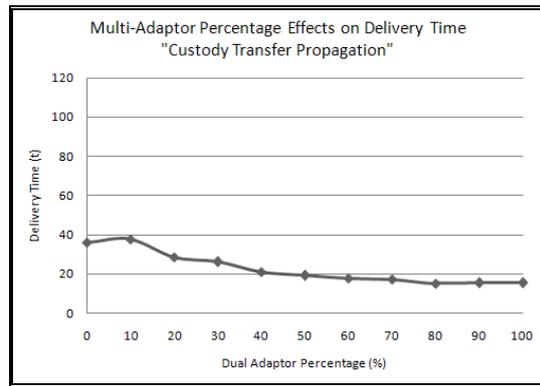


FIGURE D-47: CUSTODY TRANSFER PROPAGATION

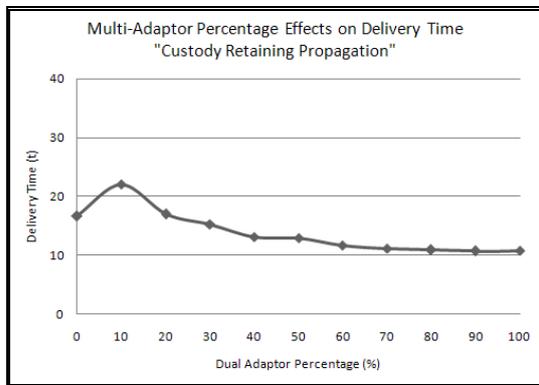


FIGURE D-48: CUSTODY RETAINING PROPAGATION

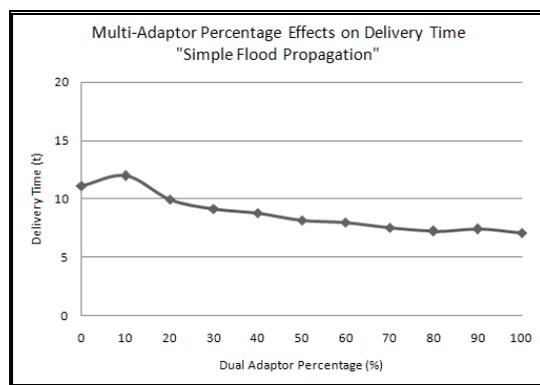


FIGURE D-49: SIMPLE FLOOD PROPAGATION

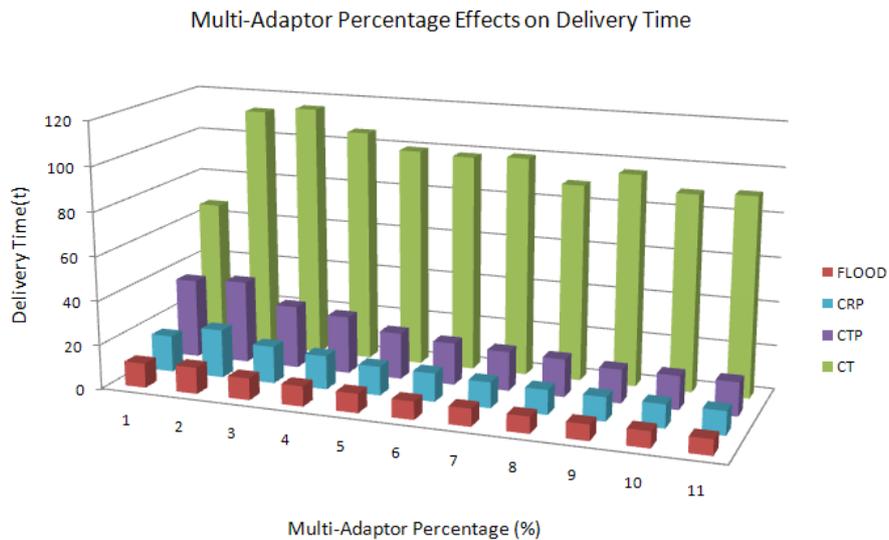


FIGURE D-50: MULTI-ADAPTOR AND ITS EFFECTS ON DELIVERY TIME

3) Multi-adaptor and its effects on message delivery routing depth

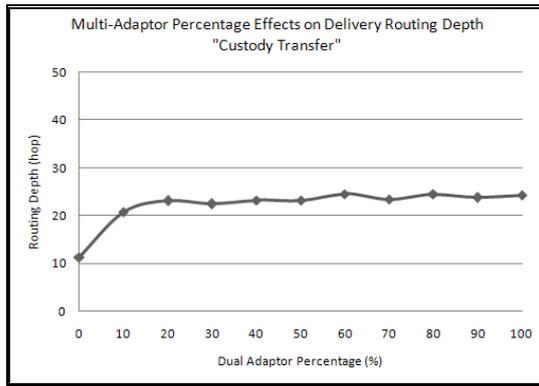


FIGURE D-51: CUSTODY TRANSFER

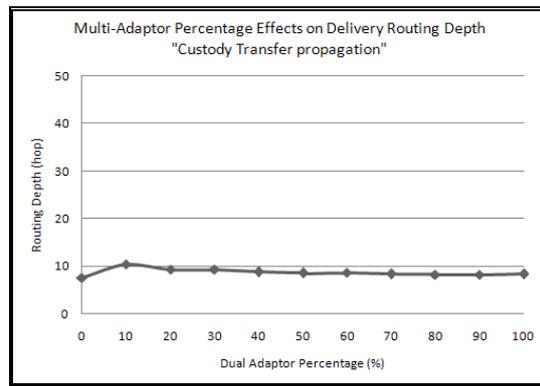


FIGURE D-52: CUSTODY TRANSFER PROPAGATION

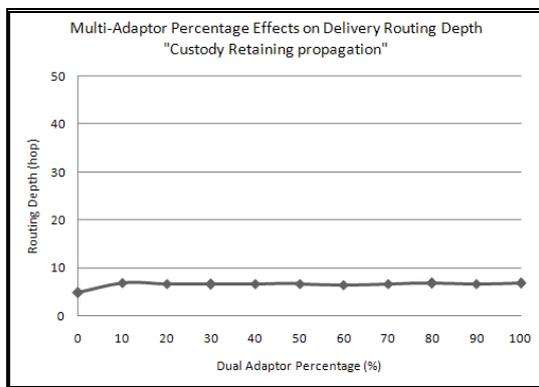


FIGURE D-53: CUSTODY RETAINING PROPAGATION

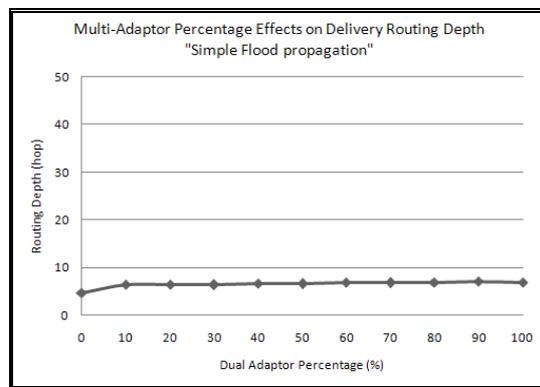


FIGURE D-54: SIMPLE FLOOD PROPAGATION

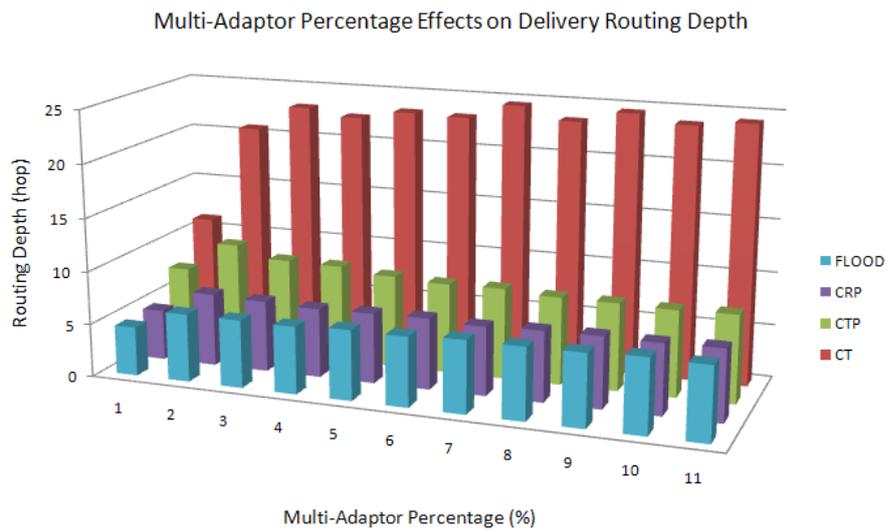


FIGURE D-55: MULTI-ADAPTOR AND ITS EFFECTS ON ROUTING DEPTH

4) Multi-adaptor and its effects on message duplication

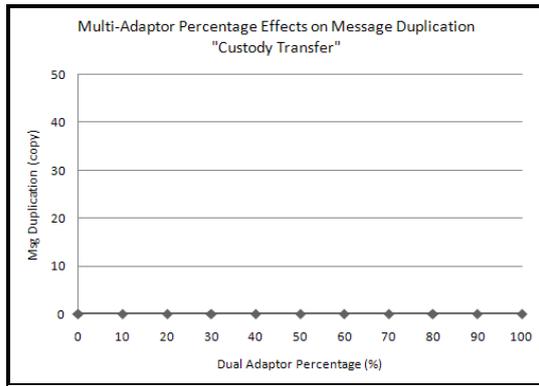


FIGURE D-56: CUSTODY TRANSFER

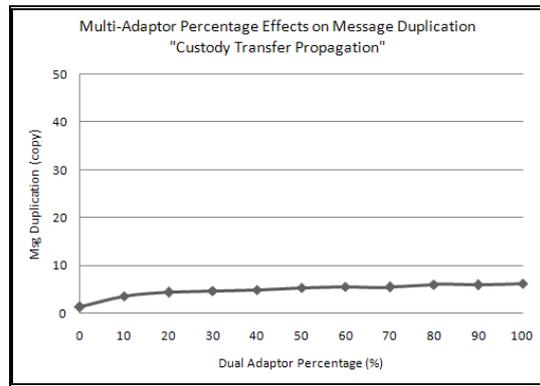


FIGURE D-57: CUSTODY TRANSFER PROPAGATION

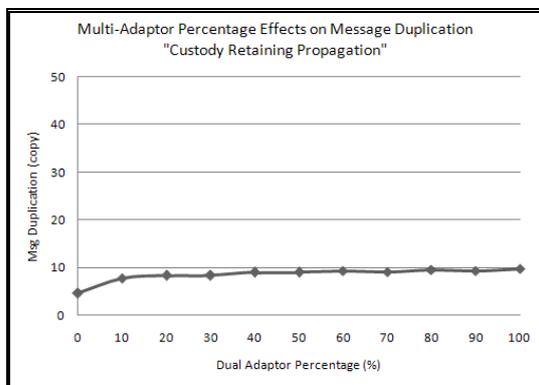


FIGURE D-58: CUSTODY RETAINING PROPAGATION

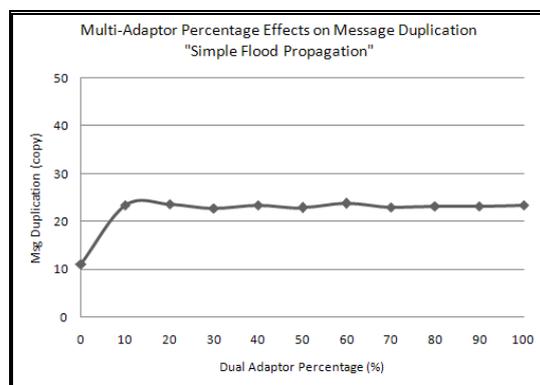


FIGURE D-59: SIMPLE FLOOD PROPAGATION

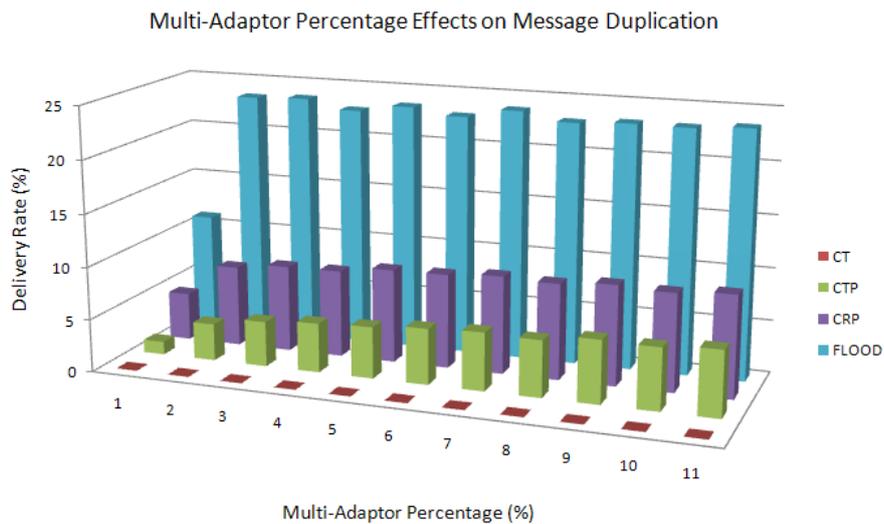


FIGURE D-60: MULTI-ADAPTOR AND ITS EFFECTS ON MESSAGE DUPLICATION

D.4 "Movement Consistency" and its Effects on Delivery

1) Peer consistency and its effects on message delivery rate

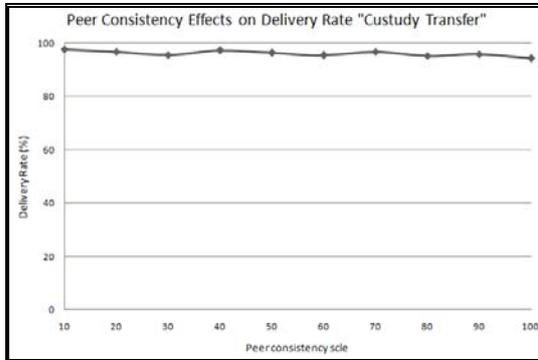


FIGURE D-61: CUSTODY TRANSFER

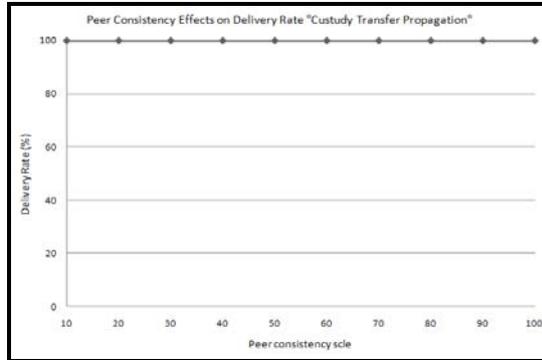


FIGURE D-62: CUSTODY TRANSFER PROPAGATION

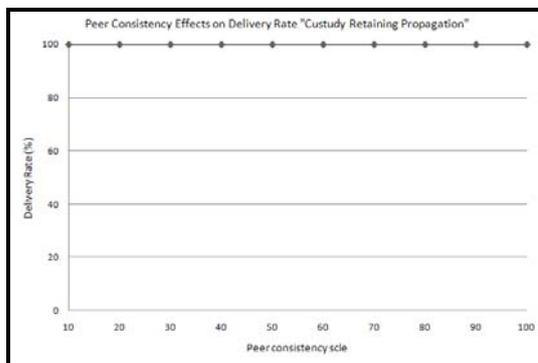


FIGURE D-63: CUSTODY RETAINING PROPAGATION

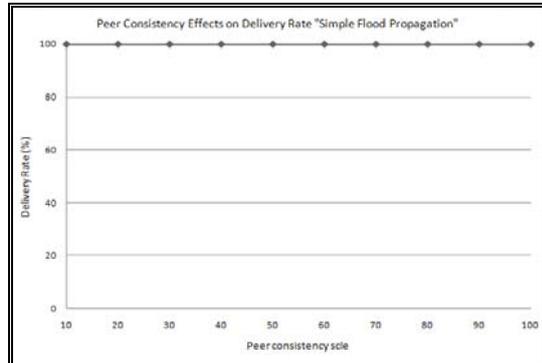


FIGURE D-64: SIMPLE FLOOD PROPAGATION

Peer Consistency Effects on Delivery Rate

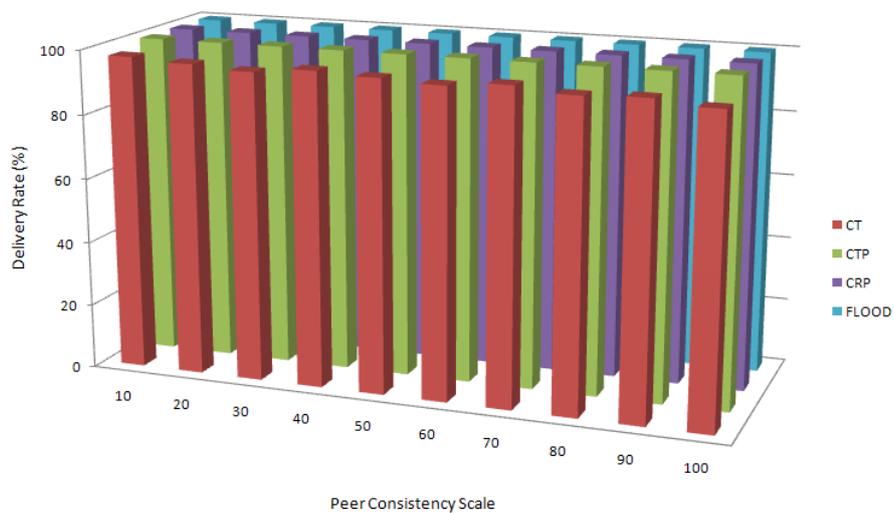


FIGURE D-65: CONSISTENCY AND ITS EFFECTS ON DELIVERY RATE

1) Peer consistency and its effects on message delivery time

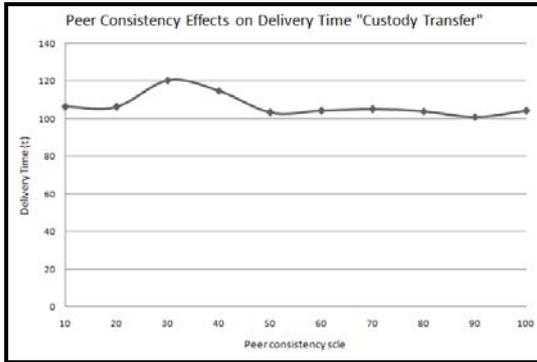


FIGURE D-66: CUSTODY TRANSFER

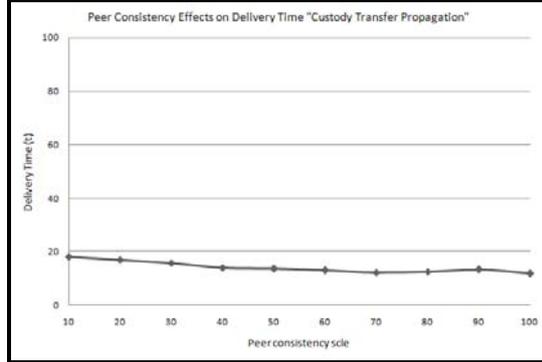


FIGURE D-67: CUSTODY TRANSFER PROPAGATION

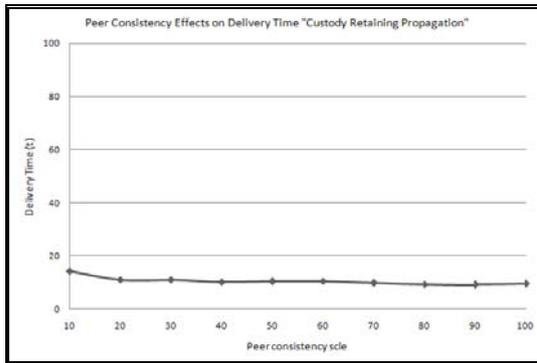


FIGURE D-68: CUSTODY RETAINING PROPAGATION

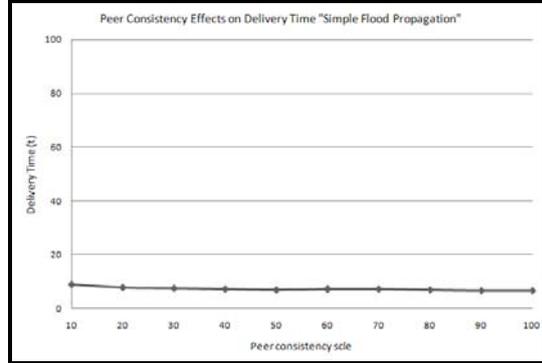


FIGURE D-69: SIMPLE FLOOD PROPAGATION

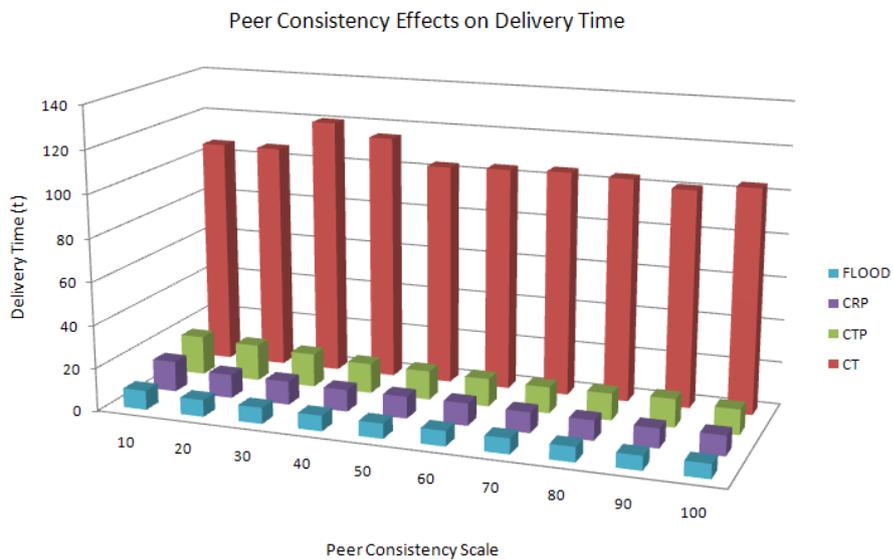


FIGURE D-70: CONSISTENCY AND ITS EFFECTS ON DELIVERY TIME

2) Peer consistency and its effects on message delivery routing depth

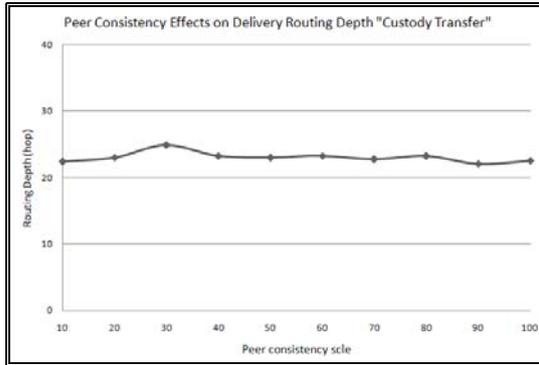


FIGURE D-71: CUSTODY TRANSFER

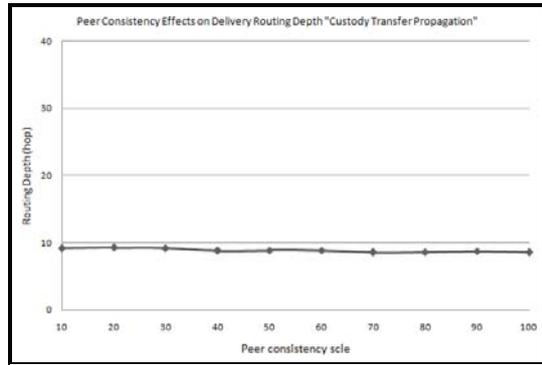


FIGURE D-72: CUSTODY TRANSFER PROPAGATION

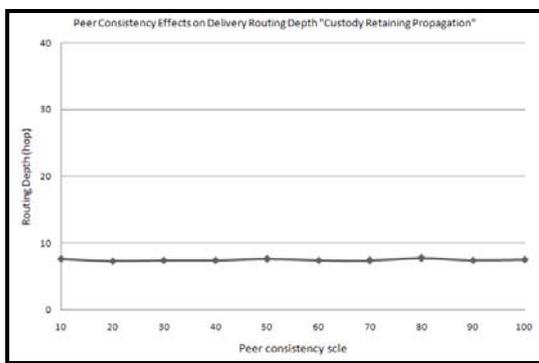


FIGURE D-73: CUSTODY RETAINING PROPAGATION

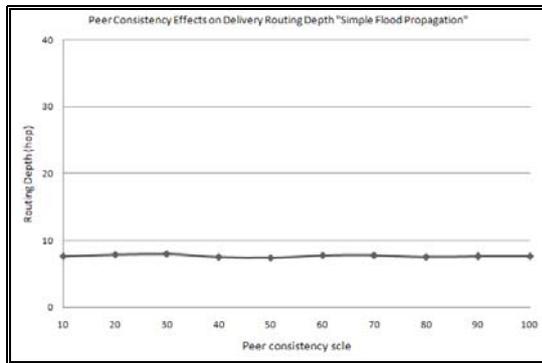


FIGURE D-74: SIMPLE FLOOD PROPAGATION

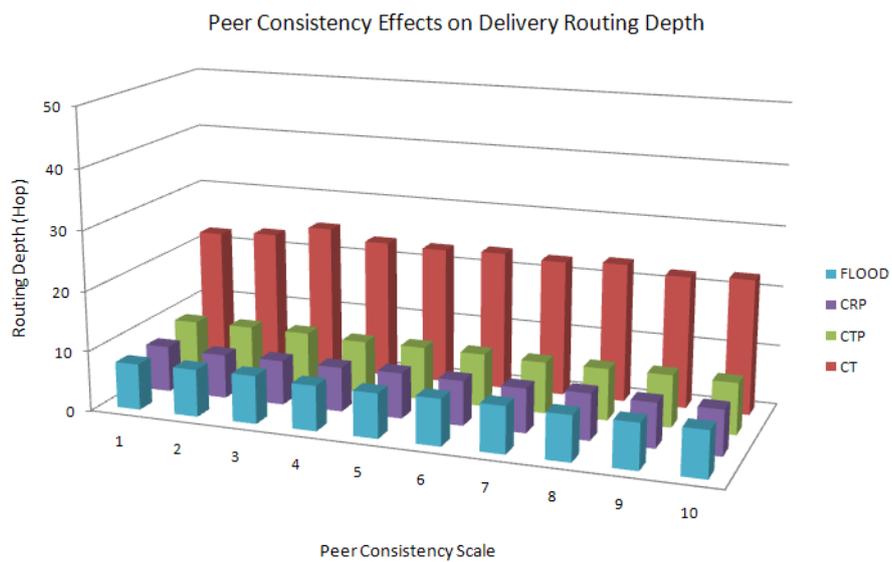


FIGURE D-75: CONSISTENCY AND ITS EFFECTS ON ROUTING DEPTH

3) Peer consistency and its effects on message duplication

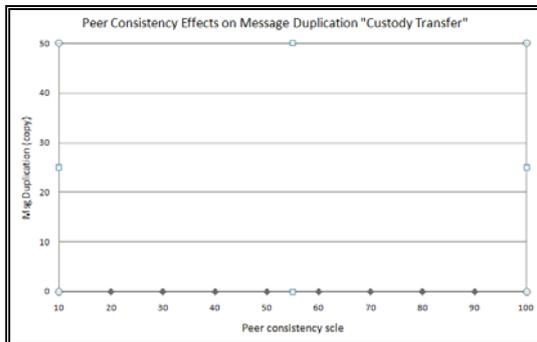


FIGURE D-76: CUSTODY TRANSFER

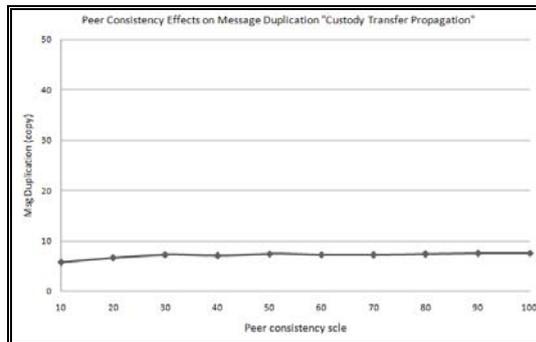


FIGURE D-77: CUSTODY TRANSFER PROPAGATION

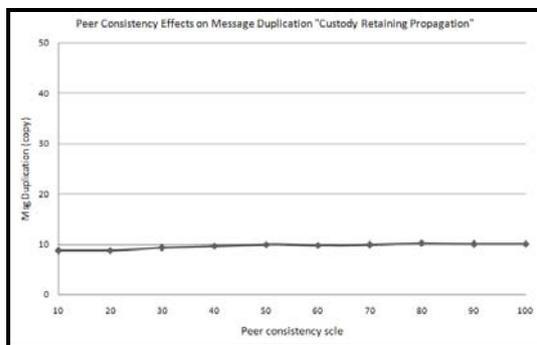


FIGURE D-78: CUSTODY RETAINING PROPAGATION

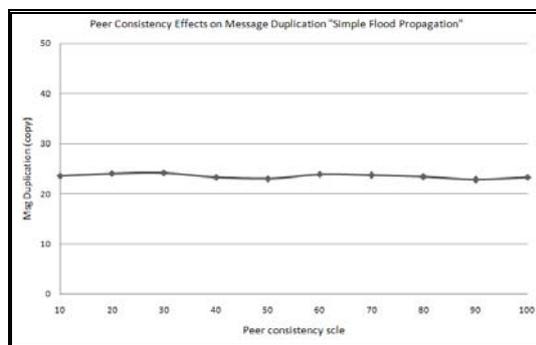


FIGURE D-79: SIMPLE FLOOD PROPAGATION

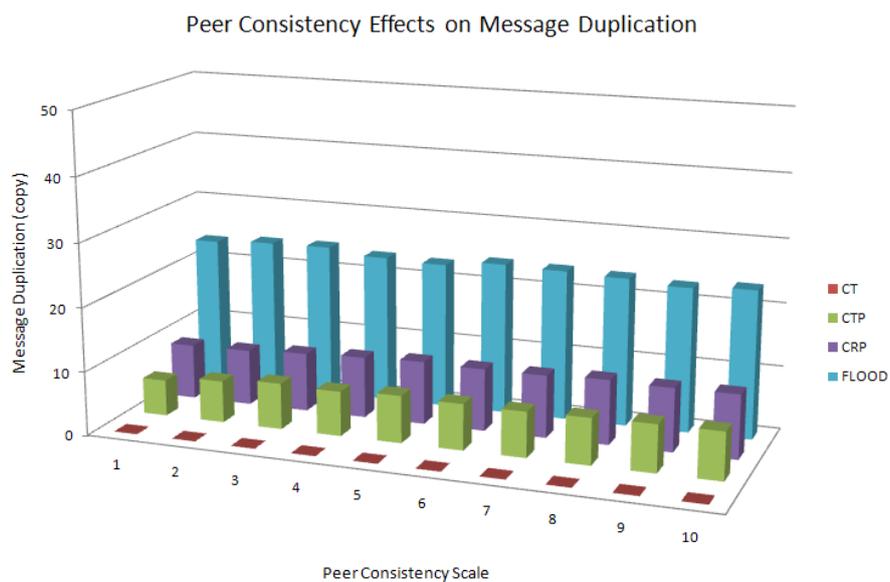


FIGURE D-80: CONSISTENCY AND ITS EFFECTS ON MESSAGE DUPLICATION

Appendix E: Element Importance Index

E.1 Performance emphasised importance Index

<u>VELOCITY</u>	PER UNIT IMPROVEMENT				<u>TERRITORY</u>	PER UNIT IMPROVEMENT			
Value	CT	CTP	CRP	Flood	Value	CT	CTP	CRP	Flood
20	1.72	1.99	0.96	0.54	100	0.65	-0.18	-0.06	-0.05
30	0.63	0.53	0.42	0.21	150	0.61	-0.04	-0.06	-0.06
40	0.05	0.44	0.16	0.13	200	0.45	0.13	-0.06	-0.02
50	0.31	0.07	0.16	0.10	250	-0.08	-0.08	-0.04	-0.03
60	-0.03	0.09	0.05	0.00	300	0.38	0.13	0.02	-0.01
70	-0.07	0.13	0.11	0.11	350	-0.16	-0.02	-0.05	0.02
80	-0.06	-0.03	-0.10	-0.05	400	-0.01	0.00	0.03	0.01
90	-0.27	-0.10	-0.02	0.00	450	0.09	-0.02	0.00	-0.02
100	-0.03	-0.21	0.01	-0.01	500	-0.27	-0.07	0.02	0.03
Average	0.25	0.32	0.19	0.11	Average	0.18	-0.02	-0.02	-0.01
<u>Reputation</u>	Per unit improvement				<u>Range</u>	Per unit improvement			
Value	CT	CTP	CRP	Flood	Value	CT	CTP	CRP	Flood
2	0.27	2.47	4.25	1.47	10	17.98	3.12	33.04	14.61
3	41.23	20.48	2.90	1.60	15	11.23	14.56	3.89	1.49
4	11.70	21.20	1.30	1.40	20	5.33	3.12	1.08	0.53
5	-7.33	13.60	2.18	0.63	25	4.13	1.97	0.87	0.37
6	29.03	6.45	0.92	0.95	30	1.85	0.94	0.56	0.41
7	-7.50	3.80	1.60	0.02	35	0.69	0.91	0.52	0.32
8	0.67	3.60	-0.40	0.38	40	1.23	0.67	0.36	0.24
9	-0.07	1.97	0.60	0.45	45	0.60	0.36	0.33	0.18
10	-15.60	2.73	0.38	-0.60	50	1.33	0.16	0.14	0.12
Average	5.82	8.48	1.53	0.70	Average	4.93	2.87	4.53	2.03
<u>Adaptor</u>	Per unit improvement				<u>Consistency</u>	Per unit improvement			
Value	CT	CTP	CRP	Flood	Value	CT	CTP	CRP	Flood
20	-0.27	0.93	0.51	0.20	20	0.01	0.13	0.32	0.12
30	0.94	0.23	0.17	0.08	30	-1.42	0.13	0.01	0.03
40	0.67	0.50	0.22	0.04	40	0.55	0.17	0.08	0.03
50	0.11	0.20	0.02	0.06	50	1.16	0.04	-0.01	0.03
60	-0.12	0.13	0.12	0.02	60	-0.08	0.06	-0.01	-0.01
70	1.03	0.07	0.05	0.04	70	-0.08	0.08	0.07	-0.01
80	-0.66	0.19	0.03	0.03	80	0.09	-0.03	0.04	0.02
90	0.68	0.00	0.02	-0.02	90	0.30	-0.07	0.02	0.03
100	-0.11	0.00	-0.01	0.03	100	-0.31	0.15	-0.03	-0.01
Average	0.25	0.25	0.13	0.05	Average	0.02	0.07	0.05	0.03

TABLE E-1: PER UNIT INCENSEMENT

CT		CTP		CRP		FLOOD	
5.82	Reputation	8.48	Reputation	4.53	Range	2.03	Range
4.93	Range	2.87	Range	1.53	Reputation	0.70	Reputation
0.25	Velocity	0.32	Velocity	0.19	Velocity	0.11	Velocity
0.25	Adaptor	0.25	Adaptor	0.13	Adaptor	0.05	Adaptor
0.18	Territory	0.07	Consistency	0.05	Consistency	0.03	Consistency
0.02	Consistency	-0.02	Territory	-0.02	Territory	-0.01	Territory

TABLE E-2: PER UNIT INCENSEMENT BASED IMPORTANT LIST

E.1.2 Overall Improvement

<u>VELOCITY</u>	OVERALL IMPROVEMENT				<u>TERRITORY</u>	OVERALL IMPROVEMENT			
Value	CT	CTP	CRP	Flood	Value	CT	CTP	CRP	Flood
Max	47.78	37.98	22.90	14.48	Max	139.35	40.88	25.25	14.73
Min	20.67	5.53	4.33	3.60	Min	38.83	29.83	12.90	6.78
Average	26.21	11.89	8.49	6.17	Average	66.72	35.41	21.33	12.48
Overall	33.15%	61.11%	55.99%	49.51%	Overall	46.96%	14.57%	27.53%	30.48%
<u>Reputation</u>	Overall Improvement				<u>Range</u>	Overall Improvement			
Value	CT	CTP	CRP	Flood	Value	CT	CTP	CRP	Flood
Max	172.25	87.90	24.38	13.80	Max	284.83	131.66	206.39	93.35
Min	97.35	11.60	10.65	6.90	Min	63.03	2.64	2.49	2.14
Average	125.09	39.80	14.88	9.21	Average	118.86	36.95	32.40	16.48
Overall	24.78%	62.79%	33.70%	29.17%	Overall	52.62%	82.40%	88.31%	84.69%
<u>Adaptor</u>	Overall Improvement				<u>Consistency</u>	Overall Improvement			
Value	CT	CTP	CRP	Flood	Value	CT	CTP	CRP	Flood
Max	115.48	37.80	22.00	11.97	Max	120.43	18.24	14.28	8.99
Min	89.05	15.25	10.66	7.11	Min	101.05	11.78	9.10	6.64
Average	99.59	21.37	13.53	8.52	Average	106.95	14.12	10.48	7.32
Overall	12.17%	36.05%	29.87%	22.67%	Overall	8.35%	19.59%	19.87%	13.94%

TABLE E-3: OVERALL IMPROVEMENT

CT		CTP		CRP		FLOOD	
52.62%	Range	82.40%	Range	88.31%	Range	84.69%	Range
46.96%	Territory	62.79%	Reputation	55.99%	Velocity	49.51%	Velocity
33.15%	Velocity	61.11%	Velocity	33.70%	Reputation	30.48%	Territory
24.78%	Reputation	36.05%	Adaptor	29.87%	Adaptor	29.17%	Reputation
12.17%	Adaptor	19.59%	Consistency	27.53%	Territory	22.67%	Adaptor
8.35%	Consistency	14.57%	Territory	19.87%	Consistency	13.94%	Consistency

TABLE E-4: OVERALL IMPROVEMENT BASED IMPORTANCE LIST

E.2 Delivery Routing Depth Optimization Metrics

VELOCITY	PER UNIT IMPROVEMENT				TERRITORY	PER UNIT IMPROVEMENT			
Value	CT	CTP	CRP	Flood	Value	CT	CTP	CRP	Flood
20	-0.03	0.10	0.01	0.03	100	-0.04	-0.03	-0.02	0.00
30	-0.05	0.06	0.03	-0.02	150	-0.02	-0.01	0.00	-0.01
40	-0.05	0.05	0.03	0.01	200	0.03	0.01	-0.01	-0.02
50	-0.01	0.03	0.04	0.04	250	-0.04	-0.01	-0.01	0.00
60	0.03	0.04	0.03	0.04	300	0.04	0.00	0.00	0.00
70	-0.05	-0.01	-0.02	0.00	350	-0.04	0.00	0.01	0.00
80	0.06	0.01	0.02	0.00	400	0.02	0.01	0.01	0.01
90	-0.03	-0.01	-0.03	0.01	450	0.01	0.00	0.00	-0.01
100	-0.10	0.01	0.00	-0.03	500	-0.03	0.02	0.00	0.00
Average	-0.03	0.03	0.01	0.01	Average	-0.01	0.00	0.00	0.00
Reputation	Per Unit Improvement				Range	Per Unit Improvement			
Value	CT	CTP	CRP	Flood	Value	CT	CTP	CRP	Flood
2	-6.00	-4.10	-1.02	-1.33	10	-2.05	-1.49	0.03	0.15
3	-3.40	-2.09	-0.65	-0.47	15	-0.19	0.74	0.08	0.00
4	-3.03	-0.31	-0.40	-0.58	20	-0.16	0.30	-0.05	-0.12
5	-2.77	0.22	-0.38	-0.52	25	0.06	0.24	0.00	-0.11
6	1.32	0.30	-0.02	-0.20	30	-0.11	0.07	-0.02	-0.08
7	-1.92	0.05	-0.15	-0.78	35	0.04	-0.01	-0.10	-0.07
8	-1.33	0.08	-0.50	-0.10	40	-0.01	-0.04	-0.17	-0.12
9	-0.87	0.25	-0.30	0.63	45	0.01	-0.01	-0.01	-0.02
10	-1.62	0.77	-0.40	-0.50	50	0.10	-0.03	-0.01	-0.02
Average	-2.18	-0.54	-0.42	-0.43	Average	-0.26	-0.03	-0.03	-0.04
Adaptor	Per Unit Improvement				Consistency	Per Unit Improvement			
Value	CT	CTP	CRP	Flood	Value	CT	CTP	CRP	Flood
20	-0.24	0.10	0.02	0.00	20	-0.07	-0.01	0.04	-0.02
30	0.06	0.01	0.02	0.00	30	-0.19	0.00	-0.01	-0.01
40	-0.08	0.05	-0.01	-0.03	40	0.16	0.04	0.00	0.05
50	0.01	0.02	-0.01	0.00	50	0.03	-0.01	-0.02	0.01
60	-0.14	-0.01	0.02	-0.03	60	-0.03	0.01	0.01	-0.03
70	0.11	0.03	-0.02	0.00	70	0.06	0.03	0.01	0.00
80	-0.11	0.00	-0.01	0.00	80	-0.05	0.00	-0.04	0.02
90	0.08	0.01	0.01	-0.02	90	0.12	-0.01	0.03	0.00
100	-0.05	-0.01	-0.01	0.01	100	-0.05	0.01	-0.01	-0.01
Average	-0.04	0.02	0.00	-0.01	Average	0.00	0.01	0.00	0.00

TABLE E-5: PER UNIT INCENSEMENT

CT		CTP		CRP		FLOOD	
0.00	Consistency	0.03	Velocity	0.01	Velocity	0.01	Velocity
-0.01	Territory	0.02	Adaptor	0.00	Consistency	0.00	Consistency
-0.03	Velocity	0.01	Consistency	0.00	Adaptor	0.00	Territory
-0.04	Adaptor	0.00	Territory	0.00	Territory	-0.01	Adaptor
-0.26	Range	-0.03	Range	-0.03	Range	-0.04	Range
-2.18	Reputation	-0.54	Reputation	-0.42	Reputation	-0.43	Reputation

TABLE E-6: PER UNIT INCENSEMENT BASED IMPORTANT LIST

VELOCITY	OVERALL IMPROVEMENT				TERRITORY	OVERALL IMPROVEMENT			
Value	CT	CTP	CRP	Flood	Value	CT	CTP	CRP	Flood
Max	12.59	10.18	7.93	7.65	Max	12.53	6.78	5.70	5.90
Min	10.28	7.33	6.55	6.55	Min	8.73	4.43	4.00	4.70
Average	11.39	8.05	7.09	7.06	Average	11.16	6.23	5.12	5.27
Overall	9.66%	14.95%	9.08%	7.47%	Overall	16.36%	18.51%	16.03%	10.75%
Reputation	Overall Improvement				Range	Overall Improvement			
Value	CT	CTP	CRP	Flood	Value	CT	CTP	CRP	Flood
Max	24.67	10.00	8.10	7.63	Max	24.41	14.40	8.19	8.24
Min	5.05	3.50	4.28	3.65	Min	12.14	6.94	6.40	5.56
Average	17.80	8.63	6.55	6.31	Average	22.69	8.87	7.18	6.94
Overall	49.74%	36.58%	26.90%	29.73%	Overall	26.78%	30.09%	11.58%	17.80%
Adaptor	Overall Improvement				Consistency	Overall Improvement			
Value	CT	CTP	CRP	Flood	Value	CT	CTP	CRP	Flood
Max	24.56	10.34	6.88	7.02	Max	24.90	9.24	7.74	7.99
Min	20.72	8.16	6.48	6.34	Min	22.04	8.55	7.33	7.46
Average	23.32	8.76	6.68	6.66	Average	23.02	8.85	7.50	7.69
Overall	8.11%	11.10%	2.88%	4.91%	Overall	5.91%	3.79%	2.70%	3.34%

TABLE E-7: OVERALL IMPROVEMENT

CT		CTP		CRP		FLOOD	
5.91%	Consistency	3.79%	Consistency	2.70%	Consistency	3.34%	Consistency
8.11%	Adaptor	11.10%	Adaptor	2.88%	Adaptor	4.91%	Adaptor
9.66%	Velocity	14.95%	Velocity	9.08%	Velocity	7.47%	Velocity
16.36%	Territory	18.51%	Territory	11.58%	Range	10.75%	Territory
26.78%	Range	30.09%	Range	16.03%	Territory	17.80%	Range
49.74%	Reputation	36.58%	Reputation	26.90%	Reputation	29.73%	Reputation

TABLE E-8: OVERALL IMPROVEMENT BASED IMPORTANCE LIST

E.3 Delivery Message Duplication Optimization Metrics

VELOCITY					TERRITORY				
Value	CT	CTP	CRP	Flood	Value	CT	CTP	CRP	Flood
20	0.00	-0.14	-0.18	-0.16	100	0.00	-0.01	-0.01	-0.03
30	0.00	-0.10	-0.09	-0.13	150	0.00	0.00	-0.01	-0.06
40	0.00	-0.11	-0.08	0.02	200	0.00	0.00	-0.01	-0.02
50	0.00	-0.12	-0.07	-0.05	250	0.00	0.00	-0.01	0.00
60	0.00	0.04	0.05	0.07	300	0.00	0.00	0.00	-0.01
70	0.00	-0.12	-0.17	-0.04	350	0.00	0.00	0.00	0.00
80	0.00	0.02	0.09	0.01	400	0.00	0.00	0.00	0.01
90	0.00	0.01	0.09	0.03	450	0.00	0.00	0.00	-0.02
100	0.00	0.00	-0.09	-0.18	500	0.00	0.01	0.00	0.01
	0.00	-0.06	-0.05	-0.05	Average	0.00	0.00	0.00	-0.01
Reputation					Range				
Value	CT	CTP	CRP	Flood	Value	CT	CTP	CRP	Flood
2	0.00	-0.15	-1.30	-7.23	10	0.00	-0.27	-0.44	-0.02
3	0.00	-0.79	-0.10	-3.65	15	0.00	-0.33	-0.30	0.11
4	0.00	-1.01	-0.65	-3.50	20	0.00	-0.25	-0.25	-0.10
5	0.00	-1.10	-0.55	-1.67	25	0.00	-0.27	-0.08	-0.01
6	0.00	-0.82	-0.57	0.07	30	0.00	-0.34	-0.14	0.21
7	0.00	-1.33	-0.58	-1.35	35	0.00	-0.44	-0.24	-0.16
8	0.00	-1.10	-0.50	0.25	40	0.00	-0.39	-0.35	0.13
9	0.00	-0.70	-1.10	1.18	45	0.00	-0.41	-0.30	-0.11
10	0.00	-0.77	-0.90	-2.53	50	0.00	-0.29	-0.27	0.09
	0.00	-0.86	-0.69	-2.05		0.00	-0.33	-0.26	0.02
Adaptor					Consistency				
Value	CT	CTP	CRP	Flood	Value	CT	CTP	CRP	Flood
20	0.00	-0.23	-0.32	-1.24	20	0.00	-0.09	-0.01	-0.05
30	0.00	-0.08	-0.06	-0.02	30	0.00	-0.07	-0.05	-0.01
40	0.00	-0.04	0.00	0.08	40	0.00	0.02	-0.04	0.09
50	0.00	-0.02	-0.06	-0.07	50	0.00	-0.04	-0.03	0.03
60	0.00	-0.04	-0.01	0.06	60	0.00	0.02	0.01	-0.09
70	0.00	-0.02	-0.03	-0.09	70	0.00	0.01	0.00	0.02
80	0.00	0.01	0.02	0.08	80	0.00	-0.02	-0.03	0.03
90	0.00	-0.06	-0.04	-0.03	90	0.00	-0.02	0.02	0.06
100	0.00	0.01	0.02	0.00	100	0.00	0.00	-0.01	-0.06
	0.00	-0.05	-0.05	-0.13		0.00	-0.02	-0.02	0.00

TABLE E-9: PER UNIT INCREMENT

CT		CTP		CRP		FLOOD	
0.00	Not Applicable	0.00	Territory	0.00	Territory	0.02	Range
0.00	Not Applicable	-0.02	Consistency	-0.02	Consistency	0.00	Consistency
0.00	Not Applicable	-0.05	Adaptor	-0.05	Velocity	-0.01	Territory
0.00	Not Applicable	-0.06	Velocity	-0.05	Adaptor	-0.05	Velocity
0.00	Not Applicable	-0.33	Range	-0.26	Range	-0.13	Adaptor
0.00	Not Applicable	-0.86	Reputation	-0.69	Reputation	-2.05	Reputation

TABLE E-10: PER UNIT INCREMENT BASED IMPORTANT LIST

<u>PEER VELOCITY</u>					<u>TERRITORY</u>				
Value	CT	CTP	CRP	Flood	Value	CT	CTP	CRP	Flood
Max	0.00	11.00	13.40	25.13	Max	0.00	2.70	4.45	13.23
Min	0.00	5.68	7.93	21.00	Min	0.00	1.85	2.70	6.65
Average	0.00	9.29	11.34	23.43	Average	0.00	2.41	3.97	11.47
Overall	0.00%	27.23%	22.74%	8.56%		0.00%	16.99%	21.40%	27.65%
<u>Reputation</u>					<u>Range</u>				
Value	CT	CTP	CRP	Flood	Value	CT	CTP	CRP	Flood
Max	0.00	7.80	10.73	23.78	Max	0.00	15.09	15.99	23.56
Min	0.00	0.03	4.48	5.35	Min	0.00	0.19	4.26	22.51
Average	0.00	3.65	7.49	18.68	Average	0.00	7.25	10.29	23.15
Overall	0.00%	76.17%	35.19%	46.41%	Overall	0.00%	74.67%	47.10%	2.25%
<u>Adaptor</u>					<u>Consistency</u>				
Value	CT	CTP	CRP	Flood	Value	CT	CTP	CRP	Flood
Max	0.00	6.25	9.70	23.72	Max	0.00	7.53	10.19	24.18
Min	0.00	1.23	4.55	11.03	Min	0.00	5.69	8.66	22.78
Average	0.00	4.82	8.53	22.14	Average	0.00	7.07	9.62	23.51
Overall	0.00%	48.73%	29.37%	28.43%	Overall	0.00%	12.80%	7.76%	2.94%

TABLE E-11: OVERALL IMPROVEMENT

CT		CTP		CRP		FLOOD	
0.00	Not Applicable	12.80%	Consistency	7.76%	Consistency	2.25%	Range
0.00	Not Applicable	16.99%	Territory	21.40%	Territory	2.94%	Consistency
0.00	Not Applicable	27.23%	Velocity	22.74%	Velocity	8.56%	Velocity
0.00	Not Applicable	48.73%	Adaptor	29.37%	Adaptor	27.65%	Territory
0.00	Not Applicable	74.67%	Range	35.19%	Reputation	28.43%	Adaptor
0.00	Not Applicable	76.17%	Reputation	47.10%	Range	46.41%	Reputation

TABLE E-12: OVERALL IMPROVEMENT BASED IMPORTANCE LIST

Appendix F: Extreme High Density Scenario

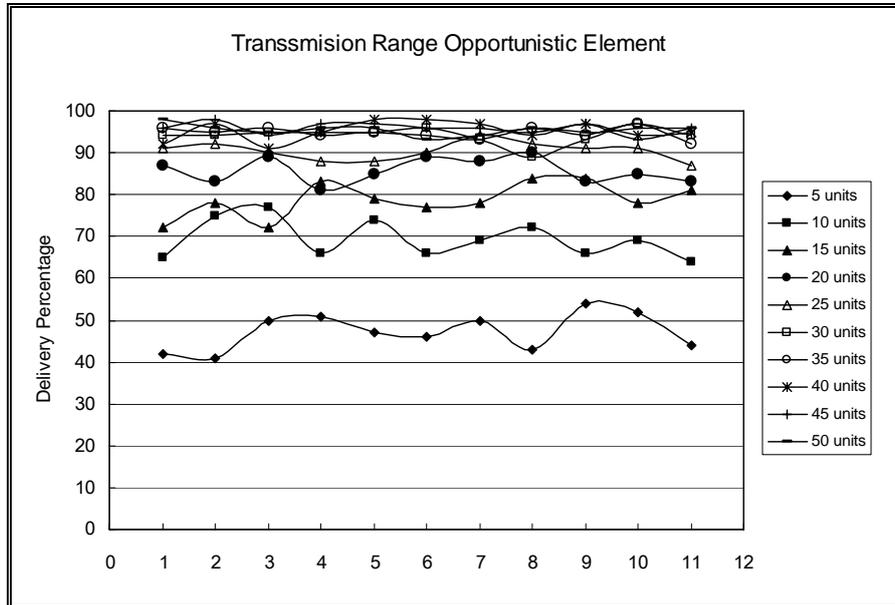


FIGURE F-1: DELIVERY RATE OVER TRANSMISSION RANGE

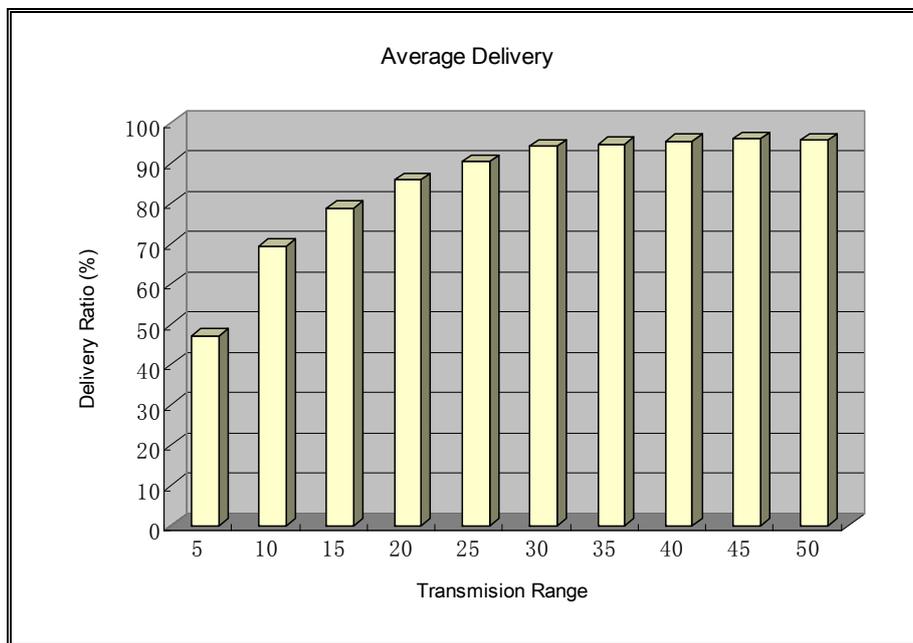


FIGURE F-2: DELIVERY RATE OVER TRANSMISSION RANGE

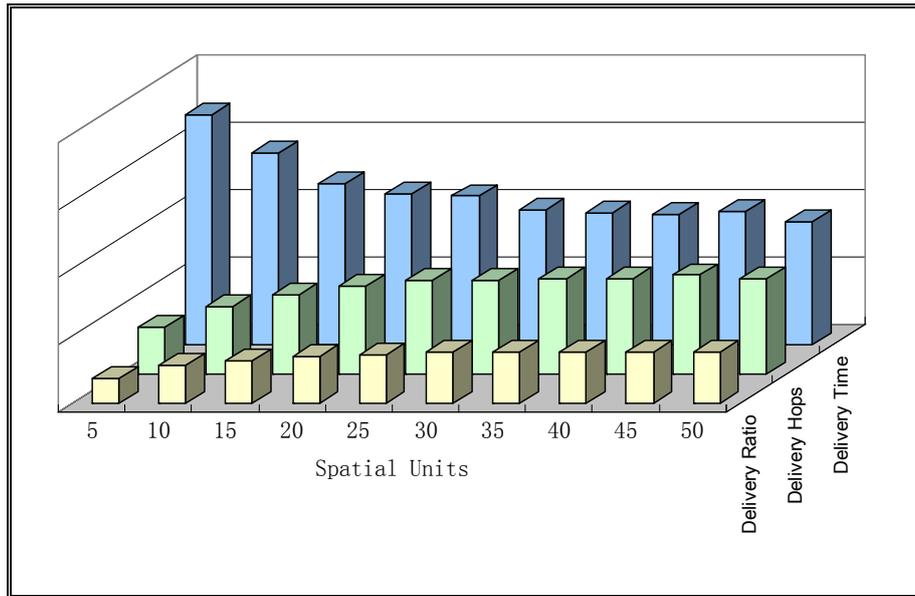


FIGURE F-3: QUALITY MATRIX OVER TRANSMISSION RANGE

(500×500 area, 1000 Peers, 2 Speed) The above two diagrams clearly show that the mobile device transmission range has very high impact on the reliability (Delivery Ratio) in this given scenario, the delivery ratio improved from 40 percent to over 90 percent delivery ratio when transmission range is increased from 5 units to 30 units. However, in this particular scenario, after the transmission range reaches 30 units, further increasing of transmission range does not provide additional significant delivery ratio gain.

Peer Density and Routing Quality			
Speed Units	Delivery Ratio	Deliver Hops	Deliver Time
1 Units	88.7 %	436.3	991.3
5 Units	95.5 %	471.9	1047.2
10 Units	96.3 %	490.7	1052.3
15 Units	97.7 %	475.7	966.5
20 Units	97.1 %	493.5	1008.2
25 Units	97.5 %	489.2	963.3
30 Units	98.4 %	496.5	992.6
35 Units	97.5 %	490.5	973.2
40 Units	97.8 %	495.0	964.5
45 Units	97.3 %	481.0	924.9
50 Units	96.6 %	492.0	942.1

TABLE F-1: PEER DENSITY AND ROUTING QUALITY

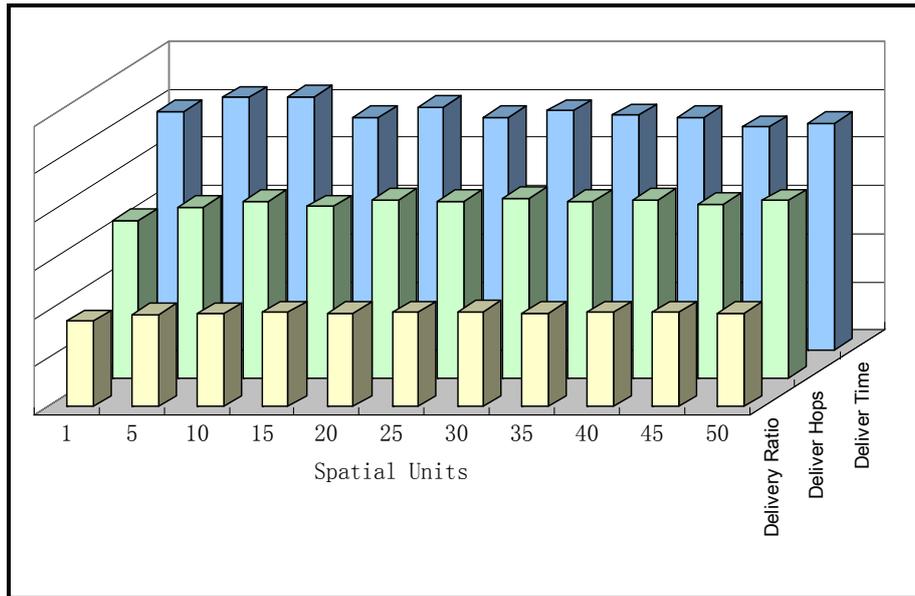


FIGURE F-4: QUALITY MATRIX OVER TRANSMISSION RANGE
(500×500 area, 1000 Peers, 25 Range)

From this diagram, it is obvious that there is hardly any fluctuation for all three measurements while peer velocity is moving from $1s_{\Delta}/per\ t_{\Delta}$ to $50s_{\Delta}/per\ t_{\Delta}$. Conclusion thus can be made that in a high density DTMANET such as this example (500×500 area, 1000 Peers, 25 Range) peer velocity does not play a significant role in terms of reliability (delivery ratio), performance (delivery time) and efficiency (delivery hops count)

By comparing data output such as delivery ratio, delivery time, total hops count and output other data if necessary; OCI will build an index – the Opportunistic Confidence Index (OCI), which suggest an optimized combination of filter selection. OCI also provides indexes that indicate the importance of each factor that will determine the successful and efficiency of MANET messaging communication of a give scenario.

Appendix G: Hardware and Software Environment

OCI-SIM is a Java-based simulator, and it runs on top of any Operating System with the support of a Java runtime environment

G.1 Hardware environment

- 1) CPU: Intel[®] Core2[™] 6300 (@ 1.86GHz × 2)
- 2) RAM: 1.98GB
- 3) Hard drive: 149GB
- 4) Graphic Card: Nvidia[®] GeForce[™] 6600

G.2 Software environment

- 1) Operating System:
Microsoft[®] Windows XP[™] Professional Edition, Version 2002, Service Pack 2
- 2) Runtime environment:
Java[®] 2 Runtime Environment 1.4.2
- 3) Development Environment
IDE: Borland[®] JBuilder[™] 2005 Foundation
SDK: Java[®] 2 Standard Edition SDK 1.4.2
- 4) Third party code/library used and/or modified
 - a) Java XML parser: NanoXML [Scheemaecker, 2001]
 - b) Combination mechanism: Combination generator [Gilleland, 2004]
 - c) Permutation mechanism: Permutation Generator [Gilleland, 2004]