Cooperative opportunistic data delivery solutions for challenging wireless networks

Aung, Cherry Ye

2017

Cherry Ye Aung. (2017). Cooperative opportunistic data delivery solutions for challenging wireless networks. Doctoral thesis, Nanyang Technological University, Singapore.

<http://hdl.handle.net/10356/69551>

<https://doi.org/10.32657/10356/69551>

**COOPERATIVE OPPORTUNISTIC
DATA DELIVERY SOLUTIONS FOR
CHALLENGING WIRELESS NETWORKS**

CHERRY YE AUNG

**SCHOOL OF ELECTRICAL AND
ELECTRONIC ENGINEERING**

2017

**COOPERATIVE OPPORTUNISTIC
DATA DELIVERY SOLUTIONS FOR
CHALLENGING WIRELESS NETWORKS**

CHERRY YE AUNG

SCHOOL OF ELECTRICAL AND ELECTRONIC ENGINEERING

A thesis submitted to the Nanyang Technological
University in partial fulfilment of the requirements for the
degree of Doctor of Philosophy

2017

Abstract

Advances in wireless communications and networking technologies are rapidly making the anywhere anytime wireless access a reality. Wireless ad hoc networks (WANETs) have been largely deployed due to its several promising benefits over the wired networks. The deployment of WANET is in various forms which include mobile ad hoc network (MANET), vehicular ad hoc network (VANET), opportunistic network or delay tolerant network (DTN), and mobile social networks. Disaster recovery communications, military/tactical communications, mobile social networks, vehicle-to-any (V2X) communications, mobile sensor networks are some major applications of WANET which have been widely deployed today.

Data delivery in WANET, especially in mobile and hostile environments, poses great challenges due to the infrastructure-less environment, mobility, unreliable wireless link nature, effect of terrain, and distributed nature of wireless devices. In this thesis, we propose data delivery solutions for challenging wireless networks with major focus on opportunistic networks. We exploit adaptive data forwarding strategies, cooperation opportunities, open-air wireless medium nature, communities and ties in social network nature, and multiple network interfaces to tackle such challenges and improve the system performance.

Firstly, we study the mobility models in WANET. The synthetic mobility models are widely used in the WANET research for network simulations and performance evaluations. Group mobility model is of particular interest as group motion occurs commonly in scenarios where WANETs are deployed, such as disaster recovery operations, military operations, and mobile social

networks. We provide a systematic and critical review of recent group mobility models proposed for WANETs.

Secondly, we propose a data delivery solution which aims to solve the long delivery delay problem and the information epidemics problem in store-carry-forward routing in opportunistic networks. The proposed solution comprises two main algorithms: store-carry-cooperative forward routing and information epidemic control. The susceptible-infected-recovered model is used to study the effectiveness of the proposed mechanism. Extensive network performance evaluation is conducted under a wide range of scenarios which include fading environments, obstacle-constrained environments, and mobile social network environments. We show that 1) the information epidemics control mechanism provides higher vaccination rate and recovery rate; 2) proactive replication incurs a number of unnecessary transmissions; 3) monitoring the vicinity and exploiting the opportunity shorten the data delivery delay; and 4) with the integrated solution, a robust data delivery is achieved and a substantial amount of unnecessary transmissions are well deterred.

Thirdly, we propose a data delivery solution which aims to enhance the system throughput for multi-hop wireless networks under multiple cross traffic flows and unreliable wireless channel. Cooperative forwarding has shown a substantial network performance improvement compared to traditional routing in multi-hop wireless network. To further enhance the system throughput, especially in the presence of highly congested multiple cross traffic flows, a promising way is to incorporate the multi-radio multi-channel (MRMC) capability into cooperative forwarding. In this work, we propose a new cooperative forwarding solution for multi-radio multi-channel multi-flow wireless networks. The performance is studied in highly congested network. Performance evaluation shows that the proposed solution reduces about 97% of the

delay and achieves about 200% of throughput improvement compared to the single-radio single-channel case.

Finally, we propose a data delivery solution for opportunistic networks from social network perspective. Given the communities and centralities nature in social network, it assures to exploit receiver diversity in such network environment, and therefore be able to maximize the given contact opportunity for the reliable data transfer. In addition, we claim that having this community nature also enables information epidemics control in such network more manageable by using efficient vaccination and recovery scheme. Motivated by these facts and the previous proposed adaptive, cooperative protocols and information epidemic control mechanism, we propose a third solution called cooperation-aided data delivery solution for opportunistic networks. The evaluation is conducted with varying network connectivity ranging from highly dense to highly sparse environments and under the presence of probabilistic propagation channels. We show that the proposed solution substantially improves the data delivery performance (i.e. higher delivery ratio while cutting down 4% to 85% of the delivery delay, 86% to 93% of the data overhead/retransmissions, and 86% to 97% of the control overhead) compared to the existing highest success rate protocol.

Acknowledgments

Five years ago, when I contacted Prof. Chong Han Joo Peter for my PhD study, I got to know that I had made an excellent choice. Throughout the study of my PhD, Prof. Chong has given me continuous support and guidance. I would like to take this opportunity to sincerely thank Prof. Chong for all the countless support without which I could not be able to complete my PhD study. His wisdom and broad experience and a number of long discussions with him and his insightful comments shape the ideas of my research. Besides the research experience, he provides me opportunities to involve in a number of projects including the current test-bed project and to visit overseas universities and present my research work at those universities. After working under Prof. Chong's supervision for all these years, I can feel great differences in me in many aspects such as research experience, working skills with several people from different backgrounds, presentation skills and the most importantly, confidence. He is one of the best mentors in my life to whom I will always be thankful for.

I would also like to express my sincere thanks to Prof. Guan Yong Liang and Prof. Tay Wee Peng for their guidance and support in my PhD studies and in the test-bed project. I would also like to thank Prof. Ivan Wang-Hei Ho for showing me how to write high quality research papers; Dr. Lingfu Xie for sharing knowledge and providing help on network simulation; Dr. G.

G. Md. Nawaz Ali for helping me to revise my manuscripts carefully and patiently; Dr. Yang Ming for giving me hardware and network resources so that I could finish all my simulations within a short period of time.

The study life during the last four years will not be that much comfortable without my senior sisters and brothers: Dr. Aye Su Yee, Dr. Thein Than Tun, and Dr. Yan Naing Aye. I would like to sincerely thank you for being there for me all the time. My special thanks goes to Dr. Aye Su Yee for her help, encouragement, and discussions on my research throughout all these years. I would also like to thank Ma Thida Than for being such a good big sister to me who always take care of me and Dr. Hnin Yu Shwe for helping me revise my manuscripts. My stay in INFINITUS lab would not have been that much memorable without my international friends and colleagues. Thanks to Mehrnaz, Zeinab, Ruojun, Zhao Ming, Xinxing, Pengty, and Heng Chuan. I would like to special thank to Mehrnaz for sharing knowledge with me and making very delicious Iranian meals for me. I would also like to thank all members of the V2X test-bed team for being good colleagues and helping me in the project when I am busy with my thesis.

There are no words I can say to describe my sincere gratitude to my mother, Daw Yi Htay. She has been both a mother and a father since my childhood. Her unconditional love, patience, and support have made me the person I am today. I would also like to express my sincere gratitude to my uncle, Col. Than Oo, for supporting me throughout my studies. My sincere thanks to my one and only sister, Philay, for being there whenever I needed it. My sincere thanks to my husband, Wang Shaobo, for giving me all the understanding and support in my life.

Contents

Acknowledgment	i
Table of Contents	vii
List of Figures	xii
List of Tables	xiv
List of Abbreviations	xv
List of Symbols	xviii
1 Introduction	1
1.1 Background	1
1.2 Motivation	4
1.3 Major Contributions	7
1.4 Thesis Organization	11
2 A Review of Group Mobility Models for Mobile Ad Hoc Networks	12
2.1 Introduction	12
2.2 Classification of Group Mobility Models	14
2.3 Point Based Group Mobility Models	15

2.3.1	PBGM with IGM Movement	16
2.3.2	PBGM with CGM Movement	19
2.4	Region Based Group Mobility Models	21
2.4.1	RBGM with IGM Movement	22
2.4.2	RBGM with CGM Movement	25
2.5	Abstraction of Design Features in Group Mobility Models . . .	27
2.6	Conclusion	32

3	Store-Carry-Cooperative Forward Routing with Information Epidemics Control for Data Delivery in Opportunistic Networks	33
3.1	Introduction	33
3.2	Related Work	37
3.2.1	Routing	37
3.2.2	Information Epidemics Control	40
3.3	Overview of SCCF	43
3.4	Live-update Link Quality	45
3.5	Routing	48
3.5.1	Forwarding Type	48
3.5.2	Opportunistic Forwarding	49
3.5.3	Packet Buffer	53
3.5.4	Network Layer Acknowledgement	54
3.5.5	Replication-based Forwarding	55
3.5.6	Handling Link Loss	56
3.6	Information Epidemics Control	57
3.7	SIR Model for the Information Epidemics in Network	60
3.7.1	SIR Model Without Vaccination	61
3.7.2	SIR Model With Vaccination	62

3.7.3	Control Signal Distribution	63
3.8	Performance Evaluation	64
3.8.1	Evaluation Methodology	66
3.8.2	Verification and Validation of Simulation Models	69
3.8.3	Study on SIR Population in the Network	70
3.8.4	Routing Performance Study	71
3.9	Conclusion	83
4	Cluster-based Cooperative Data Forwarding with Multi-radio	
	Multi-channel for Multi-flow Wireless Networks	86
4.1	Introduction	86
4.2	Related Work	90
4.2.1	Cooperative Forwarding	90
4.2.2	Integration with Multi-Radio Multi-Channel	91
4.3	Network Model	92
4.4	Cluster-to-cluster ETX (C2CETX)	92
4.5	Cluster-based Cooperative Data Forwarding (CCDF)	96
4.5.1	Distributed Agent-based Coordination	98
4.5.2	Receiving cluster diversity	100
4.5.3	Network Layer Acknowledgement	101
4.5.4	Packet Buffer	102
4.5.5	Augmentation for Data Resiliency	102
4.5.6	Handling Link Loss	104
4.5.7	Data Forwarding Example	104
4.6	CCDF with MRMC	106
4.6.1	Cluster-based Channel Assignment	106
4.6.2	Data Forwarding over Diverse Channels	107
4.6.3	CCDF-MRMC Example	109

4.7	Analysis on Coordination time in CCDF	109
4.7.1	Single flow	109
4.7.2	Multi flows	113
4.8	Performance Evaluation	114
4.8.1	Evaluation Methodology	114
4.8.2	Single Flow	117
4.8.3	Multi Flows	120
4.9	Conclusion	128
5	Cooperation Aided Data Delivery Solution for Opportunistic Networks under Group Connectivity Characteristics	130
5.1	Introduction	130
5.2	Related Work	134
5.2.1	Cooperative Forwarding	134
5.2.2	Routing in Opportunistic Networks	134
5.3	Network Model	138
5.4	Overview of C-SCCF	139
5.5	Live-update Link Quality	140
5.6	Contact Cluster Discovery	141
5.6.1	Dynamic Gateway Selection	143
5.6.2	2-hop Cluster Contact Discovery	143
5.7	Routing	144
5.7.1	Cluster-based Cooperative Data Forwarding (CCDF)	146
5.7.2	Cooperation-aided Replication	152
5.7.3	Distributed Agent-based Coordination	154
5.7.4	Packet Buffer	157
5.7.5	Network Layer Acknowledgement	158
5.7.6	Augmentation for Data Resiliency for Direct CCDF	159

5.7.7	Handling Link Loss	160
5.8	Community-aware Information Epidemics Control	161
5.8.1	Vaccine Synchronization in a Cluster	164
5.8.2	Overhead	165
5.9	Performance Evaluation	167
5.9.1	Evaluation Methodology	167
5.9.2	Overall network performance	169
5.9.3	Dense network	171
5.9.4	Relatively sparse network	173
5.9.5	Highly sparse network	174
5.9.6	Overhead	176
5.10	Conclusion	180
6	Conclusions and Future Work	181
6.1	Conclusion	181
6.2	Recommendation for Future Work	184
	Author's Publications	186
	References	201

List of Figures

2.1	Classification of group mobility models. References to these models are: RPGM [1], RVGM [2], GFMM [3], MGCM [4], RRGGM [5], VTGM [6], and CMM/ECMM [7, 8]	15
2.2	Movement of group members in RPGM.	18
2.3	Movement of groups in MGCM.	22
2.4	Movement of group and non-group nodes in VTGM.	25
3.1	Network layer architecture.	41
3.2	Data forwarding decision when a node gets contact.	44
3.3	Cooperative data forwarding.	44
3.4	Header format (a) Hello packet (b) Data packet.	45
3.5	Calculating link quality.	45
3.6	Four packet states and state transitions.	56
3.7	(a) Direct broadcast. (b) Two-hop broadcast.	57
3.8	Markov chain model of an infectious disease: (a) SIR model without vaccination; (b) SIR model with vaccination.	61
3.9	(a) Infected population under ER self-healing. (b) SIR population under SCCF self-healing; (c) without partially-immunized vaccination; and (d) with both partially-immunized and fully-immunized vaccination.	66

3.10	Illustration of the basic conceptual protocol model.	70
3.11	(a) CDF with respect to packet delivery delay; (b) zoom in view with AODV.	74
3.12	The effect of fading: (a) CDF with respect to packet delivery delay; (b) CDF with respect to transmission count.	74
3.13	Summary vector overhead in Epidmeic and SCCF routing. . .	76
3.14	(a), (b), and (c) The effect of obstacle-constrained environ- ment. (d) The effect of varying mean speed.	77
3.15	The effect of varying mean speed.	79
3.16	(a) CDF with respect to packet delivery delay; (b) CDF with respect to packet transmission count.	80
3.17	(a) CDF with respect to packet delivery delay; (b) CDF with respect to packet transmission count.	81
3.18	Delivered packet probability with respect to packet transmis- sion count.	82
4.1	Link delivery probability p	93
4.2	Hello packet header format.	93
4.3	Data packet header format.	93
4.4	Network with three clusters and <i>C2CETX</i> between clusters. .	96
4.5	Illustration of various data forwarding paths from the source/sender cluster to the next-hop or the destination cluster.	98
4.6	Illustration of data forwarding in CCDF. For CCDF-MRMC, please refer to the channel number indicated in the bracket. .	98
4.7	Illustration of augmentation for data resiliency.	104
4.8	Data forwarding over diverse channels.	108
4.9	Expected coordination time: (a) with varying link delivery probability; (b) with varying fragment size.	110

4.10	Network topology.	115
4.11	Probe packet reception ratio over distance.	117
4.12	Comparisons with AODV and OR in scenario II network model: (a) Packet delivery ratio; (b) Average end-to-end delay; and (c) Throughput.	118
4.13	Normalized MAC loads of AODV and CCDF in scenario II network model.	119
4.14	Packet delivery ratio: (a) SRSC. (b) 3R3C; and (c) 3R4C. . .	121
4.15	Average end-to-end delay: (a) SRSC. (b) 3R3C; and (c) 3R4C.	122
4.16	Throughput: (a) SRSC. (b) 3R3C; and (c) 3R4C.	123
4.17	CDF of PDR: (a) SRSC and (b) 3R3C and 3R4C.	124
4.18	Study on MAC layer transmission count in two-flow network: (a) AODV-SRSC; (b) AODV-3R3C; (c) COR-SRSC; and (d) COR-3R3C.	127
4.19	Study on MAC layer transmission count in six-flow network: (a) AODV-SRSC; (b) AODV-3R3C; (c) COR-SRSC; and (d) COR-3R3C.	128
5.1	Network layer architecture.	137
5.2	Link delivery probability p	138
5.3	Forwarding types and order.	139
5.4	Hello packet header format.	141
5.5	Data packet header format.	141
5.6	CCDF-1h forwarding approach: (a) <i>Direct</i> forwarding; (b) <i>via-Bridge-DstCluster</i> forwarding; and (c) <i>via-Gateway-DstCluster</i> forwarding. (where a_0 is the bridge node, a_2 is the gateway node, a_4 is the member-relay, and b_1 is the ACK-sender.) . . .	147

5.7	CCDF-2h forwarding approach: (a) <i>via-RelayCluster</i> forwarding; (b) <i>via-Bridge-RelayCluster</i> forwarding; and (c) <i>via-Gateway-RelayCluster</i> forwarding (where a_0 is the bridge node, a_2 is the gateway node, a_4 is the member-relay, and b_1 is the ACK-sender.). The illustration for data forwarding from relay cluster to the destination cluster is shown in Fig. 5.8.	149
5.8	Forwarding from relay cluster to destination cluster in CCDF-2h: (a) <i>Direct</i> forwarding; (b) <i>via-Bridge-DstCluster</i> forwarding; and (c) <i>via-Gateway-DstCluster</i> forwarding.	150
5.9	Cooperation-aided replication: (a) Packet state vector request and reply; (b) Data forwarding.	152
5.10	Four packet states and state transitions.	161
5.11	Probe packet reception ratio over distance.	166
5.12	Overall performance: (a) packet delivery ratio; (b) average end-to-end delay; (c) average MAC layer transmission count; and (d) normalized hello packet collision rate in ER.	169
5.13	Dense network: (a) CDF with respect to packet delivery delay; (b) CDF with respect to transmission count.	171
5.14	CDF with respect to packet delivery delay in relatively sparse networks: (a) 2 sq km; (b) 3 sq km; (c) 4 sq km; and (d) 5 sq km.	172
5.15	CDF with respect to packet delivery delay in highly sparse networks: (a) 6 sq km; (b) 7 sq km; (c) 8 sq km; and (d) 9 sq km.	174
5.16	CDF with respect to MAC layer transmission count in relatively sparse networks: (a) Epidemic Routing; (b) C-SCCF Routing.	175

5.17 CDF with respect to MAC layer transmission count in highly sparse networks: (a) Epidemic Routing; (b) C-SCCF Routing.	175
5.18 C-SCCF overhead in 2 sq km network: (a) CDF of summary vector transmissions; (b) CDF of agent vector transmissions.	177
5.19 C-SCCF overhead in 9 sq km network: (a) CDF of summary vector transmissions; (b) CDF of agent vector transmissions.	177
5.20 ER overhead: (a) in 2 sq km network; (b) in 9 sq km network.	178
5.21 Normalized MAC load in both 2 sq km and 9 sq km network: (a) C-SCCF; (b) ER.	178

List of Tables

2.1	Comparison of the models based on their design features and applications	31
3.1	Summary of Related Works	42
3.2	Reverse Link Information Table	47
3.3	Contact Information Table	48
3.4	SCCF Forwarding Types	49
3.5	Forwarding Table Format	51
3.6	Simulation Parameters	65
3.7	PDR Comparison with CORMAN	72
3.8	95% Confidence Intervals for Comparison with CORMAN Study	83
3.9	95% Confidence Intervals for Effects of Fading Study	84
3.10	90% Confidence Intervals for Study under Mobile Social Network Model	84
4.1	Simulation parameters	114
4.2	Relative packet delivery ratio improvement of CCDF-MRMC over AODV-MRMC counterpart	124
4.3	Relative end-to-end delay improvement of CCDF-MRMC over CCDF-SRSC	124

4.4	Relative end-to-end delay improvement of CCDF-MRMC over AODV-MRMC counterpart	125
4.5	Relative throughput improvement of CCDF-MRMC over CCDF-SRSC	125
4.6	Relative throughput improvement of CCDF-MRMC over AODV-MRMC counterpart	125
4.7	Average MAC layer transmission count per delivered packet	127
5.1	Contact Node Information Table	145
5.2	1-hop Contact Cluster Information Table	145
5.3	2-hop Contact Cluster Information Table	145
5.4	Node Types and Definition	146
5.5	Simulation parameters	166
5.6	Overall relative improvement of C-SCCF over ER	167

List of Abbreviations

1R1C	One-Radio One-Channel
3R3C	Three-Radio Three-Channel
3R4C	Three-Radio Four-Channel
ACK	Acknowledgement
AODV	Ad Hoc On-Demand Distance Vector
ARP	Address Resolution Protocol
C2CETX	Cluster-to-Cluster Expected Transmission Count
CaR	Cooperation-aided Replication
CCDF	Cluster-based Cooperative Data Forwarding
CCDF-1h	One-Hop Cluster-based Cooperative Data Forwarding
CCDF-2h	Two-Hop Cluster-based Cooperative Data Forwarding
CDF	Cumulative Distribution Function
CGM	Coordinated Group Member
CMM	Community-based Mobility Model
CORMAN	Cooperative Opportunistic Routing in Mobile Ad Hoc Network
CRS	Candidate Relay Set
CRP	Candidate Relay Prioritization
C-SCCF	Cluster-based Store-Carry-Cooperative Forward
DAC	Distributed Agent-based Coordination
DSDV	Destination Sequenced Distance Vector

DSRC	Dedicated Short Range Communications
DTN	Delay/Disruption Tolerant Network
E2ED	End-to-End Delay
ECMM	Enhanced CMM
ECT	Expected Coordination Time
ECT _{agt}	Expected Coordination Time required for Agent-based Coordination
ER	Epidemic Routing
ETX	Expected Transmission Count
GFMM	Group Force Mobility Model
HCMM	Home Cell Mobility Model
HSR	Hierarchical State Routing
IGM	Individual Group Member
IM	Interaction Matrix
MAC	Medium Access Control
MANET	Mobile Ad Hoc Network
MCM	Mission Critical Mobility
MCS	Modulation and Coding Scheme
MGCM	Multi-Group Coordination Mobility
MRMC	Multi-Radio Multi-Channel
NET	Network
OPPNET	Opportunistic Network
OR	Opportunistic Routing
PBGM	Point Based Group Mobility
PDR	Packet Delivery Ratio
RBGM	Region Based Group Mobility
RPGM	Reference Point Group Mobility
RRGM	Reference Region Group Mobility

RVGM	Reference Velocity Group Mobility
RWP	Random Waypoint
SA	Social Attractiveness
SCCF	Store-Carry-Cooperative Forward
SCF	Store-Carry-Forward
SIR	Susceptible-Infected-Recovered
SRSC	Single-Radio Single-Channel
SWIM	Small World in Motion
TXC	Transmission Count
V2V	Vehicle-to-Vehicle
V2X	Vehicle-to-Everything
VANET	Vehicular Ad Hoc Network
VTGM	Virtual Track Group Mobility
WANET	Wireless Ad Hoc Network

List of Symbols

p	Bi-directional link delivery probability
p_{n_i, n_j}	p between two nodes n_i and n_j
p_{n_i, C_j}	p between a node n_i and a cluster C_j
p_{fwd}	Forward link delivery probability
p_{rev}	Reverse link delivery probability
$p(e)$	Probability of event e
$p_{b_r}(i)$	Forwarding probability of node b_r with rank r for flow f_i
K	Number of interfaces a node has
X	Set of IDs of packets a node has in its primary packet buffer
Y	Set of IDs of packets with state <i>Local_fwd</i>
Z	Set of IDs of packets with state <i>Delivered</i>
$S(t)$	Susceptible node population at time t
$I(t)$	Infected node population at time t
$R(t)$	Recovered node population at time t
$v(t)$	Vaccination rate in the system at time t
v_o	Vaccination rate using VACCINE method
v_f	Vaccination rate using fully-immunized vaccine
v_p	Vaccination rate using partially-immunized vaccine
w	A time window (in seconds) during which the p value is computed
β	Node infection rate

γ	Node recovery rate
τ	Periodic hello message interval
χ	The set of all possible events
$C2CETX_{C_i, C_j}$	C2CETX between clusters C_i and C_j
ECT_{f_i}	ECT for flow f_i
ETX_{n_i, C_j}	ETX between node n_i and cluster C_j
k_{C_i}	Cluster C_i 's channel number
ID_{C_i}	Unique identifier of cluster C_i
T_{VAC}	Time instance the vaccinating process is initiated
T_{P_VAC}	Time instance the partially-immunized vaccinating process is initiated
T_{F_VAC}	Time instance the fully-immunized vaccinating process is initiated
t_δ	Amount of time taken to complete a fragment transmission
t_{data}	Data packet forwarding time
$t_{control}$	Control packet forwarding time
t_{agt}	Time required for DAC
t_{REQ}	Time taken for decision request
t_{REP}	Time taken for decision reply

Chapter 1

Introduction

1.1 Background

Advances in wireless communications and networking technologies are rapidly making the anywhere anytime wireless access a reality. These advancements have enabled the deployment of wireless ad hoc networks (WANETs) in some of the challenging wireless environments such as disaster recovery operation, military operations, vehicle-to-everything communication networks, and mobile social networks. Data delivery in WANET, especially in those hostile environments, poses great challenges due to the infrastructure-less environment, mobility, effect of terrain, effect of mobile and stationary scatterers, and distributed nature of wireless devices.

The first batch of data delivery solutions for mobile ad hoc networks (MANETs) were proposed in late 1990s and early 2000s [9, 10, 11]. These routing protocols are called traditional routing protocols. The traditional routing protocols preselect one or a few complete end-to-end path(s) before data delivery begins. Data packets are forwarded to the next-hop node in sequence over the path. Therefore, it performs the data delivery as if

it were over the wired networks. Such routing protocols provide reliable communication for the stable network environment, i.e., connected and no hostile nature. But for the challenging wireless environments, characterized by inevitable link losses and frequent disconnections, the data delivery performance becomes degraded [12, 13, 14, 15].

A batch of data delivery solutions for intermittent or partially disconnected networks also known as opportunistic networks (OppNets) or delay tolerant networks (DTNs) were proposed in early 2000s [16, 17] and new solutions are evolving [18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28]. The fundamental idea behind these protocols is the store-carry-forward (SCF) routing paradigm. It has been shown that SCF routing can achieve eventual data delivery for opportunistic networks. In SCF routing, a node with a lower ID proactively sends its summary vector to its contact node and each node in turn gets the copies of messages it has not yet seen from the other. The nodes keep the copies and replicates whenever there is a contact opportunity. Longer data delivery delay, resource consumption and information epidemics become the major problems in SCF routing. Various forms of SCF routing protocols have been proposed and can be broadly classified as epidemics-based routing [29, 30, 21], quota-based routing [31, 20, 24], probabilistic-based routing [32, 33], mobility-based routing [34, 35], and social-based routing [26, 18, 22, 36].

Several packet discarding strategies have also been proposed to control and eradicate the spread of packet copies in networks [37]. The VACCINE recovery scheme is shown to be the most effective recovery scheme in the existing works [37, 38, 39]. By using VACCINE (anti-packet), susceptible node will become immune and infected node will become recovered and will not infect other susceptible nodes when it receives the anti-packet.

In addition to the intermittent connectivity due to the mobility, the presence of stationary and mobile scatterers in the environment results in fading in the network, which further incapacitates the contact opportunities [12, 13, 14, 15]. Most routing protocols in MANETs and OppNets did not consider the impact of fading caused by stationary and mobile scatterers on the network performance. Instead, most existing protocols evaluate the performance based on the deterministic modelling of the propagation channel.

Cooperative forwarding or receiver diversity-based routing is shown to be an efficient routing mechanism in static and mobile wireless networks under fading environments [12, 40, 14, 41, 42]. It takes into account the probabilistic nature of wireless channel. In this forwarding, one or more forwarding nodes are located in between the source and destination nodes that are not in contact. Data packet is broadcasted over the air and the relay nodes forward the data packet in a cooperative fashion to the destination node. This is in contrast to generic unicast forwarding used in MANET and OppNet protocols. In mobile environment, this type of forwarding requires nodes position information or complete network topology information prior to the data delivery process [14, 41]. Therefore, this type of forwarding cannot be directly applied in opportunistic networks.

As discussed above, network topology can change dramatically due to the node mobility and probabilistic propagation channels caused by fading. As a result, it is hard to predetermine the network scenario as the node density could change drastically at any point in time. In summary, a robust data delivery solution which addresses successful data delivery, data delivery with minimum delay, buffer consumption, and information epidemics control under diverse network connectivity characteristics is needed. Hence, in this thesis, we focus on data delivery solutions which jointly addresses all these

issues for challenging wireless environments.

1.2 Motivation

Store-carry-forward routing mechanism has been widely used for data delivery in opportunistic networks. However, it has been shown that massive packet replications and unnecessary buffer consumption are inevitable in this scheme [29, 17, 37, 38, 39]. To control the packet replications, some protocols limit the number of copies made for a packet or the number of hops a packet traverses [31, 20, 24]. Some use randomly generated probabilities to decide to forward/replicate a packet by direct, two-hop, probabilistic, and epidemic forwarding approaches [32, 33]. However, it has been shown that in these approaches, the message delivery delay becomes higher compared to the pure epidemic routing when there is no constraint on buffer and bandwidth resources.

Some adaptive routing schemes, which basically are the integration of SCF routing and end-to-end MANET routing, have been proposed [28, 23, 43]. Nodes find an end-to-end path and forward the data packet. SCF-based forwarding is used when the end-to-end connection breaks or the path cannot be found. Some select multiple forwarding paths and replicate data packet copies along the predetermined multiple paths [21]. In short, these approaches combine end-to-end routing and SCF-based routing to shorten delivery delay and minimize resource consumption and information epidemic. However, as discussed in Section 1.1, network topology can change dramatically due to mobility and probabilistic propagation channels caused by fading. These protocols did not consider the impact of probabilistic propagation channel on the network connectivity. Therefore, under the situation where

the end-to-end path cannot be found, it will fall back to the SCF-based routing which therefore likely to incur longer delay due to route discovery compared to pure epidemics routing.

Recovery schemes for controlling the spread of packet replications were proposed in [37]. The VACCINE recovery scheme is shown to be the most effective one in the existing works [37, 38, 39]. In this approach, a vaccine is generated when a node successfully delivers a packet to the destination. However, in opportunistic links and dynamic topology environments, due to the effect of mobility and fading, having a direct contact with the destination is scarce. Therefore, this results in late information epidemics control and more infections.

From these observations, we focus on robust and resilient data delivery solution under challenging wireless environments with mobility and probabilistic propagation channel. The objective is to provide higher data delivery with the lower end-to-end delay, buffer consumption, and unnecessary packet transmissions for networks with opportunistic links and dynamic topologies.

We then consider the data delivery in opportunistic network from social network perspectives. It has been discussed in recent works about the presence of social nature such as communities and centralities in opportunistic networks [36, 44, 45, 26]. Some mobility models which consider the social ties between nodes and community/group movement patterns have been proposed [46, 45, 2, 47, 1]. In addition, some routing protocols known as community/group/cluster/social aware routing protocols have also been proposed [48, 18, 22, 36]. These protocols use hybrid nature of routing where multi-hop routing is used within a group and SCF routing is used between groups. Given the community/group nature, buffer sharing is used within the group and therefore relax the resource constraints. In these approaches, it assumes

that data packets can be transferred reliably without any wireless link disruptions during a meeting period with a contact. Under the probabilistic propagation nature, this type of assumption poses great challenges in data delivery.

Cooperative forwarding or receiver-diversity based forwarding has shown a substantial network performance improvement under the unreliable and lossy wireless nature [12, 40, 14, 41, 42]. It takes into account the unreliable nature of wireless channel, utilizes broadcast wireless medium and combines several weak/unreliable links to make a reliable communication link. It has been shown that cooperative forwarding provides a resilient data delivery for networks with adverse wireless links. From these observations, we claim that with the presence of communities and centralities nature in opportunistic network, it assures to exploit receiver diversity-based forwarding in such environment, and therefore be able to maximize the given contact opportunity for the reliable data packet transfer, which is the next focus of this thesis.

The fundamental design components of cooperative forwarding include candidate relay set selection, candidate relay prioritization, broadcasting the data packet to the candidate relays, and best relay selection using a coordination scheme [42, 15]. We observe that, with the existing cooperative forwarding design, in the presence of mobility and multiple cross traffic flows in the network, it is possible to have 1) duplicate transmissions caused by missing communications among forwarding nodes; 2) multiple schedules for multiple flows and inaccuracy in scheduling due to the unavailable medium; and 3) unnecessarily waiting for the better candidate node transmission when it is the only node who receives the packet correctly.

Most cooperative forwarding protocols are proposed for stationary wireless networks [42]. Some recent works study on exploiting cooperative for-

warding opportunities in mobile networks [14, 41]. However, those approaches require topology information using proactive source routing or position information to locate the destination. Although the efficacy of exploiting cooperative opportunistic forwarding under fading environment was demonstrated, there is a performance drop when the network connectivity becomes intermittent due to the increasing network size. From these observations, we first focus on the cooperative forwarding design to be able to deploy in both static and mobile wireless networks in the presence of multiple cross traffic flows and without requiring topology or position information in advance. We then focus on deploying the cooperative forwarding design in opportunistic networks under community/group connectivity characteristics. The solution is called cooperation-aided data delivery solution.

1.3 Major Contributions

In the light of the problems presented in Section 1.2, the main contributions of this thesis can be summarized as follows.

1. We propose a network layer solution called store-carry-cooperative forward (SCCF) routing with information epidemics control. It aims to provide higher data delivery with the lower end-to-end delay, buffer consumption, and unnecessary packet transmissions for networks with opportunistic links and dynamic topologies. The contribution of this work can be summarized as follows.
 - We propose an adaptive data forwarding algorithm where nodes proactively monitor the direct/2-hop cooperative forwarding opportunities and deliver the data packet. The nodes adaptively switch between the cooperative forwarding and SCF forwarding

based on the contact opportunity. The SCF forwarding here is reactive in which nodes replicate the packet copies only when the message has never been forwarded by cooperative forwarding.

- We propose a new information epidemics control algorithm where nodes continuously monitor the broadcast-based cooperative transmissions over the medium. The nodes then translate the information overheard into packet states and then build the knowledge base for the packet delivery progress. This knowledge is used to deter packet infections and vaccinate others from being infected.
 - We study the effectiveness of the proposed solution based on the susceptible-infected-recovered (SIR) model and prove that it provides higher vaccination rate and recovery rate. Due to the flexibility of the proposed scheme, it can also be integrated with any SCF-based routing.
 - We study the network performance of the proposed solution under various network scenarios which include fading, obstacles, varying speed, and varying density. The simulation study is also conducted under a social mobile network model which provides real-life scenarios such as the Cambridge/Haggle datasets.
2. We propose a cooperative forwarding protocol called cluster-based cooperative data forwarding (CCDF) for multi-flow wireless networks. The proposed protocol design aims to relax the pre-selection of candidate relay set, candidate relay prioritization, and scheduling requirements in cooperative forwarding. It also aims to solve the duplications, scheduling inaccuracy and coordination delay problems. The proposed scheme can be used in both static and mobile wireless networks. The

contribution of this work can be summarized as follows.

- The cluster-to-cluster expected transmission count (*C2CETX*) metric is introduced to compute cluster path for forwarding. Data packets are broadcasted over the cluster path. Distributed agent-based coordination is proposed to support best candidate relay selection and duplicate transmission avoidance. With the agent-based coordination, it is shown that the forwarding nodes need not require to have global/local scheduling. In addition, it also prevents several potential duplications caused by 1) missing higher priority node transmissions; and 2) scheduling inaccuracy due to unavailable medium.
- Analysis on the coordination time required by the DAC mechanism and comparison with the one required by existing timer based coordination method are conducted. It shows that the proposed mechanism takes much shorter coordination time compared to the existing coordination method. We study the performance impact when the same candidate relay set is being used for multiple cross traffic flows in the network.
- The integration of CCDF with multi-radio multi-channel (MRMC) capability is proposed to further enhance the system throughput, especially in the presence of highly congested multiple cross traffic flows. We show that the CCDF with MRMC not only retains the advantage of receiver diversity in cooperative forwarding but also minimizes the interference, which therefore further enhances the system throughput for the network with multiple cross traffic flows.

3. We propose a cooperation aided data delivery solution for opportunistic networks under community/group connectivity characteristics. The proposed solution does not require to have topology or position information in advance.
 - We propose an adaptive cooperative data forwarding algorithm. Nodes proactively monitor the direct/2-hop cluster-based cooperative data forwarding (CCDF) opportunities and deliver data packets. Based on the given contact opportunity, nodes adaptively switch between the CCDF and cooperation-aided replication (CaR)-based forwarding approaches. The CaR approach here is reactive in which nodes replicate the packet copies only when the message has never been forwarded by CCDF.
 - We propose a group/community aware information epidemics control algorithm. With the proposed control mechanism, information epidemics in community network is manageable and will never lead to outbreak. We show that the number of infections incurred is only a small fraction of the pure epidemic-based routing. We also show that the number is almost constant regardless of the network density. Due to the flexibility of the proposed scheme, it can also be integrated with any group-aware SCF-based routing.
 - We then study the network performance of the proposed solution under community/group mobility. The evaluation is conducted with varying network connectivity ranging from highly dense environment to highly sparse environment and under the presence of probabilistic propagation channels. Performance evaluation on successful data delivery rate, delivery delay, overhead, and infor-

mation epidemics are studied and discussed.

1.4 Thesis Organization

The rest of the thesis is organized as follows:

- Chapter 2 reviews the community/group mobility models widely used in the WANET research for network simulations and performance evaluations.
- Chapter 3 proposes a data delivery solution called store-carry-cooperative forwarding (SCCF) which aims to solve the long delivery delay and the information epidemics problem in store-carry-forward routing in opportunistic networks.
- Chapter 4 proposes a data delivery solution called cluster-based cooperative forwarding (CCDF) which aims to 1) relax the pre-selection of candidate relay set, candidate relay prioritization, and scheduling requirements, 2) solve the duplications, scheduling inaccuracy and coordination delay problems; and 3) enhance the system throughput under multiple cross traffic flows and unreliable wireless channel.
- Chapter 5 proposes a cooperation aided data delivery solution for opportunistic networks under community/group connectivity characteristics.
- Chapter 6 concludes the thesis and sketches out future research directions.

Chapter 2

A Review of Group Mobility Models for Mobile Ad Hoc Networks

2.1 Introduction

Mobile ad hoc network (MANET) is a form of infrastructureless mobile network with no fixed centralized controller such as base stations for coordinating the communications among the mobile terminals in the network. In MANET, all nodes can be mobile. The nodes cooperate to dynamically form routing paths and relay each other's packets for end-to-end communication [9]. For a thorough performance study of MANET routing protocols, it is important to use mobility models which accurately represent the movements of nodes in the MANET [49]. Existing MANET mobility models can be generally classified into two categories: entity and group mobility models. In entity mobility models, nodes move independently of each other. On the other hand, in group mobility models, nodes tend to cluster and move in

groups. In earlier stages of MANET research, entity mobility models were the most common, if not the only type of mobility models used [49]. However, in recent years, mobile nodes in real-life MANETs were often observed to exhibit some form of group behaviour [1, 2]. This motivates the formulation of a variety of group mobility models based on real-life military and civilian application scenarios. The former refers to battlefield operations involving movements by military units such as fighting troops and tank fleets. The latter may include disaster recovery (e.g. fire fighting, search and rescue) and law enforcement (e.g. SWAT, crowd and riot control) operations. All of these applications tend to exhibit movement patterns and connectivity characteristics that are quite different from those of entity mobility models. This necessitates the use of a different mobility model for evaluating the performance of routing protocols in such applications. Recently, a number of group mobility models have been proposed and studied with different routing protocols [1, 3, 4, 5, 6, 7, 2, 8, 50]. These studies showed that group movements can indeed significantly impact the routing performance. In the literature, there exist a number of survey papers on mobility models for wireless networks. The classification of mobility models into entity and group mobility models was first proposed in [49]. Mobility models have also been classified based on the degree of randomness of node movements such as trace-based, constrained topology based, and statistical models [51]. In [52], the authors reviewed the mobility models for tactical mobile networks, which are one of the common environments where group movements can be found. However, the authors did not focus explicitly or cover in-depth on group mobility. They classified the mobility models according to five types of movement dependencies or restrictions, among which “spatial dependencies” is one that encompasses group mobility. In this chapter, we focus on surveying the state-of-the-art in

group mobility models. Specifically, we review seven most well-known and representative group mobility models [1, 3, 4, 5, 6, 7, 2]. We first classify the models according to how the mobile nodes move in a group. We then compare the models based on six design features and their target applications. Finally, we discuss the potential of these models to be used or adapted for use in other emerging network types, including vehicular ad hoc networks, mobile sensor networks, opportunistic and delay tolerant networks.

2.2 Classification of Group Mobility Models

In group mobility models, mobile nodes are divided into groups, each having a coverage area of a certain radius. The mobile nodes located within the coverage area of a group are considered to be the members of that group. Each group can move to any location within the network, whereas members of a group can only move within the group's coverage area. The most well-known and representative group mobility models in the literature [1, 3, 4, 5, 6, 7, 2] can be classified according to how the mobile nodes move as a group. In [1, 3, 4, 5], the nodes in a group follow a lead point whose movement dictates that of an entire group. Here, the lead point can be either a real physical node, or more commonly a logical "center" that exists within a group. We refer to these models as point based group mobility (PBGM) models. In [6, 7, 2], instead of following a lead point within a group, nodes may follow a path through a sequence of regions or areas. We refer to these models as region based group mobility (RBGM) models. Both PBGM and RBGM can be further classified based on the existence of interaction among group members. In [1, 3, 6, 7], group members move independently of each other, and their movements are influenced only by their

Figure 2.1: Classification of group mobility models. References to these models are: RPGM [1], RVGM [2], GFMM [3], MGCM [4], RRGM [5], VTGM [6], and CMM/ECMM [7, 8]

lead point or targeted regions. These models are said to exhibit individual group member (IGM) movements. In [4, 5, 2], the movement of a group member may be influenced by or correlated with other group members due to some interaction or relationships that exist between them. For these models, they are said to exhibit coordinated group member (CGM) movements. In summary, the group mobility models can be classified as PBGM with IGM, PBGM with CGM, RBGM with IGM, and RBGM with CGM. This group mobility classification is shown in Fig. 2.1.

2.3 Point Based Group Mobility Models

In PBGM models, each group member follows the movements of a physical or logical lead point that exists within the group.

2.3.1 PBGM with IGM Movement

In PBGM with IGM models, each group member further moves independently of each other, only to be influenced by the movements of its lead point. The following reviews two prominent examples of such models: RPGM [1] and RVGM [3].

Reference Point Group Mobility (RPGM) Model

RPGM [1] is one of the most commonly used group mobility models for MANETs. At the center of each group's coverage area is a logical lead point whose movement defines that of the entire group, including its speed, direction, and acceleration. The trajectory of the lead point can be predefined or obtained based on a particular entity mobility model. Furthermore, each group member has its own reference point that follows the lead point and maintains a constant distance and direction from it. Whenever the reference point of a group member moves to a new location, this group member also moves to a randomly chosen location within a circular neighbourhood of radius R around its new reference point location. RPGM defines two motion vectors: i) group motion vector $\overrightarrow{GM}$, which represents the movement of the lead point that characterizes the group movement; and ii) random motion vector $\overrightarrow{RM}$, which represents the movement of a group member with respect to its reference point. The new position of a group member is computed as the sum of these vectors. Figure 2.2 gives an example of a group moving from time t to time $t+1$. When a lead point moves from time t to time $t+1$ based on $\overrightarrow{GM}$, the reference point of each group member also moves to a new location based on $\overrightarrow{GM}$. Then, based on $\overrightarrow{RM}$, the position of each group member will be updated to a new location in the neighbourhood of its new reference point location. In [1], three motion patterns based on RPGM,

namely in-place, overlap, and convention, which represent group mobility in battlefield, disaster recovery, and convention scenarios, respectively, were benchmarked against random mobility model and evaluated for their impact on three MANET routing protocols: Ad-hoc On-demand Distance Vector (AODV), Hierarchical State Routing (HSR), and Destination Sequenced Distance Vector (DSDV). The results showed that compared with random mobility, group mobility under RPGM leads to a lower rate of connectivity changes. This in turn resulted in higher throughput and lower overhead for AODV and HSR, particularly when communications were confined to within each group. DSDV failed to leverage on group mobility to improve its performance due to its inherent high routing load. RPGM can be considered as a generic group mobility model, which by changing its parameter values, can mimic group mobility of different real-life scenarios. Firstly, we can arrange the position of each reference point and adjust its radius to generate a particular moving pattern inside a group for a specific operation or activity. Secondly, we can vary the network connectivity among groups by setting the groups' radii. If the group radius is small, a sparse network can be formed. Generally, the smaller the coverage area of a group, the higher the probability the network will be disconnected. Finally, we can define the motion path of each group by specifying manually or automatically using digital maps, the sequence of check points or intermediate locations through which the group will move. This can provide the flexibility for RPGM to be applied in some user-defined scenarios.

Reference Velocity Group Mobility Model (RVGM)

RVGM [3] is an extension of RPGM by replacing the concept of reference point and motion vectors with two velocity vectors to model group mobility.

Figure 2.2: Movement of group members in RPGM.

The two velocity vectors are: group velocity of the lead point, and local deviation velocity of each group member. The former defines the movement of the lead point (and thus the group) in terms of speed and direction, and the latter defines the amount of deviation in velocity of a group member from that of the lead point. The movement of a group member is then given by the sum of these two velocity vectors. Similar to RPGM, the movement of the lead point controls the movement of the entire group, and its trajectory can be predefined or based on an entity mobility model. The occurrence and timing of network partitioning may also be predicted through identifying groups with significantly different lead point velocities. This can assist upper layer routing and other protocols to react early to potential network partitioning and thus minimize the impact of service disruption. However, the circular group coverage and time-invariant group velocity assumptions made by the prediction algorithm of RVGM may not be realistic in some real-life scenarios. A more accurate prediction could be made by using actual information about the group mobility disseminated through some scalable mechanisms.

Moreover, in order to always keep the group members to moving within the group, the distribution of the local deviation velocity should be carefully chosen. Otherwise, it may result in the groups to become increasingly loosely connected over time.

2.3.2 PBGM with CGM Movement

In PBGM with CGM model, the movement of a group member is not only influenced by its lead point, but also by other group members.

Group Force Mobility Model (GFMM)

GFMM [4] is similar to RVGM in that the velocity of a group member follows the velocity of its lead point with a small random deviation. However, forces (repulsion and attraction) between nodes are introduced to provide a mechanism for nodes to avoid physical node collision and congestion. The strength of force is based on the distance between two nodes and the force distributions. Repulsion force prevents collision between two nodes from any group. As the distance between nodes increases, the repulsion force decreases exponentially to zero and may become attraction force. The attraction force exists only between nodes of the same group. By selecting different attraction force distributions, different levels of attraction force can be generated to create different characteristics (from loose to tight grouping). In [4], three attraction force distributions, namely F, Rayleigh, and Chi-Squared, which represent loose, moderate, and tight grouping, respectively, were used to generate three GFMM mobility patterns. These mobility patterns were benchmarked against RPGM in a MANET using AODV routing protocol. The results showed that compared with RPGM, GFMM can lead to lower spatial correlation between nodes a similarity measure of the velocity and movement

direction between nodes, due to the attraction forces inducing a change in direction of the nodes to avoid collisions. For the same reason, GFMM showed a shorter average link duration as compared to RPGM. These results point towards a more volatile group structure when simulating group mobility under the more realistic GFMM. The collision avoidance mechanism of GFMM should be easily applied to other group mobility models for modelling the effects of physical interactions between nodes on the group movement in real life.

Multi-Group Coordination Mobility (MGCM)

In MGCM [5], all groups move in a coordinated fashion towards a common object. A two-level group architecture, group and main group, is introduced, in which all groups in the network form a main group. Each group has a lead point, and one of the lead points is selected as the lead point of the main group. The lead points of other groups are then treated as group members of the main group. The velocity vector of a lead point for group l , denoted as $\vec{B}_l$, can be described as follows.

$$\vec{B}_1 = \vec{B}_m + \vec{B}_p + \vec{B}_d \quad (2.1)$$

where $\vec{B}_m$ is the velocity vector of the lead point for the main group, $\vec{B}_p$ is the force exerted by other peer lead points, and $\vec{B}_d$ is a small random deviation. The force $\vec{B}_p$ is used to maintain a certain distance between lead points to avoid collision among the groups and it only exists when the distance between two lead points is smaller than a predefined threshold. Similarly, the movement of a group member is determined by that of its lead point and further influenced by its peer group members and some deviation. In Fig. 2.3, the main group consists of three groups with lead points, a, b,

and c. All three groups are moving towards a common object. Lead point a is selected as the main lead point. For lead point b, since its distance to a is smaller than the predefined threshold, a peer force from a exists to widen the distance between the groups of a and b. Hence, the velocity vector $\vec{B}_b$ of lead point b is determined by velocity vector $\vec{B}_m$ of main lead point a, force $\vec{B}_p$ from peer lead point a, and a small deviation $\vec{B}_d$. In [5], MGCM was benchmarked against RPGM and RRGM. The results on average instantaneous node speed showed that MGCM does not suffer from the well-known average speed decay problem of most random mobility models that makes them less realistic. MGCM also showed a lower rate of link changes than RPGM, but a higher rate as compared to RRGM. The latter was due to all nodes in a group moving at the same pre-defined speed, while the former was attributed to movement coordination in MGCM which led to a lower degree of freedom for nodes to select a motion direction. In MGCM, the repulsion existing among groups and group members makes MGCM a better fit for scenarios where collision avoidance is required, such as in vehicular ad hoc networks (VANET) [53]. Moreover, the group coordination exhibited in MGCM tends to keep the entire network connected because all groups are moving in unison towards the same destination object.

In RBGM models, each group member may follow a path through a predefined or real-time dynamically determined sequence of regions or areas.

2.4 Region Based Group Mobility Models

In RBGM models, each group member may follow a path through a predefined or real-time dynamically determined sequence of regions or areas.

Figure 2.3: Movement of groups in MGCM.

2.4.1 RBGM with IGM Movement

In RBGM with IGM model, each group member further moves independently of each other, only to be influenced by the location of the next region or area to which it is moving. The following reviews two prominent examples of such models: RRGGM [6] and VTGM [7].

Reference Region Group Mobility (RRGM) Model

In RRGGM [6], the movement path of each group is defined by a sequence of reference regions leading to some target destination. The reference region is a rectangular or circular area whose size is determined by user-specified node density and group size. The location of each reference region is determined in real-time and in sequence such that each is progressively closer to the target destination. Once a reference region is identified, each group member selects a location inside this region and moves towards it. On arriving, the group member will move within its reference region according to random waypoint model while waiting for other members of the same group

to arrive. After all group members have arrived in the reference region, they will remain in that region for some time (referred to as pause time), before the next reference region is determined. This process repeats until the group arrives at the target destination. At some user-specified time, a new target destination can emerge to cause splitting of an existing group into smaller groups, one of which will move towards the new destination. Furthermore, groups having arrived at their target destinations for a period longer than the pause time may move to merge into a larger group. RRGM can well reflect real-life scenarios because it introduces mobility patterns which are extracted from real mission-oriented operations such as fire fighting where nodes come together initially and then divided into small groups to execute different tasks in different areas. By providing users with a flexible means of configuring the model parameters, a wide range of realistic group movements including merging and splitting scenarios could be modelled, which broadens the applications of RRGM.

Virtual Track Group Mobility (VTGM) Model

In VTGM [7], each group member moves along a predefined path called virtual track, which originates and terminates at some junction points known as switch stations. As shown in Fig. 2.4, a number of switch stations are placed at junction points of the virtual tracks, each having a certain track width. Members of the same group are distributed on the same track and move towards the next switch station in a way similar to random waypoint model, but their movements are confined within the virtual track. In addition, the group can merge with other groups or split into smaller groups at switch stations. VTGM also introduces non-group nodes, which can remain stationary or move freely without constraints of the virtual tracks and switch

stations.

In [7], VTGM was benchmarked against random waypoint model in a MANET operating AODV routing protocol. The MANET topology was abstracted from a highway map where highways and intersections represent the virtual tracks, and switch stations, respectively. The results showed that the performance in terms of packet delivery ratio, throughput, and end-to-end delay, was generally worse in VTGM due to weaker inter-group connectivity as compared to random waypoint model. Thus, additional non-group nodes may be deployed as forwarding nodes outside of the virtual tracks to enrich the connectivity among groups and improve network performance.

VTGM can precisely mimic a number of practical scenarios involving constrained node movements. For example, switch stations can be regarded as gathering points for commuters and virtual tracks can be viewed as streets. Groups' movements can be treated as buses with on-board passengers travelling along the streets and non-group nodes can represent other motorists or pedestrians within the area. As shown in the lower left corner of Fig. 2.4, some passengers in one bus may alight (group splitting) and board the other at gathering point (switch station). Similarly, passengers of a pre-terminating bus, e.g. due to accident or breakdown, may board an alternative bus at gathering point (group merging) as shown in the upper right corner of Fig. 2.4. The use of this model is not necessarily limited to the vehicular context. The virtual tracks may also represent the planned paths to be taken by a walking tour guide leading a group of tourists to visit the various attractions (switch stations) within the locality. VTGMs concept of tracks to constrain nodes' trajectories could be made even more realistic by incorporating some additional rules for avoiding physical collisions among individual nodes and between groups.

Figure 2.4: Movement of group and non-group nodes in VTGM.

2.4.2 RBGM with CGM Movement

In RBGM with CGM model, the movement of a group member is not only influenced by the location of its current and neighbouring regions or areas, but also by other group members.

Community based Mobility Model (CMM)

CMM [2] is one of the earliest proposed mobility model based on social network theory. In CMM, each community is defined as a group of nodes that have strong social links with each other, and consequently a likelihood of being co-located together. An interaction matrix (IM) where each element has a value ranging from 0 to 1 (low to high) is used to represent the degree of social interaction between any two nodes. This IM is then converted into a connectivity matrix (CM) where each element is a binary value that indicates whether or not the associated pair of nodes should be socially linked or connected. Consequently, the highly connected set of nodes will be grouped

together to form a community. Each community is associated with one specific region inside the network area, and each member of the community is randomly placed within its community region as an initial location. In addition to identifying community or group memberships, movements of the nodes are also influenced by the social links between them. A node will move to a randomly selected location within a community region to which it is most socially attracted. The social attractiveness (SA) of a community region to a node is defined by the average strength of the social links between this node and members of that community. The target location chosen by each node for its next movement can be inside its current community region or in other community region, depending on which region exerts the highest SA on the node. Once a location is chosen, the node can move towards that location based on any entity mobility model. It is worth noting that under CMM, although nodes are grouped and co-located together by their social relationships, their movements may not result in a group movement since each node moves according to its own social relationship with other members of the same or different group. An enhanced version of CMM (ECMM) was recently proposed which incorporates a feature that encourages group mobility [8]. This is achieved by introducing a group movement encouragement factor k . A high k value strengthens the social attraction of nodes that have chosen to commute to another community region for their next movement. This increases the tendency for some nodes to be attracted to and follow those which are commuting to another community region, resulting in a group movement. However, it should be noted that ECMM only encourages and not enforces group mobility, and the resulting group movement typically does not involve all but only a subset of members of the group. In [8], the mobility traces generated by ECMM were compared against CMM

and HCMM (Home Cell Mobility Model – an extension of CMM incorporating social attractions exerted by physical spaces) as well as two sets of real mobility traces collected from human subjects in a conference context. The results showed that the mobility traces of ECMM resembled more closely to the real ones in terms of their inter-meeting time and contact duration distributions. The former refers to the time interval between two consecutive meetings between two nodes, while the latter refers to the duration for which two nodes maintain radio contact. It was further found that group mobility encouragement factor k does not affect the inter-meeting time, but prolongs the contact duration as k increases.

2.5 Abstraction of Design Features in Group Mobility Models

From our discussed models in Sections 2.3 and 2.4, six design features can be identified as follows:

1. *Spatial constraints*: These refer to constraints imposed by the nodes' spatial environment that restrict their movements in a certain way. Such constraints may include roads, walkways, rail tracks, and transportation structures such as bridges and tunnels. VTGM is an example of a model that incorporates spatial constraints in the form of virtual tracks.
2. *Collision avoidance*: This refers to controlling the movement of a node to avoid physical collision with other nodes. GFMM introduces repulsion force between neighbouring nodes to achieve this goal. MGCM similarly adopts the repulsion force approach for collision avoidance

among groups.

3. *Stable group structure*: This refers to maintaining the relative distances and positions among the group members during a group movement, such as when a group of nodes (e.g. troops, tanks, robots) move in formation, or on-board a transport carriage (e.g. bus, truck, train). RPGM can provide a stable group structure by properly placing the reference points and limiting their coverage radius. VTGM can also provide such a feature by limiting the degree of internal random mobility.
4. *Group destination*: This refers to the target destination of a moving group, which can be randomly chosen or predefined. In models such as RPGM and RVGM, the groups do not have any restrictions in choosing a target destination. Thus, they are free to move to anywhere inside the network area. In other models such as MGCM and RRGm, each group moves toward a particular destination that is usually defined a priori. In VTGM, all switch stations are potential destinations for the groups.
5. *Group coordination*: This refers to the existence of some cooperation or interaction among groups, which allows the movement of these groups to be coordinated. Such coordination exists in real-life scenarios such as that exhibited between groups when moving through a road traffic intersection. MGCM is an example of a model that incorporates group coordination.
6. *Group merging/splitting*: This refers to the provision for two or more groups to merge into one when they come together, or for a group to split into two or more smaller groups, each moving towards a different

destination. RRGm, VTGM, and ECMM are models that incorporate such a feature.

A comparison of the discussed group mobility models based on their design features and target applications is provided in Table 1. Note that none of the models include all the six design features as not all features are required by the applications for which they are designed for. Our aim is to provide users of these models with the list of design features to consider when adapting them, or creating new models with the desired set of features to suit their specific applications.

It should be noted that even though these group mobility models [1, 3, 4, 5, 6, 7, 2, 8] have been proposed mainly for the performance study of MANETs, they can be used or adapted for use in the performance study of other network types that involve group mobility. For example, in VANETs [53], vehicles entering a street or highway are modelled to move in the same direction for some time, which resembles a group movement that can be modelled using a group mobility model such as VTGM incorporated with a street map and collision avoidance. In mobile sensor networks where nodes move in groups such as in monitoring sensor tagged animals that live in groups, existing models such as ECMM that considers the social influence on node movement could be used as a basis for developing group mobility models for specific wildlife communities. The performance of opportunistic and delay tolerant networks (DTNs) [27], which are characterized by intermittent communication opportunities, frequent disconnection, and long delays, have been mostly studied based on entity mobility models. Thus, it would be interesting to also investigate the performance of such networks, their protocols and applications under different group mobility patterns. Another potential direction for future research is the design of a generic parametric

group mobility model with an automated mechanism to populate the model parameters based on mining captured datasets of real mobility traces. Such a model can facilitate the automatic modelling of any new types of group mobility patterns of existing or new mobile entities (e.g. tactical drones) that may emerge in the future.

Table 2.1: Comparison of the models based on their design features and applications

Model	Spatial Constraints	Collision Avoidance	Stable Group Structure	Group Destination		Group Coordination	Group Merge/Split	Target Applications
				Random	Predefined			
RPGM			✓	✓				Battlefield operation, disaster recovery, convention mobility
RVGM				✓				Similar to RPGM
GFMM		✓		✓				Any application requiring non-colliding interaction between nodes
MGCM		✓			✓	✓		Military applications requiring motion coordination among multiple groups
RRGM					✓		✓	Applications requiring group splitting and merging, including search and rescue, battlefield, exhibition hall visiting
VTGM	✓		✓		✓		✓	Any applications requiring nodes to move on predefined paths, including roads, trails, highway systems with intersections
CMM / ECMM					✓		✓	Scenarios where nodes gather and interact socially with each other, such as in conference venue, school, and workplace

2.6 Conclusion

In this chapter, we have reviewed some of the most representative group mobility models proposed for MANET research. We classified the models into four categories based on the movement of the group members: PBGM with IGM, PBGM with CGM, RBGM with IGM and RBGM with CGM. We first concisely described how the group and group members move in each model based on our classification, including our insights from analysing each model. We then summarized and compared the models based on six design features and their target applications. The list of design features is provided for users of these models to consider when adapting them or creating new models with the desired set of features to suit their specific applications. Finally, we discussed some directions for future research on modelling group mobility patterns not only for MANETs, but also other types of networks.

Chapter 3

Store-Carry-Cooperative Forward Routing with Information Epidemics Control for Data Delivery in Opportunistic Networks

3.1 Introduction

Opportunistic networks (OppNets) are a form of mobile ad hoc networks (MANETs) where the network topology is composed of opportunistic and random links. Some applications of opportunistic networks include mobile social networks, disaster recovery networks, and military networks. For such networks, end-to-end routing is shown to be inefficient for data delivery due to opportunistic links and dynamic topology. The store-carry-forward (SCF) based approach has become a conventional data forwarding mechanism in

such networks [16, 17].

It has been shown that SCF routing can achieve eventual data delivery for dynamic and random network topologies [17]. In SCF routing, a node with a lower ID proactively sends its summary vector to its contact node and each node in turn gets the copies of messages it has not yet seen from the other. The nodes keep the copies and replicate whenever there is a contact opportunity. Due to the lack of the data delivery information, a node will try to replicate the packet to all the neighbouring nodes it encountered [29]. Therefore, massive packet replications are inevitable in SCF routing. Moreover, it is possible that the destination node is in the vicinity of the node holding the copies under some network situation. So, simply replicating to all the neighbouring nodes causes unnecessary transmissions and may lead to longer delivery delay.

Several packet discarding strategies have been proposed to control and eradicate the spread of packet copies in networks [37]. The VACCINE recovery scheme is shown to be the most effective recovery scheme in the existing works [37, 39, 38]. By using VACCINE (anti-packet), susceptible node will become immune and infected node will become recovered and will not infect other susceptible nodes when it receives the anti-packet.

In addition to mobility, the presence of stationary and mobile scatterers in the environment results in fading in the network, which further incapacitates the contact opportunities. Most routing protocols in MANET and opportunistic networks did not consider the impact of fading caused by stationary and mobile scatterers on the network performance. Instead, most existing protocols evaluate the performance based on the deterministic modelling of the propagation channel.

Cooperative forwarding is shown to be an efficient routing mechanism in

static and mobile wireless networks under fading environments [14, 41, 12]. It takes into account the probabilistic nature of wireless channel. In this forwarding, one or more forwarding nodes are located in between the source and destination nodes that are not in contact. Data packet is broadcasted over the air and the relay nodes forward the data packet in a cooperative fashion to the destination node. This is in contrast to generic unicast forwarding used in the traditional routing. In mobile environment, this type of forwarding requires nodes position information or complete network topology information prior to the data delivery process. Therefore, this type of forwarding cannot be directly applied in opportunistic networks.

As discussed above, network topology can change dramatically due to mobility and probabilistic propagation channels caused by fading. As a result, it is hard to predetermine the network scenario as the node density could change drastically at any point of time. Hence, adaptive routing which is robust to different network scenarios and provides highest data delivery with minimum delivery delay, overhead, buffer consumption, and packet replications is needed.

In this chapter, we propose a network layer solution called store-carry-cooperative forward (SCCF) routing with information epidemics control. It aims to provide higher data delivery with the lower end-to-end delay, buffer consumption, and unnecessary packet transmissions for networks with opportunistic links and dynamic topologies. The major contributions of this work are summarized as follows.

1. We propose an adaptive data forwarding algorithm where nodes proactively monitor the direct/2-hop cooperative forwarding opportunities and deliver the data packet. The nodes adaptively switch between the cooperative forwarding and SCF forwarding based on the contact op-

portunity. The SCF forwarding here is reactive in which nodes replicate the packet copies only when the message has never been forwarded by cooperative forwarding.

2. We propose a new information epidemics control algorithm where nodes continuously monitor the broadcast-based cooperative transmissions over the medium. The nodes then translate the information overheard into packet states and then build the knowledge base for the packet delivery progress. This knowledge is used to deter packet infections and vaccinate others from being infected.
3. We study the effectiveness of the proposed solution based on the susceptible-infected-recovered (SIR) model and prove that it provides higher vaccination rate and recovery rate. Due to the flexibility of the proposed scheme, it can also be integrated with any SCF-based routing.
4. We study the network performance of the proposed solution under various network scenarios which include fading, obstacles, varying speed, and varying density. The simulation study is also conducted under a social mobile network model which provides real-life scenarios such as the Cambridge/haggle datasets.

The rest of the chapter is organized as follows. Section 3.2 reviews related works on routing and information epidemics control schemes. The overview of the proposed scheme is provided in Section 3.3. Section 3.4 details the link awareness approach. The proposed routing scheme with several protocol details are presented in Section 3.5. The proposed vaccinating algorithm and the integration with the data forwarding module are discussed in Section 3.6. Section 3.7 discusses the susceptible-infected-recovered model for information epidemics in the network. Section 3.8 evaluates the efficacy of the

proposed scheme under fading, obstacle network, and mobile social network environments. The concluding remarks are provided in Section 3.9.

3.2 Related Work

3.2.1 Routing

Extensive research on routing in mobile wireless networks has been carried out. These includes end-to-end routing [10], cooperative forwarding [41, 14], and SCF routing [19, 18, 28, 21, 30, 23, 31, 33, 43, 24, 29, 48, 54]. The summary of the related works is also given in Table 3.1.

Cooperative forwarding differs from end-to-end routing in the sense that it takes into account the unreliable nature of wireless channel. It utilizes broadcast wireless nature and combines several weak/unreliable links to make a reliable communication link. It has been shown that it provides a resilient data delivery for networks with adverse wireless links. However, it requires topology information using proactive source routing or position information to locate the destination. CORMAN [14] is one of the first cooperative forwarding protocol proposed for mobile networks. In CORMAN, it was shown that fading can dramatically jeopardize the network performance. Although the efficacy of exploiting cooperative opportunistic forwarding under fading environment was demonstrated, there is a performance drop when the network connectivity becomes intermittent due to the increasing network size.

In contrast to end-to-end routing and cooperative forwarding, SCF-based routing does not locate the destination node. Instead, it uses available contact opportunities and replicates the copies whenever there is a contact. Due to node's mobility, the relay nodes are expected to have contact with the destination at some point and deliver the packet. As it only relies on the

contact node information, this type of routing is robust to intermittent network connectivity and can guarantee eventual data delivery.

Epidemic routing [29] is the first SCF routing protocol, which is robust to link losses and network disconnection. However, it has been shown that massive packet replications and unnecessary buffer consumption are inevitable in this scheme. To control the packet replications, some protocols limit the number of copies made for a packet or the number of hops a packet traverses [31, 33, 24]. In (p, q) epidemic routing [33], packets are forwarded blindly in a probabilistic manner. Two randomly generated probabilities p and q are used to decide to forward/replicate a packet by direct, two-hop, probabilistic, and epidemic forwarding approaches. The multi-spreader routing was proposed in [24]. These schemes prove that the number of packet copy transmissions can be controlled by limiting the spreaders/forwarders. However, it is shown that the message delivery delay is higher compared to the epidemic routing.

Some adaptive routing schemes which are the integration of SCF routing (also called delay tolerant network (DTN) routing) and end-to-end MANET routing have been proposed for opportunistic networks. In [43, 28], nodes find the end-to-end path and forward the data packet. SCF-based forwarding is used when the end-to-end connection breaks. Similar to [28], HYMAD [48] is also a hybrid DTN-MANET routing protocol, which uses end-to-end routing to forward messages within cluster/group and uses SCF routing to forward messages between groups. Another hybrid MANET-DTN approach [23] showed the benefits of integrating the SCF mechanism into a proactive MANET routing protocol. In this protocol, it stores data packets when there is no end-to-end path or sends the packets if it finds a route.

Moreover, there exists some other recent adaptive routing schemes for networks with diverse connectivity characteristics. In [21], multiple forwarding

paths are selected and data packet copies are forwarded along the predetermined multiple paths. In [19], it highlights the importance of being aware of nodes' vicinity on the delivery delay. The WAIT forwarding strategy is discussed where nodes wait for the end-to-end transmission opportunities and forward data packets using multi-hop transmissions. Similar to [19], reference [18] also points out the existence of transient connected components (TCC) in the network. Packets are replicated to nodes with higher forwarding metric using multi-hop communications.

Most existing adaptive routing protocol for opportunistic networks combine multi-hop routing and SCF-based routing. In those approaches, nodes wait/discover for the multi-hop opportunities and use SCF if there is no multi-hop route. In contrast to the existing works, the proposed solution proactively monitors the 2-hop vicinity and does not require to have route-discovery. It forwards the data packet by broadcast-based cooperative forwarding whenever there is an opportunity in the vicinity. Nodes adaptively switch between the cooperative forwarding and SCF routing based on the contact opportunity. Nodes reactively replicate the packet copies only when the packet has never been forwarded by cooperative forwarding. Moreover, overhearing the cooperative transmissions over the air provides the node the ability for better information epidemics control. By having the adaptive routing capability and information epidemics control, the proposed solution works well in diverse network environments which include 1) highly dynamic MANET such as network with high mobility and unstable link connectivity due to mobility and fading and 2) opportunistic networks such as obstacle-constrained network and mobile social networks. It also works well in connected MANET as shown in the comparison with AODV [10] and CORMAN [14] in Section 3.8.4.

3.2.2 Information Epidemics Control

Recovery schemes for controlling the spread of packet replications were proposed in [37]. *IMMUNE* (*self-healing*) and *VACCINE* are the two most commonly used packet discarding strategies for information epidemics control in opportunistic networks. The *VACCINE* recovery scheme is shown to be the most effective one in the existing works [37, 39, 38]. In *VACCINE*, a node maintains information of packets which have been successfully delivered to the destination node in addition to information of packets it has received so far in a list. This list is exchanged between two nodes to avoid sending duplicate data packets and the data packets that has been delivered to the destination. In SCF-based routing, a vaccine is generated for a packet when the destination node receives the packet or when a node receives the ACK from the destination node upon delivering the packet.

Analogy to the epidemiology, the SIR model has been widely used for analysis of information dissemination in the network [37, 38, 39]. In the SIR model, a packet is considered as one infectious disease. A node is said to be susceptible if it does not have a copy of that packet. A node is said to be infected when it receives a packet copy. A node becomes recovered when it successfully delivers the packet to the destination node or receives the anti-packet.

In the existing information epidemics control approaches, a vaccine is generated only when a node successfully delivers a packet to the destination. However, in opportunistic links and dynamic topology environments, due to the effect of mobility and fading, having a direct contact with the destination is scarce. Therefore, this results in late information epidemics control and more infections. In this work, we propose a vaccinating scheme which provides early control signal distribution, and hence leverage the informa-

Figure 3.1: Network layer architecture.

tion epidemics control in the network. The proposed vaccinating scheme is integrated with the data forwarding algorithm. Due to the flexibility of the scheme, it can also be integrated with any SCF-based routing.

Table 3.1: Summary of Related Works

Protocol	Network type	Routing type	Addressing mode	Replication control
CORMAN [14]	MANET	Cooperative forwarding over the pre-determined shortest paths.	broadcast	N.A
POR [41]	MANET	Position-based cooperative forwarding.	broadcast	N.A
ER [29]	Opp Net	SCF forwarding.	unicast	Self-healing
Spray and Wait [31]	Opp Net	SCF. Nodes spread message copies to L distinct relays. Each node carrying a copy performs direct transmissions.	unicast	Self-healing. Limit the no. of copies.
(p,q) ER [33]	Opp Net	Direct, two-hop, probabilistic, and epidemic forwarding based on randomly generated probabilities p and q .	unicast	VACCINE
Multi-spreader [24]	Opp Net	SCF. Limit the max number of spreaders replicating the copies.	unicast	VACCINE
PER [30]	Opp Net	SCF. Replicate the packets to nodes which has lowest-cost route to the destination.	unicast	Self-healing
DTN+MANET [28, 43]	Opp Net	Forward the packets over the end-to-end path if there is or use SCF otherwise.	unicast	VACCINE
HYMAD [48]	Opp Net	End-to-end routing within group and SCF forwarding between group.	unicast	Self-healing
G-ER [54]	Opp Net	End-to-end routing within group and SCF forwarding between group.	unicast	Self-healing. Keep only one copy in each group.
DTN+MANET [23]	Opp Net	Send/forward the packets when there is a route or store otherwise.	unicast	N.A
R3 [21]	Opp Net	Replicate the packets over the pre-determined multi-paths.	unicast	Control the no. of paths.
Vicinity-aware routing [19]	Opp Net	WAIT for multi-hop opportunities and forward the packets when available.	unicast	N.A
TCC-aware routing [18]	Opp Net	Multi-hop communications inside TCC and SCF forwarding outside TCC.	unicast	Self-healing. Replicate to nodes with higher forwarding metric.

3.3 Overview of SCCF

In this chapter, we propose a data delivery solution for networks with opportunistic links and dynamic topologies. The proposed solution does not require any information in advance. The basic idea is 1) deploying the local topology awareness and vicinity-aware cooperative forwarding on top of the SCF routing; 2) reactively replicate the packet copies only when the message has never been forwarded by cooperative forwarding; and then 3) incorporating the vaccinating module which deters packet infections and vaccinate others from being infected.

In SCCF data forwarding, five different types of data forwarding are defined. They are *direct*, *two-hop*, *forwarder*, *forwarder-by-chance*, and *replicate*. SCCF uses broadcast transmission for the first four forwarding types and unicast for the *replicate* type. A node inserts the respective data forwarding type in the data packet header before it forwards the packet. This information is used to inform the nodes receiving or overhearing the packet so that they can process accordingly. Figures 3.2 and 3.3 illustrate how respective forwarding types are used in SCCF. Section 3.5 details the SCCF data forwarding algorithm.

In SCCF vaccinating, nodes maintain a packet state vector inside which the progress of each data packet is recorded. There are four possible packet states: *No_copy*, *Have_copy*, *Local_fwd*, and *Delivered*. The four packet states and the state transition are shown in Fig. 5.10. Section 3.6 discusses how state of each packet are transitioned from one state to another and how packet infection is deterred using partially-immunized vaccine and fully-immunized vaccine.

Figure 3.2: Data forwarding decision when a node gets contact.

Figure 3.3: Cooperative data forwarding.

Figure 3.4: Header format (a) Hello packet (b) Data packet.

Figure 3.5: Calculating link quality.

3.4 Live-update Link Quality

In SCCF algorithm, nodes conduct live-updating link qualities to monitor the unreliable and probabilistic nature of wireless channel and to have local topology awareness so as to timely exploit the forwarding opportunities in the vicinity.

Each node broadcasts a hello message periodically, at an average period of τ (we use one second in the performance evaluation). The format of the hello message is shown in Fig. 5.4. Nodes calculate bi-directional link qualities of the links connecting to the direct contact nodes. This link quality denoted as p is calculated as $p = p_{fwd} \times p_{rev}$, where p_{fwd} refers to the link quality

from a node to its contact node and p_{rev} refers to the link quality in reverse direction. p_{fwd} is called forward link quality and p_{rev} is called reverse link quality. We define that a node gets contact with another node if the link quality p is non-zero.

The reverse link quality p_{rev} with a contact node is calculated based on the received hello packet ratio during the last w seconds from that node. For each contact node, a node calculates p_{rev} using the equation shown below.

$$p_{rev}(t) = \frac{r(t-w, t)}{w/\tau} \quad (3.1)$$

where $p_{rev}(t)$ denotes the reverse link quality at time t , $r(t-w, t)$ denotes the number of hello received from the contact node during the last w seconds, and w/τ denotes the number of hello the contact node sent during the last w seconds. For each contact node, a node live-updates the $p_{rev}(t)$. We call the table recording this information reverse link information table. The format of this table is shown in Table 3.2.

The forward link quality p_{fwd} with a contact node is obtained as follows. In addition to the IP header in the hello message, nodes also add the contact IDs and the reverse link qualities with each of the contacts. These reverse link qualities in the hello message are indirectly telling each of the contact node the forward link quality from each of them. For example, in Fig. 3.5, node n_1 (n_2) computes reverse link quality with n_2 (n_1) using (1). It then shares its contact IDs and the reverse link qualities using hello message. By adding this additional info in the hello message, nodes obtain the forward link qualities to each of the contact nodes. From this p_{fwd} and above discussed p_{rev} , nodes calculate the bi-directional link qualities p with each of the contact nodes.

Besides the IP header, contact IDs, and reverse link qualities, nodes also add the bi-directional link qualities with each of the contact nodes in the hello message. For example, as shown in Fig. 3.5, in addition to IP header, node

n_2 puts its contact IDs n_1 and n_4 and bi-directional link qualities and reverse link qualities with each of them. From this one-hop contact information in the hello message, nodes in the network obtain two-hop contact information. For example, from n_2 's hello message in Fig. 3.5, n_1 knows that n_2 can reach n_4 with bi-directional link quality of $p(n_2, n_4)$. From this information, nodes build a contact information table. The format of the table is shown in Table 5.1. This table is used to monitor the data forwarding opportunities in the vicinity. The p_{fwd} value is updated when another hello message with update information from respective contact node is received. The p_{rev} value is updated whenever there is an update in reverse link information table. The bi-directional link is considered lost and the entry is deleted when there is no update for p_{fwd} value within timeout period or when the p_{rev} value becomes zero.

In a dense network, in order to reduce the overhead due to the added link information in the hello message, one can choose to add the contact only if the link quality is above a certain threshold. However, in opportunistic links and dynamic topology environments, a node is not likely to have many contact nodes.

Table 3.2: Reverse Link Information Table

contact ID	p_{rev} during the last w seconds
n_2	0.7
n_3	0.8
n_5	0.85
...	...

Table 3.3: Contact Information Table

contact ID	p_{fwd}	p_{rev}	p	one-hop contact list
n_2	0.8	0.7	0.56	$(n_1, 0.56), (n_3, 0.7), (n_4, 0.7) \dots$
n_3	0.9	0.8	0.72	$(n_1, 0.72), (n_2, 0.7), (n_4, 0.9) \dots$
n_5	0.9	0.85	0.77	$(n_1, 0.77), (n_4, 0.9), (n_6, 0.8) \dots$
...	...	...	...	...

3.5 Routing

In this section, we discuss the details of the SCCF data forwarding algorithm. We first describe the five different types of forwarding mechanism: *direct*, *two-hop*, *forwarder*, *forwarder-by-chance*, and *replicate* for data delivery. We then elaborate the details of first four forwarding mechanisms in 3.5.2 and the details of the *replicate* forwarding mechanism in 3.5.5. The procedure about when the respective forwarding mechanism is triggered is also illustrated in Figs. 3.2 and 3.3. We also present the additional components of the protocol that complement data forwarding which include network layer acknowledgement, packet buffers (primary and temporary), and handling link loss.

3.5.1 Forwarding Type

A forwarding type is defined in data packet header to inform the contact nodes the forwarding type of the data packet that is being sent. SCCF protocol forwarding types and their definitions are discussed in Table 3.4. We add this field in the data packet header, which is shown in Fig. 3.4b, so that the receiver or nodes which overhear the transmission can obtain the information and/or process accordingly. A node will modify the forwarding type in the packet header based on the type of data forwarding it will use

before passing the packet to lower layers for transmission.

Table 3.4: SCCF Forwarding Types

Forwarding type	Definition	Addressing mode
<i>direct</i>	send to destination by direct broadcasting.	broadcast
<i>two-hop</i>	send to forwarder(s) by two-hop forwarding.	broadcast
<i>forwarder</i>	send from forwarder to destination.	broadcast
<i>forwarder-by-chance</i>	send from high-chance non-candidate forwarding node to destination.	broadcast
<i>replicate</i>	send packet copies to a contact node.	unicast

3.5.2 Opportunistic Forwarding

Upon receiving a hello message from n_j and updating n_j 's contact information in the contact information table, the node n_i makes forwarding decisions to forward the packets in its buffer based on the contact information table. Firstly, the node obtains all destination IDs of the packets in the buffer which are listed as one-hop contact in its contact information table. It will then broadcast the data packets to their destination node. This type of forwarding is called *direct* broadcasting.

The node will then check if the destination IDs of the remaining data packets in the buffer are listed as contacts of its direct contact nodes in the contact information table. Suppose a node n_1 has a contact information table with contents as shown in Table 5.1 and the destination IDs are identified as n_4 and n_6 . It will then build the forwarding table for each destination ID as shown in Table 3.5. The direct contact nodes are assigned as potential forwarding nodes in the forwarding table. n_1 will select the two best for-

warding candidates based on the expected number of transmissions (ETX) to destination, which is calculated as follows.

$$ETX_{src,dst} = ETX_{first_hop} + ETX_{second_hop} \quad (3.2)$$

$$ETX_{first_hop} = \frac{1}{1 - (1 - p(src, f_1))(1 - p(src, f_2))} \quad (3.3)$$

$$ETX_{second_hop} = p_1 ETX(f_2, dst) + p_2 ETX(f_1, dst) + p_3 \min\{ETX(f_1, dst), ETX(f_2, dst)\} \quad (3.4)$$

where, $ETX_{src,dst}$ refers to the expected number of transmissions from source/sender node src (n_1 is the source/sender in this example) to destination node dst , ETX_{first_hop} refers to the expected number of transmissions required so that at least one of the two forwarders f_1 and f_2 successfully receive the data packet from src , and ETX_{second_hop} refers to the expected number of transmissions from two forwarders to destination node dst . $p(src, f_1)$ refers to the bi-directional link quality between src and f_1 . We calculate $ETX(f_1, dst) = 1/p(f_1, dst)$ and $ETX(f_2, dst) = 1/p(f_2, dst)$. p_1 (p_2) denotes the probability that forwarding node f_1 (f_2) fails to receive the packet and p_3 is the probability that both node receive the packet. We assume that p_4 , the probability of both nodes fail to receive is zero and p_1 , p_2 , and p_3 are evenly distributed, which is $1/3$.

The node will then add the candidate IDs to the packet header and broadcast the packet. We also require that the two selected candidates be in contact so that they can overhear each other and can avoid duplicated transmissions. This type of forwarding is called *two-hop* forwarding. The format of the data packet header is shown in Fig. 3.4b.

The following discussion details the cooperative data forwarding which includes 1) how the forwarders cooperatively forward the data packet from

Table 3.5: Forwarding Table Format

dst ID	fwd list (fwd ID, p between fwd and dst)
n_4	$(n_2, 0.7), (n_3, 0.9), (n_5, 0.9)$
n_6	$(n_5, 0.8)$
...	...

the source/sender node to the destination node; 2) how nodes not being assigned as forwarder join the data forwarding on-the-fly so as to have resiliency under hostile environments; and 3) how a forwarder handles when it finds link lost to the destination.

Cooperative data forwarding - before the forwarding node broadcasts the packet to the destination, it will decide which forwarding node in the list should broadcast first. Here, the decision is based on the link quality between the forwarding nodes and the destination. The two forwarding nodes f_1 and f_2 will self-evaluate the length of delay (timer-to-forward), and will broadcast the packet when the timer expires. The length of the delay at forwarding node f_1 is calculated as follows.

$$Delay(f_1) = \frac{1}{p(f_1, dst)} \times k + x \quad (3.5)$$

where $p(f_1, dst)$ is the link delivery probability from f_1 to dst . k refers to the average one-hop delay, i.e, the time taken from the transmitter network layer to the receiver network layer, which includes processing time, queuing time, and transmission time. We set $k = 0.01$ sec as it is the average one-hop delay in our simulation environment. x is the jitter which follows uniform distribution. If $p(f_1, dst) > p(f_2, dst)$, f_1 will have a smaller x (chosen from uniform distribution $[a, b)$) and f_2 will have a larger x (chosen from uniform distribution $[c, d)$) so that f_1 will be likely to broadcast earlier than f_2 and f_2 will overhear the transmission of f_1 . Note that $a < b < c < d$. In this case,

node f_2 will broadcast only if it does not overhear the transmission from f_1 . When a forwarding node overhears transmission from another forwarder, it will stop the timer-to-forward and discard the packet. By adding the parameter x , it prevents collisions due to simultaneous transmission from the two forwarders and avoids duplication by overhearing. If $p(f_1, dst)$ and $p(f_2, dst)$ are equal, x will be chosen randomly. By adding one more relay node, the success probability of data delivery from n_i to the destination dst is improved. Note that the forwarding type is set to *forwarder*.

As the node movement affects link connectivity, a node might encounter inaccuracies in the timer-to-forward self-evaluation if many forwarding candidates are involved, hence, we limit the number of forwarders to two. However, to increase the probability of successful delivery, a node can join the data forwarding on-the-fly. A node n_k can help cooperatively forward the data packet if the following conditions are satisfied. 1) Node n_k overhears the transmission from n_i ; 2) its link quality to the destination is higher than $p(f_1, dst)$ and $p(f_2, dst)$; and 3) it has non-zero link quality to both f_1 and f_2 .

If the above three conditions are satisfied, n_k chooses a jitter value x (picked from uniform distribution $[x, y)$, where $x < y < a$) and forwards the packet. Note that the forwarding type in this case is set to *forwarder-by-chance*. The two forwarders will discard the packet when it overhears the transmission from other nodes with higher chance. By doing this, the forwarding is resilient in case the link delivery probabilities of the assigned candidates are affected due to node movement or changes in environments.

We observed that due to the opportunistic links and dynamic topology environments, a node might have only one potential forwarding node available. In this case, a node will still forward the data packet using one forwarding

candidate so as to achieve data delivery in the shortest possible time. With one forwarding candidate, the expected number of transmissions to destination will become as follows.

$$ETX_{src,dst} = \frac{1}{p(src, f_1)} + \frac{1}{p(f_1, dst)} \quad (3.6)$$

As any node can join the data forwarding on-the-fly if conditions discussed above are satisfied, the probability to successfully deliver the data packet can be further increased.

In two-hop forwarding, when a forwarding node finds that the link to the destination is lost, it will first broadcast the forwarder error message. It will then check its contact information table if there is any direct contact node which has a link to the destination. The node then selects a contact node with the lowest ETX to the destination node as in (3.6) and delays the transmission until it can overhear another forwarder or *forwarder-by-chance* transmission if there is any. If it does not overhear transmission from forwarder or high-chance non-candidate for t interval (chosen from uniform distribution $[u, v)$ and $d < u < v$), it will replicate the packet to the selected node using unicast. If there is no direct contact node to the destination, the node will store the packet in its buffer and wait for future contact opportunities. Upon receiving/overhearing the forwarder error message, the receiver updates its contact information table and forwarding table. It will then inspect the interface queue and discard the packet if the erroneous forwarder is being assigned in the packet header.

3.5.3 Packet Buffer

In the proposed protocol, a node maintains two types of buffer to store data packets as follows.

Primary packet buffer (buffer) - It is used to store data packet when a node finds that the destination of that packet is not in its transmission range (not in contact) and it cannot find forwarding nodes to that destination.

Temporary packet buffer (temporary buffer) - It is used to keep copies of packets which are being broadcasted (direct broadcasting or two-hop cooperative forwarding) but waiting for acknowledgement to ensure the successful reception. After the node overhears the acknowledgement from the forwarding node or the destination, the corresponding packet copy will be removed from the temporary buffer. Otherwise, the node will retrieve the packet from the temporary buffer and store it in the primary buffer after acknowledgement timeout.

3.5.4 Network Layer Acknowledgement

In *direct*, *two-hop*, *forwarder*, and *forwarder-by-chance* transmissions, a node uses IEEE 802.11 broadcast mode to forward the data packet. In the proposed algorithm, we use network layer acknowledgement scheme as IEEE 802.11 broadcast mode does not provide any acknowledgement for the broadcast packet. When a node receives a broadcast data packet, it will send the acknowledgement if it is the destination or forwarder of that data packet. We use cumulative acknowledgement instead of acknowledgement for individual packet to reduce the control packet overhead and collisions to data packet transmissions. This acknowledgement is transmitted in broadcast mode. Therefore, it informs not only the sender about successful reception of the data packet but also other nodes who are within the communication range.

3.5.5 Replication-based Forwarding

Node n_i replicates the packets in its buffer to node n_j only if the direct and cooperative forwarding opportunities discussed above are not available. In this case, node n_i first sends packet state vector request to node n_j .

Packet state vector is an information table a node maintains to track the progress of data packet. There are four possible packet states. They are *No_copy*, *Have_copy*, *Local_fwd*, *Delivered*. Details of the packet state vector and the state transitions are discussed in Section 3.6.

Upon receiving n_i 's request, node n_j broadcasts its packet state vector. Due to broadcasting, node n_i and all the other nodes in contact with node n_j are able to receive/overhear and update their packet states as per the state transition rules shown in Fig. 5.10. For example, if the overheard packet state is *Local_fwd*, a node will update its packet state to *Local_fwd* if its current state is *No_copy* / *Have_copy*. Similarly, if the overheard packet state is *Delivered*, a node will update its packet status to *Delivered* if its current state is *No_copy* / *Have_copy* / *Local_fwd*. The detail of the information epidemics control algorithm is discussed in Section 3.6.

After updating the packet state vector based on the received vector from n_j , node n_i gets the copies of packets in the buffer as in (5.9) and send to node n_j .

$$packets_to_be_replicated = X_A + \overline{Y_A} + \overline{X_B} \quad (3.7)$$

where X_A refers to a set of IDs of packets that node n_i has in its primary packet buffer, Y_A refers to a set of IDs of packets with state *Local_fwd* in node n_i 's packet status vector, and X_B refers to a set of IDs of packets with states *Have_copy* in node n_j 's packet status vector. Note that packets in temporary buffer and packets with *Local_fwd* state are not replicated, and the proposed protocol uses unicast to transmit these packets. The forwarding type in

Figure 3.6: Four packet states and state transitions.

this case is set to *replicate*. In contrast to original epidemic routing [29] where n_i and n_j exchange the missing packets with each other, the proposed protocol does not exchange the packets as n_j may have direct or cooperative forwarding opportunities to deliver its data packets.

When a node receives a packet transmitted by replication-based forwarding, it updates the packet status in the status vector to *Have_copy* if the current state is *No_copy* or discard otherwise. The node then checks whether the destination of the packet is its contact (direct) or it has forwarder(s) to that destinations (two-hop). If direct or two-hop opportunities are available, it will perform direct or two-hop forwarding. Otherwise, it will store the packet in the buffer and wait for future contact opportunities.

3.5.6 Handling Link Loss

Due to the dynamic and random nature of wireless link, the communication link can be lost while packets are being forwarded. To stop sending out packet if the link is lost, the link layer maintains a bi-directional link information table and passes the packets to interface queue only if the link delivery probability is non-zero. In case of link loss, the link layer will discard the packet. The information in the link information table is created/updated/deleted if the corresponding link information is created/updated/deleted in the routing layer.

3.6 Information Epidemics Control

In the existing information epidemics control approach, a vaccine is generated only when a node successfully delivers a packet to the destination. However, in opportunistic links and dynamic topology environments, due to the effect of mobility and fading, having a direct contact with the destination is scarce. Therefore, this results in late information epidemics control and more infections. In this chapter, we propose a vaccinating scheme which provides early control signal distribution, and therefore leverage the information epidemics control in the network.

The vaccinating module monitors the broadcast-based vicinity-aware forwarding transmissions over the medium and translates the information overheard into packet states. We call the database maintaining the state for each packet 'packet state vector'. We define 4 packet states as ***No_copy***, ***Have_copy***, ***Local_fwd***, and ***Delivered***. The default state for each packet is set to ***No_copy***. The four packet states, their definitions, and state transitions are discussed as follows. Figure 5.10 shows the packet states and the state transitions.

Have_copy - A packet is set to *Have_copy* status when a node received

Figure 3.7: (a) Direct broadcast. (b) Two-hop broadcast.

a packet 1) from the upper layer; 2) from another node where data packet is transmitted by replication-based forwarding; and 3) from overhearing *two-hop* transmissions (i.e. the transmission from node *A* shown in Fig. 3.7b).

If a packet is in this status, a node will not accept the packet copy. That is to say, the node is immune to this packet infection. We call this information epidemics control *SCCF self-healing*.

Local_fwd - A packet status is set to *Local_fwd* when a node is aware that the data packet has been/is being forwarded by *direct* (i.e. the transmission from node *A* shown in Fig. 3.7a), *forwarder* (i.e. the transmission from node f_1 or f_2 shown in Fig. 3.7b), and *forwarder-by-chance* transmissions. This happens when a node 1) overhears the *direct*, *forwarder*, and *forwarder-by-chance* transmissions; and 2) receives/overhears acknowledgement from forwarders (i.e., the acknowledgement transmissions from node f_1 or f_2 shown in Fig. 3.7b).

Upon learning the *Local_fwd* status, a node will not accept the packet copy and will be able to vaccinate others in order not to get infected. When a node is transmitting a packet using *direct*, *forwarder* and *forwarder-by-chance* transmissions, it implies that the sender is in contact with the destination in one or two-hops. That is to say, most likely, the packet will be delivered to the destination. In addition, replicating this packet is no longer necessary as it will incur bandwidth consumption and collision to other data transmissions. Therefore, if a node happens to have a copy of the packet in its buffer, it will no longer use replication-based forwarding for this packet delivery to the destination. However, to ensure the reliability, the node will still use vicinity-aware forwarding until it gets information that the destination has received the packet. This type of information epidemics control is called vaccinating using partially-immunized vaccine.

Delivered - A packet status is set to *Delivered* when a node is aware that the data packet has been successfully delivered to the destination. This happens 1) when the destination node receives the packet; and 2) when a node receives/overhears acknowledgement from the destination node (i.e., the acknowledgement transmissions from node *Dst* shown in Figs. 3.7a and 3.7b).

Upon learning this status, a node becomes fully-immunized. It will no longer accept any incoming copy. It will also discard if there is a corresponding packet copy in its buffer. It will also be able to vaccinate others so that others can discard the packet in their buffer as well if any. This type of information epidemics control is called vaccinating using fully-immunized vaccine.

No_copy - A packet will be in *No_copy* status if a node does not have any information discussed above. A node will involve in both vicinity-aware forwarding and replication-based forwarding for this packet delivery.

In the existing protocols using VACCINE approach, it takes one bit for each packet inside the summary vector to tell if the node has the packet copy or not and another one bit for each packet inside the VACCINE vector to tell if the packet has been delivered to the destination or not. In the proposed algorithm, it uses only one vector called packet status vector and takes two bits for each packet to tell *No_copy*, *Have_copy*, *Local_fwd*, and *Delivered* packet status. As a result, the proposed algorithm requires transmission for one single vector only which results in less number of vector transmissions compared to the protocols using existing recovery schemes. In addition, each vector provides further information including advance delivery progress while using the same number of bits as in the existing protocols. Moreover, the replication-based forwarding in proposed algorithm is reactive that it is

initiated only if the direct and cooperative forwarding opportunities are not available and the packets have never been forwarded by those forwarding. Therefore, this reactive nature results in lower number of replication-based forwarding and hence, lower packet state vector transmissions. The size of the vector can be further minimized 1) by using compression mechanism [55] and 2) by setting the timer based on the desired confidence level of the packet delivery and purge the corresponding information inside the vector when timeout.

We use the IEEE 802.11 broadcast mode to send the status vector in contrast to unicast in existing SCF routing. As a result, not only the intended node but also the other nodes in the vicinity receive the progress of data delivery and update their knowledge accordingly, as per the state transition rules shown in Fig. 5.10. In the network layer, by having the information of packet state information from overhearing *direct*, *two-hop*, *forwarder*, and *forwarder-by-chance* transmissions, network layer acknowledgements, and packet state vectors, node *B* will be infected with a packet copy from node *A* only if *B*'s packet state is *No_copy* and *A*'s packet state is neither *Local_fwd* nor *Delivered*. This results in a lower number of infections/replications, lower bandwidth consumption, and lesser buffer occupancy.

3.7 SIR Model for the Information Epidemics in Network

Susceptible-Infected-Recovered (SIR) is a model used to study the spread of packet copies in the network. As information dissemination in a network is same as disease spreading in a population, the SIR model has been widely used to model the packet dissemination in Epidemic routing variants [39, 37,

38]. The Markov chain model of an infectious disease using the SIR model with and without vaccination are shown in Fig. 3.8, where S , I , and R refers to the susceptible, the infected, and the recovered population in the system, respectively.

3.7.1 SIR Model Without Vaccination

Consider the SIR model without vaccination. Let $S(t)$ refers to the susceptible node population which does not have a packet copy at time t , $I(t)$ refers to the infected node population which have the packet copy at t , and $R(t)$ refers to the recovered node population which has successfully delivered the packet to the destination at t . The total population in network N is equal to $S(t) + I(t) + R(t)$. Assume β is the contact rate for a node to meet another node (infection rate) and γ is the contact rate of a node to the destination node (recovery rate). Accordingly, the total infection rate and the total recovery rate in the network are $\beta S(t)I(t)$ and $\gamma I(t)$, respectively [37, 38]. Using the Markov chain model shown in Fig. 3.8a, we obtain the first order differential equations as in (3.8).

Figure 3.8: Markov chain model of an infectious disease: (a) SIR model without vaccination; (b) SIR model with vaccination.

$$\begin{aligned}
\dot{S}(t) &= -\beta S(t)I(t), \\
\dot{I}(t) &= \beta S(t)I(t) - \gamma I(t), \\
\dot{R}(t) &= \gamma I(t), \\
S(t) + I(t) + R(t) &= N
\end{aligned} \tag{3.8}$$

Assume that at time $t = 0$, only one node in the network (source node) has the packet. So, the initial conditions are $S(0) = N - 1$, $I(0) = 1$, and $R(0) = 0$. Therefore, we obtain the number of infected nodes as a function of time, $I(t)$ by solving the differential equations in (3.8), that is,

$$I(t) = \frac{N}{1 + (N - 1)e^{-\beta N t}} \tag{3.9}$$

Recalling that β is the contact rate for a node to meet another node which can be calculated as $\beta = \frac{2\omega RE[V^*]}{L^2}$ [38]. Note that ω is a constant specific to the mobility model, R is the radio transmission range, $L \times L$ is the network size, and $E[V^*]$ is the average relative speed between the two nodes.

3.7.2 SIR Model With Vaccination

We extend the original SIR model without vaccination to include the vaccinating process as follows. In the SIR model with vaccination, $R(t)$ represents the recovered node population which includes 1) infected nodes successfully delivered the packet to the destination and 2) the infected nodes and susceptible nodes which have successfully received the vaccine.

Incorporating the vaccinating process, we show the Markov chain model of information dissemination in the network in Fig. 3.8b. From the model,

we obtain the first order differential equations as in (3.10).

$$\begin{aligned}
 \dot{S}(t) &= -\beta S(t)I(t) - v(t)S(t), \\
 \dot{I}(t) &= \beta S(t)I(t) - \gamma I(t) - v(t)I(t), \\
 \dot{R}(t) &= \gamma I(t) + v(t)I(t) + v(t)S(t), \\
 S(t) + I(t) + R(t) &= N
 \end{aligned} \tag{3.10}$$

where $v(t)$ is the successful vaccination rate in the system at t . The infected node population as a function of time under the vaccinating process, $I(t)$, can be obtained using early-stage analysis as discussed in [39].

3.7.3 Control Signal Distribution

Assume that v_f and v_p are the successful vaccination rate of the nodes in the network using fully-immunized and partially-immunized vaccine, respectively. Let v_o be the successful vaccination rate of the nodes in the network using the original VACCINE method. Note that the vaccinating process using VACCINE approach can be initiated only when a node meets the destination. Assume T_{VAC} be the time instance the vaccinating process is initiated or the control signal distribution time in VACCINE. We describe $v(t)$, the vaccination rate in the network at time t using original VACCINE method as follows,

$$v(t) = \begin{cases} 0, & t < T_{VAC} \\ v_o, & t \geq T_{VAC} \end{cases} \tag{3.11}$$

We then describe $v(t)$ for the proposed vaccinating process as follows.

$$v(t) = \begin{cases} 0, & t < T_{P.VAC} \\ v_p, & T_{P.VAC} \leq t < T_{F.VAC} \\ v_p + v_f, & t \geq T_{F.VAC} \end{cases} \tag{3.12}$$

where T_{P_VAC} and T_{F_VAC} are the time instance for the partially-immunized and fully-immunized control signal distribution in proposed vaccinating scheme. We state that $v_f \geq v_o$ as v_f is the rate using one-to-many vaccination (broadcast) whereas v_o is the pairwise vaccination. Note that the control signal distribution using fully-immunized vaccine and the original VACCINE scheme can be initiated only when a node meets the destination. However, having the local topology information and exploiting broadcast-based vicinity-aware forwarding make a node delivering the packet to the destination much earlier compared to the pairwise packet exchange. Therefore, we claim that the control signal distribution using the fully-immunized vaccine can be initiated much earlier than the vaccination using the original VACCINE scheme (i.e. $T_{F_VAC} \leq T_{VAC}$). In addition, using the partially-immunized vaccine can further advance the vaccinating process as it is initiated when there is a local-forwarding opportunity.

We implement the integrated data delivery solution in network simulator (NS-2). We prove that having early control signal distribution provides a higher vaccination rate and recovery rate. This results in a lower infection rate and hence, requires fewer packet copy transmissions.

3.8 Performance Evaluation

Table 3.6: Simulation Parameters

Parameters	Network area ($l \times l$ m^2)	SIR study	Comparison with CORMAN [14]	Effect of fading	Effect of obstacles	Effect of mean speed	Mobile network model	social model
Network area	1000	1000	250, ..., 1000	1000	1000	1000	1600	
No. of nodes	100	100	50	25	20 to 50	25	41, 78	
Mobility model	RWP	RWP	RWP	RWP	MCM[56]	RWP	SWIM[57]	
Avg. Speed (m/s)	10	10	5	15	15	5, 10, 15, 20, 25	SWIM[57]	
Fading model	Two-ray-ground	Two-ray-ground	Nakagami	Freespace, Nakagami, Rayleigh	model used by MCM [56]	Nakagami	Freespace	
Traffic	1 pkt	1 pkt	50 pkts/sec	50 pkts/sec	50 pkts/sec	50 pkts/sec	4 pkts/sec	
No. of (src,dst)	1	1	1	10	10	10	420

Figure 3.9: (a) Infected population under ER self-healing. (b) SIR population under SCCF self-healing; (c) without partially-immunized vaccination; and (d) with both partially-immunized and fully-immunized vaccination.

3.8.1 Evaluation Methodology

We use network simulator (NS-2) to evaluate the performance of the proposed scheme. The evaluation covers two main scenarios which include 1) the susceptible-infected-recovered population study; and 2) the routing performance. In the SIR population study, let $inf(t)$ be the number of nodes that get infected at each time instance t , $rec_i(t)$ be the number of infected nodes

that get recovered at each time instance t , and $rec_s(t)$ be the number of susceptible nodes that get recovered at each time instance t . Using the simulation, we obtain $inf(t)$, $rec_i(t)$, and $rec_s(t)$ under *Epidemic self-healing*, *SCCF self-healing*, under the vaccination without having partially-immunized vaccines, and under the vaccination with both partially-immunized plus fully-immunized vaccines. Note that the recovered population in Epidemic routing is 0 as the nodes in the network continues spreading the packet copies until all the nodes in the network get infected. We then plot the SIR population in the network under each scheme using the following equations.

For Epidemic routing,

$$\begin{aligned} I(t) &= \sum_{t=0}^t inf(t), \\ S(t) &= N - I(t) \end{aligned} \tag{3.13}$$

For SCCF routing,

$$\begin{aligned} R(t) &= \sum_{t=0}^t rec_s(t) + \sum_{t=0}^t rec_i(t), \\ I(t) &= \sum_{t=0}^t inf(t) - \sum_{t=0}^t rec_i(t), \\ S(t) &= N - I(t) - R(t) \end{aligned} \tag{3.14}$$

In the routing study, we examine 1) the packet delivery ratio (PDR) which shows the amount of packets delivered over the amount of packets generated, 2) the end-to-end delay which indicates the time taken from the packet generated at the source application layer to the packet delivered at the destination application layer, and 3) the transmission count which indicates the number of times a packet is transmitted/forwarded by the nodes in the network. The transmission count here also takes into account the number of retransmissions as well. We compare the data delivery performance with the

cooperative forwarding (CORMAN) [14], end-to-end routing (AODV) [10], and SCF-based routing (Epidemic or ER) [29].

We choose AODV [10] for performance comparison in connected environment as AODV is shown to be one of the best protocols for such network. The implementation of AODV is obtained from NS-2.33 package. We choose ER [29] for comparison in disrupted/disconnected environment as it provides the highest delivery success rate with lower latency in such network, which is shown in recent studies [58, 26, 48, 18, 59]. We implement the ER protocol in NS-2.33 and verify the implementation by using the same parameter settings and reproducing the results as in [54]. We also compare the ER simulation model with the analytical model as shown in Fig. 3.9a. We choose CORMAN for comparison as it is the first cooperative data forwarding protocol proposed for mobile ad hoc network without requiring position information. We obtain the result trace files from the authors of CORMAN [14], reproduce the AODV results from the paper [14], and conduct relative performance comparison by using the same network, mobility, and traffic scenarios and propagation model as in the paper.

The evaluation covers various scenarios which include 1) the comparison with the CORMAN protocol, 2) the effect of fading, 3) the effect of obstacles, 4) the effect of speed, and 5) the mobile social network environment. We use the newly revised IEEE 802.11 MAC and PHY layer modeules [60]. The IEEE 802.11a standard is used in all the simulation studies in this work. We collect simulation results using identical traffic model with 10 different randomly generated mobility scenarios. Each simulation is run for 900 seconds except for the study under mobile social network model for which we use the same parameters as per the reference [57]. For studies on effect of obstacles and effect of speed, average network performance results are plotted with

95% confidence intervals. For studies on comparison with CORMAN, effect of fading, and mobile social network environment, confidence intervals are provided in Tables 3.8, 3.10, and 3.10. The detail simulation parameters are described in Table 5.5. The fading model parameters can be found in [60].

3.8.2 Verification and Validation of Simulation Models

We use the open source network simulator (NS-2.33) framework [61] and Eclipse Indigo integrated development environment [62] to implement the proposed protocols. An illustration of the basic conceptual protocol model implemented in NS-2 is shown in Fig 3.10. Note that this model depicts the basic components of the protocol only and specific modules such as entity/cluster-based information epidemic control, distributed agent-based coordination, integration with multi-radio multi-channel, etc. of each solution discussed in each chapter are not included in the illustration. The simulation environment or the network topology is set up using Tcl scripting as described in [61]. For the lower layer configurations, we use the model as discussed in [60]. To study the effective transmission range and loss behaviour of a wireless link, we study the probe packet reception ratio by varying the distance between the two nodes without injecting any data traffic.

Throughout the implementation of the model, we take iterative process to confirm that it is correctly implemented with respect to the conceptual model. Before proceeding to the final stage of running simulation, we conduct rigorous debugging and trace analysis. We use interactive Eclipse debugging tool [62] to verify the logic and working flow of the model. Specifically, we abort the simulation running process whenever the logic fails to meet the criteria specified in the conceptual model. At the end of a simulation, we examine output trace files and check all the packet events happened at the

Figure 3.10: Illustration of the basic conceptual protocol model.

different layers of the protocol stack. We do a number of trial runs and see if the value of each performance metric varies greatly from the mean value. If it does, we then analyse the trace file and track all the events to gain insight into the actual root cause and rectify the model. For all the simulation studies in the thesis, the results are obtained with 90% to 95% confidence intervals.

3.8.3 Study on SIR Population in the Network

The normalized infected node population (i.e., $I(t)/N$) under self-healing used in Epidemic routing is shown in Fig. 3.9a. It also shows the result with the SIR model (3.9). As a node with a copy continues infecting other nodes without a copy, the number of infected population increases over time. In addition, as it does not have the mechanism to deter the transmissions of packets that have been successfully delivered to the destinations, the recov-

ered population under self-healing Epidemic routing is zero.

The SIR population in SCCF under different proposed vaccinating algorithms is shown in Figs. 3.9b, 3.9c, and 3.9d. We observe that the infected population is substantially reduced under SCCF complete (both partially and fully immunized) vaccination compared to the SCCF self-healing and the SCCF without partially-immunized vaccines. In SCCF self-healing (Fig. 3.9b), a node gets recovered only when it delivers the packet using direct-broadcast or two-hop broadcast. Therefore, the infected population is increased dramatically in the initial stage and gradually decreased in the later stage. In SCCF without partially-immunized vaccines (Fig. 3.9c), a node also maintains packet delivery information and also shares with others. From the delivery information, a node becomes able to stop transmitting unnecessary packet copies. Thus, the infected population is greatly reduced and the vaccinated population is greatly increased compared to SCCF self-healing. In SCCF with complete vaccination (Fig. 3.9d), a node also maintains the information of packet which has been locally-forwarded, i.e. higher probability to be received at the destination. By having this information in addition to the ones in the other two schemes, a substantial amount of infected population can be further reduced.

3.8.4 Routing Performance Study

Comparison with CORMAN

For comparison with CORMAN, we use the same network, mobility, and traffic scenarios and propagation model as in [14]. The Nakagami propagation model is used to study the effect of realistic fading environment. Therefore, the transmission range between two nodes is probabilistic.

Using the same scenarios as in [14], we obtain the results for SCCF and AODV. The packet delivery ratio performance comparison against CORMAN protocol is presented in Table 3.7. We obtain the PDR of CORMAN and AODV and the relative PDR of CORMAN over AODV from refer-

Table 3.7: PDR Comparison with CORMAN

Network area(m)	SCCF	AODV	Relative	CORMAN [14]	AODV [14]	Relative [14]
250	1.000	0.958	104.424	0.981	0.781	125.577
300	1.000	0.850	117.592	0.926	0.755	122.696
350	1.000	0.821	121.828	0.911	0.790	115.335
400	1.000	0.660	151.515	0.986	0.633	155.900
450	1.000	0.595	168.188	0.937	0.579	161.853
500	1.000	0.615	162.575	0.938	0.425	220.602
550	0.999	0.509	196.139	0.789	0.247	319.707
600	1.000	0.254	394.166	0.883	0.402	219.609
650	0.986	0.194	508.494	0.818	0.390	209.481
700	0.998	0.473	211.134	0.844	0.476	177.308
750	0.982	0.270	363.646	0.758	0.253	299.848
800	0.975	0.160	607.687	0.572	0.141	406.092
850	0.989	0.381	259.376	0.553	0.183	301.464
900	0.960	0.262	365.955	0.605	0.170	355.749
950	0.961	0.244	393.810	0.763	0.406	187.657
1000	0.947	0.150	633.204	0.478	0.153	312.175
average			297.483			230.691
relative						

ence [14]. We observe that the average relative performance of SCCF over AODV achieves almost 30% improvement compared to that of CORMAN over AODV. Although the CORMAN protocol is resilient to the link quality fluctuation and the node mobility, there is a performance drop in CORMAN's PDR when the network size increases due to the intermittent network connectivity. In CORMAN, each node uses source routing and finds the shortest paths to all the other nodes in the network. This information is used by forwarding candidates to adaptively forward the data packet upon link changes. When the network density decreases, there exists partial/intermittent disconnection which causes nodes unable to locate certain parts of the network at some point of time and hence, results in lower PDR under such environments. However, in SCCF, a higher PDR is maintained regardless of the diverse network topologies.

From [14], it is noted that CORMAN takes similar delivery delay as AODV. So, for the delivery delay comparison, we compare the delivery delay of SCCF with that of AODV. The overall SCCF performance for $l = 250, 500, 750, 1000$ is given in Fig. 3.11a and the zoom-in view of both SCCF and AODV performance is shown in Fig. 3.11b. We observe that regardless of the network area, given the maximum delay taken by AODV to deliver the packets, SCCF delivers far beyond what AODV does. It is observed that as the network becomes sparser, more SCF forwarding is involved in seeking the broadcast-based vicinity-aware forwarding opportunities. Therefore, SCCF incurs longer delay in sparser network. However we see that the delay incurred in SCCF is much lower than the original SCF routing in sparse network, which is discussed in the following section.

(a)

(b)

Figure 3.11: (a) CDF with respect to packet delivery delay; (b) zoom in view with AODV.

(a)

(b)

Figure 3.12: The effect of fading: (a) CDF with respect to packet delivery delay; (b) CDF with respect to transmission count.

Effect of fading

In this study, we evaluate the SCCF and ER performance under different fading environments. To simulate the sparser and more dynamic network, we use 25 nodes in a 1000 m x 1000 m area. Each source generates maximum 300 packets. The buffer size is set to be 1500 packets for all protocols. This value is selected based on the buffer requirement study in the preliminary work [25], where it shows that ER requires 50% of the total packet for buffer space to maintain the maximum data delivery.

Firstly, we study the CDF of packet delivery with respect to delivery delay as shown in Fig. 3.12a. We observe that SCCF delivers the most number of packets with the minimum delay under the intermittent network environments. Specifically, we note that with the delay ≤ 50 seconds, 95%, 91% and 80% of the packets are delivered by SCCF under Freespace, Nakagami, and Rayleigh fading in contrast to 31%, 13% and 7% by ER, respectively. Furthermore, with the delay ≤ 100 seconds, almost 100% packets are delivered by SCCF in contrast to 24 to 75% by ER. From these facts, we observe that SCCF can deliver most packets with minimum delay and only a few percentage of packets takes extra delay. In SCCF, it scans around its contact nodes, monitors and exploits broadcast-based vicinity-aware forwarding if there is any, which significantly reduces the delivery delay. Although ER is robust to opportunistic links and dynamic topologies in the network, simply forwarding to the available contact incurs longer delay and the situation becomes severe in fading due to the higher chance of unsuccessful reception.

We then study the CDF of packet delivery with respect to the transmission count. As shown in Fig. 3.12b, at most 18 transmission count are required to maintain the packet delivery 80% in SCCF. Whereas in ER, the transmission count rises to 120 to maintain the same delivery percentage.

Figure 3.13: Summary vector overhead in Epidmeic and SCCF routing.

From these facts, we observe that SCCF is able to reduce a larger number of unnecessary transmissions/replications using the proposed replication control where nodes in the network get vaccinated and/or recovered progressively by the partially-immunized vaccines and the fully-immunized vaccines.

We then show the overhead caused by the vector transmissions under Nakagami fading. Figure 3.13 shows the CDF of PDR and the vector transmission count incurred for the delivery over the simulation time. The traffic is generated from 5 seconds to 207 seconds and the simulation is run for 900 seconds. As SCCF uses reactive form of replication, there is no summary vector transmission in the beginning when there is no traffic generated in the network. It can be observed that in SCCF routing the vector transmissions is stopped even before some packets are not yet delivered to the destination. In SCCF, replication is triggered only when the node has the packets which have not yet been forwarded by direct and cooperative forwarding. So, for

Figure 3.14: (a), (b), and (c) The effect of obstacle-constrained environment. (d) The effect of varying mean speed.

some nodes which have packet copies but know that the packet has been locally forwarded before will not use replication-based forwarding. However in ER, we can see the vector transmissions in the network even when there is no traffic in the beginning due to the proactive nature of replication. It is observed that the vector transmissions in ER continues even after the 90% of delivery due to the lack of delivered information in ER.

Effect of obstacles

In this study, we evaluate the performance of SCCF, ER, and AODV under an obstacle-constrained environment. A simplified version of Mission Critical Mobility (MCM) model [56] is used. In this model, nodes move around the obstacles in the network. For the propagation model, a modified version of the two-ray-ground model [56] which takes into account the impact of the physical obstacles on signal propagation is used. We use the same obstacle-constrained environment scenario as in reference [56]. The width and height of the four obstacles are set as (470, 210), (150, 360), (200, 490), and (320, 270) respectively.

The PDR, average delivery delay, and average packet transmission count under obstacle environment with different node densities are presented in Figs. 3.14a, 3.14b, and 3.14c. We see that the PDR of SCCF is not affected by environment with obstacles where PDR is maintained at almost 100% in both with and without obstacle scenarios. Similarly, the average transmission count of SCCF remains unaffected, with only a negligible increase in the obstacle scenario. Furthermore, we note that, in SCCF, increasing node density does not increase the transmission count unlike ER and it is just a few times more than the transmission count of end-to-end routing, due to the advantage of the proposed information epidemics control. However, for the average delay of SCCF, there is an increase in delivery delay in the obstacle-constrained scenario. This happens due to the fewer opportunities to timely exploit the *direct*, *two-hop*, *forwarder*, and *forwarder-by-chance* forwarding approaches. We also observe that ER needs higher node density to achieve better PDR and delivery delay in contrast to SCCF which maintains the similar performance in networks with low node density as well.

Effect of speed

In this study, we evaluate the performance of SCCF, ER, and AODV under different mean speed. The PDR, average delivery delay, and average packet transmission count under Nakagami fading environment with different mean speeds are presented in Figs. 3.14d, 3.15a, and 3.15b. We note that SCCF is robust to the node speed maintaining almost 100% PDR with minimum replications whereas in ER, higher contact rate among nodes (higher speed) is required to have minimum delivery delay. We also note that in ER, when the contact rate is low and the contact time is longer (this happens when the node speed is relatively low), the network incurs more unnecessary transmission/replications and also longer delay. However, in SCCF, we observe that the data forwarding is robust to high/low contact rate and contact time. This is due to 1) nodes always scan and exploit the better forwarding opportunities in their vicinity; 2) nodes also give copies to their contacts to help scan and exploit if no opportunities is found; and 3) unnecessary transmissions/replications are well controlled by the vaccinating scheme.

Figure 3.15: The effect of varying mean speed.

Figure 3.16: (a) CDF with respect to packet delivery delay; (b) CDF with respect to packet transmission count.

Study under mobile social network model

In this study, we evaluate the performance of SCCF and ER under a mobile social network model. We use the SWIM model [57] to generate the scenarios. The SWIM model was shown to have the same statistical properties of real social network traces and predict the performance of the SCF protocols accurately [63]. We use the same parameters as per the reference [57] to model the real-world traces in the Infocom05 and Infocom06 [64]. We use 802.11a with the freespace propagation model with a transmission of 160 m. We do not add the fading in this study as the model provides the inter-contact time distribution, contact distribution, and number of contacts per pair of nodes. We consider a network area of $1600 \times 1600 \text{ m}^2$ so as to keep the transmission range parameter in the SWIM model as 0.1.

We first study the Infocom05 scenario. The number of nodes in the network is 41. The CDF of packet delivery with respect to end-to-end delay and the packet transmission count are shown in Figs. 3.16a and 3.16b. We ob-

Figure 3.17: (a) CDF with respect to packet delivery delay; (b) CDF with respect to packet transmission count.

serve that most packets are delivered with lower delivery delay using SCCF compared to ER. It is noted that exploiting the cooperative forwarding opportunities whenever available and adaptively switching to SCF forwarding guarantee faster delivery compared to the flooding protocol. This encourages adding the vicinity aware forwarding on top of SCF forwarding for faster delivery. This observation is much in line with the findings in the reference [19], where it shows that a large amount of delay can be reduced using 2-hop opportunity compared to using 1-hop contact only. In addition, having early control signal distribution using partially-immunized vaccine and fully-immunized vaccine results in lower infection rate. Therefore, most packets are delivered with less number of forwarding count in SCCF compared to ER, as shown in Fig. 3.16b. We then study the Infocom06 scenario, where the number of nodes in the network is 78. Similar observations can be seen in this case as well. However, with more nodes in the network, the number of contacts and the contact rate are increased. Therefore, faster delivery is achieved for both SCCF and ER protocols, shown in Fig. 3.17a, compared to

Figure 3.18: Delivered packet probability with respect to packet transmission count.

the network with 41 nodes. For the transmission count, we note that more replications are being made in this case due to more number of contacts with more nodes. However, as shown Fig. 3.18, regardless of the node density, SCCF delivers most packets with fewer replications whereas, in ER, a large number of replications is made and it becomes more significant when the network is denser.

3.9 Conclusion

In this chapter, we have proposed a data delivery solution for opportunistic networks. The solution comprises two main algorithms: store-carry-cooperative forward routing and information epidemics control. In the data forwarding, nodes proactively scan the contact opportunities in the vicinity

Table 3.8: 95% Confidence Intervals for Comparison with CORMAN Study

Network area(m)	SCCF	AODV
250	1.000 ± 0	0.958 ± 0.0436
300	1.000 ± 0	0.850 ± 0.1243
350	1.000 ± 0	0.821 ± 0.1661
400	1.000 ± 0	0.660 ± 0.1921
450	1.000 ± 0	0.595 ± 0.2798
500	1.000 ± 0	0.615 ± 0.2209
550	0.999 ± 0.0020	0.509 ± 0.2776
600	1.000 ± 0.0001	0.254 ± 0.1988
650	0.986 ± 0.0304	0.194 ± 0.1914
700	0.998 ± 0.0040	0.473 ± 0.2821
750	0.982 ± 0.0215	0.270 ± 0.1847
800	0.975 ± 0.0222	0.160 ± 0.1596
850	0.989 ± 0.0136	0.381 ± 0.2792
900	0.960 ± 0.0468	0.262 ± 0.2092
950	0.961 ± 0.0490	0.244 ± 0.2069
1000	0.947 ± 0.0403	0.150 ± 0.1582

and use broadcast-based cooperative forwarding for data delivery whenever there is an opportunity. Nodes adaptively switch between the cooperative

Table 3.9: 95% Confidence Intervals for Effects of Fading Study

	Fading model	SCCF	ER	AODV
PDR	Freespace	0.999 ± 0.0001	0.941 ± 0.023	0.568 ± 0.075
	Nakagami	0.999 ± 0.0001	0.846 ± 0.036	0.234 ± 0.080
	Rayleigh	0.999 ± 0.0001	0.785 ± 0.031	0.138 ± 0.061
E2ED	Freespace	15.41 ± 2.501	75.25 ± 4.541	1.225 ± 0.353
	Nakagami	21.51 ± 2.601	113.35 ± 7.312	5.428 ± 2.939
	Rayleigh	33.52 ± 3.372	145.93 ± 7.549	4.375 ± 2.681
TXC	Freespace	8.445 ± 0.728	54.14 ± 3.832	3.122 ± 0.266
	Nakagami	11.90 ± 0.851	54.436 ± 3.292	2.484 ± 0.242
	Rayleigh	16.98 ± 1.410	55.39 ± 3.065	1.906 ± 0.594

Table 3.10: 90% Confidence Intervals for Study under Mobile Social Network Model

	No. of nodes	SCCF	ER
PDR	41	0.467 ± 0.0950	0.304 ± 0.0580
	78	0.564 ± 0.0904	0.446 ± 0.0751
E2ED	41	163.42 ± 40.544	200.67 ± 48.525
	78	40.52 ± 15.67	67.53 ± 12.621
TXC	41	18.44 ± 2.454	37.87 ± 7.330
	78	45.248 ± 6.219	67.12 ± 7.626

forwarding and store-carry-forward routing based on the contact opportunity. Nodes reactively replicate the packet copies only when the packet has never been forwarded by cooperative forwarding. Moreover, overhearing the cooperative transmissions over the air provides the node the ability for better information epidemics control. The susceptible-infected-recovered model is used to study the effectiveness of the proposed mechanism. The performance evaluation is conducted under various scenarios which include Nakagami and Rayleigh fading environments, obstacle-constrained environment, and mobile social network environment. We have found that 1) the proposed information epidemics control mechanism provides higher vaccination rate and recovery rate; 2) proactive replication incurs a number of unnecessary transmissions; 3) with the integrated solution, high data delivery is maintained regardless of the connectivity in fading environment where it achieves almost 30% relative PDR improvement compared to CORMAN's performance; 4) most packets are delivered with the minimum delay compared to AODV and ER; and 5) a substantial amount of unnecessary transmissions/replications are well deterred.

Chapter 4

Cluster-based Cooperative Data Forwarding with Multi-radio Multi-channel for Multi-flow Wireless Networks

4.1 Introduction

The capacity of wireless ad hoc networks is mainly limited by co-channel interference and unreliable wireless channels. Co-channel interference can be minimized by using multi-radios [65]. But it requires a proper channel assignment to assign different channels to use in different nodes. A number of works focusing on channel assignment has been done for Traditional deterministic Routings (TRs), such as DSR and AODV, for multi-radio multi-channel wireless ad hoc networks. However, it has been proved that those TRs perform poorly under unreliable and lossy wireless channels due to interference and multipath fading [13]. It has been shown that cooperative forwarding can

highly increase the reliability of data communications under lossy wireless channels [12, 40]. In order to improve the system capacity, a promising way is to employ cooperative forwarding with multi-radio multi-channel (MRMC) functions.

The basic idea of cooperative forwarding is to make use of receiver diversity (i.e. spatial diversity) and provide reliable communications over long, weak wireless links. The fundamental design components of cooperative forwarding include 1) routing metric computation; 2) candidate relay set (CRS) selection; 3) candidate relays prioritization (CRP); 4) broadcasting data packets to candidate relays; and 5) best relay selection using a coordination scheme. Source selects CRS and conducts CRP before forwarding. Timer-based coordination is mostly used in the existing works for transmission scheduling and duplicate transmission avoidance [42, 12, 40]. During the coordination phase, the candidate nodes set a waiting delay proportional to its priority as per CRP. Nodes have to listen higher priority nodes transmission and discard its schedule if the same packet is being forwarded by the higher priority node.

From this coordination nature, we observe that the following problems may arise under multiple traffic flows, especially when the same CRS is being used for different flows.

- The forwarding node being unable to hear the higher priority node transmission successfully due to the traffic coming from another source and therefore wrongly believes that the higher priority node did not receive the packet, resulting in duplicate transmissions.
- Having multiple schedules for multiple flows and inaccuracy in scheduling due to the unavailable medium, resulting in forwarding the same packet by different forwarding nodes at the same time.

- All forwarding nodes being unable to receive the data packet from the source/sender node due to collisions, resulting in unsuccessful data delivery.
- Unnecessarily waiting for the higher priority node transmission when it is the only node who receives the packet correctly.

Given the advancement of wireless radios equipped with multiple interfaces working on multiple radio frequency (RF) channels, it is a promising way to incorporate the multi-radio multi-channel (MRMC) capability into cooperative forwarding to further enhance the system throughput, especially in the presence of highly congested multiple cross traffic flows. A number of works on joint channel assignment and routing in MRMC wireless networks had been proposed for traditional routing mechanisms [66, 67, 68, 69, 70]. In the traditional routing, data forwarding takes place between a sender and a pre-selected next-hop node. No diversity is involved. In channel assignment for the traditional routing, a pair of nodes uses orthogonal channel to other neighbouring pairs or a set of nodes for one flow use orthogonal channel to other nearby flows to minimize the interference and maximize the throughput.

However, in cooperative forwarding, it has been said that "multi-carrier utilization reduces interference, but it also reduces the spatial diversity gain offered by cooperative forwarding, since the number of relays operating over a common channel decreases" [42]. In [71], it is also stated that incorporating MRMC in cooperative forwarding can be considered as a trade-off between multiplexing and spatial diversity and what choice the neighbouring nodes should make is non-trivial. In [72], it is shown that traditional channel assignment does not work well in cooperative forwarding. Therefore, in cooperative

forwarding, one has to jointly address multiple issues: radio-channel assignment, CRS selection, CRP, data broadcasting over MRMC, and best relay selection and duplicate transmission avoidance using a coordination scheme. This chapter presents a simple and efficient cluster-based cooperative data forwarding (CCDF) which jointly addresses all these issues.

The following summarizes the major contributions of this work. The cluster-to-cluster expected transmission count *C2CETX* metric is introduced to compute cluster path for forwarding. Distributed agent-based coordination (DAC) mechanism is proposed so as to support best candidate relay selection and duplicate transmission avoidance in both single-radio single-channel (SRSC) and multi-radio multi-channel (MRMC) networks. The MRMC capable cooperative data forwarding mechanism which supports both multiplexing and spatial diversity is proposed. Analysis on the coordination time required by the DAC mechanism and comparison with the one required by existing timer based coordination method are conducted. Performance of the proposed solution is evaluated in both SRSC and MRMC networks and under the situation where the same set of forwarding candidates is being used by multiple flows.

The rest of the chapter is organized as follows. Section 4.2 reviews related works on cooperative forwarding in MRMC networks. The network model used in this work is discussed in Section 4.3. Section 4.4 presents the routing metric. The proposed data forwarding scheme with several protocol details is presented in Section 4.5. Integration of proposed data forwarding with MRMC capability is discussed in Section 4.6. The analysis on the coordination scheme is presented in Section 4.7. Section 4.8 evaluates the efficacy of the proposed scheme with single-radio single-channel and multi-radio multi-channel under multiple cross traffic flows. The concluding remarks are

provided in Section 4.9.

4.2 Related Work

4.2.1 Cooperative Forwarding

Originally proposed for static dense wireless networks, cooperative forwarding is a receiver diversity based routing mechanism which exploits the advantage of broadcast nature of wireless medium [12, 40, 73, 74, 75, 42]. Its main focus is to achieve high throughput compared to unicast-based single path routing over the unreliable wireless channels. The efficiency of cooperative forwarding is first demonstrated in ExOR [12]. The fundamental design components of cooperative forwarding include candidate relay set selection, candidate relay prioritization, data broadcasting to the candidate relays, and best relay selection using a coordination scheme [42, 15].

Timer-based coordination is mostly used in the existing works for transmission scheduling and duplicate transmission avoidance [42, 12, 40]. During the coordination phase, the candidate nodes set a waiting delay proportional to its priority as per the one defined by the source. Nodes have to listen higher priority nodes transmission and discard its schedule if the same packet is being forwarded by the higher priority node. So, nodes cannot miss the transmission from higher priority nodes, or else it will make duplicate forwarding. In cooperative forwarding, sender sends data packets in bulk transfer (i.e., a fragment of packets) and candidate nodes start coordination for further forwarding at the end of the fragment transmission. The highest priority candidate will have the shortest delay before further forwarding. The subsequent candidates will have to delay for the sufficient amount of time in order not to make any duplicate transmissions which has been received at

the higher priority candidates.

4.2.2 Integration with Multi-Radio Multi-Channel

A number of works on joint channel assignment and routing in MRMC wireless networks have been proposed [66, 67, 68, 69, 70]. Existing channel assignment schemes are based on assigning a frequency on a link between two nodes or on a series of connected link between source and destination. Whereas for cooperative forwarding, a new channel assignment strategy which allows broadcast transmission to achieve receiver diversity while mitigating interference and providing multiple concurrent transmissions is required.

Most existing cooperative forwarding protocols were proposed for SRSC wireless network [42]. Consequently, research in MRMC cooperative forwarding is still in the very beginning stage [42]. A linear-programming based study on end-to-end throughput of MRMC cooperative forwarding is discussed in [71]. It shows the performance improvement achieved using the MRMC cooperative forwarding over the MRMC traditional routing. A single-radio multi-channel cooperative forwarding protocol is proposed in [72]. It uses only one radio and the channels are uniformly assigned to all nodes. The candidate relay set (CRS) per channel is formed and joint channel and CRS selection is performed to minimize the interference. The forwarding nodes switch the channel as per the destination's channel. However, it did not discuss a strategy on how to guarantee the receiver diversity when the candidate nodes are operating on different channel from source or sender. A similar protocol but with multiple radios is proposed in [76]. It uses one radio for control purpose and the others for data communications. Random channel assignment is used for each radio interface. Although it ensures at least one candidate receives the data packet, it does not guarantee receiver diversity

at most of the potential candidate nodes due to random channel assignment and channel switching. In [77], it studies the possible concurrent transmissions using MRMC. It assigns the same channel to all nodes in a flow and all forwarding nodes belong to the flow, which makes the nodes unavailable for other flows especially when a number of cross flow traffic flows are present. In [78], it extends a cooperative routing protocol to incorporate MRMC. It performs workload-aware channel assignment to candidate nodes and focuses on throughput maximization for single flow. Multiple traffic flows are not considered.

4.3 Network Model

We consider a network with C clusters and K orthogonal channels. The proposed forwarding mechanism makes use of clustering to support 1) candidate relay coordination and 2) multiplexing and diversity in multi-flow network. In this work, we focus on cooperative data forwarding under multiple cross traffic flows and then maximizing its throughput using MRMC features. So, for clustering, we assume that the network is already composed of clusters and nodes are organized by clusters according to their geographical/logical relationships. We also assume that each cluster comprises member nodes and a head and the link between member and its head has non-zero link delivery probability.

4.4 Cluster-to-cluster ETX (C2CETX)

In CCDF algorithm, nodes conduct live-updating link qualities to monitor the unreliable and probabilistic nature of wireless channel and to compute

Figure 4.1: Link delivery probability p .

IP header		
contact node ID n_j	p with n_j	p_{rev} with n_j
...		

Figure 4.2: Hello packet header format.

the cluster path for data forwarding.

Each node broadcasts a hello message periodically, at an average period of τ (we use one second in the performance evaluation). Nodes calculate bi-directional link qualities of the links connecting to the direct contact nodes. This link quality denoted as p is calculated as $p = p_{fwd} \times p_{rev}$, where p_{fwd} and p_{rev} are the link quality of forward and reverse links. The reverse link

IP header	
next-hop cluster ID	destination cluster ID
data relay node ID	ACK-sender ID
...	

Figure 4.3: Data packet header format.

quality p_{rev} with a contact node is calculated based on the received hello packet ratio during the last w seconds from that node. A node calculates p_{rev} using the equation shown below.

$$p_{rev}(t) = \frac{r(t-w, t)}{w/\tau} \quad (4.1)$$

where $p_{rev}(t)$ denotes the reverse link quality at time t , $r(t-w, t)$ denotes the number of hello received from the contact node during the last w seconds, and w/τ denotes the number of hello the contact node sent during the last w seconds.

The forward link quality p_{fwd} with a contact node is obtained as follows. In addition to the IP header in the hello message, nodes also add the contact IDs and the reverse link qualities with each of the contacts. These reverse link qualities in the hello message are indirectly telling each of the contact node the forward link quality from each of them. From this p_{fwd} and above discussed p_{rev} , nodes obtain the bi-directional link qualities p with each of the contact nodes as shown in Fig. 5.2. The illustration of hello packet header format is given in Fig. 5.4.

From the measured link delivery probabilities, nodes compute its link delivery probability to neighbouring clusters. The illustration with three clusters named A , B , and C is shown in Fig. 4.4. Clusters A , B , and C act here as a source cluster, forwarding cluster, and destination cluster, respectively. A node in cluster B computes its link delivery probability to cluster C as follows.

$$p_{b_i, C} = 1 - \prod_{j=1}^{|Z_C|} (1 - p_{b_i, c_j}), \forall b_i \in Z_B \quad (4.2)$$

where Z_B and Z_C refer to set of nodes in cluster B and cluster C . From $p_{b_i, C}$, the node computes the expected transmission count (ETX) [79] to cluster C as $ETX_{b_i, C} = \frac{1}{p_{b_i, C}}$. Nodes periodically update the measured ETX-to-cluster

values to its head. From these members update, head calculates the cluster-to-cluster ETX ($C2CETX$) as follows.

$$C2CETX_{B,C} = \sum_{e=1}^{2^{|Z_B|}-1} p(e) \min \left\{ x_{b_1}(e)ETX_{b_1,C}, \right. \\ \left. x_{b_2}(e)ETX_{b_2,C}, \dots, x_{b_{|Z_B|}}(e)ETX_{b_{Z_B},C} \right\} \quad (4.3)$$

where $C2CETX_{B,C}$ refers to ETX from cluster B to cluster C. $p(e)$ denotes the probability of each event e . The event e here refers to nodes in cluster C_i receiving the data packet from the source/sender successfully. Let χ denotes the set of all possible events, i.e. $|\chi| = 2^{|Z_{C_i}|} - 1$.

$$p(e) = \frac{1}{2^{|Z_{C_i}|} - 1}, e \in |\chi| \quad (4.4)$$

$x_{n_i}(e)$ is the binary variable which is defined as

$$x_{n_i}(e) = \begin{cases} 1, & \text{if } n_i \text{ is selected in event } e \\ 0, & \text{otherwise} \end{cases} \quad (4.5)$$

We assume that the probability of each event is evenly distributed and the probability that all nodes fail to receive the data packet from cluster A is zero. Therefore, the calculation in (5.2) serves as a rough, conservative estimate of the cluster-to-cluster transmission count.

One can compute a closer estimation of $C2CETX_{B,C}$ as follows. Consider the case where $p_{b_3,C} > p_{b_2,C} > p_{b_1,C}$. This indicates that node b_3 has highest reachability to cluster C , followed by b_2 , and then b_1 . If we know the sending node from cluster A in advance, $C2CETX_{B,C}$ can be obtained as follows.

$$C2CETX_{B,C} = p_{a_i,b_3}ETX_{b_3,C} + (1 - p_{a_i,b_3})p_{a_i,b_2}ETX_{b_2,C} + \\ (1 - p_{a_i,b_3})(1 - p_{a_i,b_2})p_{a_i,b_1}ETX_{b_1,C} \quad (4.6)$$

where a_i refer to the sending node from cluster A . This calculation requires sending node ID in advance. In CCDF, we use (5.2) to compute a rough estimate on the connectivity between clusters.

Figure 4.4: Network with three clusters and $C2CETX$ between clusters.

4.5 Cluster-based Cooperative Data Forwarding (CCDF)

In this section, we discuss the details of the CCDF data forwarding algorithm. We first describe an overview of the forwarding procedure. We then elaborate the details of the proposed distributed agent-based coordination. We also present the additional components of the protocol that complement data forwarding which include network layer acknowledgement, packet buffers (primary and temporary), augmentation for data resiliency, and handling link loss. We also provide an example illustration of CCDF data forwarding in Fig. 4.6. The discussion based on the example is also provided at the end of this section.

Each cluster head maintains a table with the $C2CETX$ to all other clusters in the network. It then updates its members the consolidated $C2CETX$ periodically. Cluster head dynamically selects a gateway node to each contact cluster based on the highest link delivery probability to the cluster. In CCDF, nodes which have direct connectivity to a neighbour cluster are called as bridge node to the cluster. So, a gateway node can be considered as one of the bridge nodes which has strongest link quality to the neighbour cluster. Note that different nodes can be assigned as gateway to different contact

clusters or a node can be assigned as gateway to more than one contact clusters. Cluster head will conduct reselection if there is a change in link delivery probability. This information is also added in the hello message. Therefore, member nodes are aware of gateway nodes to each contact cluster.

We focus on the inter-cluster traffic flow in this work as this type of traffic has to be traversed over multiple hops and has higher chance to be affected by other cross traffic flows. Suppose the source cluster is A and the destination cluster is C . A source node a_1 chooses a cluster path with the lowest $C2CETX$ value, i.e. the sum of the cluster link $C2CETX$ value, and performs the data forwarding as follows. Note that a source node chooses a next-hop cluster only if none of the member nodes in the cluster has a non-zero link delivery probability to cluster C .

If the source node finds that the source cluster A has direct connectivity to the destination cluster C , it first checks if it has direct connection to the destination cluster. If it does not have a direct connectivity, it will then find a bridge node among its reachable members to the destination cluster. If it finds that it does not have direct connectivity to any bridge node, it will then get a member node as a relay and forward the data packet to the gateway via the relay node. Upon receiving the data packet, the bridge node or the gateway node broadcasts the packet to the destination cluster directly.

If the source node finds that there is no direct connectivity to cluster C , it forwards the data packet as follows. It first check if it has connectivity to the next-hop cluster. Suppose cluster B is the next-hop cluster. If it has connectivity to cluster B , it will broadcasts the data packet to cluster B directly. Otherwise, it will forward the packet to a bridge node or the gateway node via a relay if it does not have connectivity to any bridge node. The bridge node or the gateway node will then broadcasts the packet to cluster

Figure 4.5: Illustration of various data forwarding paths from the source/sender cluster to the next-hop or the destination cluster.

Figure 4.6: Illustration of data forwarding in CCDF. For CCDF-MRMC, please refer to the channel number indicated in the bracket.

B. Upon receiving the data packet, cluster *B* performs distributed agent-based coordination as follows and further forward the packet. The above discussed various data forwarding paths from the source/sender cluster to the next-hop or the destination cluster are shown in Fig. 4.5. An illustration of data forwarding from cluster *A* to cluster *C* is shown in Fig. 4.6.

4.5.1 Distributed Agent-based Coordination

We use cluster head as an agent for making decision on selecting a candidate out of all receivers to further forward the data packet. Upon receiving the data packet, nodes update head about the reception information (*senderID*, *packetID*). We assume a packet has its unique ID. As we consider multiple

traffic flows coming from any directions, the sender ID associated to each flow is attached in the reception information. Each node holds the reception for a period of time to append all the possible incoming traffic and informs head once timeout. Upon receiving the reception information, head first selects the next-hop cluster C_i for each flow.

Head then selects a node which has the highest p_{b_i, C_i} value. If it finds that none of the receivers has connectivity to the next-hop cluster, it selects a node which has highest p_{b_i, g_i} value to the gateway node g_i . After making the decision for all the receptions, head broadcasts its decision to the member nodes. For reliability, head sends the decision for three times so that all receivers receive the decision successfully and process the data packet accordingly. This decision re-send also notifies other nodes who has late reception, so that those nodes can promptly discard the packet without further processing.

Upon receiving the decision, nodes which are not selected as forwarding node discard the packet. Therefore, all the receivers not selected as forwarding node need not require any scheduling and processing for the data packet, resulting in higher probability of resource availability for other flows. In addition, the node selected as the forwarding candidate can immediately perform further data forwarding. In Section 4.7, we analyse the time taken to perform distributed agent-based coordination and compared it with existing timer-based coordination method.

In CCDF, upon receiving the reply from the agent, nodes discard the data packet in the buffer if the ID is not listed or schedule their transmissions as follows. Nodes first sort the number of packets to be sent at each candidate node in ascending order. For the same number of packets, node with lowest ID will schedule its transmission first followed by the second lowest ID. Hav-

ing this agent-based scheduling, it brings the following benefits. 1) Nodes does not require to involve in scheduling if it is not assigned as the forwarding candidate; 2) the forwarding candidate becomes the custody of the packet to ensure the delivery instead of source; 3) there is no scheduling misalignment, duplicate transmissions, and collisions due to inaccuracy in scheduling; and 4) it shortens the coordination time especially for multi-flow network with high traffic load.

By having forwarding coordinators distributed over the network, no global priority scheduling is required. As a result, it can prevent the following situations. 1) Nodes being unable to hear the higher priority node transmission successfully due to the traffic coming from another source and therefore wrongly believe that the higher priority node did not receive the packet, resulting in duplicate transmissions; 2) having multiple schedules for multiple flows and inaccuracy in scheduling due to the unavailable medium, resulting in forwarding the same packet by different forwarding nodes at the same time; and 3) unnecessarily waiting for the higher priority node transmission when it is the only node who receives the packet correctly.

4.5.2 Receiving cluster diversity

CCDF also allows to take advantage of transmissions reaching nodes in a cluster closer to the destination cluster, other than the nodes in the next-hop cluster. That is, due to dynamic wireless link nature, it is possible that cluster C receives the data packet from cluster A over long, weak wireless links. To exploit this spatial diversity, cluster C also performs agent-based coordination and further forward the data packet to the destination cluster D . Upon overhearing the transmission from cluster C , the forwarding node in cluster B discards the ongoing transmission. It also inspects the interface

queue and drops the packet if it has passed the packet to lower layer for transmission.

4.5.3 Network Layer Acknowledgement

In CCDF, nodes use IEEE 802.11 unicast for intra-flow data transmission and IEEE 802.11 broadcast for inter-flow data transmissions. We use network layer acknowledgement scheme as IEEE 802.11 broadcast mode does not provide any acknowledgement for the broadcast packet. Prior to forwarding the data packet to another cluster, the source/forwarding/gateway node dynamically chooses a node in the receiving cluster as ACK reply node. It selects the node with which it has the highest link delivery probability. It then inserts the nodeID in the packet header as shown in Fig. 5.5 and forwards the packet. Upon receiving the data packet from another cluster, a node sends an ACK if it is assigned as the ACK reply node in the packet header. Due to the dynamic nature of wireless links, different nodes can be assigned as ACK-sender for different data packets. In case of missing acknowledgement from the next-hop cluster, the data packet sender can still be informed about the successful packet reception when it overhears the data packet transmission from the next-hop cluster.

CCDF use cluster-by-cluster acknowledgement. That is, once the source or forwarding node receives the ACK from the next-hop cluster, it discards the packet from its buffer. The forwarding node in the next-hop cluster becomes the custody for the data packet until it has successfully forwarded the packet to its next-hop cluster or destination cluster.

We use cumulative acknowledgement instead of acknowledgement for individual packet to reduce the control packet overhead and collisions to data packet transmissions. This acknowledgement is transmitted in broadcast

mode. Therefore, it informs not only the sender about successful reception of the data packet but also other nodes who are within the communication range. For ACK message reliability, the gateway node re-broadcasts the ACK it receives from next-hop cluster to make sure that the source/forwarding node are acknowledged.

4.5.4 Packet Buffer

In the proposed protocol, a node maintains two types of buffer to store data packets as follows.

Primary packet buffer (buffer) - It is used to store data packet while waiting for the head decision about the selected forwarding candidate.

Temporary packet buffer (temporary buffer) - It is used to keep copies of packets which are being forwarded to next-hop cluster but waiting for acknowledgement to ensure the successful reception. After the node overhears the acknowledgement from the ACK reply node, the corresponding packet copy will be removed from the temporary buffer. Otherwise, the node will retrieve the packet from the temporary buffer and store it in the primary buffer after acknowledgement timeout.

4.5.5 Augmentation for Data Resiliency

As data packets are sent over long, weak wireless links, CCDF adds an augmentation mechanism to ensure data resiliency. When a cluster finds that if it has a non-zero link delivery value to the destination cluster, it will forward the data packet directly without traversing a relay cluster. However, sending over the long weak links likely incurs transmission failure with higher probability. Therefore, to strengthen this data delivery, an augmentation scheme is introduced. Suppose a_2 is the sender, c_1 is the ACK-sender selected by

a_2 and clusters A and C are the source/sender cluster and the destination cluster, respectively as shown in Fig. 4.7.

Consider the case where only nodes b_0 , b_1 , b_3 , and b_4 receive the data packet successfully from a_2 . Suppose b_0 , b_1 , b_2 , b_3 , b_4 , and c_1 are listed as neighbours of a_2 and b_0 , b_2 , b_3 , and b_4 are listed as neighbours of c_1 in the most recent hello message of a_2 and c_1 , respectively. The hello message header format is given in Fig. 5.4. In this case, it can be observed that only nodes b_0 , b_3 , and b_4 will participate in the augmentation for data resiliency.

Assume that nodes participating in the augmentation process are denoted as n_k . A node n_k will further check whether 1) it has direct contact with both a_2 and c_1 , i.e. both p_{n_k,a_2} and p_{n_k,c_1} are non-zero.; 2) $p_{a_2,c_1} < 0.55$, i.e. most likely a_2 will require at least two transmissions to successfully deliver a data packet to c_1 .; and 3) $p_{n_k,c_1} - p_{a_2,c_1} \geq \alpha$. If the conditions are met, the node n_k will self-evaluate other nodes who meet the above conditions as well and calculate its rank among all potential nodes. Note that the information contained in the hello message of a_2 and c_1 is sufficient to self-evaluate other potential nodes. The node n_k will then set a timer based on the length of delay as in (5.14) and will broadcast the packet when the timer expires.

$$Delay(n_k) = r \times t_x + r \times t_{ack} \quad (4.7)$$

where r is the rank of n_k , i.e. $r = 1$ being the highest rank node. t_x is the jitter which follows uniform distribution. t_{ack} is the time taken to receive an acknowledgement from c_1 (90 ms in the simulation). The value of α is set to 0.1, i.e. a node n_k can join the augmentation process only when it has at least 10% better link delivery probability than a_2 . The node n_k will discard its ongoing transmission if it overhears the transmission from other n_k nodes or the acknowledgement from node c_1 . By having the backup transmissions, the success probability of data delivery from a_2 to the destination cluster C

Figure 4.7: Illustration of augmentation for data resiliency.

is improved.

4.5.6 Handling Link Loss

Due to the dynamic and random nature of wireless link, the communication link can be lost while packets are being forwarded. To stop sending out packet if the link is lost, the link layer maintains a bi-directional link delivery information table and passes the packets to interface queue only if the link delivery probability is non-zero. In case of link loss, the link layer will discard the packet.

4.5.7 Data Forwarding Example

An example illustration about the forwarding in CCDF is given in Fig. 4.6. There are three clusters: A , B and C in this example. Node a_1 from cluster A is the source and c_2 from cluster C is the destination. Assume none of the nodes in cluster A has non-zero delivery probability to cluster C . From the $C2CETX$ information, a_1 chooses cluster B as the next-hop cluster. Suppose the link delivery probability from a_1 to cluster B (i.e., $p_{a_1,B}$) is zero and $p_{a_2,B} > p_{a_0,B} > p_{a_3,B} > p_{a_4,B}$. Cluster head of A (assumes a_0 in this

example) assigns node a_2 as the gateway node to cluster B as it has highest delivery probability to B among all other nodes.

As a_1 has no direct connectivity to B , it unicasts its data packet to the gateway node a_2 . Upon receiving the data packet, a_2 chooses b_1 as the node to reply *ACK* for its upcoming transmission, insert b_1 's ID in the packet header and broadcasts the packet. Upon receiving the data packet from a_2 (nodes receiving a_2 's transmission are shown in the dotted line), nodes in B reply *ACK* if it is assigned as the *ACK* reply node. Note that the node a_2 , which is the current gateway node to B will re-broadcast the *ACK* reply upon receiving so as to ensure that all the nodes in A receive the reply. Nodes in B will then perform distributed agent-based coordination. Suppose cluster head of B (assumes b_0 in this example) selects node b_3 as the forwarding node as it is the best candidate among others. Upon receiving the decision, nodes not selected as the forwarding candidate will drop the packet. The forwarding node, which is b_3 in this example will then broadcast the data packet to cluster C . Here, it chooses c_1 as the *ACK* reply node for its upcoming transmission.

Upon receiving the data packet from b_3 , the *ACK* reply node, which is c_1 in this example, will first reply the *ACK* to B . As the receiving node know that the destination, which is c_2 in this example is a member in its cluster, each receiver (c_0, c_1 , and c_3) will self-evaluate the time-to-forward based on its delivery probability to c_3 using (5.14) and broadcasts once timeout. Upon receiving the data packet, c_3 replies the *ACK* using broadcast. By overhearing another receiver's transmission or the *ACK* reply from c_3 , the rest of the receivers will cancel their schedule and drop the packet.

4.6 CCDF with MRMC

Although we can now avoid coordination misalignment and scheduling inaccuracies, there still exists interference and collisions in multiple flow network. As a result, it causes all potential forwarding nodes being unable to receive the data packet from source/sender and hence, resulting in throughput drop and unsuccessful data delivery. In order to suppress potential interference and collisions and improve the system capacity, a promising way is to employ cooperative forwarding with multi-radio multi-channel (MRMC) functions. In the following, we first discuss how radios and channels are being assigned in CCDF and then present the cooperative data forwarding over diverse channels.

4.6.1 Cluster-based Channel Assignment

We use $K - 1$ channels for data transmission/reception and 1 channel for control purpose. We call $K - 1$ channels “data channel” and the one used for sending/receiving control messages “control channel. Each node is equipped with three radios namely $R1$, $R2$, and $R3$. The radio $R1$ is used for control purposes and therefore, it tunes to control channel. The second radio $R2$ is used for receiving incoming packets and the last radio $R3$ is used for transmitting outgoing packets. By allocating the transmit radio and receive radio specifically, it supports multiplexing for multiple receptions with unlimited number of channels. In CCDF-MRMC, channel assignment is required only for radio $R2$. Cluster head chooses a channel for $R2$. Assume each cluster C_i , where $C_i \in C$, has a unique ID, ID_{C_i} ; $0 \leq ID_{C_i} \leq |C| - 1$. Cluster head of each cluster conducts an initial channel assignment for $R2$ as follows.

$$k_{C_i} = ID_{C_i}(\text{mod } K - 1) \quad (4.8)$$

where k_{C_i} refers to the channel number of $R2$ at cluster C_i . Member nodes in each cluster set k_{C_i} to $R2$ upon receiving notification from the head. As we focus on the scheduling and data delivery problem in cooperative forwarding under multi-flow network, we assume that every adjacent cluster in the network uses different channel for $R2$. For the inter-flow interference estimation and channel reassignment, one can use the approach in iAWARE protocol [66, 80] and switch the channel for $R2$. As the adjacent cluster selects different channels for $R2$ and the node switch $R3$ as per the next-hop cluster $R2$'s channel for transmission, the probability of having intra-flow interference is low.

4.6.2 Data Forwarding over Diverse Channels

For intra-cluster data communications, nodes use $R2$ for both transmission and reception. For inter-cluster data communications, nodes use radio $R2$ for reception and $R3$ for transmission. All members in a cluster use the same channel on $R2$. By tuning the same channel on $R2$, the receivers diversity required for cooperative data forwarding is achieved.

Consider clusters A , B , and C are the source, next-hop, and the destination cluster respectively. Let a_i be the source/sender node in cluster A and b_i be the forwarding candidate selected using distributed agent-based coordination in cluster B . Let $R2$ of clusters A , B , and C are on channels 1, 2, and 3, respectively. The data packet from a_i to cluster C will be forwarded as shown in Fig. 4.8.

As described in Fig. 4.8, when there is no direct connectivity to cluster B , a_i uses $R2$ to reach to gateway node a_{gw} . Then the gateway node uses $R3$ and tunes the channel to the one cluster B is using for receiving. Otherwise, it uses its $R3$ radio and tunes the channel to cluster B 's receiving channel.


```

1: First cluster-hop forwarding:
2: if  $p_{a_i,B} == 0$  then
3:    $a_i \xrightarrow{ch2} a_{gw} \xrightarrow{ch3} B.$ 
4: else
5:    $a_i \xrightarrow{ch3} B.$ 
6: end if
7: Second cluster-hop forwarding:
8: if  $p_{b_i,C} == 0$  then
9:    $b_i \xrightarrow{ch3} b_{gw} \xrightarrow{ch4} C.$ 
10: else
11:    $b_i \xrightarrow{ch4} C.$ 
12: end if

```

Figure 4.8: Data forwarding over diverse channels.

Same concept applies to the second cluster-hop forwarding as well.

Using dedicated radio for transmission and for reception and switching the transmitter radio's channel based on the receivers' channel, concurrent transmissions is achieved while retaining the spatial diversity. In addition, as adjacent clusters are using different receiving channel, the interference caused by other flows coming to the adjacent cluster can be avoided. As discussed in CCDF, having distributed agent-based coordination enables to process multiple incoming flows at the same time. Incorporating MRMC into the CCDF further enhances the system capacity as forwarding candidates selected for each flow will likely be using different channels for their outgoing transmissions. Moreover, CCDF with MRMC works well for unlimited number of channels while requiring only three radios.

4.6.3 CCDF-MRMC Example

An example illustration of CCDF with MRMC is given in Fig. 4.6. Assume there are four channels $ch0, ch1, ch2$, and $ch3$ in the network (i.e., $K = 4$). Suppose R_1 of all nodes tune to channel $ch0$. Using (4.8), nodes in clusters A, B , and C choose $ch1, ch2$, and $ch3$ for their $R2$ radio respectively. Therefore, as shown in Fig. 4.6, when a node sends data packets to another cluster (i.e., inter-cluster communication), it uses $R3$ radio and tunes the channel of $R3$ to the receiving cluster's $R2$'s channel. Note that for the intra-cluster communication, it uses $R2$ radio. It is also important to point that CCDF-MRMC allows channel reuse. That is, if there are only 3 channels in the network (i.e., $K = 3$), cluster C will reuse $ch1$ and therefore, its $R2$ radio will be operating on $ch1$. We will show the performance of both cases, i.e., with reuse and without reuse, in the performance evaluation.

4.7 Analysis on Coordination time in CCDF

In this section, we conduct analysis on the time required for coordination using proposed agent-based coordination and the timer-based coordination method used by existing cooperative forwarding protocols [42].

4.7.1 Single flow

Consider a traffic flow $f(a_i, c_i)$ or f_i , where a_i and c_i are the source node and destination node, respectively. Let nodes b_1, b_2 , and b_3 are the forwarding candidates. Assume nodes a_i, b_i , and c_i belong to cluster A, B , and C , respectively.

In timer-based coordination schemes, it requires to rank candidates based on some priority metric. So, assume that candidates are prioritized according

Figure 4.9: Expected coordination time: (a) with varying link delivery probability; (b) with varying fragment size.

to their link delivery probability p_{b_i, c_i} , such that $p_{b_1, c_i} > p_{b_2, c_i} > p_{b_3, c_i}$. Upon receiving the data packet, candidates schedule the time-to-forward the data packet based on its forwarding priority/rank assigned by the source/sender. In this example, b_1 will have the first priority to transmit, followed by b_2 , and then b_3 . A candidate can forward at its turn only if it does not hear the transmission from higher priority candidates. The coordination time at each forwarding node is calculated as

$$t_{b_r} = (r - 1) \times t_\delta + t_x \quad (4.9)$$

where t_δ refers to the time interval required for suppression so as to avoid duplicate transmissions, t_x is the delay jitter, and $r = 1$ being the highest priority node (highest rank). The t_δ is calculated as the amount of time to complete a fragment transmission. A fragment may consist of multiple packets. Therefore, one can compute t_δ as follows.

$$t_\delta = N \times t_{data} \quad (4.10)$$

where N denotes number of packets per fragment and t_{data} denotes per data packet forwarding time. In SOAR [40], it limits the maximum number of packets queued on the wireless card to 3 and t_δ is set as $45ms$ (i.e. $15ms$ for each packet).

Given the above discussed coordination time at each forwarding node, the expected coordination time ECT for flow f_i can be calculated as follows.

$$ECT_{f_i} = \sum_{r=1}^n t_{b_r} p_{b_r}(i) \quad (4.11)$$

$$p_{b_r}(i) = p_{a_i, b_r} \prod_{k=0}^{r-1} (1 - p_{a_i, b_k}) \quad (4.12)$$

where $p_{b_r}(i)$ is the forwarding probability of node b_r with rank r for flow f_i when its higher rank nodes fail to receive the packet from a_i .

In CCDF, using distributed agent-based coordination, it does not require to rank/prioritize the candidates. Instead the agent node needs to get requests from candidates and then selects a node from a pool of potential candidates to be the custody, as discussed in Section 4.5.1. So, the expected coordination time required for agent-based coordination ECT_{agt} is calculated as

$$ECT_{agt_{f_i}} = t_{agt} \quad (4.13)$$

where t_{agt} is the time required for distributed agent-based coordination and it is calculated as follows.

$$t_{agt} = t_{REQ} + t_{REP} + t_x \quad (4.14)$$

$$t_{REQ} = t_{REQ_1} + T_{DEC} \quad (4.15)$$

where t_{REQ} denotes the time taken for the agent to receive decision request from all potential candidates, and t_{REP} denotes the time taken for decision reply. In CCDF, the agent sets the timer from the time it receives the first

decision request t_{REQ} till the threshold value T_{DEC} . Once timeout, it makes decision for all the request it has received within the time frame. Late request for the same packet will be discarded by the agent. A node which has not yet sent out the request but receives the decision from the agent will also drop the outgoing request. Note that with probability $1 - \prod_{k=0}^n (1 - p_{a_i, b_k})$ where n is the number of potential candidates, there will not be any coordination as no node receives the packet successfully.

Figure 4.9 shows analytical results of expected coordination time required for single flow. The results are shown with varying forwarder count, link delivery probability, and fragment size. The per data packet forwarding time t_{data} is set as $15ms$. The transmission time for control packets such as request and reply, $t_{control}$, is set as $2.6ms$ ¹. The link delivery probability p_{a_i, b_k} is chosen from uniform distribution $[u, v]$, where u and v are the minimum and maximum probabilities.

In Fig. 4.9a, we vary u from 0.1 to 0.5. The value of v is set to 1.0. The fragment size N is set to 3. Note that “ect” and “ect-agt” denotes expected coordination time required for timer-based coordination and proposed agent-based coordination. It is observed that no coordination time is incurred in timer-based approach when the forwarder count is one. With increasing forwarder count, timer-based approach incurs longer coordination time than agent-based approach, especially when the average link delivery probability is about 50%. It is noted that timer-based approach can incur even longer coordination time with the increasing fragment size as shown in Fig. 4.9b. Note that the link delivery probability is chosen from $[0.1, 1.0]$ in this case.

¹Note that in simulation, data packet size is set to 512 bytes. Request and reply packets are about 80 bytes, i.e. 6 times smaller than the data packet. Thus, the transmission time of control packets is about 6 times lower than that of data packet.

4.7.2 Multi flows

Consider a second flow $f(a_j, c_j)$ or f_j is injected into the network, where a_j and c_j are the source node and destination node for this flow. Assume b_1, b_2 , and b_3 are the forwarding candidates for this flow as well and flow f_i has higher transmission priority than f_j . Therefore, the expected coordination time for f_j using timer-based coordination method becomes,

$$ECT_{f_j} = ECT_{f_i} + \sum_{r=1}^n t_{b_r}(j) p_{b_r}(j) \quad (4.16)$$

In timer-based coordination, candidate nodes have to wait and listen the medium for transmissions from higher priority nodes, if any. Therefore, for a second flow f_j , the coordination can be initiated only when the coordination of flow f_i is completed, resulting in large delay.

For the agent-based coordination, as it involves communications between candidates and agent using control packets, it only requires t_{agt} for flow f_j and no additional waiting time is incurred.

In fact, in agent-based coordination, there is a holding period at each candidate node before it reports the reception information to the agent. A holding period, denoted by t_h , is the period of time during which the node tries to consolidate all incoming reception information at the moment so as to have less frequent reporting and more efficient coordination. Hence, if candidate nodes receive another flow packets during t_h , only one request for each candidate is required. At the agent, it will process all timeout requests simultaneously and send all decisions in one reply. Thus, control overhead is minimized using cumulative request and reply. Having this holding period is well in line with the existing cooperative forwarding protocols. In cooperative forwarding, sender sends data packet in bulk transfer (i.e., a fragment of packets) and candidate nodes start coordination for further forwarding at

the end of the fragment transmission.

Note that the agent-based coordination time is not affected by the number of flows, but by the number of packets received at the candidate nodes. When a node finds that the request has been sent out, it will insert the newly received packet information in the next request message. This request will be sent when it reaches the maximum request count or when holding period timeout. Therefore, if the request is not immediately sent out, the holding period will be added into the coordination time for that packet.

4.8 Performance Evaluation

4.8.1 Evaluation Methodology

We use network simulator (NS-2.33) to evaluate the network performance. The discussion on verification and validation of the simulation model is provided in 3.8.2. The network topology used in our simulation is illustrated in

Table 4.1: Simulation parameters

Parameters	Single flow	Multi flows
Network area ($l \times l \text{ m}^2$)	80x800	80x800
Number of nodes	15	15
Transmission range	Probabilistic	Probabilistic
Fading model	Nakagami	Nakagami
Traffic rate	100 Packets/sec	10 Packets/sec
Total number of packets	100	3000
Packet size	1000 Bytes	512 Bytes
Traffic type	Constant Bit Rate (CBR)	Constant Bit Rate (CBR)
Number of traffic flows	1	2, 3, 4, 5, 6

Fig. 4.10. The Nakagami propagation model is used to study the effect of realistic fading environment. Therefore, the transmission range between two nodes is probabilistic. The fading model parameters can be found in [60]. Figure 5.11 shows the probe (HELLO) packet reception ratio over distance. With this reception probability, it is noted that the source and destination pair (n_1, n_{12}) which are at the two ends would take 10 hops end-to-end path length under 100% reception probability. In CCDF, with the given topology, three clusters are formed with nodes n_0, n_1, n_2, n_3 , and n_4 in cluster A, n_5, n_6, n_7, n_8 , and n_9 in cluster B and $n_{10}, n_{11}, n_{12}, n_{13}$, and n_{14} in cluster C, respectively. Note that, the node with the highest packet reception ratio to all the other nodes in the cluster is assigned as the head (i.e. agent for coordination). Therefore, nodes n_0, n_5 , and n_{10} become the head of the respective cluster.

We examine 1) the packet delivery ratio (PDR) which shows the amount of packets delivered over the amount of packets generated, 2) the end-to-end delay which indicates the time taken from the packet generated at the source application layer to the packet delivered at the destination application layer, 3) the network throughput, and 4) the MAC layer transmission count which indicates the number of times a packet is transmitted/forwarded by the nodes over the air to be delivered to the destination. The transmission count here also takes into account the number of retransmissions as well. The IEEE

Figure 4.10: Network topology.

802.11 standard is used in all the simulation studies in this work. All traffic flows are injected into the network at the same time to study the multi-flow network performance. We collect simulation results using identical traffic model with at least 10 different traffic injection time. Each simulation is run for 900 seconds. The results are plotted with 95% confidence intervals. The detail simulation parameters are described in Table 5.5.

We first compare the network performance of CCDF with an existing cooperative forwarding protocol and the traditional AODV protocol under single-flow SRSC network. We implement an opportunistic routing (OR) protocol in NS-2 based on the ExOR mechanism [12]. Candidate relays are selected and prioritized based on the expected transmission count to the destination. Nodes maintain in-memory batch map which records the highest priority candidate ID which is known to have a copy of the packet. Nodes embed the in-memory batch map in the data packet header and broadcast the packet. This embedded information indirectly acknowledges other candidates of which packets were successfully received. Timer-based coordination is used for transmission scheduling and duplicate transmission avoidance. As shown in Table 5.5 single flow network model, the traffic source initially generates 100 data packets. Therefore, we set batch size for OR to 100. To the best of our knowledge, no cooperative forwarding study/mechanism on the same candidate relay set being used for multiple cross traffic flows and under MRMC environment has been conducted/proposed.

We then study the network performance of CCDF-SRSC and CCDF-MRMC in multi-flow network. In this case, we compare the performance with the end-to-end routing (AODV) [10] for SRSC network and AODV-MRMC for MRMC network. For this comparison, we implement the same radio-channel assignment for AODV. In AODV-MRMC, each node is equipped

Figure 4.11: Probe packet reception ratio over distance.

with three radios $R1$, $R2$, and $R3$. The radio $R1$ is used for control purposes and therefore, it tunes to control channel. The second radio $R2$ is used for receiving incoming packets and the last radio $R3$ is used for transmitting outgoing packets. All the control messages including route request, route reply, route error are sent over the control channel. The node uses the transmitting radio $R3$ to forward the packet. The channel for $R3$ is switched based-on the next-hop receiver's channel, same as in the CCDF-MRMC. The node can receive the data packet at the same time while transmitting as it uses $R2$ for incoming data packet, as per the CCDF-MRMC. The traffic source and destination pairs in this study are (n_1, n_{12}) , (n_{13}, n_3) , (n_4, n_{14}) , (n_{11}, n_0) , (n_2, n_{10}) , (n_1, n_{14}) .

4.8.2 Single Flow

In this study, source node n_1 sends a single batch of 100 packets to the destination node. We study the performance by varying different hop counts between the source node and the destination node. Specifically, we choose node n_7 , n_{11} , n_{10} , and n_{12} as the destination node. Packet delivery ratio, average end-to-end delay, and throughput performance are shown in Figs. 4.12a,

Figure 4.12: Comparisons with AODV and OR in scenario II network model: (a) Packet delivery ratio; (b) Average end-to-end delay; and (c) Throughput.

4.12b, and 4.12c. It can be observed that CCDF achieves packet delivery ratio almost the same as OR but with much lower end-to-end delay and slightly better throughput performance. By using agent-based coordination, nodes need not require to wait and synchronize with other nodes' transmissions. Nodes are able to schedule the forwarding immediately upon receiving the decision from the agent and forward the data packet whenever the medium is available. It can also be seen that for cases with destination nodes n_{11} , n_{10} , and n_{12} , which are generally located in the same cluster in CCDF, sim-

Figure 4.13: Normalized MAC loads of AODV and CCDF in scenario II network model.

ilar delivery delay is observed regardless of the hop count from the source. However, in OR, we observe increasing end-to-end delay which is due to additional coordination time incurred when more forwarding candidates are involved. For the end-to-end routing AODV, it is noted that the delivery performance and the throughput drop significantly with the increasing number of hops between the source node and the destination node. We observe that this performance drop in end-to-end routing is caused by probabilistic wireless channels, non-cooperative data forwarding, and high traffic bursts.

We then study the normalized control message load of AODV and CCDF. This control overhead study takes into account both network layer control messages and MAC layer address resolution protocol (ARP) and acknowledgement (ACK) messages. This control load performance does not include the hello message overhead as this message is sent periodically throughout the simulation. Note that hello message is used in all protocols (AODV, OR, and CCDF) for neighbour discovery and link metric computation. Note that CCDF with MRMC does not use any additional control packets. For OR, it does not incur any additional control overhead as the batch map is embedded

in the data packet header. As shown in Fig. 4.13, it can be observed that the overhead of CCDF in (n_1, n_7) case is very low. In this case, the source and destination clusters are the neighbours. Thus, it does not need to traverse another cluster, resulting in no agent-based coordination process involved. Therefore, the routing overhead due to agent-based coordination process are not incurred. In (n_1, n_{11}) , (n_1, n_{10}) , and (n_1, n_{12}) cases, although there is an increase in the overhead, the increment is minimal and does not vary with the increasing hop count. In CCDF agent-based coordination, the decision request and decision reply control messages are used within the cluster only. However, in AODV, the routing control messages are sent in the network-wide manner, resulting a significant increase in overhead with increasing hop count between the source node and the destination node.

4.8.3 Multi Flows

For SRSC, the packet delivery ratio, end-to-end delay, throughput, and CDF of PDR over end-to-end delay are shown in Figs. 4.14a, 4.15a, 4.16a, and 4.17a. It is observed that CCDF maintains the PDR at 90% regardless of the number of flows whereas in AODV, the PDR is decreasing with increasing number of flows. AODV usually maintains the PDR at almost 100% delivery under perfect network condition (i.e., network links with no loss rates.). However, in lossy environments, the PDR drops to about 50% under light traffic load (i.e. 2 flows) and the drop rate gets higher when more cross traffic flows are injected into the network. It is well known that cooperative forwarding works well under the lossy nature of wireless environments. Therefore, we can see that over 90% PDR is achieved with minimal delay in two-flow networks. However, with more cross traffic flows, it is observed that the network throughput is decreased. By having the agent-based distributed

Figure 4.14: Packet delivery ratio: (a) SRSC. (b) 3R3C; and (c) 3R4C.

coordination in CCDF, the forwarding node selected by the agent works as a custody to ensure the delivery for the packet. As a result, the delivery is maintained at 90% regardless of the number of cross traffic flows. However, it is noted that due to timeout and retransmission for unsuccessful delivery, the delivery delay is increased and the network throughput is reduced.

The packet delivery ratio, end-to-end delay, and throughput for 3R3C are shown in Figs. 4.14b, 4.15b, and 4.16b and for 3R4C are shown in Figs. 4.14c, 4.15c, and 4.16c. The CDF of PDR over end-to-end delay for both MRMC cases is shown in Fig. 4.17b. The relative PDR improvement over AODV-

Figure 4.15: Average end-to-end delay: (a) SRSC. (b) 3R3C; and (c) 3R4C.

MRMC is shown in Table 4.2. The relative delay and throughput improvement over CCDF-SRSC and AODV-MRMC are shown in Tables 4.3, 4.4, 4.5, and 4.6, respectively.

As shown in Fig. 4.16a, it is observed that, with more cross traffic flows injected into the SRSC network, the network becomes saturated and the throughput starts to decrease. The highest available throughput in the SRSC network is capped at about 60 Kbps. However, with MRMC capability, we observe that the more the offered traffic, the higher the total throughput achieved by the network. The throughput of one channel reuse case is shown

Figure 4.16: Throughput: (a) SRSC. (b) 3R3C; and (c) 3R4C.

in Fig. 4.16b and no channel reuse case is shown in Fig. 4.16c. The results show that throughput of CCDF-MRMC with channel reuse is almost the same as the no channel reuse case. From the studies, we see that CCDF-MRMC exploits the advantage of both multiplexing and spatial diversity. Specifically, CCDF-MRMC reduces about 97% of the delay and achieves about 200% of throughput improvement compared to the SRSC case. This improvement implies that CCDF-MRMC ensures the achievable throughput under highly congested traffic flows and it works well even under the situation where the same set of candidate nodes is being used for multiple flows.

Figure 4.17: CDF of PDR: (a) SRSC and (b) 3R3C and 3R4C.

Table 4.2: Relative packet delivery ratio improvement of CCDF-MRMC over AODV-MRMC counterpart

	2 flows	3 flows	4 flows	5 flows	6 flows
CCDF-3R3C	136.49	141.77	143.68	182.46	225.59
CCDF-3R4C	126.43	140.03	140.72	170.89	183.27

Table 4.3: Relative end-to-end delay improvement of CCDF-MRMC over CCDF-SRSC

	2 flows	3 flows	4 flows	5 flows	6 flows
CCDF-3R3C	35.34	3.84	2.36	1.60	3.62
CCDF-3R4C	32.27	3.67	2.14	1.34	2.04

We then study the MAC layer transmission count under light traffic (2 flows) and congested traffic (6 flows) scenarios. The delivered packet probability distribution based on the transmission count incurred under light traffic scenario is shown in Fig. 4.18 and under congested traffic scenario is shown

Table 4.4: Relative end-to-end delay improvement of CCDF-MRMC over AODV-MRMC counterpart

	2 flows	3 flows	4 flows	5 flows	6 flows
CCDF-3R3C	88.61	70.49	45.99	29.14	48.88
CCDF-3R4C	105.76	82.41	63.22	30.91	40.99

Table 4.5: Relative throughput improvement of CCDF-MRMC over CCDF-SRSC

	2 flows	3 flows	4 flows	5 flows	6 flows
CCDF-3R3C	101.92	159.08	232.08	321.41	311.06
CCDF-3R4C	106.32	166.73	228.59	332.95	332.99

Table 4.6: Relative throughput improvement of CCDF-MRMC over AODV-MRMC counterpart

	2 flows	3 flows	4 flows	5 flows	6 flows
CCDF-3R3C	128.35	127.90	139.14	171.45	200.46
CCDF-3R4C	126.31	138.32	133.71	169.41	169.04

in Fig. 4.19. The average transmission count per delivered packet is shown in Table 4.7. We observe that CCDF makes more transmissions in SRSC compared to MRMC cases. With the given topology, it can be seen that nodes n_5, n_6, n_7, n_8 , and n_9 will be assigned as the candidate nodes. Therefore, with more cross flow traffic injected into the network, more collisions, drops, and retransmissions are being made. As a result, 6-flow network makes about 13 transmissions per delivered packet whereas 2-flow network incurs

about 9 transmissions. This is in contrast to the AODV case. In AODV, 2-flow network makes about 12 transmissions whereas in 6-flow, it reduces to 11 transmissions. This is because of the higher drop rate and unsuccessful delivery in AODV under more contending traffic scenario.

We can see that CCDF-MRMC significantly improves the transmissions required per delivered packet over CCDF-SRSC in both 2-flow and 6-flow network as shown in Table 4.7. Specifically, in 2-flow network with CCDF-MRMC (i.e. CCDF-3R3C and CCDF-3R4C), the transmission count reduces to 67% and 65% of the one incurred by CCDF-SRSC. We can observe more significant improvement in 6-flow network, where it reduces to 47% and 42%. In CCDF, using agent-based coordination, candidate nodes need not require to have multiple schedules for multiple flows. It in fact prevents several potential duplications caused by 1) missing higher priority node transmissions and 2) scheduling inaccuracy due to unavailable medium. In addition, having receiver diversity significantly improves the chance of successful delivery, and it proves that CCDF-MRMC further enhances this capability. It is observed that AODV incurs a large number of transmissions to deliver a packet to the destination in both SRSC and MRMC cases. Specifically, under light traffic load, it incurs about 12, 11, and 10 average transmissions for SRSC, 3R3C, and 3R4C, respectively. Similarly, under congested traffic, it incurs about 11, 10, and 10 average transmissions, respectively. In AODV, although MRMC provides more bandwidth resource, the lossy wireless link environments strongly impact the AODV performance. Although the successful delivery probability is improved in MRMC, it shows that it still incurs frequent losses and retransmissions during the packet delivery.

Figure 4.18: Study on MAC layer transmission count in two-flow network: (a) AODV-SRSC; (b) AODV-3R3C; (c) COR-SRSC; and (d) COR-3R3C.

Table 4.7: Average MAC layer transmission count per delivered packet

Flow count	AODV-1R1C	AODV-3R3C	AODV-3R4C	CCDF-1R1C	CCDF-3R3C	CCDF-3R4C
2 flows	12.077	10.567	10.133	8.905	5.988	5.832
6 flows	10.906	10.369	9.721	12.977	6.104	5.566

Figure 4.19: Study on MAC layer transmission count in six-flow network: (a) AODV-SRSC; (b) AODV-3R3C; (c) COR-SRSC; and (d) COR-3R3C.

4.9 Conclusion

In this chapter, a new cooperative data forwarding solution for single-radio single-channel (SRSC) and multi-radio multi-channel (MRMC) for multi-flow wireless networks is proposed. A new cluster-to-cluster expected transmission count (*C2CETX*) is introduced to compute cluster path for forwarding. Distributed agent-based coordination (DAC) mechanism is introduced so as to support best candidate relay selection and duplicate transmission avoid-

ance in both SRSC and MRMC networks. Analysis on the coordination time required by the DAC mechanism and comparison with the one required by existing timer based coordination method are conducted. The analysis shows that the timer-based coordination method can incur longer coordination time with increasing fragment size and increasing number of traffic flows. The throughput performance shows that the proposed solution works well in both SRSC and MRMC network. It ensures the achievable throughput under highly congested traffic flows and unreliable wireless channels by exploiting the advantage of multiplexing using MRMC and spatial diversity using cooperative forwarding.

Chapter 5

Cooperation Aided Data Delivery Solution for Opportunistic Networks under Group Connectivity Characteristics

5.1 Introduction

It has been discussed in recent works about the presence of social nature such as communities and centralities in opportunistic networks [58, 26, 36, 45]. Some mobility models which consider the social ties between nodes and community/group movement patterns have been proposed [81]. In addition, community/group/cluster/social aware SCF-based routing protocols have also been proposed [58, 26, 20, 82, 48, 36, 19, 18].

In SCF approaches, it assumes that data packets can be transferred re-

liably without any wireless link disruptions during a meeting period with a contact. However, data delivery in pervasive wireless networks suffers with different levels of disruption given user mobility, mobile and stationary scatterers in the environment, and intermittent connectivity [42, 15]. Moreover, information epidemics (also known as forwarding cost) has become a critical design issue for data delivery in opportunistic networks when delivery efficiency is considered [58]. Due to the information dissemination nature of SCF routing, information epidemics control has become a necessary mechanism in SCF routing to prevent from the epidemics outbreak [83].

It has been shown that cooperative or receiver-diversity based forwarding is able to achieve a substantial network performance improvement under the unreliable and lossy wireless nature [42, 12, 40, 14, 41]. To ensure resilient data delivery, it takes into account the adverse wireless link nature, utilizes broadcast wireless medium, and combines several weak/unreliable links to make a reliable communication link.

From these observations, we claim that with the presence of communities and centralities nature in opportunistic network, it assures to exploit receiver diversity-based forwarding in such environment, and therefore be able to maximize a given contact opportunity for the reliable data transfer. We also claim that having this social nature also enables information epidemics control in such network more manageable by using efficient vaccination and recovery scheme.

Motivated by these facts, in this chapter, we propose a cooperation-aided data delivery solution for opportunistic networks under group or community connectivity characteristics. We have discussed in Chapter 4 about the issues in existing cooperative forwarding design under multi-flow networks and proposed cluster-based cooperative forwarding (CCDF) for such networks. So,

the solution in this chapter will exploit the solution proposed in Chapter 4 and then integrate with the cooperation-aided replication algorithm (this will be later discussed in this chapter) for robust and resilient data delivery. The solution is called cluster-based store-carry-cooperative forwarding (C-SCCF). The major contributions of this work are summarized as follows.

- We propose an adaptive cooperative data forwarding algorithm. Nodes proactively monitor the direct/2-hop cluster-based cooperative data forwarding (CCDF) opportunities and deliver data packets. Based on the given contact opportunity, nodes adaptively switch between the CCDF and cooperation-aided replication (CaR)-based forwarding approaches. The CaR approach here is reactive in which nodes replicate the packet copies only when the message has never been forwarded by CCDF.
- We propose a group/community aware information epidemics control algorithm. With the proposed control mechanism, information epidemics in community network is manageable and will never lead to outbreak. We show that the number of infections incurred is only a small fraction of the pure epidemic-based routing. We also show that the number is almost constant regardless of the network density. Due to the flexibility of the proposed scheme, it can also be integrated with any group-aware SCF-based routing.
- We study the network performance of the proposed solution under community/group mobility. The evaluation is conducted with varying network connectivity ranging from highly dense environment to highly sparse environment and under the presence of probabilistic propagation channels. Performance evaluation on successful data delivery rate,

delivery delay, overhead, and information epidemics are studied and discussed.

It is important to note that multi-radio multi-channel can be integrated into this solution as well by using the same radio channel assignment as discussed in Chapter 4 to enhance the system capacity, especially for the highly congested traffic scenarios. As the focus of this chapter is to show the cooperation-aided delivery in opportunistic network with group connectivity under the faded wireless channels, we use normal traffic pattern and therefore use single-radio single-channel.

The rest of the chapter is organized as follows. Section 5.2 reviews related works on group/community aware routing protocols in opportunistic networks. We describe the network model used in this chapter in Section 5.3. The overview of the proposed scheme is provided in Section 5.4. Section 5.5 briefs the link awareness approach. The procedure about the cluster contact discovery is discussed in Section 5.6. The proposed routing scheme with several protocol details are presented in Section 5.7. The proposed community-aware vaccinating algorithm and the integration with the data forwarding module are discussed in Section 5.8. Section 5.9 evaluates the efficacy of the proposed scheme with varying network connectivity ranging from highly dense environment to highly sparse environment. The concluding remarks are provided in Section 5.10.

5.2 Related Work

5.2.1 Cooperative Forwarding

As the proposed design is based on cooperative forwarding algorithm, we briefly describe the algorithm here. Detailed discussion on protocols and algorithms can be found in [42, 15]. The fundamental design components of cooperative forwarding include candidate relay set selection, candidate relay prioritization, broadcasting data packets to the candidate relays, and best relay selection using a coordination scheme [42, 15]. Most cooperative forwarding protocols are proposed for stationary wireless networks [42]. Some recent works study on exploiting cooperative forwarding opportunities in mobile networks [14, 41]. However those approaches require topology information using proactive source routing or position information to locate the destination. Although the efficacy of exploiting cooperative opportunistic forwarding under fading environment was demonstrated, there is a performance drop when the network connectivity becomes intermittent.

5.2.2 Routing in Opportunistic Networks

HYMAD [48] is a hybrid DTN-MANET routing protocol, which uses end-to-end routing to forward messages within cluster/group and uses SCF routing to forward messages between groups. Group epidemic routing (G-ER) [82] is another type of hybrid routing protocol similar to HYMAD. In G-ER, it treats a group of connected nodes as an individual node. A packet will be replicated only once to each group no matter how many nodes are within the group to alleviate the buffer occupancy.

A number of recent works focus on minimizing the cost associated with flooding while retaining high delivery performance. In [20], a quota-based

approach is proposed to control the forwarding cost, where it limits the number of copies allowed to be made for each packet. IEEE 802.11 broadcast mode is used and therefore, neighbour nodes are also able to receive the copy, which will then store and carry if the buffer space is available so as to minimize the delivery delay. In [84], a message is replicated to a node only if the joint delivery probability of two copies is greater than the delivery probability of a single copy. It assumes known knowledge about long-term inter-meeting times between nodes and computes delivery probabilities based on the knowledge. In [21], multiple forwarding paths are selected and data packet copies are forwarded along the predetermined multiple paths. In [19], it highlights the importance of being aware of nodes' vicinity on the delivery delay. The WAIT forwarding strategy is discussed where nodes wait for the end-to-end transmission opportunities and forward data packets using multi-hop transmissions.

Data delivery in social delay-tolerant networking has also been discussed in recent works [58, 26, 18]. These works focus on community formation and aggregation in the network. In [26], a replica of the data packet is forwarded to nodes with higher centrality value than the current node. Forwarding nodes are selected based on the node centrality and community labels. Upon delivering the copy to a member of the destination community, the original carrier discards the message from its buffer to prevent it from further dissemination. The member node will replicate other members with higher local centrality until the message is delivered to the destination. Similarly, in [58], nodes which are located between communities, also known as bridge nodes, are first identified and then data packet copies are forwarded to those bridge nodes. Inter-community multi-copy intra-community single-copy mechanism is used to minimize the forwarding cost. In [18], a transient connected com-

ponents (TCC) aware data forwarding strategy is proposed. A transient connected graph is formed when two nodes meet. The node with higher centrality makes forwarding decisions in the graph. It ensures that the node with the highest centrality and also a set of nodes with better forwarding capability have a copy of data packets other member nodes are holding so as to maximize the delivery performance. Multi-hop communications is used for data forwarding within a TCC.

In summary, existing works on routing in opportunistic networks with group connectivity focus on 1) single copy per group for buffer sharing and management; 2) multi-hop routing within the group to control the overhead due to intra-group communications; 3) replicating the copies to nodes with higher centralities in the community so as to minimize the forwarding cost; and 4) sharing copies to multiple nodes in one transmission (i.e. broadcast) so that multiple copies are stored in many nodes to maximize the delivery. In other words, minimizing forwarding cost and buffer occupancy have been the major focus.

In this chapter, we focus on maximizing the reliability and timeliness of the data delivery in opportunistic networks by 1) exploiting the receiver diversity brought by the community nature of the network; 2) deploying adaptive data forwarding strategies based on the contact opportunity; and 3) eliminating information epidemics problem using efficient vaccination and recovery scheme. We do not focus on community formation and aggregation. Instead, we use group mobility and therefore, the communities/groups are formed/built based on the movement behaviour of the mobility model. Specifically, we focus on cooperative/diversity-based data forwarding over the communities. Unlike the less cooperative point-to-point unicast and multi-hop unicast forwarding approaches, cooperative broadcast forwarding is used

Figure 5.1: Network layer architecture.

to deter link disruptions and maximize a given contact opportunity. In pervasive network, it is unlikely to predetermine the optimal value for quota as the value is very much dependent on network density and connectivity. So, using more quota will approach the forwarding cost of the flooding-based forwarding and less quota will cause longer delivery delay. In contrast to quota-based approaches [20, 31, 24], the proposed protocol does not require to set the quota. We also do not use inter-community multi-copy and intra-community single-copy for message copy control. Instead, we propose an efficient vaccination and recovery scheme which deters a substantial amount of both delivered and not-yet delivered packet copies. Simulation results show that it improves data delivery performance while cutting down 86% to 93% of the forwarding cost compared to flooding-based protocol regardless of the network connectivity.

Figure 5.2: Link delivery probability p .

5.3 Network Model

We consider a network with C clusters/groups/communities. Note that in this chapter, the words “cluster”, “group”, or “community” are used interchangeably. Each cluster comprises member nodes and a head. The nodes are organized by clusters according to their logical relationships. Nodes are uniformly distributed within the scope of a cluster and moves independently within the cluster. The cluster head is positioned at the logical centre of the group whose movement defines the entire cluster motion behaviour.

For the communication model, we consider the unreliable and probabilistic communication nature of the wireless link as shown in Fig. 5.2. The quality of each wireless link is determined according to Nakagami propagation model. We define that a node n_i has a contact with node n_j if the bidirectional link delivery probability between n_i and n_j (i.e., p_{n_i, n_j}) is non-zero. In this case, n_i calls n_j its “contact node”. We also define another term called “contact cluster” or “direct contact cluster” which refers to when at least one of the nodes in cluster C_i has non-zero link delivery probability to cluster C_j , i.e., $p_{n_i, C_j}, n_i \in C_i$. In this case, we call cluster C_i has a direct contact with cluster C_j . We define a third term called “2-hop contact cluster” when cluster C_i has a connectivity to cluster C_k via the cluster C_j . In

Figure 5.3: Forwarding types and order.

this case, $p_{n_i, n_k} = 0, \forall n_i \in C_i$ and $\forall n_k \in C_k$.

5.4 Overview of C-SCCF

In this chapter, we propose a data delivery solution for opportunistic network. We focus on exploiting receiver diversity given the geographical/social relationships between nodes. The aim is to provide guaranteed data delivery with minimum delay, overhead, buffer consumption, and unnecessary transmissions under unreliable contact opportunities. The network layer architecture is shown in Fig. 5.1. The basic idea is 1) deploying the cluster-based cooperative data forwarding (CCDF) protocol proposed in Chapter 4 in the vicinity (maximum 2-hop cluster); 2) adaptively switching to cooperation-aided replication (CaR) in the case where no CCDF opportunities is found and the data packet has never been forwarded by CCDF; and then 3) incorporating the community/social aware information epidemics control which deters packet infections in a community and vaccinate other communities from being infected.

In C-SCCF data forwarding, three main forwarding types are defined. They are direct CCDF, two-hop CCDF, and CaR. For the purpose of readability, we call direct CCDF “CCDF-1h” (1h denotes one hop) and two-hop CCDF “CCDF-2h”. Under each CCDF-1h and CCDF-2h, there are three sub forwarding types. All forwarding types and forwarding order are shown in Fig. 5.3. A node inserts the respective data forwarding type in the data packet header before it forwards the data packet. This information is used to inform the nodes receiving or overhearing the packet so that they can process accordingly. Section 5.7 details the data forwarding algorithm.

In C-SCCF vaccinating, community-aware information epidemics control is introduced. Nodes maintain a packet state vector inside which the progress of each data packet is recorded. There are four possible packet states: *No_copy*, *Have_copy*, *Local_fwd*, and *Delivered*. This state vector is updated whenever there is an update in the community. The cluster head works as an agent to synchronize the states in its community. This synchronized vector is shared when a node interacts with another community. By synchronizing the vector in the community and then sharing the synchronized vector, it not only prevents the infection in its community but also vaccinates other communities from being infected effectively. Section 5.8 discusses the detailed mechanism.

5.5 Live-update Link Quality

Similar to the SCCF proposed in Chapter 3, nodes conduct link quality measurements based on the periodic hello message transmissions and live-update the qualities to monitor the unreliable and probabilistic nature of wireless channel. Having live-update link quality enables to have link awareness so

IP header		
contact node ID n_j	p with n_j	p_{rev} with n_j
...	...	...
contact cluster ID C_j	$C2CETX$ with C_j	gateway node ID to C_j
...	...	...
2hop contact cluster ID C_k	$C2CETX$ with C_k	next hop cluster ID C_j
...	...	...

Figure 5.4: Hello packet header format.

IP header		
Destination cluster ID		Forwarding type
Data Relay node ID	Next-hop cluster ID	ACK relay node ID
<i>fwonlyifdirect</i>		<i>donotreplicate</i>
...		

Figure 5.5: Data packet header format.

as to timely exploit the better forwarding opportunities. For the detail on live-update link quality and contact information table maintenance, please refer to the Section 3.4 in Chapter 3.

From the contact information table constructed using the live-update link quality, a node n_i in cluster C_i can compute its link delivery probability to cluster C_j as follows.

$$p_{n_i, C_j} = 1 - \prod_{j=1}^{|Z_{C_j}|} (1 - p_{n_i, n_j}), \forall n_i \in C_i, \forall n_j \in C_j \quad (5.1)$$

where $|Z_{C_i}|$ and $|Z_{C_j}|$ refer to number of nodes in cluster C_i and C_j , respectively.

5.6 Contact Cluster Discovery

A node includes its contact information in the periodic hello message so as to share its list of direct contact nodes. The hello message format is shown

in Fig. 5.4. Upon receiving the contact information of the members, the cluster head discovers all of its direct cluster contacts. We here assume that the cluster ID that a node belongs to is known in advance. Cluster head h_i of cluster C_i computes the link delivery probabilities of its member nodes to the other clusters using (5.1). Suppose there are three member nodes namely n_1, n_2 , and n_3 and the head h_i in cluster C_i . So, h_i computes the p_{n_1, C_j} , p_{n_2, C_j} , p_{n_3, C_j} , and p_{n_4, C_j} where $C_j \in C$ and $n_4 = h_i$.

Upon completion of the link delivery probabilities computation to all other clusters, h_i computes the cluster-to-cluster expected transmission count (*C2CETX*) to each cluster as follows. The detail of the *C2CETX* also can be found in Section 4.4 in Chapter 4.

$$C2CETX_{C_i, C_j} = \sum_{e=1}^{2^{|Z_{C_i}|-1}} p(e) \min \left\{ x_{n_1}(e)ETX_{n_1, C_j}, \right. \\ \left. x_{n_2}(e)ETX_{n_2, C_j}, x_{n_3}(e)ETX_{n_3, C_j}, x_{n_4}(e)ETX_{n_4, C_j} \right\} \quad (5.2)$$

where ETX_{n_i, C_j} refer to the expected transmission count from node n_i to cluster C_j and $p(e)$ refers to probability of each event. ETX_{n_i, C_j} and $p(e)$ are computed as follows.

$$ETX_{n_i, C_j} = \frac{1}{p_{n_i, C_j}} \quad (5.3)$$

$$p(e) = \frac{1}{2^{|Z_{C_i}|-1}}, e \in |\chi| \quad (5.4)$$

The event e here refers to nodes in cluster C_i receiving the data packet from the source/sender successfully and χ denotes the set of all possible events, i.e. $|\chi| = 2^{|Z_{C_i}|-1}$. $x_{n_i}(e)$ is the binary variable which is defined as

$$x_{n_i}(e) = \begin{cases} 1, & \text{if } n_i \text{ is selected in event } e \\ 0, & \text{otherwise} \end{cases} \quad (5.5)$$

We assume that the probability of each event is evenly distributed and the probability that all nodes fail to receive the data packet from the sender is zero. Therefore, the calculation in (5.2) serves as a rough, conservative estimate of the cluster-to-cluster transmission count. A closer estimation of *C2CETX* can be computed if the source or sender node is known in advance. A more detailed discussion is stated in Section 4.4 in Chapter 4.

Cluster head periodically updates its members the cluster contact information which includes cluster contact ID and the corresponding *C2CETX* value. This information is included in its hello message. From this periodic update, member nodes obtain the entire cluster contact information of its cluster.

5.6.1 Dynamic Gateway Selection

Cluster head dynamically selects a gateway node to each contact cluster based on the highest link delivery probability to the cluster. From the link delivery probability computed as above, cluster head h_i chooses for example n_1 as gateway to cluster C_j if it satisfies $p_{n_1, C_j} > p_{n_2, C_j} > p_{n_3, C_j}$. Note that different nodes can be assigned as gateway to different contact clusters or a node can be assigned as gateway to more than one contact clusters. Cluster head will conduct reselection if there is a change in link delivery probability. This information is also added in the hello message. Therefore, member nodes is aware of gateway nodes to each contact cluster.

5.6.2 2-hop Cluster Contact Discovery

In the periodic hello message, nodes include the contact cluster information in addition to the contact node information. The contact cluster information is obtained as discussed above. Suppose node n_i in cluster C_i receives a

hello message from node n_j in cluster C_j . Assume n_j includes cluster C_k information in its contact cluster information list and n_i does not have the direct contact with cluster C_k , i.e. $p_{n_i, n_k} = 0, \forall n_k \in C_k$. Upon receiving the n_j 's hello message, n_i records C_k as its 2-hop contact cluster and C_j as its next-hop cluster in its contact information table.

In addition to contact node information and contact cluster information, nodes also add 2-hop contact cluster information in its hello message. Upon receiving the member nodes' 2-hop contact cluster list, cluster head make a list of all reachable 2-hop contact clusters and finds the best next-hop cluster using (5.6). Therefore, the cluster C_j which gives the lowest $C2CETX$ from C_i to C_k will become the next-hop cluster.

$$C2CETX_{C_i, C_k} = C2CETX_{C_i, C_j} + C2CETX_{C_j, C_k} \quad (5.6)$$

Note that cluster head also shares the complete list of 2-hop contact cluster contact with the members. Therefore, upon receiving the periodic updates from head, member nodes obtain the complete list of 1-hop and 2-hop contact clusters. The illustration of contact node information table, 1-hop contact cluster information table, and 2-hop contact cluster information table are shown in Tables 5.1, 5.2, and 5.3, respectively.

5.7 Routing

In this section, we discuss the details of C-SCCF routing module. We first describe CCDF algorithm which comprises two components: CCDF-1h and CCDF-2h. We then elaborate the details of each component. We then present the second data forwarding algorithm called CaR. We also discuss the additional components of the protocol that complement the data forwarding

Table 5.1: Contact Node Information Table

contact ID	cluster ID	p_{fwd}	p_{rev}	p	one-hop contact list	cluster contact list	2hop cluster contact list
n_2	C_1	0.8	0.7	0.56	$(n_1, p_{n_2, n_1}), \dots$	$(C_2, C2CETX_{C_1, C_2}),$...	$(C_3, ETX_{n_2, C_3}),$...
n_5	C_2	0.9	0.85	0.77	$(n_1, p_{n_5, n_1}), \dots$	$(C_1, C2CETX_{C_2, C_1}),$ $(C_4, C2CETX_{C_2, C_4}),$...	$(C_6, ETX_{n_5, C_6}),$...
n_{11}	C_3	0.9	0.8	0.72	$(n_1, p_{n_{11}, n_1}),$...	$(C_1, C2CETX_{C_3, C_1}),$ $(C_5, C2CETX_{C_3, C_5}),$...	$(C_7, ETX_{n_{11}, C_7}),$...
...	...	...	...	...	...	...	...

Table 5.2: 1-hop Contact Cluster Information Table

cluster ID	$C2CETX$	gateway node
C_2	$C2CETX_{C_1, C_2}$	n_1
C_3	$C2CETX_{C_1, C_3}$	n_4
...	...	...

Table 5.3: 2-hop Contact Cluster Information Table

cluster ID	$C2CETX$	next-hop cluster ID
C_4	$C2CETX_{C_1, C_4}$	C_2
C_5	$C2CETX_{C_1, C_5}$	C_3
...	...	...

which include distributed-agent based coordination, packet buffers, network layer acknowledgement, augmentation for data resiliency for CCDF-1h, and handling link loss. The format of the data packet header is shown in Fig. 5.5. A summary of node types and the corresponding definition is given in Fig. 5.4. All forwarding types and forwarding order is shown in Fig. 5.3.

Table 5.4: Node Types and Definition

Node types	Definition
Head/Agent	A node that handles coordination, forwarder selection, and decision making in a community.
Bridge	A node in a community which has direct connectivity to neighbour community(ies).
Gateway	A bridge node with the strongest link quality (i.e. strongest social ties) to a neighbour community.
Candidate	A node receiving the broadcast data packet from a neighbour community.
Forwarder	A candidate node selected by Head/Agent as a custody to take the responsibility for a particular data packet delivery.
ACK-sender	Before a node in community A sends a data packet to community B, it first identifies a node in community B which has strongest connectivity with community A. The selected node ID is inserted in the data packet header as the ACK-sender. Upon receiving the data packet, the ACK-sender will send the ACK for the data packet.
Member-relay	When a node in community A wants to send a data packet to community B, but not directly reachable to B and also not to any bridge to reach B, it will try to send the data packet to B through a Gateway node. However, if it finds that it is also not directly reachable to the Gateway node, it will then select a member in its community to work as a relay and send the data packet to the Gateway node through the relay. This relay node is called as Member-relay.

5.7.1 Cluster-based Cooperative Data Forwarding (CCDF)

Upon receiving a hello message from a node (either member or non-member), the node n_i updates all of its information tables accordingly. It will then make forwarding decisions to forward the packets in its buffer. The node will first try to forward the data packets with CCDF-1h. Packets for which it does not have the CCDF-1h forwarding opportunity, it will try to forward with

CCDF-2h. If there is no CCDF-1h and CCDF-2h opportunities, it will then use CaR-based forwarding. The following discusses the details of CCDF-1h and CCDF-2h algorithms.

Direct CCDF (CCDF-1h)

Figure 5.6: CCDF-1h forwarding approach: (a) *Direct* forwarding; (b) *via-Bridge-DstCluster* forwarding; and (c) *via-Gateway-DstCluster* forwarding. (where a_0 is the bridge node, a_2 is the gateway node, a_4 is the member-relay, and b_1 is the ACK-sender.)

In CCDF-1h, we define three forwarding types. They are *Direct*, *via-Bridge-DstCluster*, and *via-Gateway-DstCluster*. Suppose n_d is the destination node and n_d belongs to cluster C_d . We call C_d the destination cluster. The following discusses each of these forwarding approaches.

- *Direct*: the node n_i , where $n_i \in C_i$, first checks if it has direct contact with cluster C_d , i.e., p_{n_i, C_d} is non-zero, in its contact information table. If there is, it will broadcast the data packets to cluster C_d . This type of forwarding is called *Direct* broadcasting. The forwarding type in the

packet header is set to *Direct*. The illustration of this forwarding is shown in Fig. 5.6(a).

- *via-Bridge-DstCluster*: If the p_{n_i, C_d} is zero, i.e, there is no direct connectivity from n_i to C_d , n_i then looks for a bridge node m_i which has the highest non-zero p_{m_i, C_d} value in its contact information table. If n_i finds m_i , it will first send the data packet to m_i and then m_i will then broadcast to C_d . The transmission from n_i to m_i uses unicast and the forwarding type in the packet header is set to *via-Bridge-DstCluster*. The transmission from m_i to C_d uses broadcast and the forwarding type is set to *Direct*. The illustration of this forwarding is shown in Fig. 5.6(b).
- *via-Gateway-DstCluster*: If there is no above two forwarding opportunities, node n_i then check if C_d is listed as a contact cluster in its 1-hop cluster contact information table. If it is, it will first get the gateway node g_i , where $g_i \in C_i$, to cluster C_d . It will then get a member-relay node r_i , where $r_i \in C_i$, with highest p_{r_i, g_i} value. Once g_i and r_i are identified, n_i first forwards (unicast) the data packet to r_i . Node r_i then further forwards (unicast) the data packet to g_i . Upon receiving the data packet, the gateway node g_i broadcasts it to C_d . The forwarding type is set to *via-Gateway-DstCluster* in the first two forwarding and *Direct* in the gateway g_i transmissions. The illustration of this forwarding is shown in Fig. 5.6(c).

Two-hop CCDF (CCDF-2h)

If there is no CCDF-1h forwarding opportunities, node n_i will then try to forward the data packets using CCDF-2h approach. In CCDF-2h, we define

Figure 5.7: CCDF-2h forwarding approach: (a) *via-RelayCluster* forwarding; (b) *via-Bridge-RelayCluster* forwarding; and (c) *via-Gateway-RelayCluster* forwarding (where a_0 is the bridge node, a_2 is the gateway node, a_4 is the member-relay, and b_1 is the ACK-sender.). The illustration for data forwarding from relay cluster to the destination cluster is shown in Fig. 5.8.

Figure 5.8: Forwarding from relay cluster to destination cluster in CCDF-2h: (a) *Direct* forwarding; (b) *via-Bridge-DstCluster* forwarding; and (c) *via-Gateway-DstCluster* forwarding.

three forwarding types. They are *via-RelayCluster*, *via-Bridge-RelayCluster*, and *via-Gateway-Relaycluster*. The illustration is shown in Fig. 5.7. The following discusses each of these forwarding approaches.

- *via-RelayCluster*: Node n_i first checks if it has connectivity to a cluster which has direct reachability to the destination cluster C_d in its contact information table. If there is, n_i uses the cluster as the relay cluster for data forwarding to C_d . We denote C_r as the relay cluster. n_i chooses the best relay cluster C_r using (5.7).

$$ETX_{n_i, C_d} = ETX_{n_i, C_r} + C2CETX_{C_r, C_d} \quad (5.7)$$

Therefore, the cluster C_r which gives the lowest ETX from n_i to C_d becomes the relay cluster. Once the relay cluster is identified, node n_i will broadcast the data packets to C_r . Upon receiving the data packet, nodes in cluster C_r will perform distributed agent-based coordination

and candidate relay selection. The candidate that is selected will then forward the data packet to the destination cluster C_d . The distributed agent-based coordination and candidate selection is discussed in Section 5.7.3.

- *via-Bridge-RelayCluster*: If n_i does not have the *via-RelayCluster* opportunity, it will then look for the *via-Bridge-RelayCluster* opportunity. In this case, node n_i finds a bridge node m_i , where $m_i \in C_i$, which has reachability to C_d in two cluster hops in its contact information table. n_i chooses the best m_i using (5.8).

$$ETX_{n_i, C_d} = ETX_{n_i, m_i} + ETX_{m_i, C_d} \quad (5.8)$$

Upon identifying the bridge node m_i , n_i sends the data packets to m_i in unicast. The forwarding type is set to *via-Bridge-RelayCluster*. Upon receiving the data packet, node m_i gets the relay cluster C_r as per its contact information table and then broadcasts the data packet to C_r . Nodes in cluster C_r will then perform distributed agent-based coordination and candidate selection to further forward the data packet to C_d .

- *via-Gateway-Relaycluster*: If both opportunities discussed above are not available, node n_i will then check the 2-hop contact cluster information table and finds if C_d is listed as the two-hop contact cluster. If it is, n_i then gets the next-hop cluster C_r from the table. It then finds the gateway g_i to C_r from its one-hop contact cluster information table. Once the gateway node g_i is identified, n_i will then look for a member-relay node r_i to reach to g_i . A node with the highest p_{r_i, g_i} is selected as member-relay. Once r_i is identified, n_i will forward the

Figure 5.9: Cooperation-aided replication: (a) Packet state vector request and reply; (b) Data forwarding.

data packet to r_i , which will then forward it to g_i . The forwarding type is set to *via-Gateway-Relaycluster*. Upon receiving the data packet, g_i will then broadcast the data packet to the next-hop cluster C_r . The distributed agent-based coordination and candidate selection will then be conducted in cluster C_r for further data forwarding to C_d .

5.7.2 Cooperation-aided Replication

Upon receiving a hello message from a non-member node, i.e., n_i in C_i receiving a hello from n_j in C_j , and if both CCDF-1h and CCDF-2h above discussed opportunities are not available, n_i then replicates the packets in its buffer to cluster C_j as follows. The illustration of this forwarding is shown in Fig. 5.9.

Node n_i first sends packet state vector request to a node in C_j with which it has the highest link delivery probability than others (i.e., $\max_{n_j \in C_j} p_{n_i, n_j}$). Packet state vector is an information table a node maintains to track the

progress of data packets. There are four possible packet states. They are *No_copy*, *Have_copy*, *Local_fwd*, *Delivered*. Different from the state vector in the previous work 3, the vector here is a community-aware version which contains the consolidated information of the community. Details of the packet state vector and state transitions are discussed in Section 5.8.

Suppose n_j in C_j is the node to which n_i sends the request. Upon receiving n_i 's request, node n_j broadcasts its packet state vector. Due to broadcasting, node n_i and all the other nodes in contact with node n_j are able to receive/overhear and update their packet states as per the state transition rules shown in Fig. 5.10. For example, if the overheard packet state is *Local_fwd*, a node will update its packet state to *Local_fwd* if its current state is *No_copy* / *Have_copy*. Similarly, if the overheard packet state is *Delivered*, a node will update its packet status to *Delivered* if its current state is *No_copy* / *Have_copy* / *Local_fwd*. The detail of the information epidemics control algorithm is discussed in Section 5.8.

After updating the packet state vector based on the received vector from n_j , node n_i gets the copies of packets in the buffer as in (5.9) and broadcast to cluster C_j (i.e. the next-hop cluster parameter in the data packet header is set to C_j).

$$packets_to_be_replicated = X_{n_i} + \overline{Y_{C_i}}' + \overline{Z_{C_i}}' + \overline{X_{C_j}} \quad (5.9)$$

$$Y_{C_i}' = Y_{C_i} \cup Y_{C_j} \quad (5.10)$$

$$Z_{C_i}' = Z_{C_i} \cup Z_{C_j} \quad (5.11)$$

where X_{n_i} refers to a set of IDs of packets that node n_i has in its primary packet buffer, Y_{C_i} and Y_{C_j} refer to a set of IDs of packets with states *Local_fwd* in clusters C_i and C_j , Z_{C_i} and Z_{C_j} refer to a set of IDs of packets with states

Delivered in clusters C_i and C_j , and X_{C_j} refers to a set of IDs of packets with states *Have_copy* in cluster C_j .

From (5.9), it is noted that packets whose state is known as *Local_fwd* and *Delivered* in both clusters C_i and C_j are not replicated. In addition, packets whose state is known as *Have_copy* in cluster C_j are also not replicated. Only packets in the primary buffer of node n_i and are not included in those sets get replicated. The forwarding type in the packet header is set to *CaR*.

Note that n_i sends the packet copies in broadcast mode and hence, all the nodes in contact with n_i are able to receive the copies. Therefore, all nodes in C_j which has non-zero link delivery probability with n_i (i.e. $p_{n_i, n_j}, \forall n_j \in C_j$) will receive the packet with probability p_{n_i, n_j} . Upon receiving the data packet from n_i , nodes in C_j will then perform distributed agent-based coordination for the packet delivery.

5.7.3 Distributed Agent-based Coordination

Upon receiving the packet and if its cluster ID is assigned as the next-hop cluster, nodes in the receiving cluster will start perform distributed agent-based coordination. We use cluster head as an agent for making decision on selecting a candidate out of all receivers to become a custody for the data delivery. Upon receiving the data packet, nodes update head about the reception information. Note that the reception information is in the form of $(senderID, packetID)$. We assume a packet has its unique ID. As we consider multiple traffic flows coming from any directions, the sender ID associated to each flow is attached in the reception information. Each node holds the reception for a period of time to append all the possible incoming traffic and informs head once timeout.

Upon receiving the first reception information, the agent sets the timer

from the time it receives the first message till the threshold value T_{DEC} . T_{DEC} is set to a period of time which is sufficient to receive the reception information from the potential candidates. The agent starts make the candidate selection once timeout or the number of reported candidates exceed N_{DEC} threshold value. Suppose cluster C_j is the receiving cluster, T_{DEC} and N_{DEC} are computed as follows.

$$T_{DEC} = \lfloor \gamma \times |C_j| \rfloor \times t_{control} \quad (5.12)$$

$$N_{DEC} = \lfloor \gamma \times |C_j| \rfloor \quad (5.13)$$

where $t_{control}$ denotes the transmission time required to receive a control message from a candidate node to the agent and γ represents a fraction with value $0 < \gamma \leq 1$. We set γ to $\frac{2}{3}$ in the simulation as it ensures diversity with reasonable timeout delay. Suppose R be the set of candidate nodes. Note that $R \subseteq C_j$. The agent makes the decision based on the following conditions: 1) C_d is the direct contact cluster of C_j ; 2) C_d is the 2-hop contact cluster of C_j ; and 3) C_d is not reachable in two-hop cluster. After the decision has been made, the agent broadcasts its decision reply. The selected candidate becomes the custody for the data delivery. The non-selected candidates will take note of the decision and then discard the packet. We use cumulative decision reply instead of reply for individual packet to reduce the control overhead and collisions to data packet transmissions. The details on the coordination time incurred in distributed agent-based coordination is discussed in Chapter 4.

Case 1 - C_d is the direct contact cluster of C_j

In this case, the agent selects a candidate with highest delivery probability to C_d , i.e. $\max_{n_j \in R} p_{n_j, C_d}$. If it finds that all the candidates are not directly

reachable to C_d , i.e., $p_{n_j, C_d} = 0, \forall n_j \in R$, then the agent first identifies the gateway node g_j to C_d . It then selects a candidate with highest link delivery probability to g_j , i.e. $\max_{n_j \in R} p_{n_j, g_j}$. If none of the candidates is in one-hop contact with gateway node, it is likely that very few nodes receive the data packet. In this case, the agent randomly selects a node among the candidates. The selected candidate becomes the forwarding candidate. Upon receiving the decision from the agent, it will then try to forward the data packet to the destination cluster C_d using CCDF-1h.

Case 2 - C_d is the 2-hop contact cluster of C_j

In this case, the agent selects a candidate with lowest ETX value to C_d , i.e. $\min_{n_j \in R} ETX_{n_j, C_d}$. If it finds that all the candidates are not reachable to C_d in two cluster hops, then the agent first identifies a bridge node m_j which has the lowest ETX value to C_d , i.e. $\min_{m_j \in C_j, m_j \notin R} ETX_{m_j, C_d}$. Once the node m_j is selected, the agent then selects a candidate with highest link delivery probability to m_j , i.e. $\max_{n_j \in R} p_{n_j, m_j}$. If none of the candidate are in one-hop contact with m_j , it is likely that very few nodes receive the data packet. In this case, the agent randomly selects a node among the candidates. The selected candidate becomes the forwarding candidate. Upon receiving the decision from the agent, it will then try to forward the data packet to the destination cluster C_d using CCDF-2h.

Case 3 - C_d is not reachable in two-hop cluster

In this case, the agent selects a candidate holding the lowest number of packets in its buffer. The selected node becomes the custody for the packet. By choosing a candidate with more buffer space available, the buffer sharing in the community is achieved. Upon receiving the decision from the agent, the

selected candidate will then check if any CCDF-1h or CCDF-2h opportunity becomes available for the delivery. Otherwise, it will store the packet in the buffer and wait for future contact opportunities.

Case 4 - Destination cluster C_d receiving the data packet

In this case, no agent coordination is required. Upon receiving the data packet, the receiving nodes n_d , where $n_d \in R$ and $n_d \in C_d$, set the delay timer t_x before they broadcast the packet. The length of the delay is set based on the hop count from n_d to the destination node dst_d . If n_d is in one-hop contact with dst_d , the value t_x is chosen from the uniform distribution $[a, b)$. If it is in two-hop contact, t_x will be chosen from uniform distribution $[c, d)$. Note that $a < b < c < d$. So that nodes with one-hop reachability will be likely to broadcast earlier than the ones with two-hop. The forwarding type is set to *to-DstMember*. If the destination node dst_d is in the list of receiving nodes, i.e. $dst_d \in R$, it will then send an ACK about the destination node being received the packet. The ACK is sent in broadcast mode. The detail on the acknowledgement is discussed in Section 5.7.5. Upon receiving the ACK from dst_d , other receiving nodes n_d will discard the packet.

5.7.4 Packet Buffer

In the proposed solution, a node maintains two types of buffer to store data packets as follows.

Primary packet buffer (buffer) - This buffer is used by source or custody nodes. It is used to store data packet when a node finds that the destination cluster of that packet is not in both one-hop and two-hop contact cluster list of its cluster.

Temporary packet buffer (temporary buffer) - It is used to store data

packet while waiting for the head (agent) decision about the selected forwarding candidate. For source or custody nodes, it is also used to keep copies of packets which are being forwarded to next-hop cluster but waiting for acknowledgement to ensure the successful reception. After the node overhears the acknowledgement, the state of the corresponding packet copy will be updated, which will be discussed in detail in Section 5.8. If the acknowledgement is from the destination cluster, the corresponding packet copy will be removed from the temporary buffer. If no acknowledgement from the destination cluster is received, the node will retrieve the packet from the temporary buffer and store it in the primary buffer after acknowledgement timeout.

5.7.5 Network Layer Acknowledgement

For IEEE 802.11 broadcast transmissions, we use network layer acknowledgement scheme as IEEE 802.11 broadcast mode does not provide any acknowledgement for the broadcast packet. In CCDF-1h and CCDF-2h inter-cluster broadcast transmissions, prior to forwarding the data packet to the next-hop cluster C_j , the source/forwarding/gateway node n_i in C_i , dynamically chooses a node, n_j , in the receiving cluster, C_j , as ACK-sender. It selects a node with which it has the highest link delivery probability $\max_{n_j \in C_j} p_{n_i, n_j}$. It then sets ACK-sender node ID in the data packet header as n_j and forwards the packet. Upon receiving the data packet from another cluster, node n_j sends an ACK if it is assigned as ACK-sender in the packet header. Note that for transmissions with *CaR* forwarding type, the acknowledgement is not necessary as the ACK will not help make any packet progress update. The detail of the packet progress update will be discussed in Section 5.8.

We use cumulative acknowledgement instead of acknowledgement for in-

dividual packet to reduce the control packet overhead and collisions to data packet transmissions. This acknowledgement is transmitted in broadcast mode. Therefore, it informs not only the sender about successful reception of the data packet but also other nodes who are within the communication range. The gateway node in cluster C_i re-broadcasts the ACK it receives from the next-hop cluster C_j to make sure that the source/custody node receives the acknowledgement from C_j .

5.7.6 Augmentation for Data Resiliency for Direct CCDF

As data packets are sent over long, weak wireless links, CCDF-1h adds an augmentation mechanism to increase data resiliency. When a node finds that if it has a non-zero link delivery probability to the destination cluster C_d , it will forward the data packet directly without traversing a relay cluster. However, sending over the long weak links likely incurs transmission failure with higher probability. Therefore, to strengthen this data delivery, an augmentation scheme is introduced. Suppose 1) node n_i in cluster C_i is the sender broadcasting a data packet with forwarding type *Direct* to the destination cluster C_d and 2) node n_j in C_d is listed as the ACK-sender in the packet header.

When a node n_k in a cluster (not the destination cluster) overhears *Direct* transmissions, it will support the data delivery if the following conditions are satisfied. 1) Node n_k has direct contact with both n_i and n_j , i.e. both p_{n_k, n_i} and p_{n_k, n_j} are non-zero.; 2) $p_{n_i, n_j} < 0.55$, i.e. most likely n_i will require at least two transmissions to successfully deliver a data packet to n_j .; and 3) $p_{n_k, n_j} - p_{n_i, n_j} \geq \alpha$. If the conditions are met, the node n_k will self-evaluate other nodes who meet the above conditions as well and calculate its rank among all potential nodes. Note that the information contained in the hello

message of n_i and n_j is sufficient to self-evaluate other potential nodes. The node n_k will then set a timer based on the length of delay as in (5.14) and will broadcast the packet when the timer expires.

$$Delay(n_k) = r \times t_x + r \times t_{ack} \quad (5.14)$$

where r is the rank of n_k , i.e. $r = 1$ being the highest rank node. t_x is the jitter which follows uniform distribution. t_{ack} is the time taken to receive an acknowledgement from n_j (90 ms in the simulation). The value of α is set to 0.1, i.e. a node n_k can join the augmentation process only when it has at least 10% better link delivery probability than n_i . The node n_k will discard its ongoing transmission if it overhears the transmission from other n_k nodes or the acknowledgement from node n_j . The forwarding type here is also set to *Direct*. By having the backup transmissions, the success probability of data delivery from n_i to the destination cluster C_d is improved.

5.7.7 Handling Link Loss

Due to the dynamic and random nature of wireless link, the communication link can be lost while packets are being forwarded. To stop sending out packet if the link is lost, the link layer maintains a bi-directional link delivery information table and passes the packets to interface queue only if the link delivery probability is non-zero. In case of link loss, the link layer will discard the packet.

When a node which is assigned as next-hop node or next-hop cluster in CCDF-1h and CCDF-2h forwarding approaches finds the connection lost to its next-hop node or cluster, it sends (broadcasts) a forwarding error message. Upon receiving the error message, a node updates its contact information table. It will then inspect the interface queue and discard the packet if the

Figure 5.10: Four packet states and state transitions.

erroneous forwarder is being assigned in the packet header.

5.8 Community-aware Information Epidemics Control

In this section, we present community/cluster-aware information epidemics control algorithm which provides early control signal distribution, and therefore leverage the information epidemics control in the network. We adopt the idea of information epidemics control mechanism proposed for the entity mobility model in Chapter 3. Different from the previous work, the algorithm here is aimed for networks with group connectivity characteristics. The basic idea is to prevent infections in its own community and vaccinate other communities from being infected or to be recovered.

The vaccinating module monitors the broadcast-based CCDF-1h and CCDF-2h transmissions and acknowledgements over the air and translates the information overheard into packet states. We call the database maintaining the state for each packet 'packet state vector'. We define four packet states as *No_copy*, *Have_copy*, *Local_fwd*, and *Delivered*. The default state for each packet is set to *No_copy*. The four packet states, their definitions, and state transitions are discussed as follows. Figure 5.10 shows the packet states and the state transitions.

Have_copy - A packet is set to *Have_copy* status when a node 1) receives a packet from the upper layer; 2) becomes the custody of the packet; and 3) if a node in the same cluster is the custody of the packet.

If a packet is in this status, a node will not accept the packet copy. That is to say, the node and its cluster is immune to this packet infection. We call this information epidemics control *C-SCCF self-healing*.

Local_fwd - A packet status is set to *Local_fwd* when a node or the community is aware that the data packet has been/is being forwarded by *Direct*. This happens when a node overhears the transmissions with forwarding type *Direct*.

Upon learning the *Local_fwd* status, a node and its cluster will not accept the packet copy and will be able to vaccinate others in order not to get infected. When a node is transmitting a packet using *Direct* forwarding type, it implies that the sender is in contact with the destination cluster. That is to say, by exploiting the receiver diversity, the packet will be delivered to the destination cluster most likely. Therefore, if a node and the community has this packet status, it will no longer use CCDF-2h and CaR forwarding approaches for this packet delivery as it will incur unnecessary bandwidth consumption and collision to other data packet transmissions. However, to ensure the reliability, the source or the custody will still use CCDF-1h approach until it gets information that the destination cluster has received the packet. To achieve this purpose, we enable a flag called *fwdonlyifdirect* in the data packet header. So, nodes overhearing this packet will update the status and handle the packet accordingly. This type of information epidemics control is called vaccinating using partially-immunized vaccine.

In order to further leverage the information epidemics control, we also add another flag called *donotreplicate* in the packet header. This flag is enabled

when a node or the community is aware that the data packet has been/is being forwarded by CCDF-2h transmissions. The status of this packet is not upgraded to *Local_fwd* as the chance of the packet being delivered to the destination cluster is lower than the CCDF-1h. However, as we can assure this packet delivery to the destination cluster with some probability, nodes will no longer use CaR approach for this packet delivery to prevent the unnecessary transmission cost. Even so, to ensure the reliability, the source or the custody will still use both CCDF-1h and CCDF-2h approaches for this packet delivery.

Delivered - A packet status is set to *Delivered* when a node or the community is aware that the data packet has been successfully delivered to the destination cluster. This happens 1) when the destination cluster receives the packet; 2) when a node receives/overhears acknowledgement from the destination cluster or the destination node; and 3) when a node overhears the transmissions with *to-DstMember* forwarding type.

Upon learning this status, a node and the community become fully-immunized. It will no longer accept any incoming copy. It will also discard if there is a corresponding packet copy in the community. It will also be able to vaccinate other communities so that others can discard the packet in their communities as well, if any. This type of information epidemics control is called vaccinating using fully-immunized vaccine.

No_copy - A packet will be in *No_copy* status if the community does not have any information discussed above. A community will involve in CCDF-1h, CCDF-2h and CaR for this packet delivery.

We use the IEEE 802.11 broadcast mode to send the status vector in contrast to unicast in existing SCF routing. As a result, not only the intended node but also the other nodes in the vicinity receive the progress of

data delivery and update their knowledge accordingly, as per the state transition rules shown in Fig. 5.10. In the network layer, by having the packet state information from overhearing CCDF-1h and CCDF-2h transmissions, network layer acknowledgements, and packet state vectors, a cluster C_j will be infected with a packet copy from a cluster C_i only if all the following conditions are met. They are 1) C_j 's packet state is *No_copy*; 2) C_i 's packet state is neither *Local_fwd* nor *Delivered*; and 3) the *donotreplicate* flag of the packet is disabled. This results in a lower number of infections/replications, lower bandwidth consumption, and lesser buffer occupancy.

5.8.1 Vaccine Synchronization in a Cluster

In order to have the community-aware information epidemics control, vaccine synchronization or state vector synchronization in the community is needed. Here we use agent-based state vector synchronization. We use cluster head as an agent. The agent maintains a vector called synchronized vector or *sync-vector*. The same state definitions are used in the *sync-vector* as well. But, for the *Have_copy* status, the vector records the explicit information, that is the ID of the custody which is holding the packet. The agent updates its *sync-vector* whenever there is an update in the packet state vector. The agent broadcasts the *sync-vector* after a certain number of update is being made or when timeout.

Upon receiving the agent's *sync-vector*, members in the community updates its packet status vector as per the state transition rules. However, if a node finds that the information in the *sync-vector* does not reflect the latest packet status, it shares its latest status by piggybacking in the acknowledgement message. The agent will then consolidate the latest status of all packets and then share it with its members. By having the *sync-vector* vector, every

member in the community obtains the synchronized and latest packet status vector. Note that this message is sent by the agent for one-hop only. That is, only a handful of nodes in the entire network are sending out the message, therefore, it will not impact on the bandwidth and the network performance.

5.8.2 Overhead

In the protocols using existing vaccination approach (known as VACCINE) [33, 24, 28], it takes one bit for each packet inside the summary vector to tell if the node has the packet copy or not and another one bit for each packet inside the VACCINE vector to tell if the packet has been delivered to the destination or not. In the proposed algorithm, it uses only one vector called packet status vector and takes two bits for each packet to tell *No_copy*, *Have_copy*, *Local_fwd*, and *Delivered* packet status. As a result, the proposed algorithm requires transmission for one single vector only which results in less number of vector transmissions compared to the protocols using VACCINE. In addition, each vector provides further information including advance delivery progress while using the same number of bits as in the existing protocols. Moreover, the replication-based forwarding in the proposed algorithm is reactive that it is initiated only if the CCDF-1h and CCDF-2h opportunities are not available and the packets have never been forwarded by those forwarding. Therefore, this reactive nature results in lower number of replication-based forwarding and hence, lower packet state vector transmissions. The size of the vector can be further minimized 1) by using compression mechanism [55] and 2) by setting the timer based on the desired confidence level of the packet delivery and purge the corresponding information inside the vector when timeout. We will show the overhead of both vectors (state/summary vector and *sync-vector*) in the performance evaluation.

Table 5.5: Simulation parameters

Parameters	Value
Network area ($l \times l \text{ km}^2$)	1, 2, 3, 4, 5, 6, 7, 8, 9
Number of nodes	50
Number of clusters	10
Number of nodes per cluster	5
Transmission range	Probabilistic
Fading model	Nakagami
Mobility model	Reference Point Group Mobility Model (RPGM)
Speed	[5-15] m/s
Traffic rate	4 Packets/sec
Packet size	512 Bytes
Traffic type	Constant Bit Rate (CBR)
Number of traffic flows	27 pairs

Figure 5.11: Probe packet reception ratio over distance.

Table 5.6: Overall relative improvement of C-SCCF over ER

Area (sq km)	PDR		AE2ED		DTXC	
1	255.30	155.30%	0.48	99.52%	3.02	96.98%
2	104.52	4.52%	14.55	85.45%	6.67	93.33%
3	99.69	n/a	39.46	60.54%	9.09	90.91%
4	98.41	n/a	47.27	52.73%	10.60	89.4%
5	100.98	0.98%	50.21	49.79%	11.52	88.48%
6	101.46	1.46%	69.18	30.82%	12.72	87.28%
7	121.80	21.80%	78.09	21.91%	12.64	87.36%
8	133.81	33.81%	90.38	9.62%	13.99	86.01%
9	131.63	31.63%	95.66	4.34%	13.92	86.08%

The word “n/a” denotes not applicable. As there is no performance improvement for that particular case, we mark it as n/a.

5.9 Performance Evaluation

5.9.1 Evaluation Methodology

We use network simulator (NS-2) to evaluate the performance of the proposed scheme. The discussion on verification and validation of the simulation model is provided in 3.8.2. The evaluation covers three main network environments which include 1) dense network; relatively sparse network; and 3) highly sparse network. We examine 1) the packet delivery ratio (PDR) which shows the amount of packets delivered over the amount of packets generated; 2) the end-to-end delay which indicates the time taken from the packet generated at the source application layer to the packet delivered at the destination application layer; 3) the MAC layer data transmission count which indicates the number of times a data packet is transmitted/forwarded by the nodes

over the air to be delivered to the destination node (this metric is also called as data overhead in this chapter); and 4) the control overhead incurred by the packet vectors in both NET and MAC layers.

We choose AODV [10] for performance comparison in connected environment as AODV is shown to be one of the best protocols for such network. We choose ER [29] for comparison in disrupted/disconnected environment as it provides the highest delivery success rate with lower latency in such network, which is shown in recent studies [58, 26, 48, 18, 59].

The IEEE 802.11 standard is used in all the simulation studies in this work. The Nakagami propagation model is used to study the effect of realistic fading environment. Therefore, the transmission range between two nodes is probabilistic. The fading model parameters can be found in [60]. Figure 5.11 shows the probe (HELLO) packet reception ratio over distance. The reference point group mobility model is used to simulate a community type of network environment. Each group is composed of 5 members¹. In each group, the maximum distance from a member to the group center is set to 80 m. The randomly selected 27 source and destination pairs are used to generate the traffic. Note that source and destination nodes are selected from different clusters for generation of inter-cluster traffic. The traffic is generated from the beginning of the simulation till 600 seconds. Each simulation is run for one hour. The results are plotted with 90% confidence intervals. We collect simulation results using identical traffic model with 6 different randomly generated mobility scenarios. The detail simulation parameters are described in Table 5.5.

¹In recent studies [85], the size of clusters is shown to be usually not less than 4 in mobile social networks.

Figure 5.12: Overall performance: (a) packet delivery ratio; (b) average end-to-end delay; (c) average MAC layer transmission count; and (d) normalized hello packet collision rate in ER.

5.9.2 Overall network performance

We first study the overall network performance of C-SCCF, AODV, and ER with increasing network sparsity, shown in Fig. 5.12. We observe that AODV delivers 72% of the packets in 1 sq km, which is considered to be a dense network with the given number of nodes and the transmission range. However, as the network sparsity increases, the PDR drops dramatically, reaching almost 0% in 5 sq km. Being an end-to-end routing, it can be seen that the

performance is highly affected by the network connectivity which is influenced by mobility, network density, and probabilistic propagation channels.

We show the relative improvement of C-SCCF over ER in terms of packet delivery ratio (PDR), average end-to-end delay (AE2ED), and data transmission count (DTXC) in Table 5.6. The table shows relative value and improvement percentage for each metric. In dense network (1 sq km), we observe a significant PDR drop in ER. This drop is caused by severe hello packet collisions over the air, as shown in Fig. 5.12d. In relatively sparse networks (2 to 5 sq km), we observe a similar packet delivery ratio performance in both C-SCCF and ER. However, for end-to-end delay, we observe a significant improvement achieved by C-SCCF where it takes only 14.55%, 39.46%, 47.27%, and 50.21% of ER delay values, as shown in Table 5.6. In highly sparse networks (6 to 9 sq km), it is noted that although the end-to-end delay is approaching to that of ER, it still outperforms the ER in such sparse environment. We also observe that C-SCCF maintains the data delivery above 80% in the highly sparse network (9 sq km) whereas the delivery of ER decreases significantly, reaching almost 60% in 9 sq km. For the MAC layer transmission count, we observe that C-SCCF incurs only a small number of transmission count regardless of the network sparsity. The relative performance shows that it takes only a small fraction of the ER's transmission count in all the network environments. Therefore, it is noted that the information epidemics is well deterred in C-SCCF using proposed information epidemics control mechanism. However, in ER, it is observed to make more transmissions (infections) in denser environments due to higher contact rate compared to the sparser environment.

Figure 5.13: Dense network: (a) CDF with respect to packet delivery delay; (b) CDF with respect to transmission count.

5.9.3 Dense network

We then study the CDF of PDR over the end-to-end delay and MAC layer transmission count in dense network. We compare the performance with AODV as end-to-end routing was shown to prove promising delivery with smallest delivery delay and lowest transmission count in connected network environment. The network performance is shown in Fig. 5.13. We observe that given the maximum delay taken by AODV to deliver the packets, C-SCCF delivers far beyond what AODV does. Specifically, with the delay ≤ 5 seconds, about 95% of the packets are delivered by C-SCCF in contrast to 70% by AODV. For the MAC layer transmission count performance, it is noted that C-SCCF incurs 6 average transmission count to deliver 100% of the packets while AODV takes 4.2 average transmission count to deliver 70% of the packets. C-SCCF uses adaptive nature of routing where it proactively uses CCDF-1h and CCDF-2h whenever opportunity arises. It then switches to CaR when no opportunity is found. Having this adaptive nature brings

Figure 5.14: CDF with respect to packet delivery delay in relatively sparse networks: (a) 2 sq km; (b) 3 sq km; (c) 4 sq km; and (d) 5 sq km.

better delivery at the cost of more transmissions. Although it is well known that replication-based forwarding incurs information epidemics, especially in dense network, we observe that such epidemics is well controlled in C-SCCF, resulting in a negligible extra transmission cost for significant delivery performance.

5.9.4 Relatively sparse network

We then study the CDF of PDR over the end-to-end delay and MAC layer transmission count in relatively sparse networks which include from 2 sq km to 5 sq km. We observe that C-SCCF delivers the most number of packets with the minimum delay under all environments, as shown in Fig. 5.14. Specifically, to maintain the delivery at 80%, it incurs about 130, 370, 660, and 1000 seconds in 2,3,4, and 5 sq km, respectively. Whereas ER takes 680, 780, 1190, and 1860 seconds to deliver the same percentage. We note that in denser environment, i.e. 2 sq km, more CCDF forwarding opportunities are available and therefore, C-SCCF reduces 80% of the delay incurred by ER to provide the same delivery. In sparser environments, although delay of C-SCCF becomes higher, it still can reduce approximately 50% of the delay of ER by exploiting the CCDF forwarding opportunities proactively and making use of the receiver diversity for reliable communications over unreliable links.

The performance of CDF of PDR over the MAC layer transmission count is shown in Fig. 5.16. It is observed that C-SCCF delivers above 80% of the packets with the transmission count ≤ 30 under all the relatively sparse network environments ranging from 2 sq km to 5 sq km. Whereas for ER, to have the same delivery performance, it requires about 200 in 4 sq km and 5 sq km cases and 250 and 340 in 3 sq km and 2 sq km, respectively. It is noted that in denser network, ER incurs more transmissions due to 1) the increasing number of retransmissions caused by transmission failure as a result of mobility, channel fading, and collisions and 2) having more infections caused by higher contact rate. In C-SCCF, by exploiting receiver diversity, reliable data transfer is achieved over unreliable wireless channels. In addition, having the information epidemics control in the community prevents

Figure 5.15: CDF with respect to packet delivery delay in highly sparse networks: (a) 6 sq km; (b) 7 sq km; (c) 8 sq km; and (d) 9 sq km.

unnecessary transmission cost regardless of the contact rate in the network.

5.9.5 Highly sparse network

We then study the CDF of PDR over the end-to-end delay and MAC layer transmission count in highly sparse networks which include from 6 sq km to 9 sq km. The evaluation results for these cases are shown in Figs. 5.15 and 5.17. Generally, it is observed that C-SCCF reduces about 30% to 40% of the delay of ER under all cases for the maximum delivery provided by ER.

Figure 5.16: CDF with respect to MAC layer transmission count in relatively sparse networks: (a) Epidemic Routing; (b) C-SCCF Routing.

Figure 5.17: CDF with respect to MAC layer transmission count in highly sparse networks: (a) Epidemic Routing; (b) C-SCCF Routing.

Specifically, in 6 sq km case, C-SCCF takes 1570 seconds to make 80% of the delivery in contrast to 2130 seconds by ER. For 7,8, and 9 sq km cases, ER requires 3280, 3380, and 3350 seconds to make its maximum delivery whereas C-SCCF incurs about 1980, 2100, and 2530 seconds to deliver the same percentage. We also note that in extremely sparse cases, the delivery

of ER drops from 94% in 6 sq km to about 60% in 9 sq km, whereas C-SCCF still maintains well above 80% in such environments.

For the CDF of PDR over the MAC layer transmission count, we observe that C-SCCF requires at most 28 transmissions for 80% of the delivered packets in 6, 7, and 8 sq km networks. For 9 sq km case, it is noted that the transmission count rises to 45 to maintain the same delivery. In such extremely sparse environments, the chance of exploiting CCDF opportunities are getting smaller and most packet delivery uses CaR approach resulting in more transmissions. However, we observe that the transmissions incurred in C-SCCF is significantly smaller than the ER due to the information epidemics control. Specifically, ER makes 260, 230, 240, and 210 transmissions for its maximum deliverable packets (i.e. 95%, 77%, 66%, and 60%) in 6, 7, 8, and 9 sq km, respectively. For C-SCCF, for the same amount of delivered percentage, it incurs only 45, 26, 21, and 21 transmission count. Therefore, in general, C-SCCF reduces about 83% to 91% of ER transmission count for the same delivery rate.

5.9.6 Overhead

We then show the overhead caused by the vector transmissions in a dense environment and an extremely sparse environment. Figure 5.21 shows the normalized MAC load with 90% confidence intervals. Note that C-SCCF uses two types of vector called packet state vector (also known as summary vector) and *sync-vector* (also known as agent-vector) in its information epidemics control. Therefore we show the overhead caused by both vector transmissions. The overhead of C-SCCF and ER due to other control messages are also shown in the figure. In C-SCCF, those other control messages include summary vector request, decision request, decision reply, network layer

Figure 5.18: C-SCCF overhead in 2 sq km network: (a) CDF of summary vector transmissions; (b) CDF of agent vector transmissions.

Figure 5.19: C-SCCF overhead in 9 sq km network: (a) CDF of summary vector transmissions; (b) CDF of agent vector transmissions.

ACK, and MAC layer ACK and ARP packets. For ER, they include MAC layer ACK and ARP packets. Due to the unicast and non-cooperative nature of ER routing, a large amount of data packets suffer from packet loss and retransmissions which in turn causes a number of MAC layer ACK transmissions and therefore large overhead. Figures 5.18 and 5.20a show the CDF of

Figure 5.20: ER overhead: (a) in 2 sq km network; (b) in 9 sq km network.

Figure 5.21: Normalized MAC load in both 2 sq km and 9 sq km network: (a) C-SCCF; (b) ER.

vector transmission count incurred for the delivery over the simulation time in 2 sq km network. As C-SCCF uses reactive form of replication, there is no vector transmission in the beginning when there is no traffic generated in the network. It can be observed that the summary vector transmissions in C-SCCF is stopped even before some packets are not yet delivered to the destination. In C-SCCF, CaR forwarding approach is triggered only when

CCDF forwarding opportunities are not available and the data packets have never been forwarded by CCDF before. So, for some nodes which have packet copies but know that the packet has been forwarded by CCDF before will not use CaR approach.

In ER, we can see the vector transmissions in the network even when there is no traffic in the beginning due to the proactive nature of replication. It is observed that the vector transmissions in ER continues even after the 90% of delivery due to the lack of control mechanism in ER. Specifically, the total overhead due to both vector transmission in C-SCCF is about 11.1k transmissions over the air. Note that network layer overhead and MAC layer overhead is about the same due to no retransmission in IEEE 802.11 broadcast mode. So, the reliability is achieved with receiver diversity without requiring retransmissions. Whereas in ER, it incurs about 126k in network layer and 338k transmissions in MAC layer (i.e. over the air). Therefore, the actual overhead incurred in ER is 338k due to retransmissions. From these facts, we note that C-SCCF reduces almost 97% of the vector transmissions overhead in epidemic routing.

The overhead performance for 9 sq km network is shown in Figs. 5.19 and 5.20b. In this extremely sparse environment, we observe that C-SCCF vector transmissions continues over the time as the opportunity of using CCDF approach is lower. We also note that the summary vector transmission in both C-SCCF and ER is lower compared to the 2 sq km network due to the lower contact rate. Similar to the 2 sq km environment, a significant reduction in the overhead is achieved using C-SCCF compared to the ER. Specifically, it can be seen that C-SCCF incurs 6.15K total vector transmissions in contrast to 50K transmissions in ER, which therefore shows almost 88% reduction in vector transmissions overhead.

5.10 Conclusion

In this chapter, we have proposed a cooperation aided data delivery solution for opportunistic networks under community/group connectivity characteristics. We have shown that the reliability and timeliness of the data delivery in opportunistic networks can be maximized by 1) exploiting the receiver diversity brought by the community nature of the network; 2) deploying adaptive data forwarding strategies based on the contact opportunity; and 3) eliminating information epidemics problem using efficient vaccination and recovery scheme. We have proposed an adaptive cooperative data forwarding algorithm and a community aware information epidemics control algorithm to achieve the purpose. The performance is evaluated under both normal and extreme network environments. Specifically, the evaluation is conducted with varying network connectivity ranging from highly dense environment to highly sparse environment and under the presence of probabilistic propagation channels. We have shown that the proposed solution substantially improves the data delivery performance (higher delivery ratio while cutting down 4% to 85% of the delivery delay, 86% to 93% of the data transmission count, and 86% to 97% of the control overhead) compared to the pure flooding-based protocol which provides the best delivery with highest resource usage.

Chapter 6

Conclusions and Future Work

6.1 Conclusion

This thesis studies the data delivery solutions and information epidemics control in challenging wireless networks with major focus on opportunistic networks. We exploit adaptive data forwarding strategies, cooperation opportunities, open-air wireless medium nature, communities and ties in social network nature, and multiple network interfaces to tackle the challenges and improve the system performance.

We review some of the most representative group mobility models proposed for WANETs research. We describe the movement behaviour of each model concisely and provide our insights from analysing each model. A comprehensive comparison of the models based on the design features and target applications are provided. The list of design features is provided for users of these models to consider when adapting them or creating new models with the desired set of features to suit their specific applications.

We then propose a data delivery solution for opportunistic networks. Two main algorithms: SCCF routing and information epidemic control are pro-

posed to solve long delivery delay problem and information epidemic problem. Nodes adaptively switch between the cooperative forwarding and replication-based forwarding approaches based on the contact opportunity. Nodes reactively replicate the packet copies only when the packet has never been forwarded by cooperative forwarding. Moreover, overhearing the cooperative transmissions over the air provides the node the ability for better information epidemics control. The SIR population study shows that having both partially-immunized and fully-immunized vaccines and broadcasting the vaccines over the air give immunity to a large population of nodes and therefore deter a substantial amount of infections in the network. Network performance studies are conducted under fading, obstacle, and mobile social network environments. We found that 1) proactive replication incurs substantial forwarding cost and longer delay; 2) exploiting cooperative vicinity-aware forwarding proactively provides the delivery of most packets with minimum delay without incurring unnecessary forwarding cost; and 3) using reactive replication-based forwarding approach only for packets which are unable to be delivered in the vicinity ensures the delivery performance.

We then study the cooperative forwarding protocol design for multi-flow wireless networks. We observe that no cooperative forwarding algorithm that can work under multi-flow network where the same CRS is being used for different flows has been proposed. In addition, no performance study on the efficacy of cooperative forwarding under such environment has been conducted. In this work, we propose a simple and efficient cluster-based cooperative data forwarding (CCDF) algorithm for multi-flow networks. A new metric to compute the best cluster path is introduced. Distributed agent-based coordination (DAC) mechanism is proposed so as to support best candidate relay selection and duplicate transmission avoidance in multi-flow networks.

Analysis show that the DAC mechanism takes much shorter coordination time compared to the existing coordination method. Performance evaluation is conducted under multiple cross traffic flows. It is observed that network throughput is reduced and end-to-end delay is increased with the increasing number of traffic flows. To solve this issue, a MRMC-capable CCDF solution is proposed. By taking the advantage of clustering, MRMC can be easily integrated with the cooperative forwarding mechanism. It is observed that the integrated solution ensures the achievable throughput under highly congested traffic flows and unreliable wireless channels by exploiting the advantage of multiplexing using MRMC and spatial diversity using cooperative forwarding.

We then propose a cooperation aided data delivery solution for opportunistic networks under community/group connectivity characteristics. Given the presence of communities and centralities nature, it assures to exploit receiver diversity-based forwarding in opportunistic network so as to maximize the given contact opportunity for the reliable data transfer. In this work, we show that the reliability and timeliness of the data delivery in opportunistic networks can be maximized by 1) exploiting the receiver diversity brought by the community nature of the network; 2) deploying adaptive data forwarding strategies based on the contact opportunity; and 3) eliminating information epidemics problem using efficient vaccination and recovery scheme. We propose an adaptive cooperative data forwarding algorithm and a community aware information epidemics control algorithm to achieve the purpose. The performance is evaluated under both normal and extreme network environments. It is observed that the proposed solution substantially improves the data delivery performance (higher delivery ratio while cutting down 4% to 85% of the delivery delay, 86% to 93% of the data transmission count, and

86% to 97% of the control overhead) compared to the pure flooding-based protocol which provides the best delivery with highest resource usage.

6.2 Recommendation for Future Work

For cooperative forwarding, the author considers the issues of scheduling and coordination in existing design, especially in multiple cross traffic flows and the mobility. The author also takes into account the flexibility to integrate with multi-radio multi-channel capability so as to enhance the system throughput. From this proposed joint cooperative forwarding with multi-radio multi-channel, the theoretical throughput bound of the algorithm in the presence of multiple cross traffic flows can be studied and we leave this work as one of the future works.

To deter the unnecessary transmission cost in opportunistic networks with group characteristics, the author proposes a group/community aware information epidemic control algorithm. Based on the proposed mechanism, one can study the susceptible, infected, and recovered population in such networks with varying network density. In addition, to further enhance the information epidemics control with minimal overhead, a group/community with higher centrality can perform network coding on the packet state vectors it is overhearing if it detects any new vaccines (partially-immunized or fully-immunized) in the vector that the community in the vicinity has not yet known.

Based on the proposed data delivery solutions, we aim to deploy the cooperative data delivery solution in vehicle-to-everything (V2X) communications. V2X has become one of the emerging opportunistic network applications to advance intelligent transportation system (ITS). Data delivery

in V2X environments poses great challenges due to high mobility and faded wireless channels caused by mobile scatterers in the environment. Understanding communication link behaviour is an important factor in designing data delivery solutions for vehicular networks. For this purpose, we have done some preliminary work on vehicle-to-vehicle link level measurements in urban environments [86]. The specific focus for this future work can be described as follows.

Broadcasting the event-driven warning messages like accident and hazard warning is one of the important use cases in V2X. For example, after two vehicles collided with each other on a highway or urban road, or traffic congestion happens because of heavy rain or snow, the upcoming vehicles need to be alerted immediately. In both cases, the messages should be disseminated with only a short delay to vehicles that are up to several kilometers away, not only to prevent more possible accidents but also to enable the vehicles to make a detour as early as possible to avoid congestion. While Dedicated Short Range Communication (DSRC) allows the data transmission range of vehicles to be up to a few hundred meters, a single-hop broadcast is not sufficient to deliver the message, especially in urban environments as the link quality can be severely deteriorated and the link blackout can happen. So a cooperative forwarding is necessary to deliver the time-sensitive warning messages in such V2X networks. And we leave this deploying cooperation-aided data delivery solution in such network as one of the future works.

Author's Publications

Journal Papers

1. C. Y. Aung, B. C. Seet, M. Zhang, L. F. Xie, and P. H. J. Chong, "A review of group mobility models for mobile ad hoc networks," *Wireless Personal Communications*, vol. 85, no. 3, pp. 1317-1331, 2015.
DOI:10.1007/s11277-015-2842-z
2. C. Y. Aung, G. Md Nawaz Ali, and P. H. J. Chong, "Cluster-based Cooperative Data Forwarding with Multi-radio Multi-channel for Multi-flow Wireless Networks," *KSII Transactions on Internet and Information Systems*, vol. 10, no. 12, pp. 5149-5173, 2016.
DOI: 10.3837/tiis.2016.12.002
3. C. Y. Aung, I. W.-H. Ho, and P. H. J. Chong, "Store-carry-cooperative forward routing with information epidemics control for data delivery in opportunistic networks," to appear in *IEEE Access*.

Conference Papers

1. C. Y. Aung and P. H. J. Chong, "Cooperative Forwarding in Multi-radio Multi-channel Multi-flow Wireless Networks," to appear in *IEEE*

ICC 2017.

2. C. Y. Aung, G. Md Nawaz Ali, M. Zhao, R. J. Cai, and P. H. J. Chong, "Information epidemics control for data delivery in opportunistic networks," in Communications (ICC), 2016 IEEE International Conference on, pp. 1-6, IEEE, 2016.
3. C. Y. Aung, P. H. J. Chong, and R. J. Cai, "Hybrid opportunistic routing in highly dynamic manet," in Computer Communication and Networks (ICCCN), 2014 23rd International Conference on, pp. 1-6, IEEE, 2014.
4. C. Y. Aung, P. H. J. Chong, Z. Xinxing, A. C. Y. Pheng, N. T. Khoon, and K. S. Lai, "Hybrid routing in lossy link mobile ad hoc networks," in Information, Communications and Signal Processing (ICICS) 2013 9th International Conference on, pp. 1-6, IEEE, 2013.
5. H. Nguyen, C. Y. Aung, W. P. Tay, G. Y. Liang, and P. H. J. Chong, "Vehicle-to-vehicle communication performance measurements in residential area with high-rise buildings," in 14th ITS Asia Pacific Forum 2015.
6. R. J. Cai, G. Md Nawaz Ali, C. Y. Aung, M. Zhao, and P. H. J. Chong, "Simulation study of routing attacks under network coding environment," in Communications (ICC), 2016 IEEE International Conference on, pp. 1-6, IEEE, 2016.
7. R. J. Cai, P. H. J. Chong, and C. Y. Aung, "Poster: Trust-based routing with neighborhood connectivity to prevent single and colluded active black hole," in Communications and Networking in China (CHI-

NACOM), 2014 9th International Conference on, pp. 684-685, IEEE, 2014.

8. X. Zhao, P. H. J. Chong, and C. Y. Aung, "Delay tolerant network(DTN) based routing for mobile and hostile environment," in Proc. of OPNET-WORK 2013, 2013.

Working Papers

1. C. Y. Aung, P. H. J. Chong, H. Chan, and R. Lu, "Cluster-based Cooperative Store-Carry-Forward Protocol for Opportunistic Networks," under preparation for submission.

References

- [1] X. Hong, M. Gerla, G. Pei, and C.-C. Chiang, “A group mobility model for ad hoc wireless networks,” in *Proceedings of the 2nd ACM international workshop on Modeling, analysis and simulation of wireless and mobile systems*, pp. 53–60, ACM, 1999.
- [2] M. Musolesi and C. Mascolo, “A community based mobility model for ad hoc network research,” in *Proceedings of the 2nd international workshop on Multi-hop ad hoc networks: from theory to reality*, pp. 31–38, ACM, 2006.
- [3] K. H. Wang and B. Li, “Group mobility and partition prediction in wireless ad-hoc networks,” in *Communications, 2002. ICC 2002. IEEE International Conference on*, vol. 2, pp. 1017–1021, IEEE, 2002.
- [4] S. A. Williams and D. Huang, “Group force mobility model and its obstacle avoidance capability,” *Acta Astronautica*, vol. 65, no. 7, pp. 949–957, 2009.
- [5] K. Wu and Q. Yu, “A multi-group coordination mobility model for ad hoc networks,” in *Military Communications Conference, 2006. MILCOM 2006. IEEE*, pp. 1–5, IEEE, 2006.

- [6] J. M. Ng and Y. Zhang, “A mobility model with group partitioning for wireless ad hoc networks,” in *Information Technology and Applications, 2005. ICITA 2005. Third International Conference on*, vol. 2, pp. 289–294, IEEE, 2005.
- [7] B. Zhou, K. Xu, and M. Gerla, “Group and swarm mobility models for ad hoc network scenarios using virtual tracks,” in *Military Communications Conference, 2004. MILCOM 2004. 2004 IEEE*, vol. 1, pp. 289–294, IEEE, 2004.
- [8] N. Vastardis and K. Yang, “An enhanced community-based mobility model for distributed mobile social networks,” *Journal of Ambient Intelligence and Humanized Computing*, vol. 5, no. 1, pp. 65–75, 2014.
- [9] E. M. Royer and C.-K. Toh, “A review of current routing protocols for ad hoc mobile wireless networks,” *Personal Communications, IEEE*, vol. 6, no. 2, pp. 46–55, 1999.
- [10] C. Perkins, E. Belding-Royer, and S. Das, “Ad hoc on-demand distance vector (aodv) routing,” tech. rep., 2003.
- [11] D. B. Johnson and D. A. Maltz, “Dynamic source routing in ad hoc wireless networks,” in *Mobile computing*, pp. 153–181, Springer, 1996.
- [12] S. Biswas and R. Morris, “Opportunistic routing in multi-hop wireless networks,” *ACM SIGCOMM Computer Communication Review*, vol. 34, no. 1, pp. 69–74, 2004.
- [13] D. Aguayo, J. Bicket, S. Biswas, G. Judd, and R. Morris, “Link-level measurements from an 802.11 b mesh network,” *ACM SIGCOMM Computer Communication Review*, vol. 34, no. 4, pp. 121–132, 2004.

- [14] Z. Wang, Y. Chen, and C. Li, “Corman: A novel cooperative opportunistic routing scheme in mobile ad hoc networks,” *Selected Areas in Communications, IEEE Journal on*, vol. 30, no. 2, pp. 289–296, 2012.
- [15] K. Zeng, W. Lou, and M. Li, *Multihop Wireless Networks: Opportunistic Routing*, vol. 25. John Wiley & Sons, 2011.
- [16] K. Fall, “A delay-tolerant network architecture for challenged internets,” in *Proceedings of the 2003 conference on Applications, technologies, architectures, and protocols for computer communications*, pp. 27–34, ACM, 2003.
- [17] Z. Zhang, “Routing in intermittently connected mobile ad hoc networks and delay tolerant networks: overview and challenges,” *Communications Surveys & Tutorials, IEEE*, vol. 8, no. 1, pp. 24–37, 2006.
- [18] X. Zhang and G. Cao, “Efficient data forwarding in mobile social networks with diverse connectivity characteristics,” in *Distributed Computing Systems (ICDCS), 2014 IEEE 34th International Conference on*, pp. 31–40, IEEE, 2014.
- [19] T. Phe-Neau, M. D. De Amorim, and V. Conan, “The strength of vicinity annexation in opportunistic networking,” in *INFOCOM, 2013 Proceedings IEEE*, pp. 3369–3374, IEEE, 2013.
- [20] H. Chen and W. Lou, “Gar: Group aware cooperative routing protocol for resource-constraint opportunistic networks,” *Computer Communications*, vol. 48, pp. 20–29, 2014.
- [21] X. Tie, A. Venkataramani, and A. Balasubramanian, “R3: robust replication routing in wireless networks with diverse connectivity character-

- istics,” in *Proceedings of the 17th annual international conference on Mobile computing and networking*, pp. 181–192, ACM, 2011.
- [22] L. F. Xie, P. H. Chong, and Y. L. Guan, “Leader based group routing in disconnected mobile ad hoc networks with group mobility,” *Wireless personal communications*, vol. 71, no. 3, pp. 2003–2021, 2013.
- [23] C. Raffelsberger and H. Hellwagner, “A hybrid manet-dtn routing scheme for emergency response scenarios,” in *Pervasive Computing and Communications Workshops (PERCOM Workshops), 2013 IEEE International Conference on*, pp. 505–510, IEEE, 2013.
- [24] T. Kimura, T. Matsuda, and T. Takine, “Multi-spreader routing for sparsely populated mobile ad hoc networks,” *Wireless networks*, vol. 20, no. 1, pp. 155–175, 2014.
- [25] C. Y. Aung, P. H. J. Chong, and R. J. Cai, “Hybrid opportunistic routing in highly dynamic manet,” in *Computer Communication and Networks (ICCCN), 2014 23rd International Conference on*, pp. 1–6, IEEE, 2014.
- [26] P. Hui, J. Crowcroft, and E. Yoneki, “Bubble rap: Social-based forwarding in delay-tolerant networks,” *Mobile Computing, IEEE Transactions on*, vol. 10, no. 11, pp. 1576–1589, 2011.
- [27] H. Dang and H. Wu, “Clustering and cluster-based routing protocol for delay-tolerant mobile networks,” *IEEE Transactions on Wireless Communications*, vol. 9, no. 6, pp. 1874–1881, 2010.
- [28] J. Lakkakorpi, M. Pitkänen, and J. Ott, “Adaptive routing in mobile opportunistic networks,” in *Proceedings of the 13th ACM international*

- conference on Modeling, analysis, and simulation of wireless and mobile systems*, pp. 101–109, ACM, 2010.
- [29] A. Vahdat, D. Becker, *et al.*, “Epidemic routing for partially connected ad hoc networks,” 2000.
- [30] R. Ramanathan, R. Hansen, P. Basu, R. Rosales-Hain, and R. Krishnan, “Prioritized epidemic routing for opportunistic networks,” in *Proceedings of the 1st international MobiSys workshop on Mobile opportunistic networking*, pp. 62–66, ACM, 2007.
- [31] T. Spyropoulos, K. Psounis, and C. S. Raghavendra, “Spray and wait: an efficient routing scheme for intermittently connected mobile networks,” in *Proceedings of the 2005 ACM SIGCOMM workshop on Delay-tolerant networking*, pp. 252–259, ACM, 2005.
- [32] A. Lindgren, A. Doria, and O. Schelén, “Probabilistic routing in intermittently connected networks,” *ACM SIGMOBILE mobile computing and communications review*, vol. 7, no. 3, pp. 19–20, 2003.
- [33] T. Matsuda and T. Takine, “(p, q)-epidemic routing for sparsely populated mobile ad hoc networks,” *Selected Areas in Communications, IEEE Journal on*, vol. 26, no. 5, pp. 783–793, 2008.
- [34] T. Spyropoulos, K. Psounis, and C. S. Raghavendra, “Performance analysis of mobility-assisted routing,” in *Proceedings of the 7th ACM international symposium on Mobile ad hoc networking and computing*, pp. 49–60, ACM, 2006.
- [35] W. Zhao, M. Ammar, and E. Zegura, “A message ferrying approach for data delivery in sparse mobile ad hoc networks,” in *Proceedings of*

- the 5th ACM international symposium on Mobile ad hoc networking and computing*, pp. 187–198, ACM, 2004.
- [36] W. Gao, Q. Li, B. Zhao, and G. Cao, “Multicasting in delay tolerant networks: a social network perspective,” in *Proceedings of the tenth ACM international symposium on Mobile ad hoc networking and computing*, pp. 299–308, ACM, 2009.
- [37] Z. J. Haas and T. Small, “A new networking model for biological applications of ad hoc sensor networks,” *Networking, IEEE/ACM Transactions on*, vol. 14, no. 1, pp. 27–40, 2006.
- [38] X. Zhang, G. Neglia, J. Kurose, and D. Towsley, “Performance modeling of epidemic routing,” *Computer Networks*, vol. 51, no. 10, pp. 2867–2891, 2007.
- [39] P.-Y. Chen, S.-M. Cheng, and K.-C. Chen, “Optimal control of epidemic information dissemination over networks,” *Cybernetics, IEEE Transactions on*, vol. 44, no. 12, pp. 2316–2328, 2014.
- [40] E. Rozner, J. Seshadri, Y. A. Mehta, and L. Qiu, “Soar: Simple opportunistic adaptive routing protocol for wireless mesh networks,” *Mobile Computing, IEEE Transactions on*, vol. 8, no. 12, pp. 1622–1635, 2009.
- [41] S. Yang, C. K. Yeo, and B. S. Lee, “Toward reliable data delivery for highly dynamic mobile ad hoc networks,” *Mobile Computing, IEEE Transactions on*, vol. 11, no. 1, pp. 111–124, 2012.
- [42] N. Chakchouk, “A survey on opportunistic routing in wireless communication networks,” *Communications Surveys & Tutorials, IEEE*, vol. 17, no. 4, pp. 2214–2241, 2015.

- [43] J. Ott, D. Kutscher, and C. Dwertmann, “Integrating dtn and manet routing,” in *Proceedings of the 2006 SIGCOMM workshop on Challenged networks*, pp. 221–228, ACM, 2006.
- [44] Z. Zhang, H. Wang, C. Wang, and H. Fang, “Modeling epidemics spreading on social contact networks,” *Emerging Topics in Computing, IEEE Transactions on*, vol. 3, no. 3, pp. 410–419, 2015.
- [45] M. Musolesi and C. Mascolo, “Designing mobility models based on social network theory,” *ACM SIGMOBILE Mobile Computing and Communications Review*, vol. 11, no. 3, pp. 59–70, 2007.
- [46] M. Musolesi, S. Hailes, and C. Mascolo, “Social networks based ad hoc mobility models,” 2005.
- [47] M. Musolesi, S. Hailes, and C. Mascolo, “An ad hoc mobility model founded on social network theory,” in *Proceedings of the 7th ACM international symposium on Modeling, analysis and simulation of wireless and mobile systems*, pp. 20–24, ACM, 2004.
- [48] J. Whitbeck and V. Conan, “Hymad: Hybrid dtn-manet routing for dense and highly dynamic wireless networks,” *Computer Communications*, vol. 33, no. 13, pp. 1483–1492, 2010.
- [49] T. Camp, J. Boleng, and V. Davies, “A survey of mobility models for ad hoc network research,” *Wireless communications and mobile computing*, vol. 2, no. 5, pp. 483–502, 2002.
- [50] F. Bai, N. Sadagopan, and A. Helmy, “Important: A framework to systematically analyze the impact of mobility on performance of routing

- protocols for adhoc networks,” in *INFOCOM 2003. Twenty-second annual joint conference of the IEEE computer and communications. IEEE societies*, vol. 2, pp. 825–835, IEEE, 2003.
- [51] Q. Zheng, X. Hong, and S. Ray, “Recent advances in mobility modeling for mobile ad hoc network research,” in *Proceedings of the 42nd annual Southeast regional conference*, pp. 70–75, ACM, 2004.
- [52] N. Aschenbruck, E. Gerhards-Padilla, and P. Martini, “A survey on mobility models for performance analysis in tactical mobile networks,” *Journal of Telecommunications and Information Technology*, pp. 54–61, 2008.
- [53] J. Härri, F. Filali, and C. Bonnet, “Mobility models for vehicular ad hoc networks: a survey and taxonomy,” *Communications Surveys & Tutorials, IEEE*, vol. 11, no. 4, pp. 19–41, 2009.
- [54] L. F. Xie, P. H. Chong, Y. L. Guan, and B. C. Ng, “Routing for group mobility in mobile ad hoc networks with delay-tolerant approach,” in *Wireless and Mobile Networking Conference (WMNC), 2010 Third Joint IFIP*, pp. 1–6, IEEE, 2010.
- [55] Z. Ren, W. Liu, X. Zhou, J. Fang, and Q. Chen, “Summary-vector-based effective and fast immunization for epidemic-based routing in opportunistic networks,” *Communications Letters, IEEE*, vol. 18, no. 7, pp. 1183–1186, 2014.
- [56] C. Papageorgiou, K. Birkos, T. Dagiuklas, and S. Kotsopoulos, “Modeling human mobility in obstacle-constrained ad hoc networks,” *Ad hoc networks*, vol. 10, no. 3, pp. 421–434, 2012.

- [57] A. Mei and J. Stefa, “Swim: A simple model to generate small mobile worlds,” in *INFOCOM 2009, IEEE*, pp. 2106–2113, IEEE, 2009.
- [58] K. Wei, D. Zeng, S. Guo, and K. Xu, “On social delay-tolerant networking: Aggregation, tie detection, and routing,” *IEEE Transactions on Parallel and Distributed Systems*, vol. 25, no. 6, pp. 1563–1573, 2014.
- [59] Q. Yuan, I. Cardei, and J. Wu, “An efficient prediction-based routing in disruption-tolerant networks,” *IEEE Transactions on Parallel and Distributed Systems*, vol. 23, no. 1, pp. 19–31, 2012.
- [60] Q. Chen, F. Schmidt-Eisenlohr, D. Jiang, M. Torrent-Moreno, L. Delgrossi, and H. Hartenstein, “Overhaul of iee 802.11 modeling and simulation in ns-2,” in *Proceedings of the 10th ACM Symposium on Modeling, analysis, and simulation of wireless and mobile systems*, pp. 159–168, ACM, 2007.
- [61] T. Issariyakul and E. Hossain, *Introduction to network simulator NS2*. Springer Science & Business Media, 2011.
- [62] “The eclipse integrated development environment.” <https://eclipse.org/indigo/>. Accessed: 2016-10-17.
- [63] S. Kosta, A. Mei, and J. Stefa, “Large-scale synthetic social mobile networks with swim,” *Mobile Computing, IEEE Transactions on*, vol. 13, no. 1, pp. 116–129, 2014.
- [64] J. Scott, R. Gass, J. Crowcroft, P. Hui, C. Diot, and A. Chaintreau, “Crawdad trace cambridge/haggle/imote/infocom (v. 2006-01-31),” *Download from http://crawdad.cs.dartmouth.edu/cambridge/haggle/imote/infocom*, 2006.

- [65] M. Kodialam and T. Nandagopal, “Characterizing the capacity region in multi-radio multi-channel wireless mesh networks,” in *Proceedings of the 11th annual international conference on Mobile computing and networking*, pp. 73–87, ACM, 2005.
- [66] A. P. Subramanian, M. M. Buddhikot, and S. Miller, “Interference aware routing in multi-radio wireless mesh networks,” in *Wireless Mesh Networks, 2006. WiMesh 2006. 2nd IEEE Workshop on*, pp. 55–63, IEEE, 2006.
- [67] M. Alicherry, R. Bhatia, and L. E. Li, “Joint channel assignment and routing for throughput optimization in multi-radio wireless mesh networks,” in *Proceedings of the 11th annual international conference on Mobile computing and networking*, pp. 58–72, ACM, 2005.
- [68] A. Raniwala, K. Gopalan, and T.-c. Chiueh, “Centralized channel assignment and routing algorithms for multi-channel wireless mesh networks,” *ACM SIGMOBILE Mobile Computing and Communications Review*, vol. 8, no. 2, pp. 50–65, 2004.
- [69] R. Draves, J. Padhye, and B. Zill, “Routing in multi-radio, multi-hop wireless mesh networks,” in *Proceedings of the 10th annual international conference on Mobile computing and networking*, pp. 114–128, ACM, 2004.
- [70] K. N. Ramachandran, E. M. Belding-Royer, K. C. Almeroth, and M. M. Buddhikot, “Interference-aware channel assignment in multi-radio wireless mesh networks,” in *Infocom*, vol. 6, pp. 1–12, 2006.

- [71] K. Zeng, Z. Yang, and W. Lou, "Opportunistic routing in multi-radio multi-channel multi-hop wireless networks," *Wireless Communications, IEEE Transactions on*, vol. 9, no. 11, pp. 3512–3521, 2010.
- [72] A. Zubow, M. Kurth, and J.-P. Redlich, "An opportunistic cross-layer protocol for multi-channel wireless networks," in *Personal, Indoor and Mobile Radio Communications, 2007. PIMRC 2007. IEEE 18th International Symposium on*, pp. 1–5, IEEE, 2007.
- [73] Y. Yuan, H. Yang, S. H. Wong, S. Lu, and W. Arbaugh, "Romer: resilient opportunistic mesh routing for wireless mesh networks," in *IEEE workshop on wireless mesh networks (WiMesh)*, vol. 12, 2005.
- [74] D. M. Kim, Y. J. Hwang, S.-L. Kim, and G.-J. Jin, "Testbed results of an opportunistic routing for multi-robot wireless networks," *Computer Communications*, vol. 34, no. 18, pp. 2174–2183, 2011.
- [75] H. Liu, B. Zhang, H. T. Mouftah, X. Shen, and J. Ma, "Opportunistic routing for wireless ad hoc and sensor networks: Present and future directions," *Communications Magazine, IEEE*, vol. 47, no. 12, pp. 103–109, 2009.
- [76] W. Jingrong, W. Muqing, L. Bo, and W. Dongyang, "Opportunistic cooperative routing in multi-radio multi-channel wireless sensor networks," in *Communications Workshops (ICC), 2013 IEEE International Conference on*, pp. 276–280, IEEE, 2013.
- [77] H. Shiming, Z. Dafang, X. Kun, Q. Hong, and Z. Ji, "A simple channel assignment for opportunistic routing in multi-radio multi-channel wireless mesh networks," in *Mobile Ad-hoc and Sensor Networks (MSN), 2011 Seventh International Conference on*, pp. 201–208, IEEE, 2011.

- [78] F. Wu and N. Vaidya, “Workload-aware opportunistic routing in multi-channel, multi-radio wireless mesh networks,” in *Sensor, Mesh and Ad Hoc Communications and Networks (SECON), 2012 9th Annual IEEE Communications Society Conference on*, pp. 344–352, IEEE, 2012.
- [79] D. S. De Couto, D. Aguayo, J. Bicket, and R. Morris, “A high-throughput path metric for multi-hop wireless routing,” *Wireless Networks*, vol. 11, no. 4, pp. 419–434, 2005.
- [80] M. M. Buddhikot, S. C. Miller, and A. P. Subramanian, “Interference aware routing in multi-radio wireless mesh networks,” Sept. 10 2013. US Patent 8,532,023.
- [81] C. Y. Aung, B. C. Seet, M. Zhang, L. F. Xie, and P. H. J. Chong, “A review of group mobility models for mobile ad hoc networks,” *Wireless Personal Communications*, vol. 85, no. 3, pp. 1317–1331, 2015.
- [82] L. F. Xie, P. H. J. Chong, and Y. L. Guan, “Routing strategy in disconnected mobile ad hoc networks with group mobility,” *EURASIP Journal on Wireless Communications and Networking*, vol. 2013, no. 1, pp. 1–12, 2013.
- [83] C. Y. Aung, G. M. N. Ali, M. Zhao, R. Cai, and P. H. J. Chong, “Information epidemics control for data delivery in opportunistic networks,” in *Communications (ICC), 2016 IEEE International Conference on*, pp. 1–6, IEEE, 2016.
- [84] C. Liu and J. Wu, “On multicopy opportunistic forwarding protocols in nondeterministic delay tolerant networks,” *IEEE Transactions on Parallel and Distributed Systems*, vol. 23, no. 6, pp. 1121–1128, 2012.

- [85] J. Leskovec, K. J. Lang, A. Dasgupta, and M. W. Mahoney, “Statistical properties of community structure in large social and information networks,” in *Proceedings of the 17th international conference on World Wide Web*, pp. 695–704, ACM, 2008.
- [86] C. Y. Aung, N. Hieu, W. P. Tay, Y. L. Guan, and P. H. J. Chong, “Vehicle-to-vehicle communication performance measurements in residential area with high-rise buildings,” in *14th ITS Asia Pacific Forum 2015*.