

Network Framework For Multi-UAV Guided Ground Ad Hoc Network

A Thesis

submitted in partial fulfillment of the requirements for the award of the degree of

Doctor of Philosophy

by

Vishal Sharma

(Reg no: 901203003)

Under the Supervision of

Dr. Rajesh Kumar
(Professor, CSED)

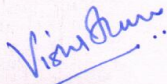


Computer Science and Engineering Department

Thapar University
Patiala, Punjab-147004, India
June, 2016

Certificate

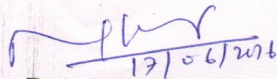
I hereby certify that the work which is being presented in this thesis entitled "**Network Framework For Multi-UAV Guided Ground Ad Hoc Network**", in partial fulfillment of the requirement for the award of degree of "Doctor of Philosophy" submitted in Computer Science and Engineering Department of Thapar University, Patiala, is an authentic record of my own work carried out under the supervision of Dr. Rajesh Kumar and refers other research works which are duly listed in the reference section. The matter presented in this thesis has not been submitted for the award of any other degree of this or any other university.



(Vishal Sharma)

Regn. No. 901203003

This is to certify that the above statement made by the candidate is correct and true to the best of my knowledge.



(Dr. Rajesh Kumar)

Professor

Computer Science and Engineering Department
Thapar University

Patiala, 147004

Punjab, INDIA.

Acknowledgements

First, I would like to express my deep gratitude to my supervisor, **Dr. Rajesh Kumar**, Professor, Computer Science and Engineering Department, Thapar University, for his invaluable advice and encouragement at every step of my PhD program. Without his unfailing support and belief in me, this thesis would not have been possible. His contribution to this thesis goes well beyond his role as an academic supervisor and includes constant support on a personal level without which this journey may never have been completed. And for this, I am truly grateful. He is a great mentors for my life as well.

I would like to express my gratitude to Professor & Head, Computer Science and Engineering Department **Dr. Deepak Garg**, and Senior Professor & Dean-RSP **Dr. O.P. Pandey** for their constant motivation and encouragement. I would like to thank to **Dr. Ilsun You** of Soonchunhyang University, South Korea who guided for the quality research work and self motivation. I would like to thank **Dr. Neeraj Rathore**, **Dr. Ravinder Kumar** and **Dr. Prashant Singh Rana** for their support. I also wish to thank my research committee members and non-teaching staff of the institute for their help and support. I would also like to thank to my AOL teachers and friend from whom I learn the art of happiness and never give up approach.

I would like to give special acknowledgements to my fellow Ph.D scholars Mr. Karun Verma, Mr. Mohd. Abuzar Sayeed, Mr. Kuldeep Sharma, Mr. Lohit Kapoor, Ms. Nidhi J. Kansal, and Ms. Seemu Sharma for their support and motivation.

I would like to thank to my friends Prashant Sharma, Varun Bhalla, Keshav Saini, Kritish Sharma, Takshi, Sahil Singla, Aman Kataria and Varun Chauhan for their support, motivation and lots of love.

Finally, I thank almighty, and would like to express my sincere and deep gratitude to my parents and family for their love, encouragement and care, without their support, I could not complete my Ph.D. program.

Vishal Sharma

*.....dedicated to Dr. A.P.J. Abdul Kalam, my parents,
family, friends, and all young generation researchers*

Abstract

Unmanned aerial vehicles (UAVs) have gained lots of potential over the years, providing a vast range of applications in existing networks. UAVs have the ability to cooperate, and can fly autonomously or can be operated without human intervention, thus, provides a versatile, and a flexible implementation. Integration of these vehicles can solve various issues concerning both civilian activities as well as military operations. One of the possible integrations of these vehicles is with ad hoc networks. Ad hoc networks are low cost infrastructure based networks that operate on the ideology of forming intermittent networks on demand. Traditional ad hoc networks do not require any centralized device for data forwarding. Every node themselves act as transmitter, receiver, and router. Formation of collaborative network between the aerial nodes and the ground nodes provides a vast range of applications in areas of civilian and military activities. These collaborations can lead to formation of guidance systems that can be used to provide look ahead information to ground units. The work presented in this thesis considers the collaboration between these two different ad hoc units as a research aspect, and provides efficient strategies for enhanced transmission. In the initial phase, a cooperative framework is developed which forms a guidance system comprising of aerial and ground nodes operating in ad hoc mode. This framework forms a collaborative task oriented network, which carries search and tracking without any redundancy. The operating time of this framework is quite low. Also, it acts as a guidance system for ground nodes, thus, providing information of user stations on the ground. The proposed framework utilizes the Bayesian probabilistic model, neural networks, and Quaternion Kalman Filter for its successful operations. The proposed framework acts as a base for network formation between the aerial and the ground nodes.

In order to completely utilize the functions of the proposed framework, an inter-networking layer is incorporated into it. A new routing protocol based on the Fruit Fly Optimization (FFO) algorithm is proposed. This protocol utilizes the smell index property of fruit flies for selection of optimal route, maintenance, and rehabilitation. The proposed protocol provides an efficient approach for data/cognitive transmission in multi-UAV guided ad hoc networks. A detailed comparative analysis of the proposed protocol with some of the existing routing protocols is also presented. Further, this formation is improved by providing a service coordination facility over the proposed network framework. This service provisioning is achieved in three parts. In the first part, a middleware is proposed that provides an initial layout of layers functioning for efficient service coordination. In

the second part, this layered middleware is subjected to cross layer formation for efficient service dissemination. In the third part, a complete model is presented that provides a 3-tier approach for efficient service coordination utilizing the initially proposed cooperative framework. The proposed 3-tier neural based service provisioning model provides a robust, reliable, and fault-tolerant connectivity between the ground ad hoc and the aerial ad hoc networks.

Utilizing the service enhancement, focus is driven towards the improvement of Quality of Service (QoS) to end users. Aiming at guaranteed QoS to end users, a queue scheduler is proposed for both ground as well as aerial assisted ad hoc networks. This approach is also developed in two parts. In the first part, a queue scheduler using the estimation and prediction strategies formulated over the Quaternion Kalman Filter is proposed. The proposed approach allows efficient estimation of network delays and bandwidth consumption. In the second part, this model is extended over UAV assisted network formations. This part also utilizes the features of existing Satisfied Importance Analysis (SIA) approach. The proposed approach is capable of providing guaranteed QoS to end users with improved connectivity.

Stabilization and broadcast storm are the other issues considered in this work. These issues hinder the network formation, and can cause serious deadlocks. Stabilization over UAVs provide continuous connectivity even in case of node failures. For stabilized connectivity, a self healing neural model is proposed that uses the concept of dummy neurons to provide stabilization against uncertain failures over networked UAVs. Broadcast storm causes redundancy and excessive utilization of resources for the same data. This is a serious issue in ad hoc formations between the aerial and the ground nodes. In order to eliminate this issue, a logical proximity graph bases parameter sensitivity routing (PSR) is proposed. The proposed approach not only solves the broadcast storm issue, but also improves the connectivity by lowering the chances of network partitioning.

In a cooperative network, security is one of the major issues. Here, security is defined in terms of localization, corridor privacy, and cognitive transfers. For this, a secure framework developed on the backbone of the initially proposed cooperative framework is proposed. This framework utilizes the α - β - γ filter to form a Teredo Tunnel for ad hoc network formation between the aerial and the ground nodes. The proposed framework withstands the network anomalies and faults, and proves robust and fault-tolerant by preventing any network hindrance during intrusions.

Table of Contents

Title	Page No.
Certificate	ii
Acknowledgement	iv
Abstract	vii
Table of Contents	ix
List of Figures	xiii
List of Tables	xvii
List of Notations	xix
List of Abbreviations	xxiii
Chapter 1 Introduction	1
1.1 Background	1
1.1.1 UAV Ad Hoc Networks	3
1.1.2 Ground Ad Hoc Networks	4
1.1.3 Cooperative Network Formations	4
1.2 Multi-UAV Guided Ground Ad Hoc Formations	4
1.2.1 Network Components and Architecture	5
1.2.2 Design Issues	7
1.2.3 Characteristics	8
1.2.4 Challenges	10
1.2.5 Applications of UAV Guided Ad Hoc Networks	11
1.3 Thesis Organization	13
Chapter 2 Literature Review	15
2.1 Taxonomy of UAV Networks	15
2.2 Literature Classification	16
2.2.1 Single UAV System	16
2.2.2 Multi UAV System	19

2.2.2.1	Cooperative Networks	20
2.2.2.2	Opportunistic Relaying	23
2.2.2.3	Task Allocations	25
2.2.2.4	UAV Aided Routing and Positioning	26
2.3	Comparison of Existing Frameworks/Models	27
2.3.1	Preface to Nature-Inspired Algorithms and Routing Protocols . .	32
2.3.2	Preface to TCP Variants for Congestion Control	32
2.4	Preface To Service Provisioning	33
2.5	Preface To QoS Scheduling In Ad Hoc Networks	36
2.6	Control Issues Over Networked UAVs	38
2.6.1	Stabilization	38
2.6.2	Topological Broadcasting	39
2.7	Security Issues Over Networked UAVs	41
2.8	Problem Formulation	43
2.9	Objectives	44
2.10	Conclusion	44
Chapter 3	Cooperative Network Framework	45
3.1	Proposed Framework	46
3.1.1	Area Cognitive Maps	47
3.1.2	Topology Organizing Map for Ground Ad Hoc Network	50
3.1.3	UAVs Topology Organizing Map	53
3.1.4	UAVs Ad Hoc Network Formation and Merging of Cognitive Maps	54
3.1.5	Modified ART2 Algorithm For Combining Cognitive Maps	57
3.2	Performance Evaluation	60
3.2.1	Analysis of Cooperative Network Formation	61
3.2.2	Analysis of Neural Schema used for Cooperative Network	64
3.3	Conclusion	66
Chapter 4	Routing and Cooperative Relaying Over UAV Guided Networks	67
4.1	Network Model	67
4.2	Proposed Protocol: OFFRP	71
4.2.1	Relay Selection and Route Discovery	71
4.2.2	Route Maintenance	73
4.2.3	Route Rehabilitation	78
4.2.4	Data Structures and Beacon Configurations	78
4.3	Performance Evaluation	81

4.3.1	Taxonomy of Network Parameters Used For Analysis	81
4.3.2	Performance Evaluation	86
4.3.3	Comparison With Existing TCP Algorithms	90
4.3.4	Variation For Multiple Simulations	90
4.4	Conclusion	92
Chapter 5	Service Provisioning Over UAVs Coordinated Networks . .	93
5.1	Service Middleware For UAVs Guided Networks	95
5.2	Service Dissemination Using Cross Layer Design	95
5.2.1	System Model For Service Dissemination	96
5.2.2	Opportunistic Cross Layer Formation	98
5.3	3-Tier Neural Model For Efficient Service Dissemination	99
5.3.1	Components of Tier-1 and Tier-3	102
5.3.2	Neural Schema For Service Layer Integration	105
5.3.3	Neural Estimator	107
5.3.4	Constraint Over Neural Model	108
5.3.5	System Fault Tolerance	109
5.3.6	Integration of Proposed Model With Cooperative Framework . . .	110
5.4	Performance Evaluation	112
5.4.1	Analysis of Proposed 3-Tier Service Model	114
5.4.2	Analysis of Cooperative Network Formed Using Proposed Model .	116
5.5	Conclusion	120
Chapter 6	QoS Scheduling For End Users In Ground and UAV Ad Hoc Networks	121
6.1	Quaternion Filter Based Network Formation	122
6.2	Proposed Model	129
6.3	Bandwidth and Delay Analysis	132
6.3.1	Bandwidth Evaluation	132
6.3.2	Delay Evaluation	133
6.3.3	Network Throughput	135
6.3.3.1	Common Channel-Based Network	135
6.3.3.2	Variable Channel-Based Network	137
6.4	Estimation Analysis Over Ground Network	138
6.5	UAV Assisted QoS Provisioning For Ground Ad Hoc Networks	141
6.5.1	System Model	141
6.5.2	Scheduling Cost Method For UAVs Allocation	143
6.5.3	Queue Scheduling	145

6.5.4	Theoretical Analysis	147
6.6	Performance Evaluation	148
6.7	Conclusion	150
Chapter 7 Stabilization and Topological Broadcasting Over Networked UAVs		151
7.1	Self-Healing Model For Stabilization Against Failures Over Networked UAVs	152
7.1.1	Proposed Self-Healing Neural Model	154
7.1.2	Performance Evaluation	157
7.2	Topological Broadcasting Over UAV Networks	159
7.2.1	Proposed Solution To Broadcast Storm	161
7.2.2	Network Parameter Sensitivity	161
7.2.3	Formation of Logical Proximity Graphs	164
7.2.3.1	Proximity-Sensitivity Routing (PSR)	165
7.2.3.2	Managing Network Partitioning and Congestion Window	166
7.2.4	Performance Evaluation	168
7.3	Conclusion	170
Chapter 8 Security Enhancement of Cooperative Network Framework		171
8.1	Proposed Framework For Secure Network Formation	171
8.1.1	Component Model For Secure Network Formation	173
8.1.2	System Model	176
8.1.3	Teredo Tunneling For Secure Connections	185
8.2	Performance Evaluation	188
8.2.1	Assessment Metrics and Baseline	189
8.2.2	Teredo Formation Analysis	191
8.2.3	Comparative Study	194
8.3	Conclusion	196
Chapter 9 Conclusion and Future Scope		197
9.1	Summary of Work Done	197
9.2	Conclusion	198
9.3	Limitations of Work	199
9.4	Scope of Future Work	200
References		201
List of Publications		227

List of Figures

Figure No.	Title	Page No.
1.1	UAVs Oriented Network	3
1.2	Network Architecture For UAV Guided Ad Hoc Formations	5
2.1	Taxonomy of UAV Networks	16
2.2	Overview of Literature Based on Network Taxonomy	17
3.1	Area Division and Mapping	47
3.2	Neural Mapping of Node Parameters	51
3.3	Neural Schema For Cooperative Ad Hoc Formation	59
3.4	Neural Layered Structure For Cooperative Ad Hoc Formation	59
3.5	Concept Value Prediction Without Kalman Filter	61
3.6	Concept Value Estimation Using Kalman Filter	61
3.7	Location of Virtual Concepts	62
3.8	Topology Organizing Map for Node Neurons	62
3.9	Average Weight Mapping For Seeds and Min. Achievable Rate	62
3.10	Neural Map For Node(Neurons)-Concept Mapping	62
3.11	Final Topology Organizing Map for Nodes	62
3.12	Power Spectral Graph for Normalized Concepts	62
3.13	3-D Plot For Yaw (15 UAVs)	63
3.14	Contour Representation of Area Before Traversal	63
3.15	Computational Time Analysis	63
3.16	Final Area Lattice After Traversal	63
3.17	Self Organizing Map for Virtual Concepts	65
3.18	Error Histogram For Neural Mapping	65
3.19	Response For Neural Mapping	65
3.20	Error Auto-correlation For Neural Mapping	65
3.21	Input Error Correlation for Neural Mapping	66
4.1	Route Discovery and Path Establishment	80
4.2	Route Maintenance, Recovery and Congestion Control	80
4.3	UAVs:1 (Neural Delays (D_N)	86
4.4	UAVs:5 (Neural Delays (D_N)	86
4.5	UAVs:1 (Link Reliability (L_R))	87

4.6	UAVs:5 (Link Reliability (L_R))	87
4.7	UAVs:1 (Cognitive Delivery Ratio)	87
4.8	UAVs:5 (Cognitive Delivery Ratio)	87
4.9	UAVs:1 (Error Rate (E_R))	87
4.10	UAVs:5 (Error Rate (E_R))	87
4.11	UAVs:1 (Maps Shared)	88
4.12	UAVs:5 (Maps Shared)	88
4.13	UAVs:1 (Throughput(T_H))	88
4.14	UAVs:5 (Throughput(T_H))	88
4.15	Packet Drops	89
4.16	Throughput (bps)	89
4.17	Packet Delays (ms)	89
4.18	Jitters (ms)	89
4.19	Variation : Cognitive Delivery Ratio	90
4.20	Variation : Error Rate (E_R)	90
4.21	Variation : Link Reliability (L_R)	91
4.22	Variation : Maps shared	91
4.23	Variation : Neural Delays (N_D)	91
4.24	Variation : Network Throughput	91
5.1	Cross Layer Design For Service Dissemination	94
5.2	Network Model For Aerial-Ground Ad Hoc Formation	100
5.3	3-Tier Service Neural Oriented Model For UAV Guided Networks	100
5.4	Application Layer	101
5.5	Guider Layer	101
5.6	Collaborative Layer	102
5.7	Decision Layer	102
5.8	Control Layer	103
5.9	Neural Interfaced Layered View	104
5.10	Integration of Service Model With Cooperative Network Framework	108
5.11	Confidence Interval Evaluation of Service Time.	113
5.12	Time Consumption and Memory Requirement Analysis of Service Model.	114
5.13	Confidence Interval Evaluation of Service Time.	116
5.14	Confidence Interval Evaluation of Service Usability.	116
5.15	Network Parameter Analysis For 3-Tier Cooperative Framework Collaboration.	117
5.16	Comparison For Accuracy (%) by Proposed Model vs. Existing Cooperative Framework and Greedy Approach.	117

5.17	Comparison For Identification of Faults by Proposed Model vs. Existing Cooperative Framework and Greedy Approach.	119
5.18	Comparison For Hindrance Cost and Fault-tolerance Offered by Proposed Model vs. Existing Cooperative Framework and Greedy Approach. . . .	119
6.1	Queue Transmission	130
6.2	Operational View of Proposed Scheme	130
6.3	Queue Estimation Comparison	138
6.4	Statistical Analysis of Queue Estimation using Quaternion Kalman Filter	138
6.5	Bandwidth Estimation Comparison	139
6.6	Statistical Analysis of Bandwidth Estimation using Quaternion Kalman Filter	139
6.7	Delay Estimation Comparison	140
6.8	Statistical Analysis of End to End Delay Estimation using Kalman Filter	140
6.9	Position Accuracy vs. Ground Nodes	149
6.10	Processing Time vs. Ground Nodes	149
6.11	Delays vs. UAVs	149
6.12	Network Throughput vs. UAVs	149
6.13	Packet Delivery Ratio (PDR) vs. UAVs	149
6.14	Probability of Guaranteed QoS vs. UAVs	149
7.1	Operational View of Proposed Model	155
7.2	Operational View of Healing Phase	155
7.3	State Wise Functionality of Proposed Self-Healing Neural Model	156
7.4	Healing Function h_f Range And State Categorization	157
7.5	Variation of h_f over 30% Faults	158
7.6	Stabilization Rate and Learning Rate Analysis over Variable Faults . . .	158
7.7	Mean Square Error over Benchmark Training Algorithms	158
7.8	Gradient over Benchmark Training Algorithms	158
7.9	Broadcast Storm Issue	160
7.10	Broadcasts vs. Ground Nodes	168
7.11	PDR vs. Ground Nodes	168
7.12	Link Recovery Time vs. Ground Nodes	168
7.13	Broadcast Overheads vs. Ground Nodes	168
7.14	Redundancy vs. Ground Nodes	169
7.15	Saved Rebroadcasts vs. Failures (%)	169
8.1	Security Pillars For Guided Flying Ad Hoc Network	172

8.2	Network Interaction Model For Cooperative Network Framework	173
8.3	Service-Oriented Security Model For Cognitive Sharing	174
8.4	Teredo Tunneling For Secure Connection Over Network Framework	185
8.5	System Fairness vs. UAVs	191
8.6	Data Delivery Ratio vs. Nodes	191
8.7	Network Throughput vs. Nodes	191
8.8	Average Delays vs. Nodes	191
8.9	Connection Accuracy vs. Nodes	192
8.10	Cognitive Delivery Ratio vs. Nodes	192
8.11	Cognitive Failures vs. Nodes	192
8.12	Fault Identification Time vs. Nodes	192
8.13	Connection Forging vs. Nodes	193
8.14	Cognitive Overheads vs. Nodes	193
8.15	Packet Faults Identified vs. UAVs	193
8.16	Network Sensitivity vs. Nodes	193
8.17	Identification Time vs. Nodes	194
8.18	Classification Time vs. Nodes	194
8.19	Teredo Formation Time vs. Nodes	194
8.20	Network Trigger Time vs. Nodes	194
8.21	System Fairness vs. UAVs	195
8.22	Data Delivery Ratio vs. Nodes	195
8.23	Network Throughput vs. Nodes	195
8.24	Average Delays vs. Nodes	195

List of Tables

Table No.	Title	Page No.
1.1	Comparison Between Ad Hoc Networks	2
2.1	Comparison of Existing UAV Based Frameworks/Models/Approaches . .	28
2.2	Various Nature Inspired Algorithms	32
2.3	Various TCP Variants	33
2.4	Comparison of Existing Service Architecture/Middleware for UAVs Ad Hoc Formations	35
2.5	Comparison of Existing Approaches For Solution To Broadcast Storms .	40
3.1	Parameter Configuration For Cooperative Framework Analysis	60
4.1	OFFRP-Relaying Components	78
4.2	OFFRP-Flash Feedback Components	78
4.3	OFFRP: Fruit Fly Based Packet Format	79
4.4	Routing Protocols Considered For Comparative Analysis	80
4.5	Parameter Configurations For Routing Protocol Analysis	86
5.1	Parameter Configurations For Service Provisioning Analysis	112
5.2	Service Parameter Configurations	113
5.3	Results For Services Operating Over Neural Model	114
5.4	Percentage Enhancement In Cooperative Framework Using Proposed Ser- vice Model	118
6.1	Queue Configurations	131
6.2	Parameter Configurations For QoS Analysis	137
6.3	Queue Rating (X*:-Variable Entities)	146
6.4	Parameter Dominance	146
6.5	Parameter Configurations For UAV Assisted QoS Provisioning	148
7.1	Parameter Configurations For Self-Healing Model	157
7.2	Self-Healing Model Analysis For 10 UAVs	158
8.1	Parameter Configurations For Secure Framework	188
8.2	Percentage Variation of Parameters with Attackers	195

List of Notations

Chapter 3

Θ	Orientation
β	Learning rate over prior and likelihood
η	Network learning rate
ρ	Threshold hamming distance
K_c	Kalman gain
R_{min}	Minimum achievable rate
$T_{rate(m)}$	Measured transfer rate
$T_{rate(o)}$	Observed transfer rate
$\mathbf{w}$	Waypoints over map
$_{H,S}\mathbf{w}_{ui}^L$	Rate of change of local frame
$_{S,L}R_{ij}^H(\bar{w}_{ui})$	Rotational matrix
$\bar{Z}_{ui}(t)$	System state equation

Chapter 4

ACK	Acknowledgement
B_C	Bandwidth
β	Network deviation
CDR	Cognitive delivery ratio
$cwnd$	Congestion window size
D_N	Neural delays
E_R	Error rate
F^N	Fly number
F^S	Food source number
F_R	Fidelity ratio
FF^I	Fruit fly index
RTT	Round trip time
S^E	Seeds
S^I	Smell index
T_H	Throughput

Chapter 5

CH_{co}	Number of channels
N_{CO}	Number of corridors
R_H	Resources available
r_{min}	Minimum achievable rate
C_{state}	Cooperative model state
ϵ''	error
ϵ_r	Error rate during recovery
F_t	Fault tolerance
γ, β	Hinderance constants
H_c	Hinderance cost
P_{state}	Predictable system state
HSR_U^t	Rate of change
S_i^g	Service generated for i^{th} control input
SN_{state}	Service neural state
S_N	Services for N Nodes

Chapter 6

α	Difference in queue rate due to mobility
β	Bandwidth allocation
δ	Pathloss exponent
ϵ	Queue observation error
$\frac{1}{\mu_p}$	Packet arrival rate
γ_s	Mean size of each PU
λ_s	Service distribution
λ_u	Unhandled users
Ω	Vector product matrix
K	Kalman gain
N_{rt}	Number of retransmissions
S_{cf}	Scheduling cost function
$\overline{Q_g}$	UAV network governing equation
${}_G\overline{\mathbf{w}}'$	State governing equation

Chapter 7

c_T	Total connections
cx^t	Faulty connections
h_f	Healing function
L_A	Link availability
L_R	Approximated link reliability
L_S	Link speed
n^t	Number of UAV active
P_S	Parameter sensitivity
R_B	Geographical radio range
RSS	Received Signal Strength
W_m	Average weight

Chapter 8

A_{AGL}	Relative localization accuracy
A_{AL}, A_{GL}	Localization accuracy
A_{CG}, A_{CA}	Active connections
AD_P	Network adaptability
C_{OF}	Corridor failures
C_{OG}	Cognitive transfers
C_{OP}	Corridor privacy
C_{OT}	Total connections
F_{IN}	Fault identification inconsistency
L_C	Network localization
N_{EA}	Number of accurate neural iterations
N_{ES}	Neural sensitivity
N_{IT}	Number of interconnections
N_R	Network repair time
N_{RSP}	Network responsiveness
S_{ID}	Service identification

List of Abbreviations

ABC	Artificial Bee Colony Optimization
ACO	Ant Colony Optimization
AGPS	Assisted Global Positioning System
AODV	Ad Hoc On Demand Distance Vector
AOMDV	Ad Hoc On Demand Multipath Distance Vector
ART2	Adaptive Resonance Theory Type 2
CBR	Constant Bit Rate
CORBA	Common Object Request Broker Architecture
CSMA/CA	Carrier Sense Multiple Access/ Collision Avoidance
DGPS	Differential Global Positioning System
DSDV	Destination Sequenced Distance Vector
DSR	Dynamic Source Routing
FANET	Flying Ad Hoc Network
FFO	Fruit Fly Optimization
GPS	Global Positioning System
HAPS	High Altitude Platform System
IMU	Inertial Measurement Unit
LRS	Local Reference System
MAC	Media Access Control
MANET	Mobile Ad Hoc Network
MAODV	Multicast Ad Hoc Distance Vector
MBNP	Mobile Backbone Network Protocol
MDART	Multipath Dynamic Address Routing
OLSR	Optimized Link State Routing
PSO	Particle Swarm Optimization
PUMA	Protocol For Unified Multicasting Through Announcements
QUBO	Quadratic Unconstrained Binary Optimization
SMAVNET	Swarming Micro Air Vehicle Network
SOA	Service Oriented Architecture
SOM	Self Organizing Map
TORA	Temporarily Ordered Routing Algorithm
UAV	Unmanned Aerial Vehicle
VANET	Vehicular Ad Hoc Network

Chapter 1

Introduction

Cooperative communication has extended the applications of existing ad hoc networks. Ad hoc networks are the infrastructure-less networks that operate without any centralized device. A collaborative utilization of resources from two or more networks can provide a strong backbone for the implementation of the new set of services aiming at improved communication. Mutually operating networks can provide non-complex solutions for border surveillance, disaster monitoring, traffic monitoring, cooperative network formations, and home area networks. Collaborative ad hoc formations involve network nodes with varied type, make, and configurations. Using aerial vehicles as relays, and formation of ad hoc network between these vehicles is one of the novel ideology that enhances the capacity of existing networks.

1.1 Background

Unmanned aerial vehicles (UAVs) have gained lots of potential over the years, providing a vast range of applications in existing networks. UAVs have the ability to cooperate, and can fly autonomously or can be operated without human intervention, thus are versatile and flexible in implementation. One of the possible collaborations of these vehicles is with ground ad hoc networks. Ad hoc networks are low cost infrastructure networks that operate on the ideology of forming intermittent networks. Traditional ad hoc networks do not require any centralized device for data forwarding. All the nodes themselves act as transmitter, receiver, and a router. However, with changing scenarios of communications, the existing ad hoc formations have certain limitations that prevent them from being used in dynamic conditions. Therefore, using UAVs as an intermediate node in existing ad hoc networks can resolve complex tasks efficiently. These complex tasks include cooperative

search, object tracking, data acquisition, and data analysis. UAVs can be used as a single entity or in group formations. A single UAV system exists in literature, and has been successfully coordinated with existing ad hoc formations. But, single UAV coordinated networks face a major issue of scalability, and can provide only limited surveillance. Thus, it becomes important to unite multiple UAVs together to form a self-sufficient network in the air that can form a collaborative system with existing ground networks. These aerial nodes can resolve the issues such as coverage, scalability, and efficiency of existing networks. The multi-UAV system can easily incorporate with existing ad hoc networks to form a guidance system that is capable of providing a look-ahead information to ground nodes. This information can be used to traverse the areas with high risk factors, and are beyond the reach of humans. Thus, UAVs form an aerial swarm that tracks user stations on the ground, and coordinate with them to take surveillance to the next level. UAV ad hoc formations are also termed as “Flying Ad Hoc Networks (FANETs)”. FANETs are the sub-category of existing Mobile Ad Hoc Networks (MANETs) and Vehicular Ad Hoc Networks (VANETs) with difference in type of nodes, configurations, topology, and high mobility rate. A comparison between MANETs, VANETs, and FANETs as

Table 1.1: Comparison Between Ad Hoc Networks

PARAMETER	MANETs	VANETs	FANETs
Mobility Model	Random	Regular	Special Random Models
Node Density	Low	High	Very low
Topology Change	Slow	Fast	Fast
Radio Propagation Model	LOS not available	LOS not available in all case	LOS available
Power Consumption	Energy Efficient	Not Needed	Type Dependent
Network life time	Low	High	High
Computational Power	Limited	High	High
Localization	GPS	GPS, AGPS, DGPS	GPS, AGPS, DGPS, IMU
Localization/Estimation	-	Local Sensors	Global and Local Sensors

provided by Bekmezci *et al.* [1] is given in Table 1.1. Multi-UAV networks can enhance the network performance to much extent in comparison with single UAV system. This is due to provisioning of higher number of interconnections between the UAVs and the ground nodes. The network formations between the aerial vehicles and the ground nodes is categorized into three formations, namely, UAV Ad Hoc Networks, Ground Ad Hoc Networks, and Cooperative Network Formations/Guided Ad Hoc Formations, as shown in Figure 1.1.

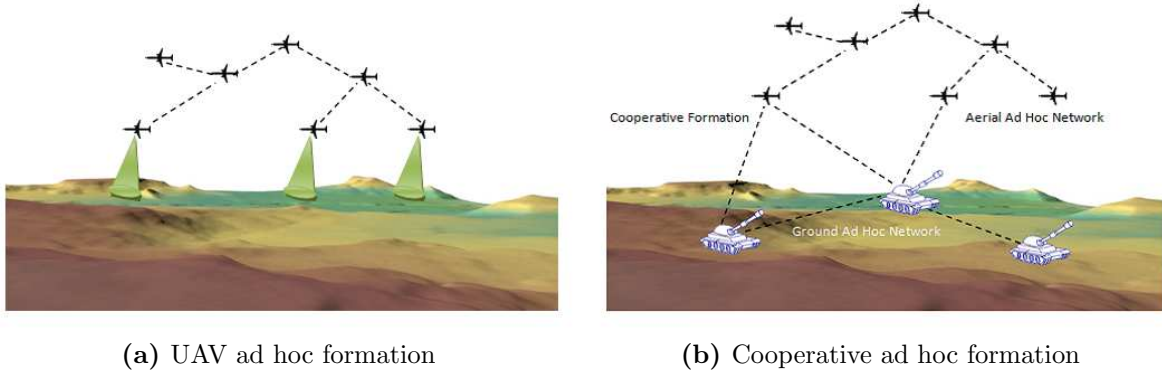


Figure 1.1: UAVs Oriented Network

1.1.1 UAV Ad Hoc Networks

Multi-UAV system can form various types of networks depending upon the requirements and deploying environment. These types of networks use UAVs as key role players in the network formation. These networks are formed considering the configurations and features of UAVs. Thus, these types of networks are also termed as UAVs Oriented Network. The multi-uav system can be a simple collection of UAVs with autonomous capabilities or controllable by remote pilots. Remote pilots are those which uses a predatory cockpit to control the maneuvers of the UAVs from a remote location. Such combination of the remote pilots, cockpit and UAVs are termed as an unmanned aerial system (UAS). However, the ad hoc formations between the UAVs do not rely on the UAS system rather they use the autonomous capability of the UAVs. Temporary connectivity between these vehicles not only enhance the flexibility in network formations, but also allows extensive control in case of varied environmental conditions and failures. UAVs ad hoc formations eliminate the complete dependency of all UAVs over satellite, rather a single UAV serves as a pivot between the infrastructure (satellite) and the other aerial nodes. Further, this type of network formations utilizes the privilege of Local Reference System (LRS) rather than only depending upon the Global Positioning System (GPS). Such ad hoc formations can play a pivotal role in existing networks with provisioning of service enhancement with prolonged connectivity.

1.1.2 Ground Ad Hoc Networks

These involve mobile ad hoc units, vehicular ad hoc units, or even sensor nodes. With flat grid topological formations, these networks provide vast range of applications. Using least infrastructure, the nodes belonging to these networks can form intermittent connections. These networks have seen a lot of development in the past few years. But, still there are lots of limitations in complete utility of these networks; such as requirement of wider susceptibility, and error free connectivity. Enhanced coordination and facilitated formation can extend the applications of these networks to a larger extent.

1.1.3 Cooperative Network Formations

Often termed as “Collaborative Flying Ad Hoc Networks” or “Hybrid Ad Hoc Networks”, these are formed using two variedly operating ad hoc units. In this thesis work, these two ad hoc units are the ground ad hoc networks and the aerial ad hoc networks. Coordination between two simultaneously operating networks can increase the scalability, and reduce the network cost for surveillance, navigation, and reconnaissance. Cooperative network formation can readily solve the complex tasks such as cooperative search, tracking, and data acquisition.

1.2 Multi-UAV Guided Ground Ad Hoc Formations

UAVs operating in collaboration offer a wide range of applications in various areas such as border surveillance, inland-security, and formation of guidance units. The guidance units include multiple ad hoc networks that operate simultaneously in cooperation with each other. These guided ad hoc networks allow formation of distributed networks. Multi-UAV guided ad hoc network are specialized ad hoc network that aims at provision for robust map transfers which provides look ahead information. Guided ad hoc network allows efficient and coordinated planning amongst network nodes to achieve complex tasks [2] [3]. Multiple nodes, operating using common algorithmic approach and frame-

works, have greater susceptibility and adaptability in terms of network conjunctures for prediction of geographical environment. This also allows estimated evaluation of the network parameters which increases performance.

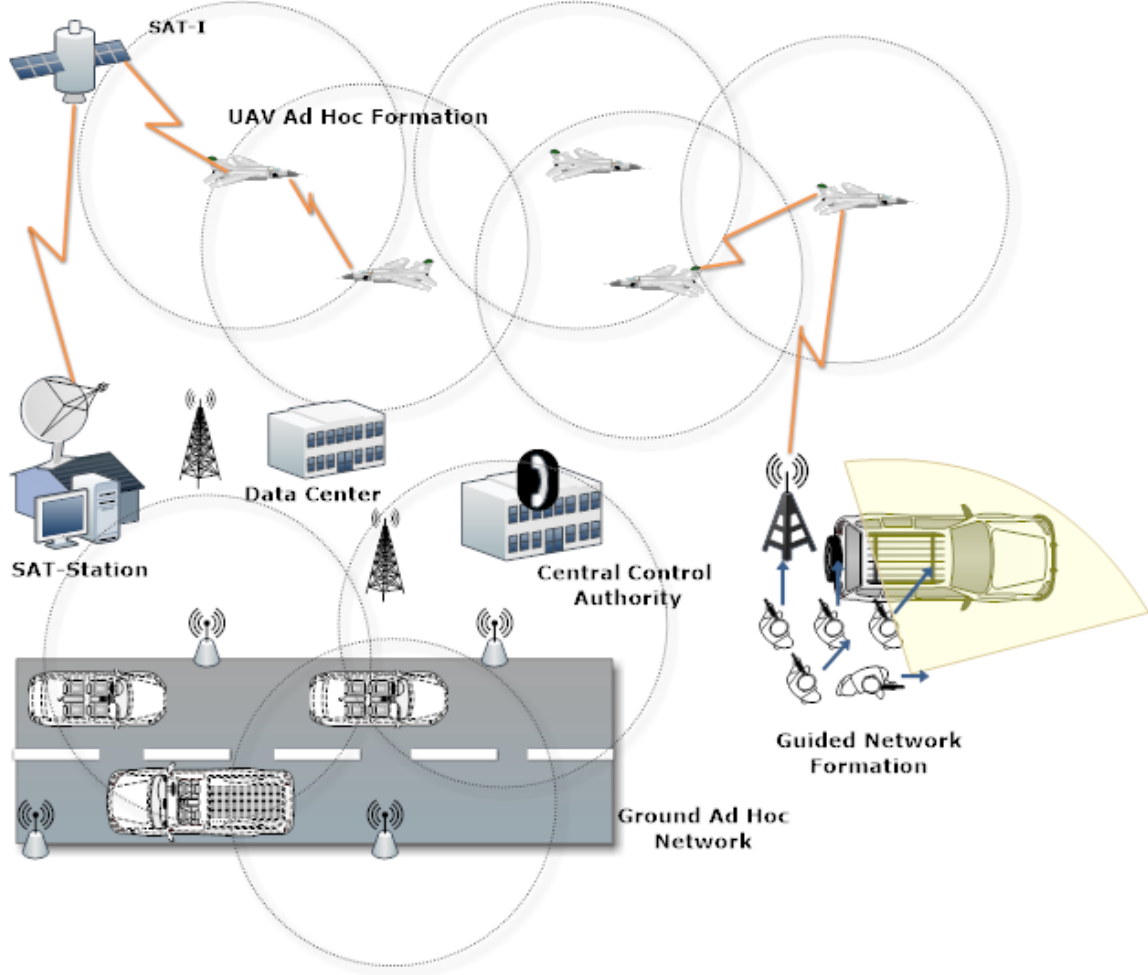


Figure 1.2: Network Architecture For UAV Guided Ad Hoc Formations

1.2.1 Network Components and Architecture

Multi-UAV guided system aims at provisioning services between the ground and the aerial nodes, allowing the formation of an efficient guidance system. A sample network architecture showing application and formation of guided ad hoc networks is shown in Figure 1.2. This guidance system comprises of following components:

- **Ground Nodes:** Ground units comprise of varied types of nodes be it the ground stations, personal devices, or the vehicles. These nodes are GPS enabled devices

that allow efficient tracking and coordination. Provision of Assisted Global Positioning System (AGPS) and Differential Global Positioning System (DGPS) helps in improved coordination and localization. These nodes are also equipped with onboard processing devices that are required to perform timely computations. External battery is required for the operations of onboard devices. Onboard sensors for directional, speed, location, and collision control are also required. For relaying, a communication setup is also implanted on the nodes that follows the standards prescribed for communication in hybrid environments.

- **Aerial Nodes:** Aerial nodes usually account for similar configurations. However, their make and type may vary. Aerial vehicles have high end facilities that can support communication with ground stations, satellite, satellite station, and the ground nodes. A high end battery is required for continuous running of these communication devices. Local and global sensors such as proximity, GPS, AGPS, DGPS, Inertial Measurement Unit (IMU) are a key part of an aerial node. Based on the specific requirements, customized sensors and devices can be implanted on aerial node.
- **Ground Station:** This accounts for control and regulation of the network formation between the aerial nodes and the ground nodes. This is also responsible for handling the satellite communication, and performs command and control operations. Ground stations are also termed as satellite-control stations. However, their function is not limited to satellite control only. This station also maintains the updates regarding the mission and tasks assigned to UAVs/aerial nodes. Dynamic configurations, periodic updates, and node failures are handled by the ground stations.
- **Data Center:** These are explicitly remotely located data warehouses. These are not directly related to formation of UAV guided networks. However, these centers act as a common warehouse for all types of data collected during a particular task. Mostly, UAV oriented networks require high end data analysis. These computations cannot be performed using onboard computational units. Thus, in these scenarios, utility of data centers come handy, and can provide analysis of huge set of data within no time.

- **Central Control Authority:** Central control authority is responsible for the regulation of all the ground units. It does not directly engage with the aerial nodes, rather it interacts with the ground station for any particular command and control operations. In a ground unit hierarchy, it is the top most node. Despite that ground nodes can directly interact with the aerial nodes, central control authority being a highly crucial node does not form any direct communication with aerial nodes.

1.2.2 Design Issues

UAV ad hoc networks as stated can either operate individually, or can operate in integration with other networks. A UAV oriented networks comprise of aerial vehicles that have tendency to fly autonomously. Data in the whole network is subjected to different transmission band selected on the basis of requirement and resources available [4] [5] [6]. Such network though simple, yet are complicated in design and implementation. Some of the key issues that are to be taken care of while configuration and implementation of these networks are explained below:

- **Adaptability:** The network architecture should be capable enough to withstand with the dynamic changes that may be encountered while transmission. A network should be able to adapt to changes with replanning and redeployment.
- **Scalability:** Any form of network is meant to be highly scalable. This reduces the cost involved in replanning and reconfiguring of networks. A network with ability to scale itself in terms of nodes, resources, and other design constraints is efficient, and can provide a vast range of applications.
- **Bandwidth Requirement:** Another design consideration to be handled efficiently is the selection of appropriate band for transmission. Network bandwidth should be selected appropriately as it improves the tradeoff which exists between the energy efficiency and the bandwidth efficiency [7]. A network with the requirement of high transfer rate with low bandwidth provisioning is of no use. Such issues should be resolved pre hand while initial configurations of the network. Also, the onboard processors and sensors on the aerial nodes should be able to support variable bands

while transmissions.

- **Latency:** It is almost impossible to aloof all types of network delays. However, network latency can be minimized and controlled to go beyond certain limits. A delay in such networks can cause devastating effects. Thus, network architectures should possess key features of handling network latency and unpredicted delays.
- **Reliability:** A network while operations may be prone to various network attacks. Such activities can hinder the ongoing process as well as can degrade the network performance. A model used for formation of an aerial network should be fault-tolerant, robust and reliable. Reliability can be determined by various parameters over which the particular transmission task depends. Using various delay-tolerant features, recovery mechanisms, and network pre-analysis can provide a highly reliable network.

1.2.3 Characteristics

UAV oriented networks can either operate independently or can perform in coordination with each other. Autonomous as well as controlled network formations with these aerial vehicles provide vast range of characteristics. The major characteristics of such networks are explained below:

- **Coordinated Formations:** UAVs have the capability of both autonomous operations as well as collaborative network formations. UAV oriented networks provide better coordination between the nodes. With the efficient availability of the bands, UAVs can easily coordinate with the ground stations and at the same time can efficiently handle the satellite based communication to form a highly stable coordinated network.
- **Controlled Mobility:** Maneuvers of the aerial vehicles can be controlled efficiently during any time of the transmission. With controllable network formations, these vehicles can provide a vast range of applications in both civilian as well as military activities. Also, the pre-configurations of the tasks in such nodes can easily be uploaded using a series of controllable algorithms. Waypoints for movement can

be pre-decided, or can be dynamically configured depending upon the network dynamics or environmental conditions.

- **Ease of Upgrades and Deploying:** Such networks are easily deployable. Non-platform dependency of these networks make them easy to upgrade and maintain. Nodes can be configured at any moment during the network operations. These networks are easy to deploy as topology is required in case of particular missions. The only major requirement of these networks is generation of non-overlapping waypoints.
- **Improved Connectivity:** Unlike the existing ad network formations such as MANETs, VANETs, and WSNs, flying networks are less susceptible to connection loss due to line of sight. Also, the band range applicable on such networks is of extreme quality that provides less hindered connectivity between the nodes. Further, a wide channel is available for corridor connectivity which also improves the connectivity between the nodes of such networks.
- **High Transfer Rate:** With improved connections between the nodes, such networks are capable of providing high bandwidth which, in turn, due to lesser delays is capable of providing high transfer rate. Lower delays make these networks efficient with high performance and delivery ratio.
- **Effective Security:** With a wide range of bandwidth available, it is easy to spread the spectrum without reallocation of channels. Higher bandwidth provides a wide range of security applications to be implemented efficiently. Also, corridor privacy is easy to maintain in such networks.
- **Sustainability:** Sustainability refers to the network survivability in adverse conditions. UAV oriented networks are highly dynamic. With certain improvements in deployment algorithms, these networks can easily be made sustainable to dynamic environments. Also, these networks are highly sustainable in terms of applications. A single network with multiple UAVs can be used for different applications without reconfigurations of the whole network.

1.2.4 Challenges

UAVs assisted networks are a hybrid version of existing ad hoc networks. Thus, major network challenges in these networks are similar to existing networks. These challenges involve various network problems that are to be taken into account while deploying. Some of the major challenges in the formation of coordinated ad hoc networks are:

- **Topological Arrangement:** One of the basic requirement of the ad hoc networks is the topology arrangement. Selection of efficient neighbours and arrangement of nodes in an efficient manner to enhance the performance of the networks is one of the major challenges in these networks. Recovery in topology during uncertain failures of some nodes, rearrangement for a particular missions, energy efficient topology formations are some other key aspects that are to be taken care of, while forming such networks [8] [9].
- **Transmission Channel:** Aerial networks are capable of providing line of sight in most of the cases. Having multiple nodes with line of sight facility makes it less complicated for selection of appropriate band for transmission between any two network nodes at a particular time. Also, selection of band for communication is to be done efficiently to provide higher transfer rates [10].
- **Link-Layer Issues:** Distributed operations, synchronization, and exposure to hidden terminals are some of the key issues to be handled while applying these networks. Delays, resource reservation and allocations are the other set of challenges that can hinder their performance [11] [12]. Media access management and error control are to be handled efficiently in these networks [13]. Existing ad hoc based MAC protocols cannot handle the high end computing involved in such networks. Therefore, new MAC protocols are required that can offer a wide channel selection with enhanced security.
- **Routing:** Routing is the one issue that remains active in all types of networks. An efficient routing provides a high delivery ratio and low delays during network operations [14] [15]. For UAV networks, more robust and fault-tolerant routing

protocols are required which can provide minimum delay during route selection, efficient route reconfigurations, fast recovery, and higher provisioning for quality of services to end users [16] [17].

- **Energy Management and Power Control:** UAVs have a disadvantage of limited available energy. Thus, efficient designs and configurations are required that consumes lesser energy and provides efficient power control. Device configurations and on-board units can be efficiently designed to lower the consumption of on-board energy units [18] [19].
- **Channel Security:** Node hijacking, man in the middle, masquerade attack, black holes, sink holes, grey holes, nodes coagulation, denial of service, node impersonation, energy wastage, buffer overflows, and heap managements are the some of the possible security threats in such networks. Efficient strategies are required to handle any misuse of such network formations [20].
- **Deployment and Maintenance:** Another key challenge in network formation with the aerial vehicles is the deployment and maintenance. The network should be easy to deploy and configure during network operations. Cost and time for deployment and maintenance should be extremely low.

1.2.5 Applications of UAV Guided Ad Hoc Networks

UAV Guided ad hoc networks have enhanced the utilization of existing ad hoc units. Limitations of existing networks can easily be overcome by utilization of aerial nodes. Apart from capacity enhancement, UAVs have gained a pivotal role in some of the crucial applications of existing networks [21]. These network formation can be used for improved surveillance, reconnaissance, navigation, tracking, and data acquisition. Description of some of the important applications of UAV guided ad hoc formations is provided below:

- **Usability in military expeditions and classified activities:** UAV guided networks can be used in military expeditions. These can be used to form a self-reliant system that can be deployed in areas that may cause loss of life. Further, border

and homeland security can easily be attained by the formation of these guided networks. These networks can provide an aid in enemy tracking, and user station identification [22] [23].

- Emergency operations such as disaster management: UAV guided networks can attain the features of cellular networks in case of non-operations of existing infrastructure particularly during a disaster like scenarios. These networks can be used to set up temporary fully-functional network with provision of complete services as that of static infrastructure-based networks.
- Patient monitoring by formation of aerial ambulance: UAV guided system can be used to form an aerial ambulance system. Driven by the GPS and mobile devices, these network formation can provide immediate remedies to patients.
- Ecological danger detection: Ecological dangers such as forest fires and vegetation losses can be notified, and data can be acquired for analysis. UAV guided system can form a monitoring system that can prevent such losses by identifying the cause of these incidents which otherwise would not have been possible.
- Exploring cyclone evolution: Cyclones are one the highly devastating natural calamity. However, their point of origin and path can be traced using ad hoc formation between the UAVs. A continuous monitoring of origin from a short distance can easily be attained using these vehicles.
- Exploring areas untraceable by humans: A large part of the earth is still unexplored, either due to geographical barriers or possible loss of life. These vehicles can be utilized to form an aerial mesh that can explore these areas that might have not been possible by humans alone [24].
- High altitude platform systems: This is one the important utilization of the aerial ad hoc networks. These networks can be used for internet facility in remote areas. Aerial mesh can act as access points providing network connectivity for multimedia support in areas without ground stations, or difficult geographical terrains.
- River banks analysis: Most of the rivers are being polluted due to unnoticed sources.

A network formation between the aerial nodes and the ground nodes can be used for data acquisition along the river banks to identify the source of pollution.

- On road traffic analysis: One of the major issues faced by the modern cities is the traffic. UAVs can provide a better, and low-complex solutions for on road traffic regulations. UAV networks can integrate with ground sensors to monitor the driver activities.

1.3 Thesis Organization

The thesis has been organized into 9 chapters. A brief outline is given below:

Chapter 1: Introduction

Beginning with introduction to ad hoc networks, this chapter provides an overview of ground ad hoc networks, UAVs ad hoc formations, and cooperative formations between the ground ad hoc networks and the aerial (UAVs) ad hoc networks. This chapter also provides an overview of various constraints in designing these type of cooperative formations and challenges involved in its deployment.

Chapter 2: Literature Review

This chapter provides a comprehensive survey of the existing UAVs oriented network formations. This chapter covers the utility of such ad hoc formations based on the previously developed work. This chapter classifies the ad hoc formations using aerial vehicles, and presents the comparative study between various existing cooperative frameworks and network models.

Chapter 3: Cooperative Network Framework

This chapter presents the cooperative framework developed for cognitive map transfers between the multiple unmanned aerial vehicles (UAVs) and the ground ad hoc networks. Formation of a guided network, in which UAVs are assigned a task of non-redundant search of user stations on the ground, is the key motive of this framework.

Chapter 4: Routing and Cooperative Relaying Over UAV Guided Networks

In this chapter, a new routing protocol for relaying over cooperative network framework is presented. This protocol provides a support for transmission between the ground nodes

and the aerial nodes in the ad hoc mode. The protocol is developed using the cooperative network framework as its backbone. This protocol is termed as “OFFRP: Optimized Fruit Fly Routing Protocol”.

Chapter 5: Service Provisioning Over UAVs Coordinated Networks

This chapter presents the facility of service provisioning using the proposed cooperative framework over the UAVs coordinated networks. This chapter includes an overview of the new middleware for UAVs coordinated network. Then, it presents a service dissemination strategy using cross layer formation over this middleware. In the later part, a fully functional 3-tier neural based service model developed on the backbone of this middleware is presented.

Chapter 6: QoS Scheduling For End Users In Ground and UAV Ad Hoc Networks

This chapter proposes a new Quality of Service (QoS) provisioning over ground ad hoc networks-MANETs, and the UAVs ad hoc networks. The proposed QoS scheduling is developed in two parts. The first part presents an estimation-based queue scheduling to provide QoS to end users in mobile ad hoc networks. Second part utilizes this estimation strategy to form an UAV assisted queue scheduling approach.

Chapter 7: Stabilization and Topological Broadcasting Over Networked UAVs

In the first part, this chapter presents the novel strategies for stabilization against failures over networked UAVs. In the next part, broadcast storm issue over the coordinated network formation between the ground ad hoc and the aerial ad hoc units is considered.

Chapter 8: Security Enhancement of Cooperative Network Framework

In this chapter, security issues over cooperative network framework is handled using a Teredo Tunneling approach. The proposed approach provides secure transmission between the UAVs and the ground nodes. The network model uses prediction and estimation procedures to identify its states, and forms a decision system in case of intrusions.

Chapter 9: Conclusion and Future Scope

The thesis concludes with this chapter with a brief overview of the proposed framework, routing protocol, service provisioning over cooperative framework, QoS scheduling, stabilization, topological broadcasting, and security enhancement using Teredo Tunneling approach. Open issues and future perspectives are also highlighted in this section.

Chapter 2

Literature Review[†]

Multi-UAV system comprises of aerial vehicles autonomously collaborating with each other to form an aerial swarm. These vehicles can form an intermittent network in the air that can relay with another unit operating on the ground. The two types of network formations considered includes the UAV ad hoc formation and the ground ad hoc formations. The UAV ad hoc formations are also termed as the “Flying ad Hoc Networks” as coined by Bekmezci *et al.* [1]. A collaboration between these two networks can form a self-sustaining guidance system that provides a vast range of applications. The work presented in this thesis focuses on network framework formation for UAV-guided networks, routing over the cooperative framework, service provisioning, Quality of service (QoS), stabilization of framework in case of failures, topological broadcasting, and service security over the proposed framework. Following the similar order, a brief summary of recent literature highlighting these issues is presented in this section.

2.1 Taxonomy of UAV Networks

UAV networks have gained a lot of success in its implementation over the years. UAVs can be used to form different networks based upon the scenarios and requirements. There are several design issues in implementation of UAV networks such as latency, scalability and adaptability. Latency can prove to be fatal as that can lead to network failures [25] [26]. The mean packet delay is very crucial for such type of mission critical networks. Number of nodes in a network plays a vital role for large coverage, either in the form of distance or

[†]Vishal Sharma and Rajesh Kumar, “Cooperative Frameworks and Network Models For UAV Ad Hoc Networks: A Survey”, Concurrency and Computation: Practice and Experience, Wiley- SCIE Indexed,[Under Review], [I.F. 0.997].

more node traversals [27] [28]. Scalability and adaptability defines the limits of a network up to which it can extend and adapt the dynamic changes [29] [30] [31]. In case of UAVs, environment is the key role player as it affects the trajectories of vehicles, and thus, can affect the parameters like latency and bandwidth. On the basis of formations, UAV network can be classified into two categories, namely, single-UAV system and multi-UAV system, as shown in Figure 2.1.

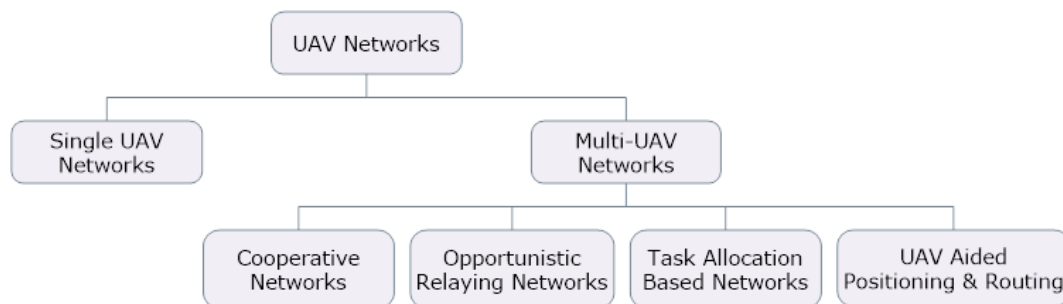


Figure 2.1: Taxonomy of UAV Networks

2.2 Literature Classification

Networking using aerial vehicles is relatively a newer field as compared to other ad hoc networks. Forming a collaborative network using these aerial vehicles have gained a lot of popularity, and many researchers have started focusing on utilization of these networks to attain various applications. The literature available can be broadly classified into the taxonomy provided in Figure 2.2

2.2.1 Single UAV System

Single UAV system started with complete focus on navigation and control for monitoring remote targets. The main aim of such networks was to increase the range and depth upto which it is not possible to perform human controlled maneuverabilities. Also, in case of unknown geographical territories and flight conditions, airborne vehicles are required that can prevent any loss of pilots, thus leading to the formation of a single UAV system. Such systems comprise of an aerial vehicle that requires human intervention to perform

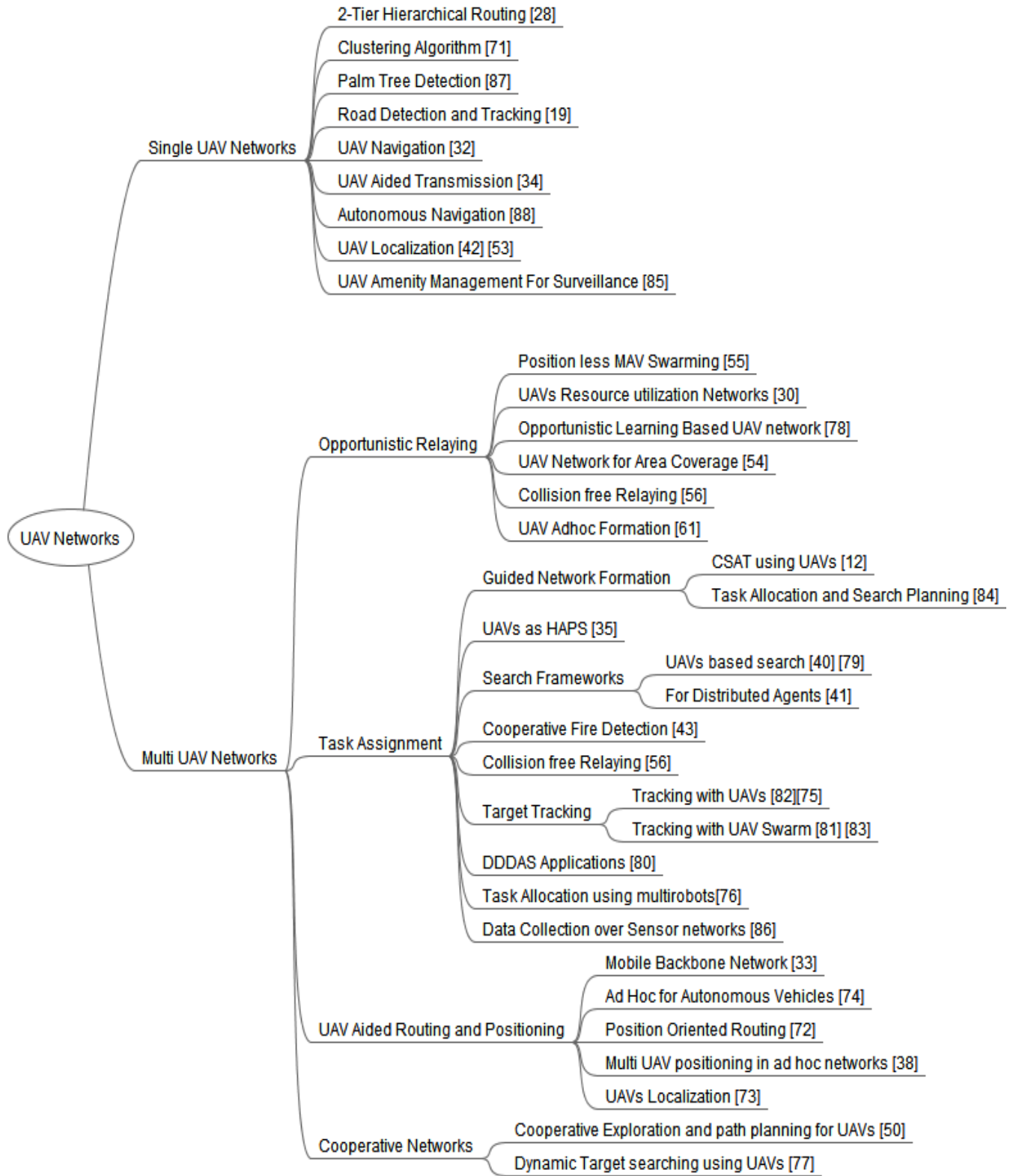


Figure 2.2: Overview of Literature Based on Network Taxonomy

tracking and searching. Nowadays, their applications have been extended with the design of drones which have improved payload capacity as compared to simple monitoring vehicles. It is important to clearly understand the difference between multi and single UAV system.

Doitsidis *et al.* [32] has proposed a framework based on fuzzy logic, which is capable of handling navigation using single UAV. The framework used six degrees of freedom based non-linear trajectory control aerial vehicle using Matlab. The authors proposed usage of two different fuzzy controllers; one for altitude, and another for latitude and longitude. Rubin *et al.* [33] has proposed a different strategy based upon the backbone networks. The authors proposed to use UAVs as network backbones for mobile data transfer in cluster based networks. The network was based on a spatial power control algorithm that used CSMA/CA based protocols. To allow robust routing in such networks, a mobile backbone based network protocol (MBNP) has been implemented. Palazzi *et al.* [34] has proposed a model based upon usage of High Altitude Platforms (HAPS) for urban area monitoring with HAPS/UAVs acting as satellite. In urban areas, it is difficult to get exact localization based on GPS alone. The authors proposed that usage of HAPS/UAVs as hot spots can allow usage of TCP (Transmission Control Protocol) for data sharing in urban areas.

In extension to their work, Palazzi and Roseti *et al.* [35] enhanced transport layer capability by developing an integrated environment between HAPS and satellite communication. Multiple HAPS/UAVs were also highlighted in their extended work that laid the potential growth of multi UAV system. Authors used Adaptive Bandwidth Share Estimator for evaluation of eligible rates. Gu *et al.* [28] has proposed an intelligent routing scheme using single UAV as broadcasting unit, dividing the network into a layered structure based on Hierarchical State Routing (HSR). A 2-level architecture is formed to allow packet transmission through direct wireless links. Authors formulated intelligent source routing with HSR that was capable of handling delays and latency variants that might hinder the performance of UAV based networks.

2.2.2 Multi UAV System

Multi UAV systems are a complex network structures that require robust and efficient network planning before implementation. Such systems are capable of handling complex tasks, but require rigorous and highly dynamic and efficient architectures to handle various issues in multiple UAV usage. This complexity varies from path planning to task allocation amongst the aerial vehicles [36] [37]. A multi-UAV network can be described as a network comprising of n number of UAVs as nodes with wireless links between them as edges so that there is no interference, and each node is aware of position and strategy of another node with the ability to take dynamic decision regarding trajectory and task assignments.

Chandler *et al.* [38] has discussed the complexity in the multi-UAVs cooperative control. The authors discuss the complexity issues in cooperative control of autonomous aerial vehicles. Flow and auction algorithms are used to evaluate the performance of such multiple systems. Several issues regarding cooperative maneuverabilities of UAVs have been addressed and discussed based on simulation results.

Flint *et al.* [39] has formulated a problem for a generation of trajectory for simultaneous operations using multiple autonomous vehicles. Dynamic programming is used to solve this cooperative problem. The authors realized optimal trajectory planning for UAVs using intelligent and dynamic programming concepts, and compared it with Zamboni search technique. A decision model is created and decision problem is solved using a conditional probabilistic approach based on number of targets searched and remaining points. Vincent and Rubin [40] have proposed a framework for cooperative UAVs based on swarm intelligence. The authors designed a reliable and adaptive system that can handle the search task mutually amongst ‘ n ’ number of aerial vehicles. Framework performs coordinated flight maneuvers, and improves sensing capabilities of aerial swarm. It is a five paradigm based model based on detection probability, detection time (estimated), number of UAVs, control information sharing, and combined trajectory plan.

The literature discussed in this section formulated the basis for usage of multi UAVs for various operations and applications. Many research proposals for cooperation amongst

UAVs have been proposed. In the next section, these multi-UAV networks have been further discussed in detail, and classified on the basis of cooperative formations, opportunistic relaying, task assignment, and UAV aided routing.

2.2.2.1 Cooperative Networks

A cooperative network is a specialized graph in which each node performs with respect to functionality of another node. Here, the term cooperative denotes communal operability of UAVs to perform complex tasks. A cooperative network can be a static network of nodes with pre-defined functionality, or can be a dynamic network in which node behaviour is unknown, and network changes on the basis of various parameters that affect its performance. Thus, UAVs in cooperative mode can similarly form a static or dynamic network either forming a fixed topology network or on basis of ad hoc formation. Several network models and architectures exist in the literature that have utilized the cooperative nature of UAVs to achieve tedious tasks.

Polycarpou *et al.* [41] has proposed a framework for dispersed agents to perform cooperative exploration. The distributed agents used in their research were vehicles with controllable maneuverabilities. The authors proposed cooperative pattern for specialized search missions. Inner and outer loop structures are formed that provide guidance and control to aerial vehicles. The trajectory of vehicles is controlled by using learning and decision making component that operates in the outer loop of the framework. Bellur *et al.* [42] has proposed a network for self-governing vehicles. Their work aims at the formation of ad hoc network using multi-UAVs. The authors initially discussed the causes of retraction of UAVs from network topology, and then proposed a solution by formation of intra and inter teams with multi-UAVs. IPV6 router configuration is used on each UAV node, and testing is performed over TCP using netperf tool.

Merino *et al.* [43] has proposed a framework for unmanned vehicles that is able to detect fire. The UAVs used in this framework are heterogenous in nature. The proposed framework is the integration of infrared sensors and data fusion by fetching values from computer aided modules. The use of UAVs eradicates the functional complexity due to

various false alarms. Richards *et al.* [44] has presented a genetic programming based cooperative strategy for a team of UAVs. The model has capability of performing search and track operation in heuristic environment. The model uses tree based structure to form a search map for UAVs. As the technique is based on genetic programming, the authors computed a fitness function for successful and unsuccessful missions.

How *et al.* [45] has discussed rapidly growing independence of UAVs. The authors determined that the increasing autonomous behaviour of UAVs suggested two important factors regarding cooperative search and tracking operations, namely, situational awareness and conflict free task assignment. These operations are coordinated using dispensed sensors. The major objective of their framework was to limit the coordination required between the UAVs and user agents. The authors used Bayesian framework for planning of UAVs in cooperative module. Capitan *et al.* [46] has proposed a non-centric filter that is able to perform cooperative tracking. The authors analyzed the filter over disaster monitoring scenarios. The filter is highly scalable and vigorous to failures in communication set up. The data fusion proposed in the framework is based on Bayesian probabilistic model. The Bayesian model is defined over Distributed Kalman Filter that divides the Bayesian set up into prior, updated and likelihood states.

Muller [47] has studied multi-UAV network as a flying ad hoc network. The author aimed at determining user stations using Mini Aerial Vehicles (MAVs) which are controlled by the base stations. The approach is based on the swarm behavior of MAVs. The methodology used is based on neural implementation. Wang *et al.* [48] has proposed a technique based on binary optimization for improving relaying capabilities of UAV nodes in cooperative ad hoc formation. The authors have discussed a UAV problem similar to the mobile facility problem. The authors recognized this problem as facility placement problem for optimized covering and improved efficiency of UAV networks.

Bekmezci *et al.* [1] [25] and Sahingoz [49] have studied cooperative ad hoc networks independently as Flying ad hoc networks. Bekmezci *et al.* has discussed the highlights of the aerial ad hoc formation with the pre-existing MANETs and VANETs structure. Arising from UAV swarms, a newer network in the form of aerial ad hoc has been introduced, with detailed description of its formation, operability and issues. The authors have sur-

veyed layer by layer formation of this network and have also identified various test beds and simulators for their implementation and testing. In extension to it, Sahingoz [49] further discusses the various provocations and concepts in flying network models. The author has also presented an initial virtual layout of network model for the aerial ad hoc formation. Various issues like scalability, survivability, adaptability, and latency in data transfer have also been discussed.

Liu *et al.* [50] has proposed a cooperative scouting for multi-UAVs with ability to form paths based on their organization in the network. A cognitive map is built using Dempster-Shafer theory. These maps helps in identification of soft and hard targets in the battle-fields. The authors have also discussed the formation of cognitive map based upon the image sensor data that will help in generation of Bayesian maps. Miles *et al.* [51] has proposed a methodology for the determination of trajectory for estimating the position of guiding signals in MANETs. Authors formulated a pseudo-problem for localization of beacons. The estimation process is based upon the position identification of beacons and node states during transmission in the network. The range of states varies from initial to localization covering estimations and indeterministic states. This formation allows small robots to carry out adjustments in their movements to maintain network connectivity for enhanced transmission.

Another enhancement in UAV estimated network is proposed by Benini *et al.* [52] in their Kalman Filter based UAV indoor localizations. Authors operated mini-UAV in ultra-wide band and used inertial measurement unit and performed localization using extended Kalman Filter to provide values closest to exactness. This produced a high fidelity in UAV localizations. Visual odometry proposed by the authors can be used for identifying the potential landing sites for UAVs.

Eroglu and Guray [53] have proposed a localization algorithm for UAV based on terrain type using binary searching. Terrain navigation and digital elevation model are used for analysis of localization in UAVs. The authors initially discuss the basic terrain reference model and used standard altimeters for its analysis. Various terrain reference models available have also been discussed in their work. Finally, binary searching is applied to a trajectory control system of UAVs, and an estimated path of UAV flight is computed.

Schleich *et al.* [54] has proposed new paradigms for designing intelligent algorithms for multi-UAVs. These include speed, integrity and debilitation of UAVs. Authors have proved a tradeoff between network connectivity and area coverage by UAV team. Cell based division is performed for estimating area scanning, and speed is also defined on a time constraint for area traversal. Based on the speed and traversal time, the integrity of the network is computed based on the fairness that depends on the cell ratio computed over a number of cells remaining and last visited.

2.2.2.2 Opportunistic Relaying

Relaying allows formation of temporal networks that can sustain sudden failures. In multi-UAV networks, relaying is not similar to that in mobile ad hoc networks or vehicular ad hoc networks. This is due to reason that in multiple UAV network, data forwarding is not the only task, various other issues are also involved primarily the cooperative formations. For this, rather than focussing on relaying, opportunistic relaying is required. This allows enhanced utilization of network parameters and resources. This carpetbagger type of network allows formation of fault tolerant aerial relaying networks.

Hauert *et al.* [55] has proposed a UAV swarm for ad hoc relaying using micro aerial vehicles. Authors termed this type of UAV ad hoc as SMAVNET. This is an artificial intelligence based approach that actually formulated the basis of what is discussed today as “Flying Ad Hoc Network” by Muller and Bekmezci *et al.*. In this SMAVNET formation, the authors considered micro-aerial vehicles rather than mini-UAVs due to requirement of more complex hardware in micro-aerial vehicles. One of the limitations of this network study is that authors only considered two-dimensional analysis that depicts the relative positioning of aerial vehicles which is based upon assumptions rather than use of control algorithm for cooperative relaying formation.

Yang *et al.* [31] has proposed an opportunistic methodology for aerial relaying that allows cooperative formation for achieving complex tasks. This opportunistic relaying forms planning modules for search maneuverabilities and trajectory formation. This approach is derived from Polycarpou *et al.* [41] in which authors have discussed various

search operation modules, and designed a new approach for efficient traversal using cooperative UAVs. The authors discussed the formation of the cooperative framework for relaying based opportunistic searching. A search problem is formulated for cooperative relaying. The opportunistic relaying based cooperative algorithm is supported by various propositions derived by the authors.

Irrespective to UAV networks, Lilien *et al.* [29] has proposed an algorithm for opportunistic relaying for special types of temporary networks. This relaying is based upon adequate usage of network resources. The authors proposed a framework, and validated it using Oppnet Virtual Machine. The applications of this infrastructure based framework are being utilized for serving as test beds for various hard bounded simulating networks. The framework is based on the formation of controllers that includes seeds for the network, helper modules created under Oppnet, resource reservoir and user utility program. Cetin *et al.* [56] has proposed a UAV cooperative relaying to avoid any collisions with dynamic obstacles. The approach is based upon creation of artificial prospective fields, termed as potential fields. Communication set up formulated is based upon identification of path loss models in the environment of UAV trajectory. The authors created a vector field for identification of network hindrance. NS-2 and Matlab are used for creating UAV modules and generating traffic and obstacle environment. Atrocious force based obstacles are used that generate potential field model using various harmonic functions.

Lilien *et al.* [30] in extension of their work, have studied the resource relaying for opportunistic networks. It is a simulation based analysis that used UAVs as the nodes of the network. Perimeter surveillance is identified as one of the major applications of such networks. Authors have used ad hoc networking for multi-UAV systems and formed a virtual environment for tracking nodes in the network periphery. The approach studied by the authors is absolutely a cooperative resource utilization approach as it distributes the Inertial Navigation System (INS) into various modules such as helpers, detectors, with the help of intelligent detection system which is reacted upon by network of guided teams.

2.2.2.3 Task Allocations

Task allocations are one of the important applications that needs to be incorporated using UAVs in aerial ad hoc formation. Autonomous UAVs can be monitored using controlled algorithms, and then can be operated in cooperative mode [57]. Task assignment is applicable to vehicles that have the neural capacity for sensing operations of each other and then taking decisions regarding various assignments in the network. It includes simple collision avoidance algorithms as well as cooperative solutions for complex missions [58] [59] [60]. In the literature, various cooperative frameworks exist which primarily focus only on task assignment for multiple UAV network, thus, making another class for analysis of autonomous aerial vehicles.

Ben-Asher *et al.* [61] has proposed an ad hoc communication set up for UAVs based upon the decisiveness and control. The authors proposed an algorithm for cooperative work handling for UAVs using randomized examination based upon tree data structures. UAV localization is carried using these tree formations. Authors focussed the analysis over distribution and robustness of UAV network. The whole algorithm is divided into a series of layers, namely, control layer, communication layer, and the objective formulation layer.

Liu *et al.* [50] has proposed a UAV scrutiny based on intellectual path organization. This task oriented reconnaissance is targeted towards identification of intimidating marks and combatant locations. Authors focussed their work on path identification in complex task related to surveillance. The approach suggested is cost oriented that covers both short and long spell cost using adjustable weights. Another task oriented approach for cooperative aerial vehicles is given by Hu *et al.* [62]. The authors have proposed a cell division based search methodology for group of UAVs. This surveillance is based upon the computation of probability maps for each cell which is formed by the division of underlying area. The standard Bayesian rules are linearized by authors to form a linear updatable system. Target formation in the discussed problem is based upon the Bernoulli distribution. The approach first aims at calculation of probabilistic values for each cell under traversal by individual vehicles without any cooperation. Then, using linear Bayesian updating

rule, area probability map is formulated using cooperative communication amongst the vehicles. McCune *et al.* [63] has worked over cooperative hunting using UAVs. UAVs were collaborated to perform a simultaneous enemy hunting task over a defined area. Bing *et al.* [64] has proposed a cooperative swarm formation using multiple UAVs for achieving complex tasks. Flocking analysis for cooperative mode are performed by the authors to evaluate the performance of their approach.

Another task assignment problem is placing aerial node in a existing mobile ad hoc network. Wang *et al.* [48] addressed this issue and developed an approach for possible solutions towards placing an aerial node using binary optimization. Some of the other task based UAVs network formation are considered in the form of coverage map formation by Morse *et al.* [65], smart target search by Altshuler *et al.*[66], data fusion based path planning by Shen *et al.*[67], and trajectory tracking by Yamasaki *et al.*[68]. All of these issues can be considered as a research problem, and more optimized approaches can be developed to provide a non-complex solution for these problems.

2.2.2.4 UAV Aided Routing and Positioning

Routing is always a challenging task in case of heterogenous environment. In case of UAV guided networks, it becomes more complex to perform routing. As two different networks are involved, and a common interface is required to support data forwarding in such scenarios. Many researchers have identified the potential in UAV networks to perform routing, whether in static mode or in dynamic mode. The focus of this study is primarily over dynamic ad hoc based routing using UAV as routing aids in mixed network structure.

Niculescu and Nath [69] have proposed a dynamic vector based routing using the positioning of dynamic node in an ad hoc environment. Authors reviewed the global positioning system and then formulated the positioning system for ad hoc networks. The major focus of this positioning system is stabilized movement and reconfigurability of the network model. The authors used multiple propagation model for analyzing vector based localization of nodes. Iordanakis *et al.* [70] has proposed a routing strategy for aeronautical

networks in ad hoc mode. However, from discussions, it is clear that such routing can form the basis of data forwarding in autonomous networks only, but cannot be directly implemented. This routing strategy is more of an infrastructure based routing due to which it cannot be implemented as UAV networking. The routing protocol suggested is an extension of AODV protocol, which is modified in its interfacing with MAC layer. Zang *et al.* [71] has given a clustering algorithm for networking using UAVs as nodes. Authors termed this algorithm as mobility prediction algorithm. The proposed routing algorithm is based upon the calculation of link expiration time that denotes how long a connection exists between two nodes. The algorithm aims at a selection of cluster head based upon the largest weight amongst each node.

Lin *et al.* [72] has proposed a position oriented routing for aerial vehicles. This is a type of location aware routing strategy. The dynamic environment that can be a constraint in such network is controlled using Gauss-Markov mobility model. The target of the algorithm is to select next hop for relaying using Euclidean distance. Beacon messages are used for identifying position of the nodes based upon the velocity and time stamp. Wang *et al.* [73] has worked on placement of aerial node in ground ad hoc environment. This is more like a mobile facility problem. Authors have used linear and quadratic programming for solving the constraints of aerial node placement based upon binary optimization techniques. Authors termed this framework as “QUBO”. This QUBO model is a quadratic model which is solved using linear binary optimizations. The major objective of this approach is to improve the quality of the downlink that refers to connectivity between the aerial and the ground nodes.

2.3 Comparison of Existing Frameworks/Models

UAV frameworks irrespective to their specifications, have been compared on the basis of some standard parameters. These parameters include ideology, type, classification, routing support, topology management, scalability, parameters targeted, cooperative support and guided network support. The detailed comparison is shown in Table 2.1.

Table 2.1: Comparison of Existing UAV Based Frameworks/Models/Approaches

Approach/ Model	Author (Year)	Ideology	N/W Type	Classification	Parameters Modules	Routing Support	Topology Management	Scalability Adaptability	Cooperative Formation	Guided N/W Formation
2-Tier Hierarchical Routing	[Gu <i>et al.</i> 2000] [28]	Hierarchical State Routing	Single-UAV	UAV-Aided Routing	Topology Formation, Propagation Delays	Yes	Yes	Medium	No	-
Search Framework for Distributed Agents	[Polycarpou <i>et al.</i> 2001] [41]	Cooperative Search	Multi-UAV	Task Assignment	Search Patterns, Uncertainty	No	No	Medium	Yes	-
Ad Hoc For Autonomous vehicles	[Bellur <i>et al.</i> 2002] [74]	Intra-Inter ad hoc formation	Multi-UAV	UAV-Aided Routing	Link Type, Throughput, Delays	Yes	Yes	High	Yes	-
UAVs for satellite coverage	[Palazzi <i>et al.</i> 2004] [34]	High Altitude Platforms for urban areas	Single-UAV	UAV-Aided Transmis- sion	Throughput, Transmission Time	Yes	No	-	No	-
Search Framework for UAVs	[Vincent and Rubin 2004] [40]	Motion Planning and control	Multi-UAV	Task Assignment	Reliability, Motion Planning	No	Yes	-	Yes	-
UAV Navigation and Control	[Doitsidis <i>et al.</i> 2004] [32]	Fuzzy Logic based Framework	Single-UAV	Task Assignment	Elevation, Altitude	No	No	-	No	-
Mobile Backbone Network	[Rubin <i>et al.</i> 2004] [33]	Using UAVs as mobile backbones	Multi-UAV	UAV Aided Routing	Throughput, Delivery Ratio, Delays	Yes	Yes	Medium	No	-
UAVs as HAPS	[Palazzi <i>et al.</i> 2005] [35]	Multi-TCP oriented architecture	Multi-UAV	Task Assignment	Average Throughput, Time to transmit, PER	Yes	-	High	Yes	-
Cooperative Fire Detection	[Merino <i>et al.</i> 2005] [43]	Data fusion using sensors	Multi-UAV	Task Assignment	Uncertainties, Image Processing, Alarm Tracking	No	No	High	Yes	-
Target tracking with UAVs	[Zhan <i>et al.</i> 2005] [75]	Kalman Filter based estimations	Multi-UAVs	Task Assignment	Real Estimation Error, UAV trajectory	No	No	-	No	-
Task Allocations using Multirobots	[Zhu and Yang 2006] [76]	Neural Schema based SOM creations	Multi- Robot/UAVs	Task Assignment	Target tracking, Movement Tracing	-	Yes	High	Yes	-
Dynamic Target Searching with UAVs	[Bertuccelli and How 2006] [77]	Uncertainty Map building for target tracking	Multi-UAV	Cooperative Network	Target identification, Trajectory tracing	-	-	High	Yes	-

Continued on next page

Table 2.1 – continued from previous page

Approach/ Model	Author (Year)	Ideology	N/W Type	Classification	Parameters Modules	Routing Support	Topology Management	Scalability Adaptability	Cooperative Formation	Guided N/W Formation
Opportunistic Learning based UAV network	[Yang <i>et al.</i> 2007] [78]	Cooperative Search and Path Planning	Multi-UAV	Relaying	Training time, Search paths, Search Efficiency	-	Yes	High	Yes	-
UAV ad hoc formation	[Ben-Asher <i>et al.</i> 2008] [61]	Decentralized task assignment and flocking algorithm for ad hoc communica- tion set up in UAVs	Multi-UAV	Ad Hoc Network formation Relaying	Rooted Spanning Tree Formation, Number of hits, Number of UAVs, Cooperation Method	-	Yes	High	Yes	-
Position less MAV swarming	[Hauert <i>et al.</i> 2009] [55]	Swarm intelligence based relaying	Multi-UAV	Relaying	Hop count, Swarm size, Speed, Communication Range	-	Yes	High	Yes	-
CSAT using UAVs	[How <i>et al.</i> 2009] [45]	On board modules based maneuver- abilities	Multi-UAV	Task Assignment	Mission Time, Search Area, Target information	-	Yes	High	Yes	Yes
Clustering Algorithm	[Zang <i>et al.</i> 2011] [71]	Based on prediction of UAV mobility	Single-UAV	UAV-Aided Routing	Number of Cluster heads, Speed, Reaffiliations	Yes	-	Low	-	-
Search using UAV Alliance	[Manathara <i>et al.</i> 2011] [79]	Decentralized algorithm based on swarm intelligence	Multi-UAV	Task Assignment	Completion Time, Coalition Assignment, Computational Time	-	Yes	High	Yes	-
DDDAS Applications	[Madey <i>et al.</i> 2012] [80]	Created Simulator Using SOA work flows for Control and Mission Planning	Multi-UAV	Task Assignment	End to End Service, Number of vehicles	-	-	High	Yes	-
Position Oriented Routing	[Lin <i>et al.</i> 2012] [72]	Routing based upon geographi- cal position and mobility of UAVs	Multi-UAV	UAV Aided Routing	Delivery Ratio, Delay, Average Hops	Yes	Yes	Medium	Yes	-
A Swarm Attack	[Cevik <i>et al.</i> 2013] [81]	UAV based electronic warfare	Multi-UAV	Task Assignment	Jamming-to- signal Ratio, Screened UAV Range	-	Yes	High	Yes	-

Continued on next page

Table 2.1 – continued from previous page

Approach/ Model	Author (Year)	Ideology	N/W Type	Classification	Parameters Modules	Routing Support	Topology Management	Scalability Adaptability	Cooperative Formation	Guided N/W Formation
Ground Vehicle's behaviour identification using UAVs	[Oh <i>et al.</i> 2013] [82]	Fuzzy oriented vehicle estimations and trajectory identifications	Multi-UAV	Task Assignment	Trajectory planning, Level of Alertness, Speed, Behaviour analysis, Geographical Segmentations	-	Yes	-	Yes	-
Target Tracking with UAV swarm	[Jensen <i>et al.</i> 2013] [83]	Tagged Fish localization using Kalman Filter	Multi-UAV	Task Assignment	Maximum Range, Tag positions, Distance B/w estimated and actual tags, Targets identified	-	-	Medium	Yes	-
Task Allocation and Search Planning	[Kim <i>et al.</i> 2013] [84]	Probabilistic Maps for UAV threshold decision for maneuverabilities	Multi-UAV	Task Assignment	Probability map, Certainty Map, Number of targets, Percentage Destroyed	-	Yes	High	Yes	Yes
UAV amenity management for surveillance	[Ortiz-Pena <i>et al.</i> 2013] [85]	Optimizations of resources for mission planning and control	Single-UAV	Cooperative Network	Potential Information Gain, Route identification	-	-	-	No	-
UAVs resource utilization Network	[Lilien <i>et al.</i> 2013] [30]	Simulative analysis of amenity management for UAVs	Multi-UAV	Opportunistic Relaying	Fighter Speed, Sensor Range, Delays, Helper Integration, Success Rate, Number of helpers, Detection time	-	Yes	High	Yes	-
Multi-UAV positioning in ad Hoc Networks	[Wang <i>et al.</i> 2013] [73]	Based upon facility positioning using Binary optimization	Multi-UAV	Positioning	Facility Location, CPU time	-	Yes	High	-	-
Cooperative Exploration and path planning for UAVs	[Liu <i>et al.</i> 2013] [50]	Information sharing based probability map for path identifications	Multi-UAV	Cooperative Network formation	Residual Uncertainty, Targets found	-	Yes	-	Yes	-

Continued on next page

Table 2.1 – continued from previous page

Approach/ Model	Author (Year)	Ideology	N/W Type	Classification	Parameters Modules	Routing Support	Topology Management	Scalability Adaptability	Cooperative Formation	Guided N/W Formation
UAV Localization	[Benini <i>et al.</i> 2013] [52]	Ultra-wide band oriented visual odometry based UAV confinement	Single-UAV	Cooperative Network	Data Rate Sampling, Trajectory, Fault Identification, UAV Ranging, Visual Odometry	-	-	Low	No	-
UAV Localization	[Eroglu <i>et al.</i> 2013] [53]	Binary Search based geo- territory oriented UAV localization	Single-UAV	Localization	Number of Measurements, Accuracy in estimation	-	-	Low	-	-
UAV Network for Area Coverage	[Schleich <i>et al.</i> 2013] [54]	Localization algorithm for mobility control of UAVs	Multi-UAV	Relaying	Exhaustiveness, Covering Speed, Fairness, Connectivity	-	Yes	Medium	Yes	-
Collision Free Relaying	[Cetin <i>et al.</i> 2013] [56]	Artificial Potential Field for Obstacle identifica- tion	Multi-UAV	Task Assignment Relaying	Potential Field, Vector field, Obstacle identification	-	Yes	High	Yes	-
Data Collection Over Sensor Networks	[Jawhar <i>et al.</i> 2014] [86]	Data Collection over WSNs with least energy con- sumption	Multi-UAVs	Task Assignment	Delivery ratio, Delays Buffer size	Yes	Yes	-	Yes	-
Palm Tree Detection	[Malek <i>et al.</i> 2014] [87]	Identification of palm tress using UAVs imaging	Single- UAVs	Task Assignment	Producer accuracy User accuracy True positive detection	No	No	-	No	-
Autonomous Navigation	[Lugo and Zell 2014] [88]	High level autonomous navigation with AR.Drone	Single- UAVs	Localization Task Assignment	Trajectory Image sensing	No	Yes	-	No	-
UAVs localization	[Wan <i>et al.</i> 2014] [89]	Cooperative Localization without GPS	Multi-UAVs	Localization	Localization Position accuracy	No	Yes	Yes	Yes	-
Road Detection and Tracking	[Zhou <i>et al.</i> 2015] [19]	UAV based road identi- fication	Single- UAVs	Task Assignment	UAV imaging Detection accuracy	No	No	-	-	-

2.3.1 Preface to Nature-Inspired Algorithms and Routing Protocols

Many routing protocols based on the nature inspired algorithms [90] have been developed which uses traditional setup of a single ad hoc formation for transmission [91] [92]. Most of them use travelling salesman concept driven by ant colonies, bee colonies, and particle swarm optimization, etc. Some of the popular algorithms that can be used for route selection includes Ant Colony Optimization (ACO) [93], Artificial Bee Colony Optimization (ABC) [94] [95], Particle Swarm Optimization (PSO) [96], Firefly Optimization (FO) [97], Cuckoo Search (CS) [98], Bat Algorithm (BA) [99], and Fruit Fly Optimization (FFO) [100], etc. Some of the popular routing protocols based on these nature inspired algorithms are shown in Table 2.2.

Table 2.2: Various Nature Inspired Algorithms

Protocol	Proposed By	Approach	Key Concepts
Bee Cluster	[Karaboga <i>et al.</i> 2012] [94]	Hybrid	Energy efficient routing
AntNet	[Di and Dorigo 1998] [101]	Proactive	Stabilized paths
GPSAL	[Camara and Loureiro 2000] [102]	Location	Decreased Overhead
Ant-AODV	[Marwaha <i>et al.</i> 2002] [103]	Hybrid	Low end-to-end delay and route discovery latency
PERA	[Baras and Mehta 2003] [104]	Reactive	Reduced end-to-end delay
ARAAI	[Yuan and Yan 2005] [105]	Hybrid	Stabilized paths and high delivery ratio
AntHocNet	[Di <i>et al.</i> 2005] [106]	Hybrid	Low overheads
NNNA	[Ahmed <i>et al.</i> 2012] [107]	Hybrid	High packet delivery ratio and reduced end-to-end delay
PACONET	[Osagie <i>et al.</i> 2008] [108]	Reactive	Efficient delivery ratio
HOPNET	[Wang <i>et al.</i> 2009] [109]	Hybrid	Low end-to-end delay and high delivery ratio
LBMRAA	[Elmoniem <i>et al.</i> 2011] [110]	Reactive	Low routing overhead, buffer overflow and end-to-end delay
ANTALG	[Singh <i>et al.</i> 2014] [111]	Hybrid	Low delays and overheads
BeeHive	[Wedde <i>et al.</i> 2004] [112]	Hybrid	Fault-tolerant routing
BeeAdHoc	[Wedde <i>et al.</i> 2005] [113]	Hybrid	Less energy consumption and low delays

2.3.2 Preface to TCP Variants for Congestion Control

Congestion control has individually been focused by many researchers over the last decade. Primarily, most of them have only focused over the development of an indi-

Table 2.3: Various TCP Variants

TCP Variants	Proposed By	Network Type	Key Concepts
Reno	[Jacobson 1990] [114]	Wired/Wireless	Fast retransmit, reduce congestion window to half instead of 1
New-Reno	[Floyd and Henderson 2004] [115]	Wired/Wireless	Detect multiple Packet loss
Vegas	[(Brakmo and Peterson 1995] [116]	Wired/Wireless	Congestion avoidance mechanism
Westwood	[Mascolo <i>et al.</i> 2001] [117]	Wireless	New faster recovery mechanism
Veno	[Fu 2001] [118]	Wireless	Detect type of loss congestion or random
HTCP	[Leith and Shorten 2004] [119]	High speed networks	Different incremental factor for different type of network
BIC	[Xu <i>et al.</i> 2004] [120]	Long distance networks	Bic combines two schemes Additive increase and Binary search increase
Cubic	[Ha <i>et al.</i> 2008] [121]	Long distance networks	Enhances fairness property of BIC with Cubic window growth function
Hybla	[Caini and Firrincieli 2004] [122]	Heterogeneous networks	Solved RTT disparity problem, Implemented with sack option to recover multiple packet losses
Scalable	[Kelly 2003] [123]	High speed networks	Modified congestion window update algorithm which improves throughput in high speed and wide area networks
Illinois	[Liu <i>et al.</i> 2008] [124]	High speed networks	Uses packet loss information to conclude whether window size be increased or decreased and delay to calculate the amount of increase or decrease
Yeah	[Baiocchi <i>et al.</i> 2007] [125]	High speed networks	Fast and slow as mode of working. In fast mode uses aggressive rule to increment the congestion window and in slow mode it acts as TCP Reno

vidual TCP-oriented congestion control algorithms. Various variants and their features available for possible congestion control are shown in Table 2.3.

2.4 Preface To Service Provisioning

Guided ad hoc network is an integration of aerial ad hoc network and ground ad hoc network. Both the networks operate in heterogeneous environment, but with common objectives. The highly dynamic environment of aerial network provides coordinated data transfers to terrestrial nodes that build a guidance map to generate look ahead information. In recent years, maneuverabilities using the single as well as cooperative ad hoc networks have been studied. Much emphasis has been shifted towards service based cooperation amongst hybrid nodes to achieve complex tasks [126]. This has provisioned the formation of ambient network using aerial vehicles [127].

Kim [128] has designed a framework that operates using activities that generates work

flows for cooperative system. Designed framework uses a fragmentation approach to form a collaborative system. Such framework can also be used for areas with higher dynamic constraints. Choi *et al.* [129] has designed a service middleware architecture for providing autonomous services using robots. It is a layered architecture that utilizes client-server paradigms to analyze the service requirements of the system. Paunicka *et al.* [130] [131] has designed a broker architecture that serves as a middleware for UAVS. The middleware utilizes the object oriented features to provide controlling mechanism for aerial vehicles. The middleware is termed as CORBA that utilizes the features of the open control program. Floch *et al.* [132] has discussed the utility of service architectures and framework for run time facilities. Authors have focussed on software based decision systems that can be used for dynamic service providers. Pastor *et al.* [133] has designed a software controlled hardware for performing UAV maneuverabilities. The authors designed a generalized architecture to provide a flight control system for UAVs.

Kramer and Magee [134] have discussed the challenges involved in the formation of self-managed systems. Authors have discussed the various research issues and controlling factors mandatory in the design of real time self controlled systems. Suri *et al.* [135][136] has designed a service oriented architecture for ad hoc network using supply computing. Experimental based analyses have been presented in the paper that shows the utilization of service based resource utilization in ad hoc networks. Lopez *et al.* [137] has designed a middleware solution for UAVs. The middleware utilizes the avionic system of aerial vehicles. The main function of the middleware is to control the maneuverabilities and provide support for real time applications using UAVs. Frew and Brown [138] have discussed the networking issues using UAVs. Various challenges involved in networking with small UAVs have been highlighted in this paper. Pena *et al.* [139] has integrated aerial vehicles to form an information management system for traffic management. The design focuses on connecting unmanned vehicles with regular traffic. Some of the other service architectures include service reference model by Heisey *et al.* [140], collaborative UAVs services by Mohamed *et al.* [141], real time service architecture by Marconato *et al.* [142], and collaborative service model by Dimitriou *et al.* [143]. The Comparison of existing service architectures for UAVs ad hoc formations is shown in Table 2.4.

Table 2.4: Comparison of Existing Service Architecture/Middleware for UAVs Ad Hoc Formations

Approach	Author	Ideology	Parameters	API Support	Network	Ad Hoc Formation	Scalable	Message Control
DDAS Application Middleware	Madey <i>et al.</i> [80]	SOA Work-flow For Mission Control	Nodes, E2E service	Yes	Multi-UAVs	-	Yes	Yes
Multi-layered Middleware	Choi <i>et al.</i> [129]	Service Based Robot Localization	Localization, Web Services, Workflow Scheduler	Yes	Multi-Robots	-	Yes	Yes
CORBA Middleware	Paunicka <i>et al.</i> [130]	Open Control Platform for UAVs	Locking, Overheads, Process-Control	Yes	Multi-UAVs	-	Yes	Yes
Hardware/Software Middleware	Pastor <i>et al.</i> [133]	Mission Controller for UAVs	Remote Telemetry, Payload, Link Control	Yes	Single-UAV	-	-	Yes
Payload Middleware	Lopez <i>et al.</i> [137]	Payload and Mission Control for UAVs	Event Scheduling, Remote Invocation, File Transfers	Yes	Single-UAV	-	-	Yes
Reference Software Architecture	Heisey <i>et al.</i> [140]	Supporting UAVs in airspace	Latency, Localization	Yes	Multi-UAV	-	Yes	Yes
Reconfigurable Flight Architecture	Deng <i>et al.</i> [144]	Flight Control System Architecture	Service provisioning, Interoperability	Yes	Multi-UAVs	Yes	Yes	Yes
Service Middleware For UAVs	Mohamed <i>et al.</i> [145]	Service Computing For UAVs	Service provisioning, Sensor range, Localization	Yes	Multi-UAVs	Yes	Yes	Yes
Cooperative ISR Architecture	Chen <i>et al.</i> [146]	Cooperative Search and Navigation	Localization, Sensor management, Surveillance	-	Multi-UAVs	-	Yes	-
Middleware Between UAVs-WSNs	Freitas <i>et al.</i> [146]	Surveillance between UAVs-WSNs	Node tracking, connectivity time	-	Multi-UAVs	-	Yes	-

2.5 Preface To QoS Scheduling In Ad Hoc Networks

Real time data provisioning requires the facility for transmission of heavy traffic over ad hoc networks [147] [148]. Number of nodes and intermediate relaying approach play a vital role in enhanced service provisioning to end users [149] [150]. Various models developed for real time usage of mobile ad hoc networks and UAV networks have been discussed in this section. These models have been modified as per usage of the network, but the major drawback that appears is the weakening of the quality of services at the end user. Many traffic scheduling algorithms have been developed which combines the features of admission control and traffic control algorithm [151] [152]. Admission control procedures can easily regulate the control rate of traffic [153]. But, these procedures use absolute differential service-based methods of data transmission for real time packets, thus, cannot be relied in terms of performance under extensive conditions such as dynamic topology changes, and distributed network formation.

Geng *et al.* [154] has designed a protocol over MAC layer that ensures QoS over ad hoc networks. The protocol operates over IEEE standards. It considers reservation system for QoS enhancement based on channelization. The protocol controls the functionality of network nodes depending upon the requirement of reserved slots for transmission. Another model given by McLaughlin *et al.* [155] is based upon IP packet scheduler. The paper includes the circuit implementation for the weighted fairness model for improving network QoS. The model utilizes the capabilities of general purpose sharing for efficient and fair allocation of bandwidth. Shreedhar *et al.* [156] has designed another model which is termed as a stochastic fairness queue model. This model is based upon round robin technique, which is only applicable for lesser number of packets and low traffic networks. This model is capable of providing quality of service for end-users and is based upon hashing technique. This model classifies the packets in the queue on the basis of probabilistic approach. However, this model increases the overheads as hashing technique would account for residing the packets coming from the same source into different queue depending upon the generated hashing function.

Further, a model for improving the end user-based quality of service is given by Vara-

prasad *et al.* [157] and is termed as a weighted fair queue model (WFQ). This model serves the queue with lower weight. Hence, weight was assigned on the basis of presentation units (multimedia datagram or packets). It reserves bandwidth for particular users and divides the working of the MANETs transmission into a set of classes with weights assigned to them. The model uses differential services for classification of traffic. Sun *et al.* [158] has designed a modified proportional fairness model with multi hop queue scheduling for cognitive radio-based MANETs. The model is cross layer-based model that restricts QoS for end users in MANETs. The model is MAC layer-based service distributor that ensures that QoS is not affected during transmission. It computes the availability of channels and buffer. Then, it uses a MAC analyzer to compute packet priority and packet scheduling before selecting the transmission strategy.

Peng *et al.* [159] has designed a virtual routing protocol that offers QoS in terms of bandwidth and topology management over OLSR protocol. The authors implemented a virtual cross layer approach to share information between the nodes of the network and used it as a decisive parameter during routing, thus, offering QoS over wireless mesh network. Sondi *et al.* [160] has designed QoS over OLSR as guidelines and operated it as an integrated approach rather than a fixed underlying algorithm. Authors designed governing rules for MANETs like networks that can improve their QoS without using particular algorithms. The approach used in their paper demonstrates the identification of factors based on heuristics that optimizes the performance of the network.

Kuo *et al.* [161] has implemented a cross layer approach to offer multimedia services over MANETs. It is link to link connectivity-based implementation. Authors integrated dynamic hash table and IPV6 to provide multimedia transmission over networks that have frequent topology changes and unreliable wireless links. The designed schema of authors operates in three paradigms. One as application layer, second as routing layer, and third as intractable cross layer. In an another approach by Bouk *et al.* [162], authors emphasized on selection of appropriate gateway node to provide seamless QoS to end users over MANETs. Taking into account the path dynamics, a feedback system is incorporated with the network, which is capable enough to handle end to end delays. Authors focussed on link expiration time to evaluate the gateway during path planning

on the basis of system dynamics in the network.

Yu *et al.* [163] has used game theory approach to provide security along with QoS over MANETs. The Bayesian theoretic approach is incorporated to identify the winning paradigm for transmission. Collaborative communications are provided over MANETs by selection of an appropriate relaying mechanism based on the network inputs in terms of throughput and security analysis performed in the system. The approach focusses on attaining equipoise over the network to improve its performance. Wen et al. [164] derived a queuing theory-based approach for service discovery in MANETs. The approach includes a $M|M+1$ model for service discovery on MANETs nodes.

Further, collaborative formations between the aerial network and the ground network can resolve certain complex issues such as scalability, coverage, throughput, and guaranteed quality of service (QoS) to end users. A network with provisioning of high range of QoS is capable of providing a highly sustainable network with higher data rate and less delays. It is considerable that a network operating in an assisted formation is capable of improving the connectivity, but is also liable to suffer from delays [165].

2.6 Control Issues Over Networked UAVs

Unmanned aerial vehicles (UAVs) have evolved as an important aspect in modern day networking. Applications involving use of multiple UAVs are dependent upon the fully functional network formation between these vehicles. One of the most important use of these vehicles is in autonomous networking [166].

2.6.1 Stabilization

These network formations are always under the threat of sudden failures due to various issues such as network hijacking, node compromising, node failures, physical collision, congestion, etc. Efficient network models are required that can handle such uncertain failures. A possible approach for handling such dynamic configurations can be the use of artificial neural networks. Treating each network node over a prescribed set of parameters

can regulate the flow and manage any loss of vehicle. A network capable of self-learning can enhance its understanding regarding the dynamic conditions which can be further utilized to attain a stabilized state [167] [168].

Continuity and integrity are the other key metrics which are to be handled efficiently for stabilized network formations. Continuity refers to the continuation of the network operations despite the failures and network losses, whereas the integrity deals with the reliable communication. Continuity can be classified as the sub-category of fault-tolerance and integrity can be determined in terms of secure transmissions. Although heuristics are considered efficient for dynamic conditions, but these cannot recover the system from damages [169]. Formation of self-aware, self-balancing, self-recoverable neural networks have seen a lot of research over the past few years [170] [171]. These approaches, however, utilize the pre-learning facilities which recovers the system from damages [172]. Recovery which is based only on learning and system configurations, is not an efficient approach. This slows down the whole system. Liu *et al.* [173] discussed that for critical networks, selection of appropriate utility function can provide an efficient learning for networks. Approach developed by them was applicable for congestion relief only. Eriksson *et al.* [174] came up with the idea of smart grids with properties of self-recovering from damages. Although efficient, their approach had some limitations in recovery and self-configurations against failed nodes.

2.6.2 Topological Broadcasting

The broadcasting problem leads to redundancy in the packet transmissions among nodes in the wireless ad hoc network which may lead to replica of messages and packet collisions [195]. Broadcast storm is not a new issue, it existed from the starting of networks using omnidirectional antennas that uses radio waves broadcasting for route discovery and data transmission [176] [175] [184]. With nodes being mobile, this issue becomes severe, resulting in adverse effects such as network breakdown, node failures, and session loss etc. [177]. Further, networks involving aerial nodes and ground nodes as part of common ad hoc units cannot be kept aloof from such issues. Therefore, efficient approach is required that can provide a topological arrangement at the initial stage of network for-

Table 2.5: Comparison of Existing Approaches For Solution To Broadcast Storms

Approach	Author	Ideology	Parameters Improved	Applicability
Distributed Broadcast Suppression	Tonguz <i>et al.</i> [175]	p-persistence, Slotted 1-persistence and Slotted p-persistence	Packet loss rate, End to end delay	Wireless Ad-hoc Networks
Adaptive Threshold Selection	Tseng <i>et al.</i> [176]	Adaptive Location-based Scheme, Adaptive Counter based Scheme	Reachability, No. of transmissions, Neighbourhood coverage	Mobile Ad-hoc Networks
Probabilistic and Timer based Approach	Wisitpongphan <i>et al.</i> [177]	p-persistence, Slotted 1-persistence and Slotted p-persistence	Message delay, Packet loss rate, Message reachability, Message overhead	Vehicular Ad-hoc Networks
Selective Reliable Broadcast protocol	Vegni <i>et al.</i> [178]	Cluster based approach to detect next hop forwarder	Detection of clusters, Detection of next hop forwarders, Throughput, Average Message Propagation	Vehicular Ad-hoc Networks
GOSSIP	Haas <i>et al.</i> [179]	Fixed Probabilistic Approach	Optimal forwarding probability	Mobile Ad-hoc Networks
AutoCast scheme	Wegener <i>et al.</i> [180]	Adaptive Probabilistic Approach	Retransmission probability, Adjusted broadcast interval, Reliability	Vehicular Ad-hoc Networks
Hypergossiping	Khelil <i>et al.</i> [181]	Adaptive Broadcasting Scheme	Intra-partition forwarding, Retransmissions	Mobile Ad-hoc Networks
GOSSIP_DIR2	Shen <i>et al.</i> [182]	Directional Antennas and Euclidean Distance	Forwarding probability	Mobile Ad-hoc Networks
NCPR	Zhang <i>et al.</i> [183]	Self-Pruning Mechanism	Forwarding delay, Rebroadcast delay, Connectivity	Mobile Ad-hoc Networks
DV-CAST	Tonguz <i>et al.</i> [184]	Distributed Broadcast Protocol	Reliability, Efficiency, Scalability	Vehicular Ad-hoc Networks
BRACER	Song <i>et al.</i> [185]	Distributed Broadcast Protocol	Successful broadcast ratio, Broadcast delay, Collision avoidance	Cognitive Radio Ad-hoc Networks
AHBP	Wei <i>et al.</i> [186]	Broadcast Relay Gateway	Broadcast redundancy, Broadcast collision, Congestion	Mobile Ad-hoc Networks
EGBB	Touzene <i>et al.</i> [187]	2-dimensional Grid Cell View	End to end Delay, Packet collision ratio, No. of saved rebroadcasts, Reachability, Hop count	Mobile Ad-hoc Networks
Broadcast Storm	Tseng <i>et al.</i> [188]	Probabilistic, Counter based, Location based, Cluster based and Distance based schemes	Redundant rebroadcasts, Reachability, Average latency	Mobile Ad-hoc Networks
Reduction of Broadcast Redundancy	Peng <i>et al.</i> [189]	Topology Information, Statistical Information	Redundant messages, Network Bandwidth, Energy, Reliability	Mobile Ad-hoc Networks
Discovery Phase of routing protocols	Reina <i>et al.</i> [190]	Jaccard Distance	Forwarding Delay, Retransmission delay	Mobile Ad-hoc Networks
Hybrid Flooding Scheme	Reina <i>et al.</i> [191]	Expansion Metric	Reachability, Forwarding probability	Mobile Ad-hoc Networks
CAREFOR	Mostafa <i>et al.</i> [192]	Irresponsible Forwarding	Collision probability	Vehicular Ad-hoc Networks
EAG	Huang <i>et al.</i> [193]	Residual Energy of Nodes	Forwarding probability, Reachability	Mobile Ad-hoc Networks
Distributed GOSSIP	Huang <i>et al.</i> [194]	Density based Scheme	Coverage level, Successful transmission ratio	Mobile Ad-hoc Networks

mation resulting in lower overheads and decreased redundancy in case of link failures. A detailed comparison of some of the popular approaches that provide successful eradication of storming problem is presented in Table 2.5.

2.7 Security Issues Over Networked UAVs

The involvement of hybrid nodes in a guided network raises the security issue of the underlying ad hoc units. These security issues deal with the logical as well as with the physical partitioning of the networks. Intrusions in any form can adversely affect the network operations, and can cause serious threats due to involvement of sensitive and mission configured nodes [196] [197] [198]. Security requires authentication and all-time availability to offer better connectivity between the two ad hoc units. This can be achieved using situational awareness, data monitoring, and prediction model that can predict the current state of the network. This, in turn, can be used to formulate a decision system that can offer secure relays amongst them. Further, the channel security in multiple ad hoc units depends upon the amount of network reparability offered, organizational time consumed, and modifications required. The guided network must have the ability to self-organize, and adapt to the concurrent changes that occur during data transfers [199] [200] [165]. These changes can be either logical changes dealing with connections and data, or physical changes dealing with the node localization. The formation of the ambient system using multi-guided network should also provide awareness in terms of situational classification and enhanced network operations for faster recovery in case of network intrusions and faults. Data security, authentication, and trust aware policies [201] [202] already exist in the literature, but these approaches provide security to end users only.

In an approach by Yokoyama *et al.* [203], authors have presented an architecture that aims at secure localization of aerial vehicles using the received signal strength metrics. This architecture aims at the formation of a secure UAV system. Rodrigues *et al.* [204] has presented a case study over the UAVs involving usage of secure web services providing reliable and fast communications. Using the knowledge base of avionics, the authors developed the service oriented architecture that provides reliable solution to UAV network

formation. Birnbaum *et al.* [205] has also worked on the security aspects of the unmanned aerial vehicles. The work primarily focuses on the identification of network attacks and hardware failures that might hinder the network operations. The model presented is based on the prototypes of the various monitoring tasks and several services running onboard. Mammadov *et al.* [206] has worked over the transmission range of the aerial vehicles. The authors have developed a framework that provides long range communication setup with physical security to UAVs operating with multiple agents.

Sahingoz [207] has given a multi-level key management approach using sensor network and unmanned aerial vehicles. The use of UAVs provides key management and key distribution over the sensor network. Data encryption and authentication are attained using the UAV based key management scheme. In a similar approach, Hadjichristofi *et al.* [208] has implemented the security applications over the ad hoc networks that provide secure routing, efficient resource management, and continuous network monitoring. Friedewald *et al.* [209] has analyzed the security and privacy over the ambient systems. The authors have presented a secure analytical framework that focuses over the privacy, control and secure system formation that is capable to remain aloof from any malicious activity. Kong *et al.* [210] has worked over the secure routing formation over the ad hoc network that can provide security against anonymous threats over the network. On demand routing is incorporated with security measure to make it capable of detecting possible threats over the ad hoc network. Paruchuri *et al.* [211] has presented the security applications against multiple network anomalies. The authors presented an efficient heuristic methodology to enhance the security based on the Bayes-Nash equilibrium. Paruchuri *et al.* [212] has also presented the security aspects over the aerial vehicles using randomization policies. Linear programming and randomized single agent based security applications are presented in contrast to UAVs. Kong *et al.* [213] has presented a security model for multilevel ad hoc network that provides highly adaptive network formations. The security enhancement is provided over the mobile ad hoc networks using aerial vehicles that form the mobile backbone. Crypto-systems and several authentication mechanisms are applied over these terminals to form a secure UAV-aided mobile ad hoc network. Some of the other work includes, time series based gene expression data by Goyal *et al.* [214], security

over spatial coordination between UAVs by Kim and Seo [215], fault-tolerant model by Freeman *et al.* [216], secure gateway selection by Luo *et al.* [217], mutual anonymity using structural-based tunneling by Naghizadeh *et al.* [218], effective multi-casting in hybrid environments by Aye and Siddiqi [219], Botnets security over UAVs by Muzzi *et al.* [220], secure transmission using identity protection over UAVs by Li *et al.* [221], and a secure Hubenko architecture for UAVs by Phillips *et al.* [222].

2.8 Problem Formulation

- (i). UAV ad hoc networks relatively being a newer family of ad hoc networks have several research issues. The network architecture available are not efficient enough to simultaneously handle transmission between ground ad hoc networks and UAV ad hoc networks. There is a requirement of an efficient network model that can form guidance system for user stations and mobile nodes on the ground to improve search and tracking scalability and performance. As discussed in the earlier sections, different research communities have highlighted different research issues related to UAVs, but, the problem formulated in this proposal is distinguished as no current solution has highlighted the issue. Further, a cooperative network between the two different setups will increase the rescue rate and will prove efficient in unmanned parameter surveillance, disaster monitoring, rescue monitoring, and various other military and civilian applications.
- (ii). In UAV guided ground based cooperative ad hoc network, it is mandatory to efficiently solve control problems like cooperative path planning, task allocation, cooperative formation control and situational awareness. Also, it is required to include certain estimation model so that redundancy in path and duplicated efforts can be reduced. This will allow selection of different search algorithms for each UAV based on its situation, thus, minimizing the uncertainties in search, tracking and acquisition. Based on these requirements, a problem is formulated to propose an efficient cooperative network framework for multi-UAV guided ground ad hoc networks. To its extension, a routing protocol is also required with the provisioning of network

services that can help in maximizing this coordination.

2.9 Objectives

The main objectives of the work are:

- (i). To review and analyze the current scenarios for UAV based cooperative networks on various performance parameters like scalability, adaptability, latency, and bandwidth requirement etc.
- (ii).
 - a. To design and implement an efficient cooperative network framework.
 - b. To design a routing protocol that can perform efficiently over proposed framework.
- (iii). To validate the proposed framework through standard/custom simulators or hardware in loop as compared to existing solutions.

The tools used in the analysis of the proposed work presented in the thesis include, NS-2, Matlab, SPSS, Open Pilot, Smart Draw, Visual Studio, and a live testbed. The details of the simulator setup and configurations are discussed in the analysis part of each chapter.

2.10 Conclusion

In this chapter, the existing UAV frameworks have been independently studied and analyzed for cooperative network formation. Various parameters and modules are identified on the basis of their functionality in cooperative network formation, and their design issues for mutual coordination in simultaneously operating dependent networks have also been discussed. Also, comparison for various models and approaches proposed over the years is presented with detailed discussion. This chapter accomplishes the first objective which includes the review and analysis of the current scenarios for UAV-based cooperative networks on various performance parameters like scalability, adaptability, latency, and bandwidth requirement etc.

Chapter 3

Cooperative Network Framework [†]

Ad hoc networks are low cost based networks that have the capabilities to carry on transmission without any centralized device. With the changing communication scenarios, it has been possible to carry out communication between the vehicles. The communication standards proposed so far are not only liable to grounded or pedestrian networks, rather they also have wide application in unmanned aerial vehicles (UAVs) networking. Further, UAVs can fly autonomously or can be operated without intervention of human operators and are versatile and flexible in implementation [71]. UAVs have wide applications such as border surveillance, traffic monitoring, disaster monitoring, and relaying ground based mobile ad hoc networks [1] [45] [74]. Single UAV based ground monitoring already exists in the literature and is practically implemented, however, these systems suffer from a major issue of scalability providing only limited surveillance [31]. Thus, in order to extend the capability of UAV controlled network, multi-UAV system is required. Also, ad hoc networking between multi-UAVs can solve the problems such as short range, limited distance coverage, network failure, dead nodes or limited guidance to mobile ad hoc networks which arise in single UAV system. UAVs in air can form a guidance swarm that can interact with the mobile ad hoc networks on ground by detecting user stations, thus, taking network surveillance to next level [55]. Multi-UAVs guided mobile ad hoc network is all about simultaneously making two types of ad hoc networks: one on the ground and another in the air, and making a common interface to allow interaction between these two ad hoc networks.

This chapter presents work on the formation of cooperative network, that allows UAVs launched from different sites to form an ad hoc network, and then perform search and

[†]Vishal Sharma and Rajesh Kumar, “A Cooperative Network Framework For Multi-UAV Guided Ground Ad Hoc Networks”, Journal of Intelligent Robotic Systems, Springer- SCIE Indexed, 1;77(3-4), 2015, pp. 629-652, [I.F. 1.178].

tracking operations to provide look-ahead information of ground data and node locations to authorized ground nodes that also forms an ad hoc network. The whole process is controlled using a neural network schema that allows UAVs to autonomously perform search operations and update waypoints based upon their learning rate. Two types of Kalman Filters have been used for formation of cooperative networks. Extended Kalman Filters allow information updating for grounded nodes and Quaternion based Kalman Filter allows estimations regarding UAV motion and waypoints.

3.1 Proposed Framework

The proposed framework aimed at establishing multi ad hoc networks and creating an interface between them to share data to perform search and tracking operations. The proposed framework is based upon neural schema that allows the formation and sharing of multiple cognitive and topology organizing maps. The efficiency of the framework depends upon updating of these maps with estimated learning rate. Extended Kalman Filter was used for estimating initial values of grounded nodes data and formation of their cognitive maps. Bayesian Kalman Filter was used for estimation of UAVs path and learning rate. The overall framework can be defined as a search model derived from integration of estimation theory and neural networks. The aim of each trial during the operation run in the framework is to provide a final search map at time stamp (T). The work included following process in the formation of final search maps:

- (i). Cognitive map of the area.
- (ii). Cognitive and topology organizing map of nodes.
- (iii). Cognitive and topology organizing map of UAVs.
- (iv). Ad hoc network formation between UAVs.
- (v). Identifying waypoints and performing non-redundant search by UAV ad hoc network.

Each of the above processes starts with estimations and then keep on updating as a cooperative framework improves its learning rate and starts obtaining correct observed values.

3.1.1 Area Cognitive Maps

Application of neural networks to derive cognitive maps requires certain centric values. In neural networks, these centric values are termed as concepts. While dealing with real world scenarios, these concept values are not derivable from existing information. For this, virtual concepts were created in the area. Thus, the initial aim was to divide the area in such a manner that it provides the location and normalized values of the virtual concepts. For this, the whole area was divided into a series of regular octagons. The reason for choosing octagons as dividing shape was that set of four regular octagons are able to form a square as a void between them. Initially, these squares were treated as virtual concepts. On final formation of search maps, these concepts pointed to location of user agents. Another reason is that during the whole mission of search and tracking, UAVs are allowed with 45° orientation in possibly eight directions. User agents can be explained as friendly nodes or various nodes found during tracking operations. Steps for deriving normalized concept values from defined area are as follows:

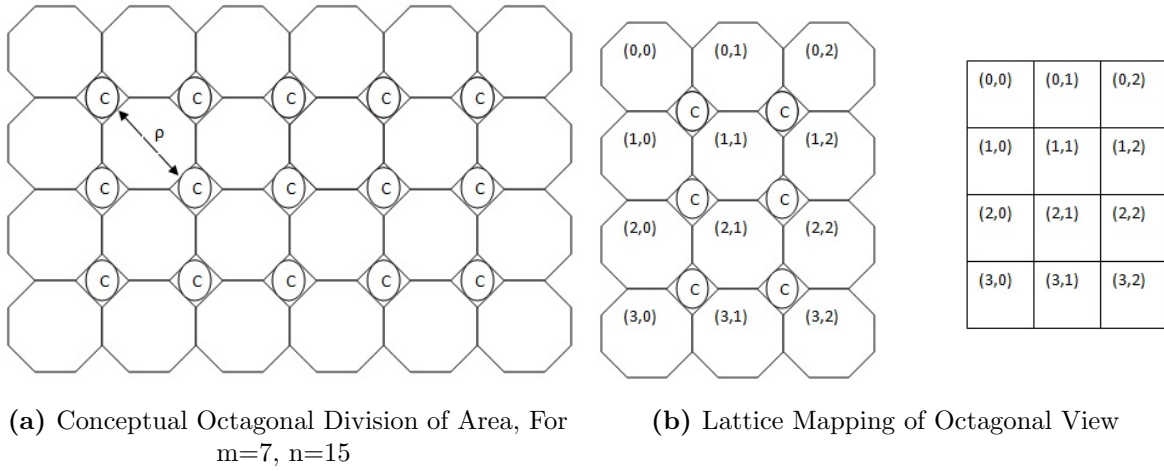


Figure 3.1: Area Division and Mapping

- (i). Area is divided into a series of octagons and virtual concepts from a given range of coordinates as shown in Figure 3.1. It is to be noted that each of the octagons should be regular with cross section range such that,

$$n = 2m + 1, \quad (3.1)$$

where n is the total number of virtual concepts and,

$$m = \frac{Area_to_be_searched}{Cross_section_Area}, \quad (3.2)$$

where $Cross_section_Area$ is the area formed with combination of four octagons and one virtual concept such that,

$$Cross_section_Area = 4s^2(3 + 2\sqrt{2}), \quad (3.3)$$

where s is the range of the octagonal cross section.

- (ii). The concept value can be selected either as the x-coordinate or the y-coordinate from location of each concept i.e.

$$C_i = Val(X_coordinate(Concept_i)), \quad (3.4)$$

or

$$C_i = Val(Y_coordinate(Concept_i)), \quad (3.5)$$

where i varies from 1 to n , and C is the assigned concept value of each of the void/concept formed in the defined area.

- (iii). Mark value of each concept is derived on the basis of its value to a lattice defined by a mapping function M_i^t at time t and i^{th} concept such that:

$$M_i^t \Rightarrow f(W_{ij}, M_i^t) \forall i, j > 0 \ \& \ i \neq j, \quad (3.6)$$

where $i \neq j$ means that in formation of a concept lattice no concept recurses and W_{ij} is the weight matrix defined over mapping function.

- (iv). Initial mapping provides with real values, and formation of cognitive map is based on the normalized value in the range $[0,1]$. These normalized values allow application of Bayesian Kalman Filter to perform concept value prediction which is given as:

$$X_i^t = H_i^t A_i^t + E(t), \quad (3.7)$$

where, X_i^t is the coordinate location along x-axis of i^{th} concept at time t, A_i^t is the mapping function that denotes the state of nature of i^{th} concept that needs to be normalized, and can be given as:

$$A_i^t \Rightarrow f(W_{ij}, A_i^{t-1}) \forall i, j > 0 \text{ \& } i \neq j, \quad (3.8)$$

H_i^t is the known value of area distribution which is a radial bias function and defined as:

$$H_i^t = e^J, \quad (3.9)$$

where,

$$J = \left[- \sum_{i=1}^n \frac{(W_i(t) - X_i(t))^2}{2\rho^2} \right], \quad (3.10)$$

where ρ is the threshold value that denotes minimum hamming distance between two concept values such that $\rho \geq 3s$, and

$$A_i^t = G_i^t A_i^{t-1} + E'(t), \quad (3.11)$$

where $E(t), E'(t)$ are normally distributed, and denote system and process errors, respectively. Here, G_i^t is the Poisson distribution, denoting the presence of concept in a particular area such that:

$$G_i^t = \frac{e^{-f(t)t} [f(t)t]^n}{n!}, \quad (3.12)$$

where $f(t)$ denotes the probability of occurrence of concept in defined area at time t .

- (v). Process error is computed using measured concept value $A_i^t(m)$ and estimated conceptual value $A_i^t(c)$, which is given by:

$$E'(t) = A_i^t(c) - A_i^t(m), \quad (3.13)$$

and system error is computed by:

$$E(t) = K_c(t)E'(t), \quad (3.14)$$

where

$$K_c(t) = PA_i^{tT}P_c, \quad (3.15)$$

and it denotes the Kalman gain for area concept. Also,

$$P_c = A_i^t(o)PA_i^{tT} + O^n, \quad (3.16)$$

where P_c denote the covariance matrix of a system error, $A_i^t(o)$ is the observed concept value at time t and O^s is the overall system residual obtained from error correction from updated data.

(vi). The updated values of A_i^t are normalized using a trivial solution as:

$$A_i^t(N) = \frac{A_i^t - \min(A^t)}{\max(A^t) - \min(A^t)}, \quad (3.17)$$

or by Bayes probability definition [223] based upon likelihood and prior. This would increase complexity of the simple normalization process, but provides with a non-trivial solution for normalization process as:

$$P(A_i^t|X_i^t) = \frac{P(E_i'(t)|A_i^t, X_i^{t-1}) \times P(A_i^t|X_i^{t-1})}{\int P(E_i'(t)|A_i^t, X_i^{t-1}) dA(t)}, \quad (3.18)$$

where $P(E_i'(t)|A_i^t, X_i^{t-1})$, $P(A_i^t|X_i^{t-1})$ are the likelihood, and prior, respectively.

3.1.2 Topology Organizing Map for Ground Ad Hoc Network

Ground ad hoc network is the combination of MANETs and VANETs nodes. The search mission aimed at providing the correct path for grounded nodes to perform search maneuverabilities and also, it aimed at providing a virtual cognitive map to each intended node regarding location of each other and waypoints of each UAV involved in search operations. The basic highlight of topology organizing map is that it helps in deciding the

location of each node w.r.t. to virtual concepts identified during formation of area cognitive maps. The steps involved in formation of neural network based topology organizing maps are:

- (i). Identify the parameters to be used for node identification and configuration.
- (ii). Select the best possible value of each configuration for every node. Also, initialize parameter priority to provide neurons with average parameter affect rate.
- (iii). Normalize the value of each configuration in range $[0,1]$.
- (iv). Create number of neurons for input layer: Number of neurons for each node are equal to the number of configurable parameters.
- (v). Constant parameters are not to be considered for developing topology organizing maps.
- (vi). Unlike self organizing maps, topology organizing maps allow hidden layer formation for mapping between the input and the output layer.
- (vii). N number of neurons of each node are mapped to M number of clusters that forms the apparent hidden layer. Further, out of M number of clusters, only one is selected as nearest concept for a particular node as shown in Figure 3.2.

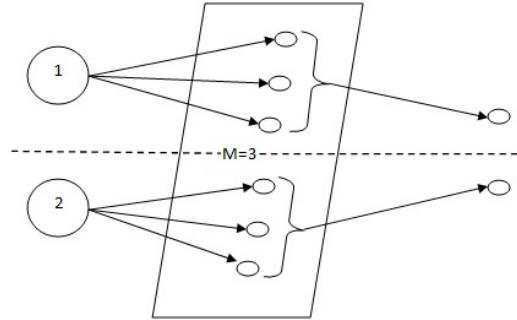


Figure 3.2: Neural Mapping of Node Parameters: $N=2$, $M=3$ (Each Node), Output=1(Each Node)

- (viii). Not more than two nodes are allowed to be present in the surrounding four octagons of a particular concept. Also, number of nodes should be less than number of octagons, and $N \leq M/2$. Maximum number of nodes allowed in an area under traversal should not be larger than 4 times the number of octagons.
- (ix). In case, two nodes belong to the same concept value, then both the nodes should not be present in adjacent octagons. The initial position of each node is based upon

the selection of octagon whose value is maximum for first two concepts i.e. if nodes 1 and 2 are to be placed around concept C_i , and node 1 has highest concept value for M_k cluster and node 2 has highest cluster value for M_j cluster, from parameter affect rate defined during priority identification stage, if $M_j \leq M_k$ then node 1 will be set first in the octagon adjoining the concept C_i and node 2 will be placed in the octagon opposite to octagon where node 1 is positioned.

- (x). In case, more than two nodes belong to same concept value, then select any two nodes arbitrarily and perform organization based upon step (ix). Now, take the next node, identify the distinguishable concept for this node from already organized nodes. Assume that C_l is the concept value different for this node from other node and l^{th} parameter of currently selected node is greater than l^{th} parameter of already positioned nodes, then check if the two out of four octagons around the concept C_l is occupied or not. If not, then set the node according to step (ix), otherwise identify next concept value and repeat step (x). Further, there can be a case that there exist no such concept value, then place the node in any octagon opposite to pre-occupied octagons with any parameter value.

Formation of Weight Matrix

Given the value of each base neuron and x, y be the initial coordinates of each node, at each change of coordinates, weight matrix is recomputed. Change here defines sufficient duration to cause alterations in results. It is derived over learning rate of each neuron defined for particular parameter value. The next step involves computing values offered to each concept location based upon range provided and is computed as normal vector distance.

- (i). Distance $D(n)$ is calculated in terms of variation of values w.r.t. weight as:

$$D(n) = \sqrt{\sum (W_{ij} - X_i)^2} \left(1 - \frac{1}{n}\right), \quad (3.19)$$

where n is the number of neurons defined for each parameter of ground node.

- (ii). Based upon the normalized values derived from equation (3.17), weight matrix is

updated using equation (3.20).

$$W_{ij}(new) = W_{ij}(old) + |\beta [\gamma_1 + \gamma_2]|, \quad (3.20)$$

where

$$\gamma_1 = (W_{ij}(old) - D(n)), \quad (3.21)$$

and

$$\gamma_2 = (X_i - W_{ij}(old)), \quad (3.22)$$

where β is the learning rate defined as the probability that denotes the rate at which correct updates can be attained based upon prior and likelihood as given in equation (3.18).

3.1.3 UAVs Topology Organizing Map

UAVs are launched from ground nodes that had already been initially deployed using a topology organizing map. Minimum number of UAVs is considered to perform search operations. Each UAV is able to perform orientation (Θ) in eight directions such that $\Theta \in (\Theta_L, \Theta_R, \Theta_F, \Theta_B) \pm 45^\circ$. For N number of UAVs launched, following computations are to be performed:

- (i). Set waypoints for each UAV.
- (ii). Define a non-redundant path of search.
- (iii). Form UAV ad hoc network to share cognitive maps.
- (iv). Transfer data to ground nodes to control maneuverability.

Waypoints are fixed using Kalman Filter based on Quaternion Algebra. Let w be the vector defining the waypoint for each UAV such that:

$$\mathbf{w} = m_1x + m_2y + m_3z + m_4, \quad (3.23)$$

where m_1, m_2, m_3 are the vectors denoting cognitive map for nodes, area and area-node integrated cognitive map, m_4 denotes the scalar that is derived from current sensor read-

ings and x, y, z are the hyper-imaginary numbers [224] .

$$\bar{w} = \begin{bmatrix} m_1 & m_2 & m_3 & m_4 \end{bmatrix}^T. \quad (3.24)$$

For efficient searching and tracking operations, collision avoidance matrix is calculated that prevents overlapping of waypoints of two or more UAVs at same time instance. Let ${}_{L}^{ij}w_{ui}^{(t)}$ be the collision avoidance matrix based on local reference (L) for i^{th} UAV such that:

$${}_{L}w_{ui}^{(t)} = {}^Hw_{ui}^{(t)} \cdot {}_{S,L}R_{ij}^H(\bar{w}_{ui})^{(t)} \cdot w_{ui}^{(t)}, \quad (3.25)$$

where ${}_{S,L}R_{ij}^H(\bar{w}_{ui})^{(t)}$ is the rotational matrix defined w.r.t. H, S, L i.e. high altitude platform, sensor readings, and local reference sensor, respectively, where

$$\bar{w}_{ui} = \begin{bmatrix} \mathbf{w} \\ 0 \end{bmatrix}. \quad (3.26)$$

Also, normalizing the value of each UAV at time t, and using quaternion time derivative, rate of change of local frame w.r.t. high altitude platform and sensor reference will be computed as:

$${}_{H,S}\mathbf{w}_{ui}^{L(t)} = \lim_{\Delta t \rightarrow 0} \frac{1}{\Delta t} \left({}_{H,S}\mathbf{w}_{ui}^{L(t+\Delta t)} - {}_{H,S}\mathbf{w}_{ui}^{L(t)} \right), \quad (3.27)$$

and

$$R_{min} \left({}_{H,S}\mathbf{w}_{ui}^{L(t)} \right) = Min \left({}_{H,S}\mathbf{w}_{ui}^{L(t)} \right)_{i,j} \text{ for } i \neq j, \quad (3.28)$$

where R_{min} is the minimum rate of change of UAVs w.r.t. local, global and sensor data.

3.1.4 UAVs Ad Hoc Network Formation and Merging of Cognitive Maps

Cognitive map formed by each node and UAV regarding search and tracking operations is limited in range and can cause redundancy in search missions. It is necessary to share information available with ground nodes and UAVs. The most efficient approach in such

type of network is to form a cooperative ad hoc network. Each UAV based upon its initial waypoints will perform collaborative rendezvous and form a link state based ad hoc network by creating a collaborative table. All the data is routed through this ad hoc formation both in the air as well as on the ground. Each of the UAV involved in formation of ad hoc network, derives an updated waypoint set from the Quaternion based waypoints. After time Δt_o , each UAV will have an updated cognitive map with new waypoints. These updates are based upon:

- (i). Number of UAVs active.
- (ii). Number of users detected.
- (iii). Number of octagons traversed.
- (iv). Number of concepts locked during search and tracking.
- (v). Process/traversal errors.

The updating process revolves around the prediction of learning rate. In generic neural networks, learning rate is pre-determined as the system equations remain the same and error rate is also constant. But, in ad hoc formation, system equations and error values keep on changing. Thus, the learning rate prediction will allow a stable cooperative network formation.

Learning Rate Estimation

Let $T_{rate(m)}$, $T_{rate(o)}$ be the measured transfer rate and observed transfer rate. Using system state based noise model [224],

$$T_{rate(m)} = T_{rate(o)} + \omega_c + b + E_s, \quad (3.29)$$

where b is the bias defined over noise model [224] and E_s is the search error,

$$\omega_c = \frac{R_{min} \left({}_H, S \mathbf{W}_{ui}^{L(t)} \right) + Transfer_Overheads(t)}{Channels_used}, \quad (3.30)$$

where

$$Transfer_Overheads(t) = \frac{N_{rt} \cdot U_{relay} \cdot N_D}{Link_speed}, \quad (3.31)$$

here, N_{rt} , U_{relay} , N_D are the number of retransmission, number of relaying UAVs, and number of data units that are retransmitted at time t . It is to be noted that with variation in system equation, there will be change in transfer rate and so will be in learning rate. From Quaternion [224] defined over rotational frame matrix, the system's state equation will be given by:

$$\bar{Z}_{ui}(t) = \begin{bmatrix} H, S \mathbf{W}_{ui}^{L(t)} \\ R_{min} \left(H, S \mathbf{W}_{ui}^{L(t)} \right) \end{bmatrix}, \quad (3.32)$$

and state noise equation will be given as:

$$Z'_{ui}(t) = \frac{N_{act} \bar{Z}_{ui}(t) \gamma_{ui}}{2}, \quad (3.33)$$

where

$$\gamma_{ui} = (T_{rate(m)} - \omega_c - b - E_s - \Delta t), \quad (3.34)$$

here, N_{act} is the number of active connections, Δt is the actual search time defined as:

$$\Delta t = \frac{E_s}{\eta} T_s, \quad (3.35)$$

where η , T_s are the network learning rate and total search time without errors, respectively. Applying the Bayesian probabilistic model defined over Kalman Filter [223], normalized learning rate can be computed as:

$$P(\eta_i^t | T_{rate(o)}^t) = \frac{P(Z'_{ui}(t) | \eta_i^t, T_{rate(o)}^{t-1}) \times P(\eta_i^t | T_{rate(o)}^{t-1})}{\sum_{i=1}^N P(Z'_{ui}(t) | \eta_i^t, T_{rate(o)}^{t-1})}. \quad (3.36)$$

3.1.5 Modified ART2 Algorithm For Combining Cognitive Maps

Adaptive resonance theory type 2 (ART2) neural algorithm provides with the approach to evaluate winning unit for locating user agents [225]. ART2 algorithm can operate over real values and then these values can be normalized for transmission between neurons of different layers. In relation to proposed schema, ART2 is modified in its approach of calculation of weight matrix and mapping function. After formation of cognitive maps for area, topology organizing maps for ground nodes and UAVs, modified ART2 is used to produce a three layer interface that serves as a virtual mapping for cooperative network. Let F1, F2, and F3 denote the neural layers for area, ground nodes and UAVs, respectively, with n number of virtual concepts in layer F1 defined over user agents, and F_M and F_N be the number of neurons in the layer F2 and F3, respectively. Unlike ART2 algorithm, reset condition is defined variably at each layer i.e. F1 with a threshold satisfying the condition that minimum hamming distance between any two virtual concepts is greater than equal to 3 times ρ . For layer F2 and F3, reset signals are defined over η_n and η_u i.e. learning rate for ground nodes and UAVs, respectively. The modified ART2 algorithm aims at solving neural schema such that $[td]_{ij} = 1 \forall i, j$. The reset condition checks for normalization of learning rate at each stage. If the learning rate is not normalized to range $[0,1]$, then all the neurons are reset at that particular layer irrespective to ART2 in which reset signal causes termination of all neurons. Following are the steps for implementing modified ART2 algorithm with a modified input/output conditions:

- (i). Mapping F3 to F2 layer: Let $B_{m,0}$, $B_{m,1}$ be the bottom-up weight matrix for ground ad hoc network and UAV hoc network, respectively, such that

$$B_{m,0} = [\alpha_1] \Rightarrow \alpha_1 I_{F_N \times F_M}, \quad (3.37)$$

where

$$\alpha_1 = \frac{\eta_n}{(1 - d_1)\sqrt{F_N}}, \quad (3.38)$$

and

$$B_{m,1} = [\alpha_2] \Rightarrow \alpha_2 I_{F_N \times F_M}, \quad (3.39)$$

where

$$\alpha_2 = \frac{\eta_u}{(1 - d_2)\sqrt{F_M}}, \quad (3.40)$$

here $d_1, d_2 \in [0,1]$ such that:

$$\frac{\eta_n d_1}{(1 - d_1)} \leq 1, \text{ and } \frac{\eta_u d_2}{(1 - d_2)} \leq 1. \quad (3.41)$$

- (ii). Input/Output of Layer Mapping: Initial input to each layer is given using ART2 algorithm [225] as:

$$w_i = \eta f(O_{layer}(i)) + I \text{ where } \eta \in (\eta_n, \eta_u), \quad (3.42)$$

after each iteration by different layers, the value $O_{layer}(i)$ is passed to next neuron of connecting layer as:

$$O_{layer}(i) = \frac{W_i + \eta_{i-1}}{N_e + \eta_i ||W_{i-1}||}, \quad (3.43)$$

where i denote the iteration, N_e is the network iteration error. Also, weight is added with a biased value obtained using a radial function as defined in equation (3.9). The top-down and bottom-up matrix are modified using the same approach of ART2 [225]. Figure 3.3 depicts neural schema based upon modified ART2 algorithm for cooperative ad hoc formation. The figure also illustrates the mapping of different neural layers defined for ground and aerial ad hoc networks. Layer F3 consists of F_M number of neurons that are mapped to F_N number of neurons in F2 layer. This is 1 to x mapping scheme where $x \in F_N$. Function F maps input/output of all the hidden layers present in layer F2. Further layer F2 is mapped to F1 layer which is based on conceptual division of area. This is 1 to 1 mapping scheme. Both F1 and F2 layers have different Reset condition which is tested on each pass. Function G maps input/output of all the hidden layers present in layer F1. F1 and F2 layers form a lower level hierarchy and is termed as a conceptual view of cooperative framework, whereas F3 layer is an external view of network hierarchy that maps to conceptual view.

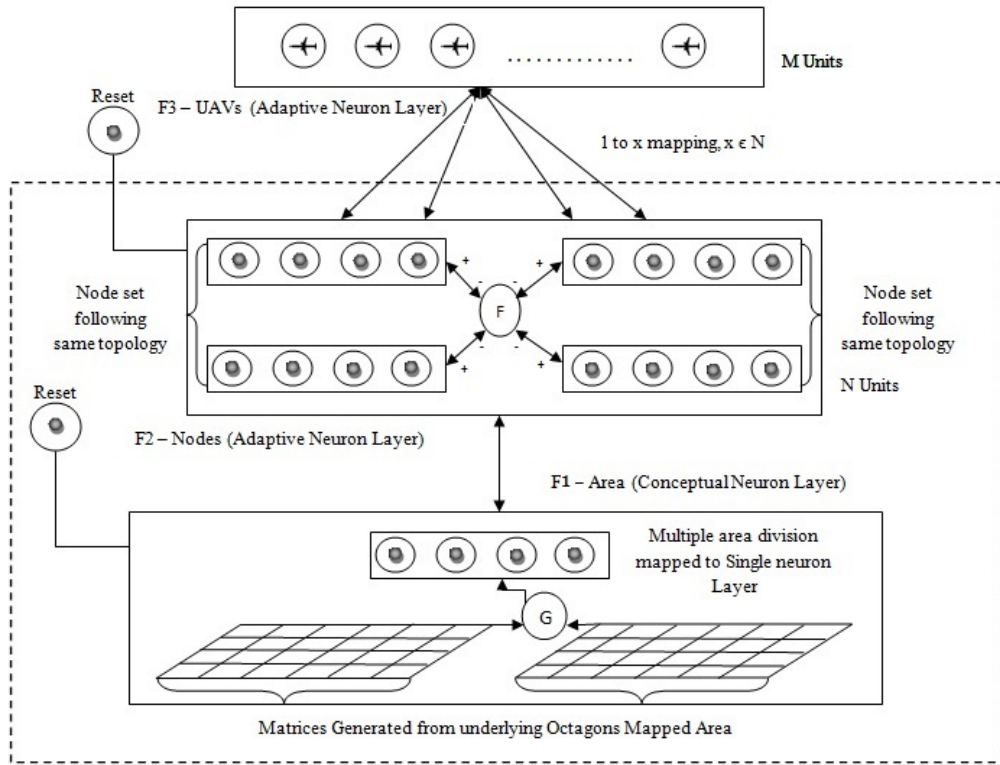


Figure 3.3: Neural Schema For Cooperative Ad Hoc Formation

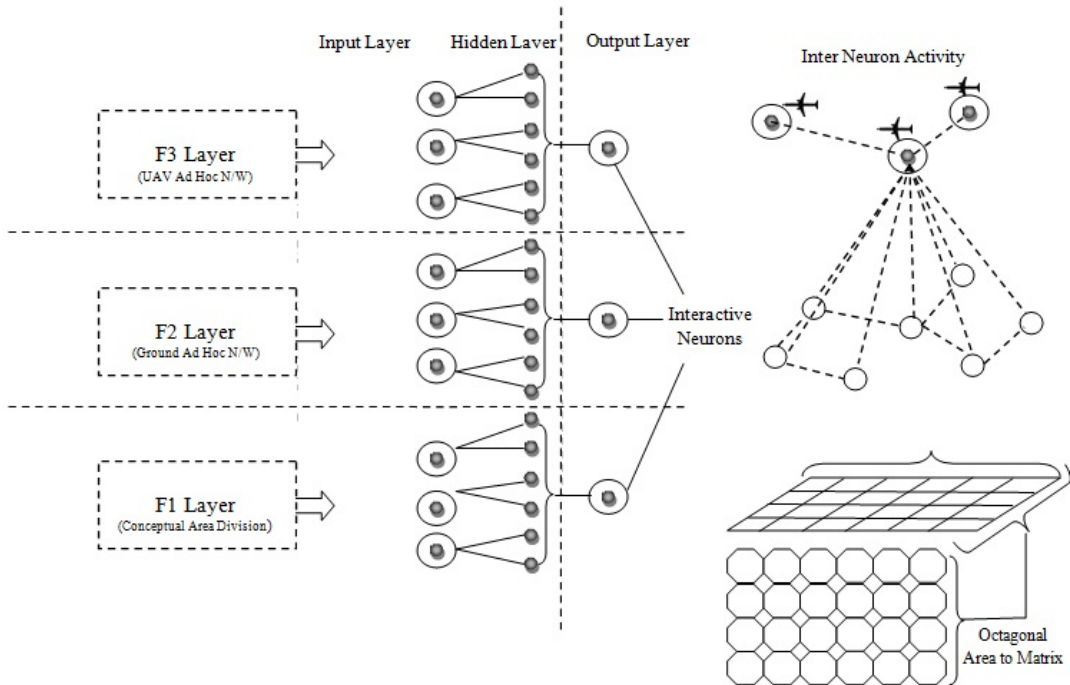


Figure 3.4: Neural Layered Structure For Cooperative Ad Hoc Formation

Figure 3.4 illustrates the final network connectivity for search maneuvers. Neurons of UAV layer perform computations for system equations and forms an updatable system by interacting with neurons of the ground nodes.

Table 3.1: Parameter Configuration For Cooperative Framework Analysis

PARAMETER	VALUE	DESCRIPTION
Dimensions	250000,500000,1000000 sq. m	Area for search and tracking
Number of Ground Nodes	15,25,30	Node sets for evaluation
Number of UAVs	5,10,15	UAVs for CSAT
Traffic Type	CBR	Traffic over TCP
Packet Size	1024 bytes	Average Packet Size
Mac Layer	802.11 b	IEEE Standard for wireless over ground nodes
Buffer Size	5000,10000,15000	Buffer Capacity of each Node
Propagation Radio Model	Two Ray Ground	Topological Propagation Model
Maximal Speed	10 m/s	Node mobility
Pause Time	2 s	Halt time for Node
Interval Time To send	2 p /s	Time to wait before sending
Transmission Range	500 m	Transmission Distance of Node
Ground Node Mobility Range	0-2.5 m	Node Steps during movement
Initial Bandwidth	256 kbps	Initial Transmission rate
Accelerator	Virtual BMA 150	Range upto ± 8 g
Variable Parameters	Seeds, Min. Achievable Rate	Base for neural Networks
Number of neurons	2 for each variable parameter	Neurons for F2 Layer
Initial Learning Rate	$\eta_n = 0.5, \eta_u = 0.5$	Learning rate before updations
Map Size For Neural Network	15,25,30	Neural Network Clustering Tool
Maximum number of hidden neurons	15	neurons ≤ 21 for effective Cognitive formation
Time Delay for Time series training	1-2 units	Time lag for generating output
Number of Training Iterations	1000	With interval of 2 epochs
Training, Validations and Testing	50%, 25 %, 25%	Time Series generation data

3.2 Performance Evaluation

A virtual environment was created to perform simulations for proposed framework using a drone path planner, NS-2 and Matlab. Drone path planner based on Matlab was used to set waypoints based upon the underlying neural schema. NS-2 was used to trace the movement of ground nodes and create an ad hoc network between them. Simulations can vary for same data set as formation of concepts is purely random and can take any of the coordinate as its initial value. This also make simulations dynamic and also, provides a virtual real environment for such task allocation problems. Simulations presented here are based on the peak values of each of the parameter i.e. 30 ground nodes with 15 UAVs, initial area of 1000000 sq.m., two neurons for each configurable parameter are taken for each node with 30 being the map size for neural based topology clustering tool. Other parameters considered and configured for creating a simulation environment are shown in Table 3.1. These parameters are defined for standard simulations performed to analyze the cooperative network formation.

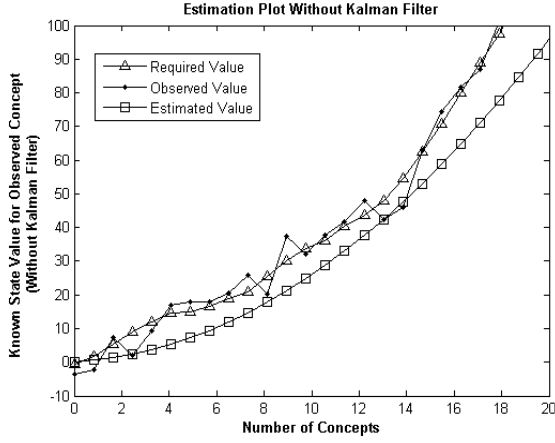


Figure 3.5: Concept Value Prediction Without Kalman Filter

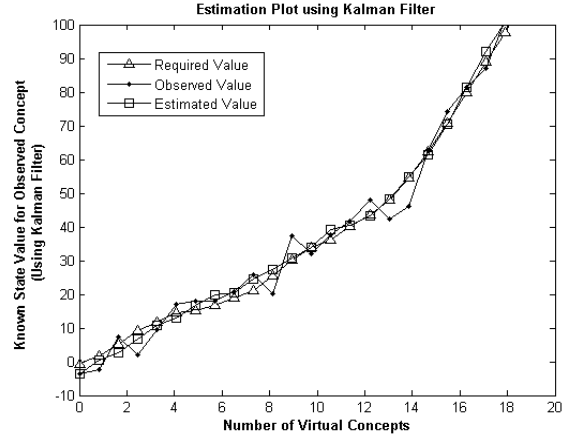


Figure 3.6: Concept Value Estimation Using Kalman Filter

3.2.1 Analysis of Cooperative Network Formation

Figure 3.5 shows plot for prediction of virtual concept formed in the area under the search. Without Kalman updates, required position of concepts cannot be estimated correctly. Whereas, Figure 3.6 shows an estimation based upon use of Kalman based updates from underlying cooperative network. The estimated and the required value almost coincides with each other, showing that Kalman Filter provides with the correct reading in terms of state values for particular concepts. It can be noticed that as more and more updates are received during the initial free run of UAVs, the graph becomes more and more steeper with values overlapping with each other representing the exact state values that are required for efficient search trajectory. With more numbers of trained values and more number of iterations, such behaviour of graph can be attained at lower computational level also. The training set used for simulations vary from 200 to 1000 iterations with epoch varying from 2 to 4. Figure 3.7 shows the coordinated location of each virtual concept formed during a trial state of the cooperative network. These locations are updated as soon as UAVs start locating user agents such that, each concept points to the location of tracked user agents or ground nodes. Figure 3.8 represents the topology organizing map for node neurons. During simulations, the number of seeds and minimum achievable rate are considered to be variable parameters. Thus, two neurons for each of these parameters are created for every node. The figure shows the hamming distance based connectivity

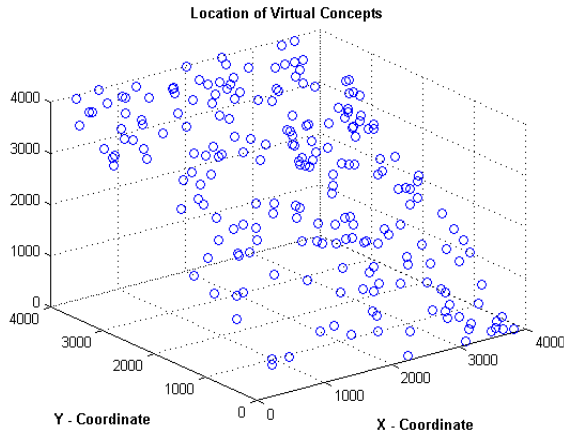


Figure 3.7: Location of Virtual Concepts

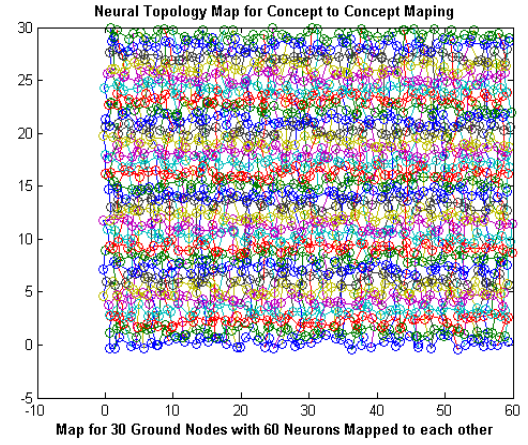


Figure 3.8: Topology Organizing Map for Node Neurons : Seeds Mapped to Min. Achievable Rate

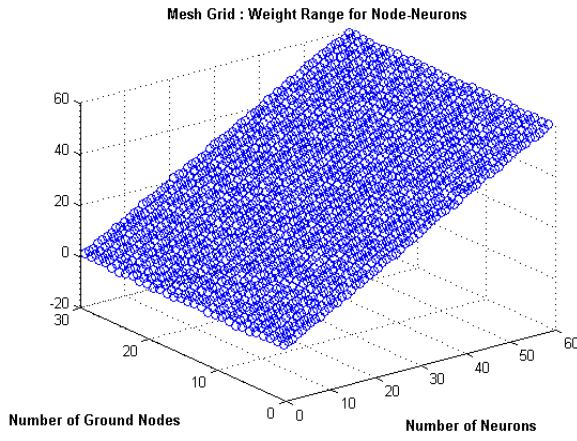


Figure 3.9: Average Weight Mapping For Seeds and Min. Achievable Rate

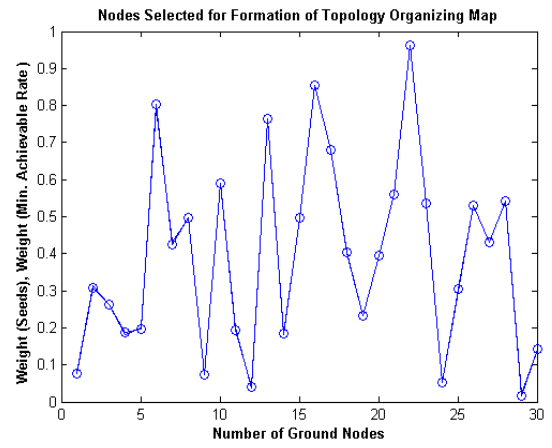


Figure 3.10: Neural Map For Node(Neurons)-Concept Mapping

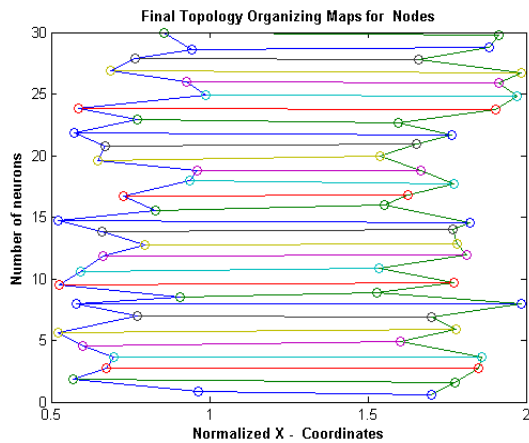


Figure 3.11: Final Topology Organizing Map for Nodes

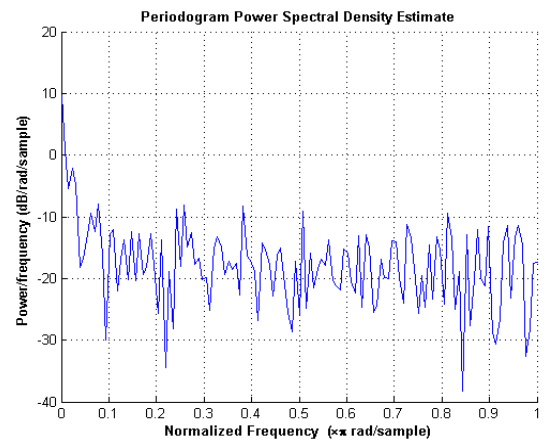


Figure 3.12: Power Spectral Graph for Normalized Concepts

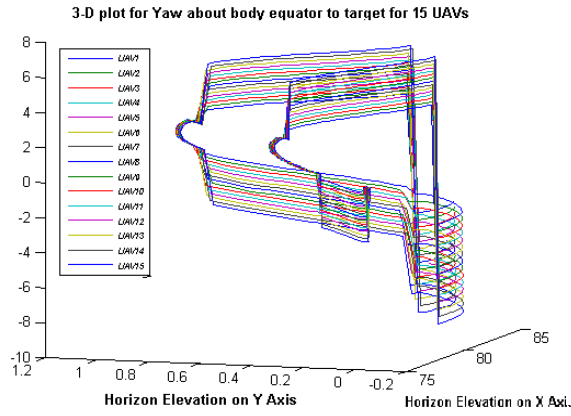


Figure 3.13: 3-D Plot For Yaw (15 UAVs)

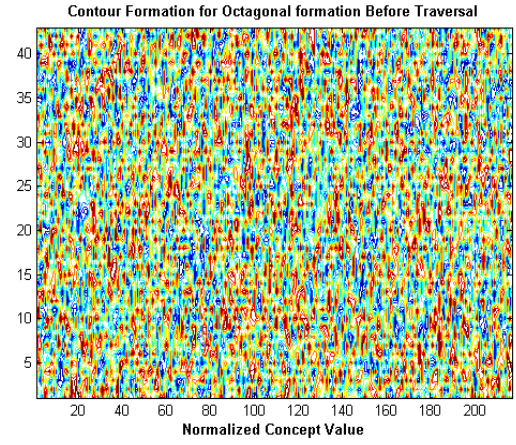


Figure 3.14: Contour Representation of Area Before Traversal

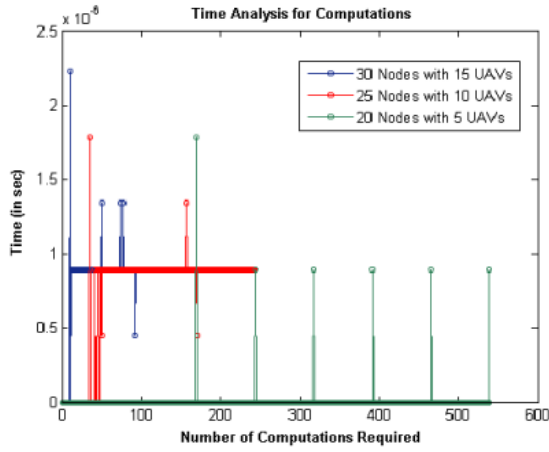


Figure 3.15: Computational Time Analysis

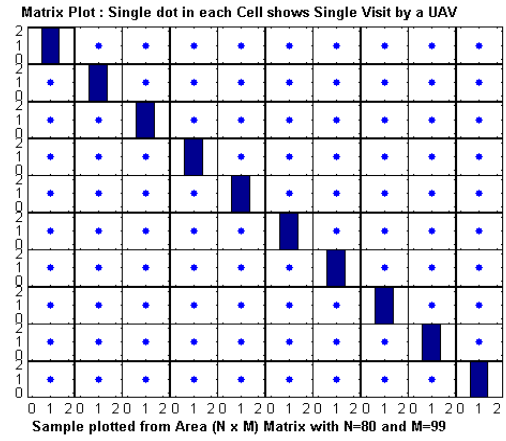


Figure 3.16: Final Area Lattice After Traversal

between each of these neurons. Figure 3.9 shows the average weight mapping between the number of seeds and minimum achievable rate for each node. Figure 3.10 plots the neural map of node neurons and underlying virtual concept mapping. Correct division can be evaluated only if this mapping follows a linear relation. Figure 3.11 shows the final topology organizing maps formed after getting updates from each of the UAV that formed an ad hoc network. Unlike self organizing maps, these maps are derived using hidden layer of neurons with the number of hidden neurons less than 21. Figure 3.12 displays the periodogram for power spectral analysis based on transmitting frequency for normalized concept value with respect to normalized frequency requirement. Figure 3.13 presents UAVs motion analysis, which is recorded during search and tracking operations. The input values for UAVs are provided using standard gyros and accelerometers. Yaw,

pitch and roll allows in prediction of behaviour of search operations, and it also allows estimation to be made regarding the physical conditions of geographical territories.

Figure 3.14 shows the contour representation for initial area. Different drift and colours in contour represents the non-traversed zone of the area. It distinguishes the searched area from non-searched territories. The contours are formed using normalized values of octagonal formation and virtual concepts after obtaining final search maps. The computations required in the formation of cooperative network is very important. More computation time might lower the network performance. Similarly, more number of computations per unit might require extra hardware in terms of energy consumption for continuous operation of the network. Thus, life time of such network depends upon computational analysis of network formation and network operations. Figure 3.15 shows the computational analysis performed for 20, 25, 30 nodes, each with number of UAVs, 5, 10, 15, respectively. Time analysis shows that with increasing number of UAV, number of computations decreases, but time of computations increases. This means that with more number of ground nodes in ad hoc network formation, tasks get cooperatively distributed amongst nodes, thus, requires a lesser number of computations. However, more updates are received from each other during such cooperative task allocations, this increases the computation time. During search and tracking operations, the main focus is to provide UAVs with non-redundant waypoints. This makes searching fast, and consumes less computational time. Also, an output formation is required that can highlight the octagons mapped with $N \times M$ lattice which are traversed more than once as shown in Figure 3.16. Each cell represents the underlying conceptual-octagon division. Presence of a single dot in each cell shows that each octagon has been traversed only once.

3.2.2 Analysis of Neural Schema used for Cooperative Network

Self-organizing maps (SOM) for the area's conceptual division were also realized using Matlab's clustering tool designed for neural networks. Figure 3.17 shows the SOM with neighbour distance formed using trial weight using bias rules with 200 iterations, map size

of 20 and 500 training set. The neural schema was also fed into time series to evaluate errors that are encountered during network formation. Figure 3.18 represents the error histogram for neural mapping between normalized concept and selected neuron value. These analyses were performed using 50% training, 25% validations and 25% testing. The errors are obtained over the difference between output obtained and set target values. Figure 3.19 represents the response time of neural mapping for training, validation and testing of targets and outputs. Figure 3.20 and Figure 3.21 show the autocorrelation and input error correlation for this mapping schema, respectively. Since, each of the units in network formation is based upon the updates process, zero correlation is attained only when no updates are either obtained or used for performing neural computations.

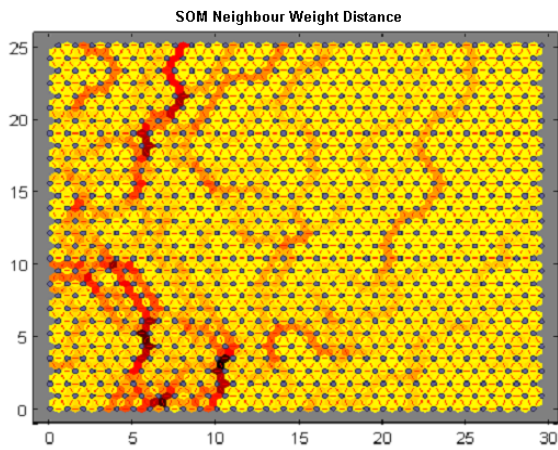


Figure 3.17: Self Organizing Map for Virtual Concepts

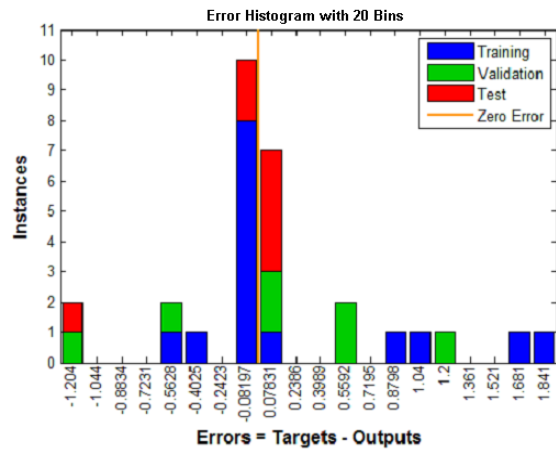


Figure 3.18: Error Histogram For Neural Mapping b/w Normalized Concept and Selected Neuron Value

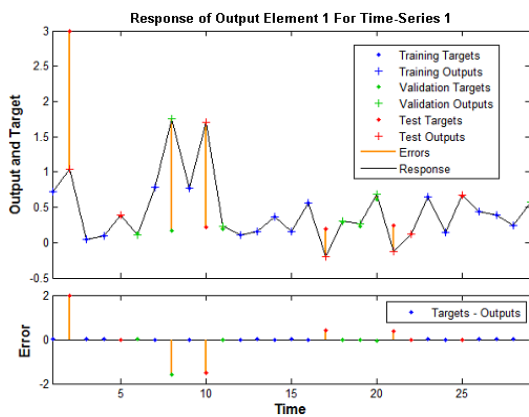


Figure 3.19: Response For Neural Mapping b/w Normalized Concept and Selected Neuron Value

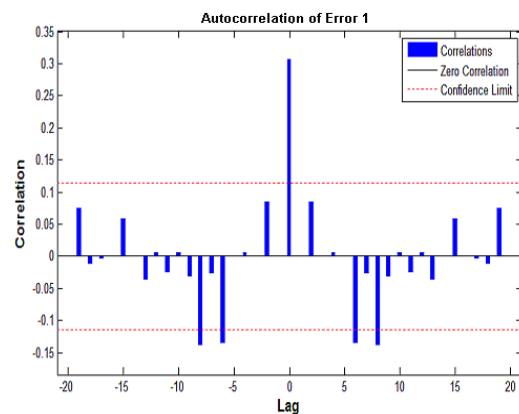


Figure 3.20: Error Auto-correlation For Neural Mapping b/w Normalized Concept and Selected Neuron Value

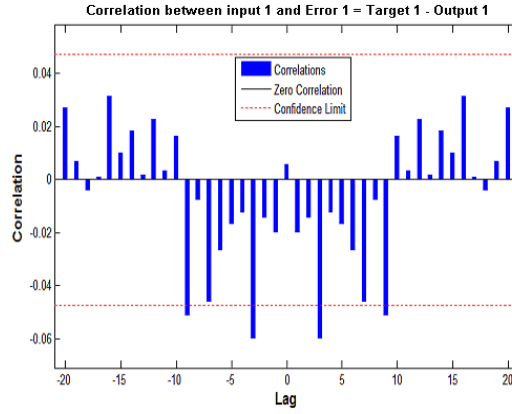


Figure 3.21: Input Error Correlation for Neural Mapping b/w Normalized Concept and Selected Neuron Value

3.3 Conclusion

This chapter presents an integrated neural approach for combining two different ad hoc networks, namely, ground ad hoc networks and aerial ad hoc networks. Kalman Filter is used to perform periodic updates that are used to enhance the learning of nodes. The neural model used for interfacing allows formation of a cooperative network framework. A task of non-redundant search is assigned to UAVs for tracking ground nodes with less complexity. The proposed framework provides a non-redundant search with less search time and less computations. The proposed framework provides large scalability in terms of cooperative search and task allocations with increased adaptability and lower latency. This chapter accomplishes the objective-2 partially which includes the design and implementation of an efficient cooperative network framework for the multi-UAV guided networks.

Chapter 4

Routing and Cooperative Relaying Over UAV Guided Networks [†]

In this chapter, a new routing protocol for relaying over cooperative network framework is presented. This protocol provides a support for transmission between the ground nodes and the aerial nodes in the ad hoc mode. The protocol is developed using the cooperative network framework as its backbone. This protocol is termed as “OFFRP: Optimized Fruit Fly Routing Protocol”.

The proposed protocol utilizes the concept of nature inspired algorithm. It is developed using the properties of fruit fly optimization algorithm that utilizes smell index to find food sources. The similar concept is used by the nodes to find path treating other nodes as food sources. The proposed nature-inspired (Fruit Fly Optimization) routing protocol provides an optimized selection of a path between the source and the destination. It also provides congestion control. Further, a prediction model is formulated based on the parameters dominance which is computed using random forest algorithm. The proposed congestion free protocol selects a path with less probability of occurrence of congestion and with efficient routing facility.

4.1 Network Model

Model considered comprises of two variedly operating networks with N number of UAVs and M number of ground units. This distributed ad hoc network operates in coordination

[†]Vishal Sharma, Ilsun You, Rajesh Kumar and Varun Chauhan, “OFFRP: Optimized Fruit Fly Based Routing Protocol With Congestion Control For UAVs Guided Ad Hoc Networks”, International Journal of Ad Hoc and Ubiquitous Computing, Inderscience-SCIE Indexed [accepted], [I.F. 0.554].

with each other and thus, forms a coordinated network with the requirement of intra-network and inter-network routing. The focus of this formation is to unite the routing structure of these two variedly operating networks, and form a protocol which is capable of providing congestion free routing. The whole model is governed by the fruit flies and their smell index S^I . While searching for hops or intermediates, source nodes act as object of fruit flies, whereas the nodes to be considered for relaying are the object of food sources. All the nodes acting as food sources compute their smell index, which is used by the source nodes to derive the fruit fly index FF^I towards each node. This FF^I is the basic criteria used in the selection of relays over the both intra and inter networks. Flies with FF^I lower than the certain threshold values keep aloof from routing. With the facility of channels, FF^I will vary for each of the incoming connections. For connection between node i and node j , $FF_{i,j}^I$ will be computed at node i for $S_{i,j}^I$ received from node j such that:

$$FF_{i,j}^I \propto S_{i,j}^I, \quad (4.1)$$

$$FF_{i,j}^I \propto \frac{1}{\beta_{i,j}}, \quad (4.2)$$

and,

$$FF_{i,j}^I = c_{i,j} \frac{S_{i,j}^I}{\beta_{i,j}}. \quad (4.3)$$

Here, c is the routing constant such that $0 \leq c \leq 1$. c varies with the number of incoming requests to a particular food source. β is the deviation of the network from its regular operations computed in terms of transfer rate, delivery ratio and jitters. Smell index S^I is the objective function with functional dependency given as:

$$S^I \longrightarrow (L^Z, S^E, P^E, R^T, L^S, R^D, L^A), \quad (4.4)$$

where L^Z is the localization defined in terms of distance between the two nodes, S^E is the number of seeds to the relay nodes, P^E denotes the corresponding peers to the relay node, R^T is the transfer rate provision by the node, L^S is the link speed, R^D is the drops over the nodes, and L^A is the link availability between the two nodes. For intra-UAV network formation, L^Z does not affect the selection of relays as line of sight is available most of

the time. Whereas, if line of sight is not available, previous connections are terminated and new peers are identified. For intra-network connectivity between the ground nodes, $L^Z \leq R^R$, where R^R is the radio range configured for the all the nodes. Lower the distance, higher will be the smell index, thus, for a given network, $S_{i,j}^I \propto \frac{1}{L_{i,j}^Z}$. For an efficient connectivity, node with more number of peers P^E and seeds S^E will have a better degree of connectivity and can offer a relatively higher number of transfers by selecting alternate paths in case of failures. Also, for enhanced connectivity, $S^E \geq P^E$. Networks with a single source or single hosting servers have limited seed count. However, in UAVs the data are in the form of information relatively collected to solve a particular task; therefore, in that case, probability of availability of seeds is more. A UAV connecting data path to the ground nodes must have a high rate for seed to peer as this will not limit the channel utilization. This will provide multiple transfers via multiple channels. Hence, seed to peer ratio can itself be treated as an optimization problem in such distributed networks. For the considered mode, maxima of seed to peer ratio S^P is considered i.e.

$$\lim_{f_s \rightarrow \max.} S^P \rightarrow \text{maximum}, \quad (4.5)$$

where

$$S^P = \frac{S^E}{P^E}, \quad (4.6)$$

and f_s are the food sources or the relay nodes. In a network, the transfer rate is directly related to network throughput. A threshold is defined for a network to sustain. Thus, at time t , if node i is capable of providing a transfer rate R^T greater than equal to the minimum prescribed rate of the overall network, smell index will have relatively higher value. The flies based network model forms a feedback mechanism throughout the connectivity. The feedback mechanism helps assign a drop index to each food source/relay node. A node within the range, but with poor connectivity and lower seeds to peer ratio will have higher drop value. Also, this accounts to the number of packets previously dropped by the node. This drop index accounts for overheads in the network. A threshold is defined for computing this drop index. During transfers, drop percentage should not

be greater than the threshold percentage γ i.e. in a network under loss,

$$R^D \left(= \frac{\text{packet_loss}}{R^T} \times 100 \right) \leq \gamma. \quad (4.7)$$

Another key aspect of the model considered is the $\text{link}_{i,j}$ which is the functional dependency over link speed L^S and link availability L^A . Better availability of the link speed and connectivity will enhance the smell index of the node. Link availability is affected by the mobility of nodes and is much more affected if line of sight LOS is not available. For network, if LOS is available, then link availability is only affected by the link speed. In case of UAV oriented networks, KU-band is used to judge the connectivity between the nodes. In a network with N number of UAVs, $L_{i,j}^A \propto$ number of channels available for connectivity to node i . More the number of channels available, more is the link availability. With UAVs as node, corridor available for connectivity directly affects the transmission. Thus, for such networks, link availability will be computed as:

$$L^A = \frac{1}{e^{\frac{1}{C^H} \sum_{i=1}^{C^H} (Pr_{lost} - Pr_{cmax})^2}} \times (C^O \times L^S), \quad (4.8)$$

where C^H are the channels available, C^O are the corridors available for connectivity, Pr_{lost} is the probability of lost packets and Pr_{cmax} is the maximum probability allowed to drop over a distributed network. Final smell index S^I is considered to be exponentially increasing with proper flow between the two nodes. This exponential growth principle is motivated from TCP's exponentially growing congestion window rule. Using this rule, $S_{i,j}^I$ will be computed as:

$$S_{i,j}^I = e^{\left(\frac{L_{i,j}^S + R_{i,j}^T}{\|W\|} + \frac{S_{i,j}^P}{\eta_1} \right)^2} + e^{-\eta_2 \left(\frac{L^Z + R^D}{L^Z R^D} \right)^2}. \quad (4.9)$$

The smell index forms the objective function which regulates the selection of relays between the source and the destination. Nodes with maximum values are selected as relays and used for feedback mechanism.

4.2 Proposed Protocol: OFFRP

The proposed congestion free routing protocol utilizes the properties of a fruit fly to form a network with a feedback mechanism. The proposed protocol is capable of handling delays which might occur if network suffers from traffic congestion. The nodes acting as source are the flies, whereas the relays to be selected for path formation between the source and destination are treated as food sources. Thus, a node to itself is a fly whereas for others it is a food source. In the traditional fruit fly algorithm, random search is used to initialize the location of the flies and then smell based objective function is used to identify the location of food sources. Using the similar concept with certain modifications, OFFRP is capable of providing congestion free routing over UAVs guided networks. The major operations of the protocol include following phases: route discovery, route maintenance, which are characterized by controlling the size of congestion window (*cwnd*) and predicting the network losses that might occur due to uncertain congestions. This phase uses a probabilistic model to quantify the occurrence of network congestions and varies the window size with respect to it. Next phase is the route recovery. In a network having nodes operating under uncertain environmental conditions, chances of failures are predicatively high. In such scenarios, the routing should be robust and fault tolerant with capability of handling uncertain failures without affecting the performance of the network. The functionality of the proposed protocol is given below:

4.2.1 Relay Selection and Route Discovery

Route discovery forms the basis of every routing protocol. Optimized selection of path can form an efficient network with higher performance. The proposed OFFRP uses fruit fly concepts for identifying the relays with the maximum facilities for forwarding the traffic towards the destination. The whole process of route discovery and relay selection is governed by the smell index S^I . This index value is used to prioritize the nodes acting as intermediate for routing. The steps for relay selection and route discovery are given below:

- (i). After initialization of the network with source and destination, route discovery starts with sending hello packets as broadcast up to the maximum radio range of each node. These hello packets are generated by each of the node acting as fly, which is in the process of maintaining a routing table for data transfers. Hello packets are marked with special signatures of the nodes with a particular fly number F^N . This fly number is used to identify the particular node in the whole network. Another indexing used to identify the food sources is the fruit source number F^S . Each time a node receives a hello packet from a node, it uses the prescribed network model equations to compute its smell index. In reply to the hello messages, these smell indexes are also delivered along with the acknowledgment to the requesting node.
- (ii). On receiving the acknowledgment from a particular food source, flies compute fruit fly index FF^I for the particular link between itself and the food source from where acknowledgement is received. Now, the nodes acting as source arrange the relays available in decreasing order of their fruit fly factor. The greater value of FF^I increases the chances of a node to be selected as relays. Here, two different approaches can be used for selection of relays. A node can directly select the relay with highest FF^I value, or it can again beacon the particular relays and request for its next neighbours FF^I value to select the distance based nodes towards the destination. This is, however an optimal approach, but adds up to the overheads of multiple requests and wait for next acknowledgement. In the proposed approach, selection of nodes is based on the highest fruit fly index value.
- (iii). While other nodes finalize the path between themselves and next hop, all the previously connected nodes share beacons to form a feedback network which helps in identifying the current state of the network. This state of the network is used during route maintenance phase to recover the network from uncertain failures.
- (iv). After each iteration, rather than using stand-alone acknowledgments, much information regarding utilization of channels, number of incoming connection request and possible connections that can be handled by the relays are transferred towards the source node. This helps in recomputing the updated fruit fly index using the deviation in network operations β (Ref. Section 4.1, equation 4.3).
- (v). Unlike traditional routing protocol, two major tables are used by all the nodes

involved in relaying of data. One is the relaying table to maintain the forward connectivity and second is the flash feedback table used to store the path for acknowledgement and feedback mechanisms. A routing table is altered with alterations in relays and movement of nodes, whereas flash feedback table is initialized after each iteration. This is a temporary table and does not require excessive memory every time.

- (vi). While in operations, it is likely to occur that the same relays are requested by two or more nodes. In that case, if the nodes are capable of handling the multiple connections, their smell index is recomputed and retransferred to the source node which updates its routing table and takes decision of selecting the particular node again for transfer based on the network threshold criteria.
- (vii). In order to optimize the route discovery and relay selection process, a threshold value can also be set for selection of relays. This would greatly reduce the complexity of the selection algorithm. However, in such scenarios, route maintenance and recovery would require extra algorithms to make network aloof from delays. Thus, using the threshold based mechanism is used on the basis of particular application oriented network formations. The algorithm for route discovery and path selection is given in Algorithm 4.1.

4.2.2 Route Maintenance

After finalizing the route between the source and destination, route stabilization is the most important aspect to be handled at that moment. A continuous connected route will offer better transfer rate and will enhance the performance of the network. Each time the nodes update their moment, their smell index changes. Thus, route maintenance has to be swift enough such that delays which might occur during these hand-offs should not affect the network performance. Route maintenance in OFFRP is governed by three basic principles. First is to update the entries in the routing table if there is any shift in values of smell index for already registered relays, the second is the identification of drop probability during the current network operations, and third is the handling of congestion window to eliminate the occurrence of the congestion without affecting the

Algorithm 4.1 Route Discovery and Path Selection

Require: $Nodes \rightarrow Up, source, destination, F^N, F^S$
Ensure: $relaying_table, flash_feedback_table \leftarrow initialized$
while $route(source \leftrightarrow destination) \neq true$ **do**
 $j = F^S$
 $i = 1$
 while $i \leq j$ **do**
 $i = F^N$
 broadcast(hello) $\rightarrow F^S$
 $ACK_i = S^I \leftarrow smell(F^S)$
 $FF_i^I \leftarrow FF(ACK_i)$
 $i = i + 1$
 if $ACK_i = true$ **then**
 continue
 else
 break
 end if
 end while
 $relaying_table \leftarrow FF^I$
 update $flash_feedback_table$
 if $(FF^I \leftarrow relaying_table) \geq FF_{TH}^I$ **then**
 $path \leftarrow (relaying_table)^P$
 else
 discard
 end if
end while
transmit()

network performance. The steps involved in route maintenance are as follows:

- (i). Route is maintained based on the feedback attained from the acknowledgments and the periodic updates a network undergoes. These alterations are identified as network updates and smell index of the corresponding flies are updated based on their current state.
- (ii). Stack mechanism is used to identify the flies that undergo change in current operations. A separate beacon is broadcasted by all the nodes that undergo changes in their current operations.
- (iii). This mechanism is capable of handling network updates effectively with provision for prediction and estimation of congestion window size for the current TCP-oriented transmission.
- (iv). This phase also manages the maximum number of food sources that can interact with the flies i.e. a watch-dog policy is used to keep allocation of relays to a node under certain limit. For a network undergoing maximum transfers, number of relays allocated to a node should not be greater than $\frac{N}{C^H}$ provided that number of channels C^H should not be greater than N .

Further, route maintenance provides the congestion control mechanism which prevents network from undergoing a state of congestion and also predicts the congestion rate based on the feedback mechanism of the flies. Existing TCP-protocols can be used to operate with the proposed routing protocol to maintain the size of the congestion window applicable enough to handle routing overheads. However, being a relatively modified nature inspired routing protocol, a modification in congestion window initialization and alterations is performed. The proposed approach uses two-way prediction and control mechanism. This approach does not randomly decrease the size of congestion window neither decreases or increases its size by constant. Rather, it uses a smell index to fix the initial length of the congestion window, and also affects the decrease and increase of window size using the smell probabilistic model.

The smell probabilistic model identifies the congestion possibilities based on the smell index of node and their feedback mechanism. A food source or relay node is under the threat of facing the congestion after certain time interval when the possibility of channel utilization approaches maximum. The prediction of this possibility helps in identifying the rate of congestion and required alterations with respect to smell index. The smell probabilistic model operates on the basis of parameter dominance which is computed using random forest algorithm [226]. This random forest algorithm forms the backbone of the network in identifying the probabilistic change in index w.r.t. certain parameters. The dominance is computed after certain interval which is equivalent to the propagation time of successfully delivered packets. The smell probabilistic model aims at lower possibility of congestion to occur in order to sustain the network for longer duration. During the initial phase, the congestion window is operated over only round trip time RTT . But, after receipt of first acknowledgement, the size of the congestion window is continuously increased towards the maximal limit by adding smell constant to it. This increase in congestion window size $cwnd$ is given by:

$$cwnd = cwnd + \frac{RTT}{e^{S^I}}. \quad (4.10)$$

Along with this increase in the congestion window size, the nodes predict the dominance of parameters affecting the transmission rate and control them beyond a limit to handle

Algorithm 4.2 Congestion Prediction and Control

Require: $Nodes \rightarrow Up, source, destination, F^N, F^S$

Ensure: $relaying_table, flash_feedback_table \leftarrow initialized$

```

while transmission=true do
   $flash\_feedback\_table \rightarrow update$ 
   $fetch^i \leftarrow relaying\_table$ 
   $K \leftarrow path(source \leftrightarrow destination)$ 
  while  $L \leq K$  do
    if  $FF_L^I \geq FF_{TH}^I$  then
       $continue$ 
    else
       $fm \leftarrow fail(L)$ 
       $retries(fm)$ 
      if  $active(fm) \rightarrow Up$  then
         $connect\_path \leftarrow fm$ 
         $update\ relaying\_table$ 
         $cwnd = cwnd + \frac{RTT}{eS^I}$ 
      else
         $initiate\ beacons$ 
        if  $network\_recovered = true$  then
           $continue$ 
        else
          if  $R^T(cwnd_L) < R^T$  then

$$D^P \leftarrow \left[ P(D^P/S^I) = 1 - ||e^{\left(\frac{D_S^P - D_C^P}{S^I}\right)^2} || \right]$$


$$cwnd = cwnd - \frac{RTT}{P(D^P/S^I)}$$

          else
             $re\_initialize\ beacons$ 
          end if
        end if
      end if
    end if
     $L=L+1$ 
  end while
end while

```

the delays and congestion before their occurrence. This phase is termed as identification of congestion using feedback mechanism. The procedure for congestion prediction and control is given in Algorithm 4.2.

Prediction and Identification of Congestion: Each of the nodes sends its smell index to the dominance computing model which is already embedded in node configurations. With functional dependencies as input and smell index as output, each of the parameters dominance is computed using random forest algorithm based on the mean square error and node priority. Congestion prediction is based on the network conditions in the congestion state C^O and the current state S^C . The dominance set formed in the both states is exclusively matched to identify the chances of congestion by identifying the differences in the parameter values in both the states. In case of matching, the network transmitter is intimated well-ahead of the collision and the system state is altered by gradually altering

the state parameters towards the initial stabilized state. However, there might be cases when state matrices are mutually exclusive and still congestion occurs. In that case, a probabilistic model based on the smell dominance is used to alter the size of the congestion window. Let F^R be the random forest model such that:

$$S^I(= f(L^Z, S^E, P^E, R^T, L^S, R^D, L^A)) \longrightarrow F^R. \quad (4.11)$$

Now, using the decision tree approach of random forest algorithm, priority matrix is formed for the input made to this model such that:

$$F_{output}^R = (L^Z, S^E, P^E, R^T, L^S, R^D, L^A)^P, \quad (4.12)$$

where P marks the priority order of the parameters over which smell index depends. Now, at the time of congestion, decreasing probability is computed which alters the dominant parameters to control the congestion. This probabilistic decrease is dependent upon the C^O and S^C . Let D^P be the dominant parameter of the network. Now, probabilistic variation required is computed as:

$$P(D^P/S^I) = 1 - ||e^{\left(\frac{D_S^P - D_C^P}{s^I}\right)^2}||, \quad (4.13)$$

where D_S^P is the value of the dominant parameter in the current state, and D_C^P is the value of the dominant parameter during congestion state. Using this probabilistic variation, the window size of the network under congestion will be varied as:

$$cwnd = cwnd - \frac{RTT}{P(D^P/S^I)}. \quad (4.14)$$

Also, it is to be noted that the value of the parameter is not decreased beyond the certain limit as this may hinder the network performance. Rather, after reaching minima for a particular parameter, another dominant parameter is considered and altered using similar probabilistic variation. These steps are repeated till the network congestion is handled and network attains a stabilized congestion free state. This state based smell variation model helps in attaining a congestion free routing between the source and the destination.

Algorithm 4.2 presents steps for route maintenance and congestion control.

4.2.3 Route Rehabilitation

A network undergoing uncontrollable changes requires efficient recovery procedure for continuation of its operations. The route rehabilitation is carried using the same procedure as that used in route establishment via transmission of HELLO messages and acknowledgement. The route rehabilitation is carried by five possible mechanisms. First is the congestion window management. In case of congestion based link breakage, congestion prediction and probabilistic model defined in sub-section 4.2.2 is used. Second is the retransmissions. The number of retransmissions is set to an upper limit of half the total sum of smell index of the overall network. Third is the table broadcast. This allows overriding of table in existing data structures for selected nodes. This saves a lot of recompilations, however, chances of recovery by this approach is minimum. Fourth is the retracing the nodes and finding alternate paths between the source and the destination. This process is equivalent to route discovery process and has same complexity. Fifth strategy is used when the network does not respond at all. This is done by recomputing and reconfiguring the network nodes. This strategy is not the part of the proposed routing protocol rather a formal approach in the worst scenarios.

Table 4.1: OFFRP-Relaying Components

Parameter	Description
<i>Relay_location</i>	Coordinate location of next node
F^N	Fly Number
<i>Seq.No.</i>	Routing Sequence Number
<i>LastUp</i>	Up Time of Node
S^I	Smell Index
FF^I	Fruit Fly Index

Table 4.2: OFFRP-Flash Feedback Components

Parameter	Description
<i>Source</i>	Initial Requesting Node
<i>Next_hop</i>	Next Intermediate Node
<i>Relay_Count</i>	Number of Intermediate Nodes
<i>Interface_Id</i>	Corridor Selected For Transmission
S^I	Smell Index
<i>Incoming_requests</i>	Requests For Coordination

4.2.4 Data Structures and Beacon Configurations

Proposed routing protocol (OFFRP) uses ideology driven from exiting fruit fly optimization algorithm which uses smell index to identify the neighbouring node which can relay efficiently. The proposed approach does not consume much memory as it uses a hybrid

mechanism with alterations in existing routing protocols. OFFRP can easily be incorporated into any of the existing routing protocol by alterations to its routing table and beacon configurations. Initial table used for routing is the smell index table. This table tracks the smell index computed by each of the nodes which is shared during “hello” exchanges between the nodes. The smell table used to identify the efficient nodes available for relays is shown in Table 4.1. This table tracks the values and places them in descending order of fruit fly factor FF^I . This table also tracks the record for last up time $LastUp$ for the nodes tracked. Also, a sequence number is allocated to each of the nodes. Another table used for transmission is the fruit-source mapping routing table or the flash feedback table. This table finally records the actual path to be used for transmission between the source and destination. All the entries constraint to the final movement of flies are recorded in this table. This table also keeps an account for lost food sources or relays. Numbers of channels available and number of incoming requests also record in this table. This table gets updated after each iteration between the source and the next hop selected for relays. This table is also used for feedback mechanism required to control the congestion and predict the next state for possible congestion avoidance. The components of this fruit-source mapping routing table is shown in Table 4.2. The whole network model is driven by the beacons shared between each network node. These beacons include specially configured packets used in fruit fly based routing. The components of packet format used in the proposed fruit fly based routing are shown in Table 4.3.

Table 4.3: OFFRP: Fruit Fly Based Packet Format

Parameter	Description
$R_{(min)}$	Rate sustainable to all nodes
C^O	Inter network corridors
$Max_{(m)}$	Maximum Mobility
$Min_{(m)}$	Minimum Mobility
L^Z	Node Localization
S^P	Seed to Peer Ratio
L^S	Link Speed
R^D	Router Drops
L^A	Link Availability
RTT	Round Trip Time
F_PKT_LN	Packet length
F_PKT_T	Packet Type
F_PKT_P	Priority
ReT	Retires
F_ADR	Current Address
$cwnd$	Window Size

The whole system is driven by the table alterations each time the smell index is updated. The memory requirement for the proposed approach depends upon the number of flies

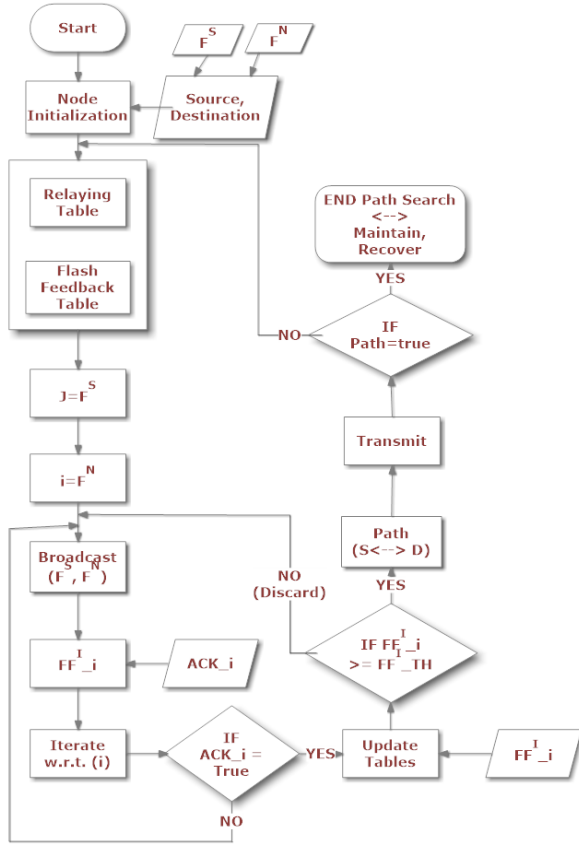


Figure 4.1: Route Discovery and Path Establishment

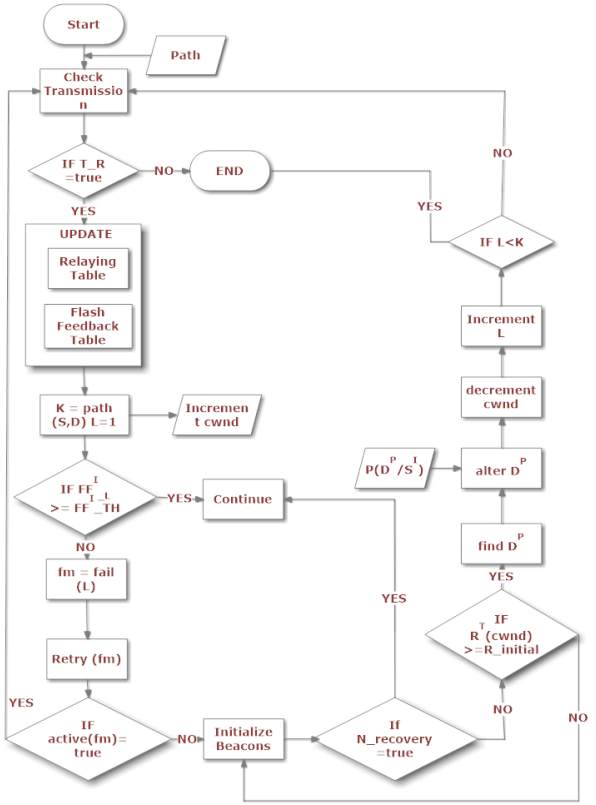


Figure 4.2: Route Maintenance, Recovery and Congestion Control

used by the source to track the relays or the number of incoming connections handled simultaneously by each node. More the number of incoming requests accepted, more is the memory requirement. Thus, for an optimized solution, each node must not be connected to more than $\frac{M^E}{F_count}$ of the incoming connections in order to efficiently handle the network operations. Here, M^E is the total memory available with each node and F_count is the total number of flies considered in the network. The complete overview of all the process, including route finding, maintenance, recovery, and congestion control are shown as a flowchart in Figure 4.1 and Figure 4.2.

Table 4.4: Routing Protocols Considered For Comparative Analysis

Protocol	Abbreviated For	Description
DSDV	Destination Sequenced Distance Vector Routing [227]	Proactive
AODV	Ad Hoc On Demand Distance Vector Routing [228] [229]	Reactive
DSR	Dynamic Source Routing [230]	Reactive
AOMDV	Optimized Ad Hoc On Demand Multipath Distance Vector Routing [231]	Multipath Reactive
MDART	Multipath Dynamic Address Routing [232]	Hierarchical
PUMA	Protocol For Unified Multicasting Through Announcements [233]	Multicast
TORA	Temporally Ordered Routing Algorithm [234]	Reactive
MAODV	Multicast Ad Hoc Distance Vector Routing [235]	Multicast

4.3 Performance Evaluation

A direct communication set up for multi-UAV guided networks is not available. Neither a direct simulation tool is available that can provide such dynamic simulation environment for emulating analysis of such network model. For this, multiple networking simulator and analysis tools were combined using a common interface. NS-2 and Matlab were used for simulating cooperative environment. Existing framework does not provide any routing support, thus, a routing layer was created that provided a platform that can sustain the connectivity and allows the inter-connectivity of the two different layered networks. Protocols considered for performance comparison are shown in Table 4.4.

4.3.1 Taxonomy of Network Parameters Used For Analysis

For analyzing the performance of routing protocols over multiple ad hoc formation, initially proposed cooperative neural based network model [166] is used. The model used for analysis includes formation of a routing layer for UAVs in ad hoc mode, and the guidance layer for sharing of information between two different ad hoc networks. Various routing protocols considered for comparative analysis have been evaluated on following parameters:

- (i). **Neural Delays (D_N):** These are the type of network delays that occurs during transmission process. These delays are evaluated on the basis of neural formation that operates over transfer of cognitive map over the multi-ad hoc formation. Neural delays are dependent upon the transmission delays, cognitive delays, propagation delays and traversal delays. For Multi UAV guided ground ad hoc formation, neural delays are evaluated as:

$$D_N = \frac{\int_{t \in T} D dt}{C_x T_c}, \quad (4.15)$$

where T is the total operational time, T_c is the total neural connectivity time, C_x are the number of connected neurons obtained from mapping function X such that

$$X \Leftarrow f(C_u, C_g),$$

here, C_u and C_g are the number of neurons for UAVs and ground nodes, respectively, and D at time t' is computed as:

$$D = d_{transmission} + d_{cognitive} + d_{propagation} + d_{traversal}. \quad (4.16)$$

- (ii). **Cooperative Link Reliability (L_R)**: For a multi ad hoc network, link reliability is the major evaluating parameter as the transfer of corrected information amongst the node of same as well as different network is of high importance. Link Reliability computations depend upon the functional division of the network i.e. whether the network is operating in serial formation or parallel. For a cognitive transmission process, the serial connectivity is considered. Thus, L_R of ground nodes w.r.t. UAVs will be computed as:

$$L_R(relative) = \frac{L_u}{L_g}, \quad (4.17)$$

where L_u and L_g are the link reliability of UAV network and ground network, respectively, that can be evaluated using Binomial distribution as:

$$L_u = \int_0^{T_c} {}^N C_X (U_F)^X (1 - U_F)^{N-X} dt, \quad (4.18)$$

and

$$L_g = \int_0^{T_c} {}^M C_Y (G_F)^Y (1 - G_F)^{M-Y} dt, \quad (4.19)$$

where N is the number of neurons for UAVs, M is the number of neurons for ground nodes, U_F is the probability of failures in UAV network, G_F is the probability of failures of ground network, X and Y is the neuron connectivity failures in UAV network and ground network, respectively.

- (iii). **Bandwidth Consumption (B_c)**: In a network comprising of multiple ad hoc networks, number of channels available are limited. This means that channel capacity needs to be optimally utilized for efficient network formation. Channel capacity directly relates to the bandwidth consumption. In a network consuming high bandwidth efficiency leads to high network lifetime. Thus, for cooperative ad hoc

network acting as a guidance system, bandwidth consumption can be computed by two ways:

- (a) Network operating on variable bandwidth requirement:

$$B_c = B_u \times B_g, \quad (4.20)$$

where B_u and B_g are the bandwidth consumption of UAV network and ground network, respectively, that are computed as:

$$B_u = \frac{B_u(initial) + \frac{data_Loss}{T_c}}{U_{ac}}, \quad (4.21)$$

and

$$B_g = \frac{B_g(initial) + \frac{data_Loss}{T_c}}{G_{ac}}, \quad (4.22)$$

where U_{ac} and G_{ac} are the active number of nodes for UAV and ground ad hoc network, respectively.

- (b) Network operating on familiar bandwidth requirement:

$$B_c = \frac{B_n^2}{2}, \quad (4.23)$$

where

$$B_n = \frac{B(initial) + \frac{cognitive_Loss}{T_c}}{I_c}. \quad (4.24)$$

Here, $cognitive_Loss$ denotes the number of map data lost during transmission and I_c denotes the number of interconnected nodes between the aerial and the ground network.

- (iv). **Cognitive Delivery Ratio (CDR)**: The ratio between the number of successful cognitive value transmission to the total number of cognitive values transmitted is termed as a cognitive delivery ratio. CDR is calculated to know the percentage of data shared between the two networks. More the CDR, more is the accuracy in map formation and higher is the network connectivity. For a multi network model,

cognitive delivery ratio is given as:

$$CDR = \frac{Cog_{success}}{Cog_{total}}, \quad (4.25)$$

where $Cog_{success}$ and Cog_{total} denotes the successful cognitive value transfer and total cognitive values transferred, respectively.

- (v). **Error Rate (E_R):** Network error can be analyzed in terms of number of failures occurring during network operations. These include loss due to node failure, mobility and router drops. Thus, the error rate can be evaluated as:

$$E_R = \frac{N/W_{Loss} + R_{drops}}{T_{Loss}}, \quad (4.26)$$

where T_{Loss} is the duration for which errors occurred.

- (vi). **Number of Map shared (M_S):** During the cooperative network formation, site maps are generated by UAVs traversing over the particular area. These traversal maps are formulated into certain cognitive values which are shared amongst each of the ground node to form a hybrid cognitive map for area unknown to a particular node. In such network formations, numbers of cognitive values required to form a minimum map is calculated and then, total cognitive transfers are evaluated to analyze the performance of the network. This can be given as:

$$M_S = \frac{Cog_{success}}{Cog_{Min}} \times C_x, \quad (4.27)$$

where Cog_{Min} is the minimum number of cognitive values required to form a guidance map.

- (vii). **Network Correlation:** The network model is based upon the formation of neural schema that creates neurons for each of the interacting layers of the model i.e. UAV layer and the ground node layer. These mappings are trained using certain training algorithms. Thus, based on the number of iterations, percentage for testing, validations and training, correlation between the neurons of two different layers is calculated. This correlation is termed as network correlation and is evaluated by plotting auto-correlation, error histogram, evaluation time and correlation between

them.

- (viii). **Fidelity Ratio (F_R)**: During network transmissions and sharing, despite of data transfers, not every value is useful or in other ways it can be said that there are certain network error that hinders the formation of guidance system. Be it the poor connectivity, node failures, lost data integrity and incorrect routing; for performance evaluation, fidelity ratio is required that can be expressed as a ratio of network sustaining time N_{ST} to the total connectivity time T_c i.e. :

$$F_R = \frac{N_{ST}}{T_c}. \quad (4.28)$$

- (ix). **Throughput (T_H)**: Network throughput is the measure of the actual transfer rate offered by the network over particular channel. Throughput is the measure of traffic flow in the network. In a cooperative multi-UAV guided network, it can be evaluated as:

$$T_H = C_N \times \left[\frac{Cog_{success}}{\frac{CDR \times A(D_N)}{L_R} + D_N} \right], \quad (4.29)$$

where $A(D_N)$ is the average network delays, N_T is the total network nodes, $Data_{Transfer}$ is the total data shared between the nodes including the acknowledgments and verification data and L_R is the link reliability.

- (x). **Total Computational Time (T_{CT})**: Time consumed in the formation of traversal maps, network connectivity, data sharing, routing, maintainable time during transmission and network analysis time is included in evaluation of total computational time. It gives the time analysis from staring of the network till the completion of each process. It can be denoted as:

$$T_{CT} = \sum_{i=1}^{N_s} (S_i) + \sum_{j=1}^{N_p} (P_j), \quad (4.30)$$

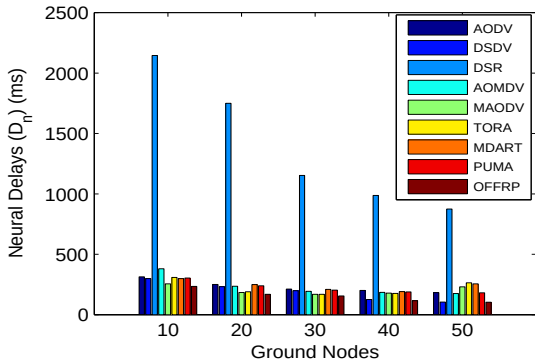
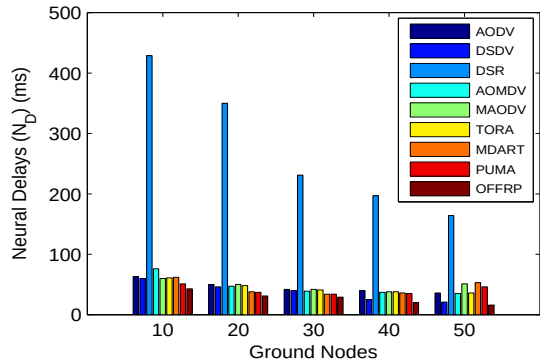
where S_i and P_i are the computational time for i^{th} serial and parallel process, respectively, and N_s , N_p are the number of serial and parallel processes, respectively.

Table 4.5: Parameter Configurations For Routing Protocol Analysis

PARAMETER	VALUE	DESCRIPTION
Area	1500x1500 sq. m	Area for search and tracking
Ground Nodes	10,20,30,40,50	Node sets for evaluation
UAVs	1,5,8	UAVs for Cooperative network
Traffic Type	CBR	Traffic over TCP
Packet Size	1024 bytes	Average Packet Size
Mac Layer	802.11 b	IEEE Standard for wireless
Buffer Size	20000 bytes	Buffer Capacity of each Node
Propagation Radio Model	Two Ray Ground	Topological Propagation Model
Speed	50kmph	Node mobility
Pause Time	0.5s	Halt time for Node
Interval Time To send	2 p /s	Time to wait before sending
Transmission Range	500 m	Transmission Distance of Node
Ground Node Mobility Range	0-2.5 m	Node Steps during movement
Initial Bandwidth	256 kbps	Initial Transmission rate
Simulation Time	1000s	Total time for network analysis
Configurable Parameters	10	Neural controlled parameters
Total UAV links	235	Active links in UAVs
Total Ground links	1143	Active links in ground nodes
Active Connections	5,10,15,20,25	Transmitting Nodes
Seeds	1.0	Source at time instance t
Minimum Rate	512 Mbps	Minimum Transmitting Rate
Minimum Speed	10 m/s	Minimum Mobility Rate
Simulation Runs	100	Number of Iterations

4.3.2 Performance Evaluation

Single and multi-UAVs systems were analyzed and evaluated over the identified metrics. Multi-UAVs network was evaluated using 3, 5, and 8 UAVs with ground nodes- 10, 20, 30, 40, and 50. The parameters configured for simulative analysis are shown in table 4.5. Separate analyses were traced for single UAV controlled and multi-UAV controlled networks. The results were traced for 100 simulation runs. A task oriented analyses were performed between the aerial and the ground nodes. UAVs were assigned to share maps as cognitive data between the source and destination. Since, the underlying coopera-

**Figure 4.3:** UAVs:1 (Neural Delays (D_N))**Figure 4.4:** UAVs:5 (Neural Delays (D_N))

tive framework used for analysis is based on the neural structure, thus, the protocols were initially analyzed for neural delays. Single UAV network faced more neural delays than

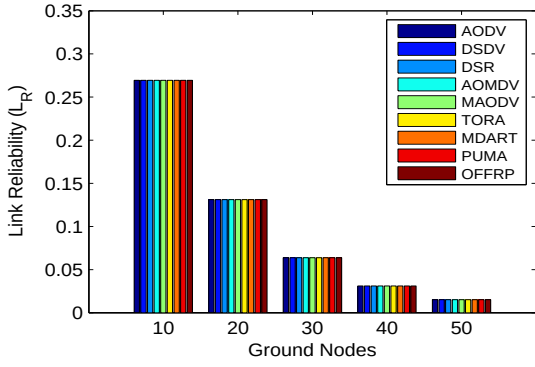


Figure 4.5: UAVs:1 (Link Reliability (L_R))

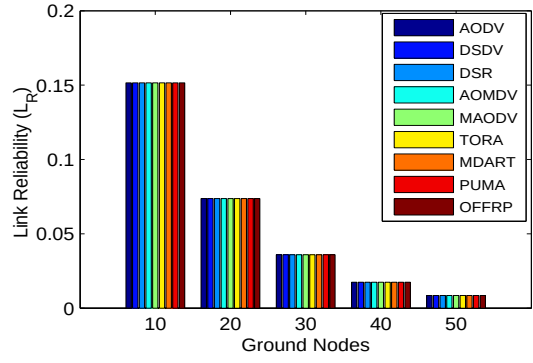


Figure 4.6: UAVs:5 (Link Reliability (L_R))

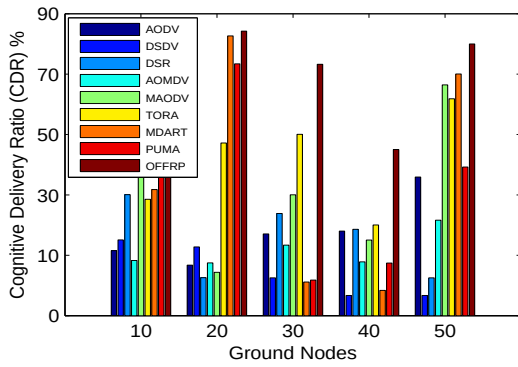


Figure 4.7: UAVs:1 (Cognitive Delivery Ratio)

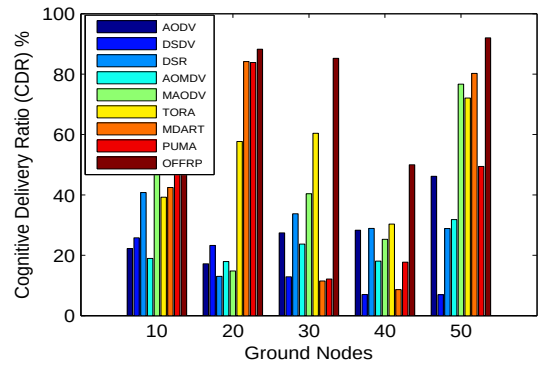


Figure 4.8: UAVs:5 (Cognitive Delivery Ratio)

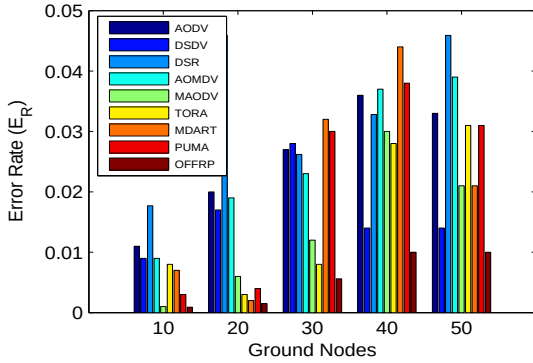


Figure 4.9: UAVs:1 (Error Rate (E_R))

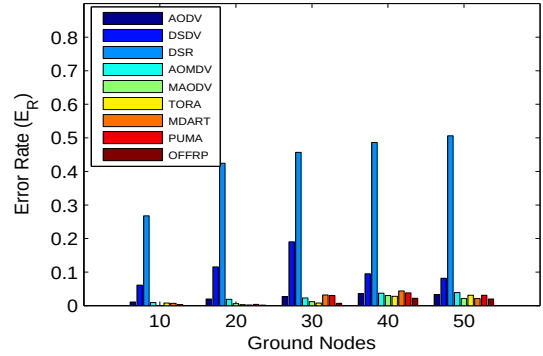


Figure 4.10: UAVs:5 (Error Rate (E_R))

multiple UAVs as more channels are available for connectivity. Proposed routing protocol -OFFRP performed better than the considered protocols with neural delays gradually decreasing with increasing number of ground nodes even in a single UAV system because of an optimized route selection between the source and destination. The comparative analysis for neural delays is shown in Figure 4.3 and Figure 4.4. Another metric for evaluation was link reliability. It is directly related to link quality available during net-

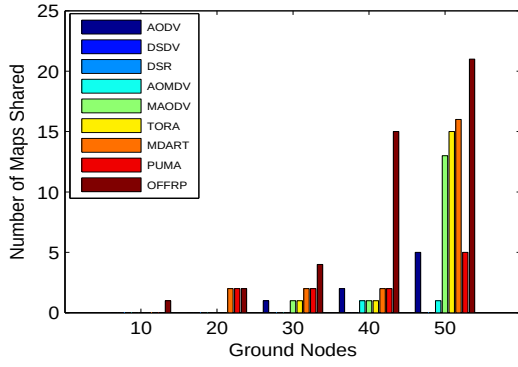


Figure 4.11: UAVs:1 (Maps Shared)

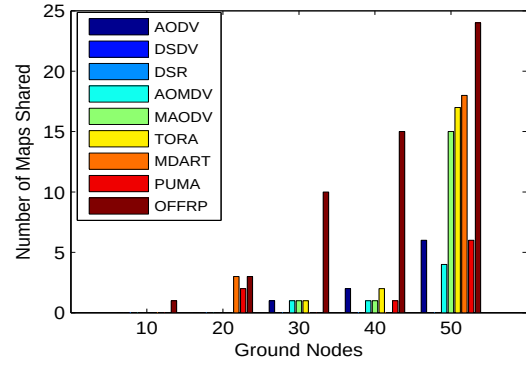


Figure 4.12: UAVs:5 (Maps Shared)

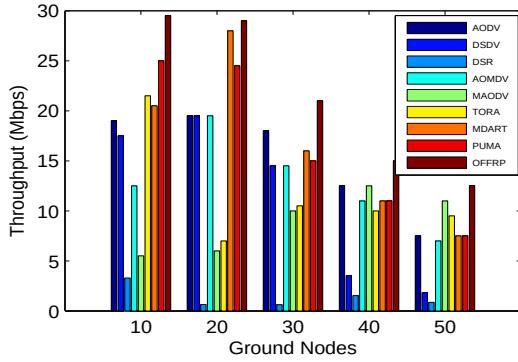


Figure 4.13: UAVs:1 (Throughput(T_H))

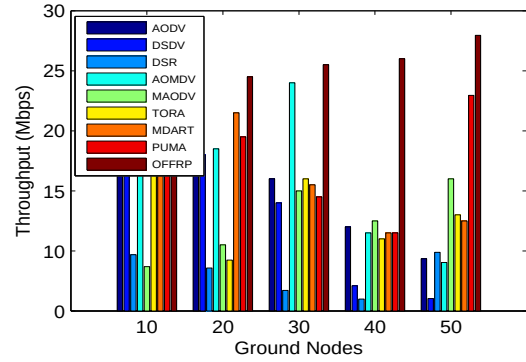


Figure 4.14: UAVs:5 (Throughput(T_H))

work operations. With increasing number of nodes, congestion is liable to occur, thus reliability decreases. However, reliability relates to configuration of nodes, thus, remains similar for all the nodes. Graphical traces for link reliability over single and multi-UAV system for considered protocols are shown in Figure 4.5 and Figure 4.6. Next, protocols were analyzed for delivery ratio in form of cognitive. Cognitive were used to evaluate the packet delivery ratio of the routing protocols. Proposed OFFRP performed better than the existing routing protocols followed by MDART and PUMA. Comparative analysis of the cognitive delivery ratio can easily be understood from Figure 4.7 and Figure 4.8. Another metric important to analyze is the error rate of the network using a particular protocol. Traces show that the proposed protocol being on demand and purely reactive, had low overheads, thus, the error rate was minimum for it. DSR suffered a maximum error rate because of memory overheads. Also, with the increase in number of UAVs, error rate decreases as more alternate connections are available in case of network failures. The comparative plots for error rate are presented in Figure 4.9 and Figure 4.10. A task

assigned to each node in the network was to share the maps between each other. Thus, the network was traced for successful receipt of maps over the configured network. With the maximum cognitive delivery ratio, OFFRP shared more number of maps between the network nodes and the UAVs. Variation of maps shared between the nodes is presented in Figure 4.11 and Figure 4.12. Further, the network was traced for throughput. With increasing number of UAVs and increasing number of nodes, more channels and corridors are available, thus, higher throughput is offered by the protocols. Single UAV networks are less efficient, but the proposed OFFRP provided better throughput than the existing protocols even with a single UAV. However, the single UAV system is not capable of sustaining the growing number of ground nodes and throughput decreases in such scenarios as shown in Figure 4.13. OFFRP outperformed all the existing routing protocols with throughput consistently increasing with increase in the number of UAVs with respect to the ground nodes as shown in Figure 4.14.

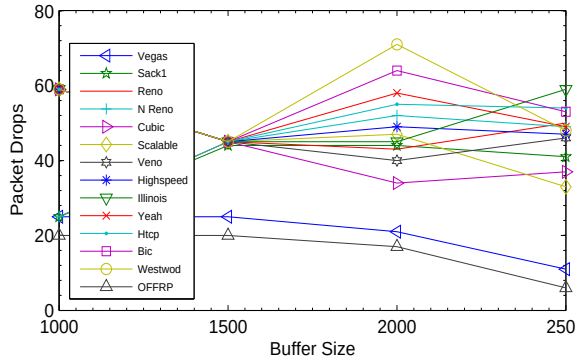


Figure 4.15: Packet Drops

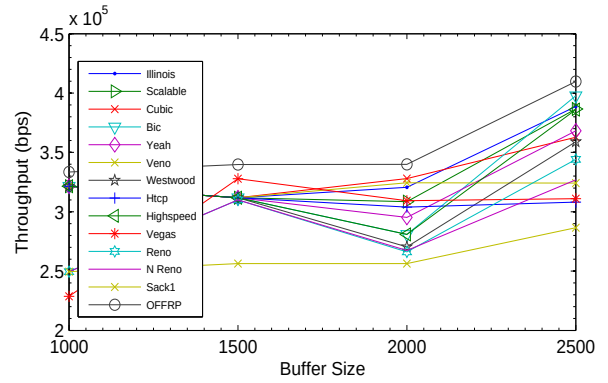


Figure 4.16: Throughput (bps)

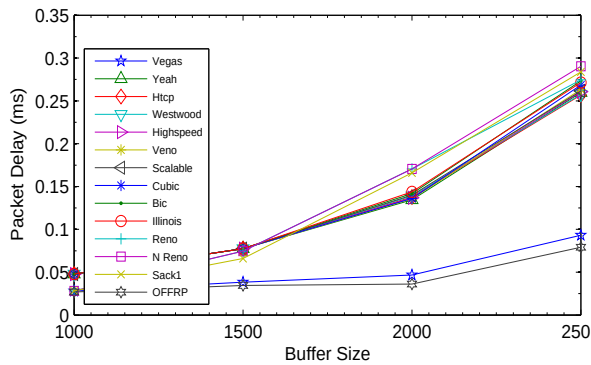


Figure 4.17: Packet Delays (ms)

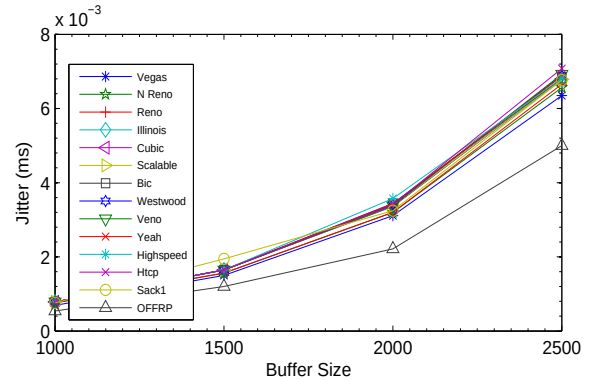


Figure 4.18: Jitters (ms)

4.3.3 Comparison With Existing TCP Algorithms

OFFRP being a routing protocol primarily focuses over congestion control with a fast recovery mechanism based on the smell index used in the initial phase to find the routes with best possible node configurations. The lower packet drop was recorded for the proposed OFFRP followed by the Vegas and cubic as shown in Figure 4.15. Also, the throughput was also high for the proposed protocol. Throughput increased consistently with the increase in buffer size till congestion is predicted. On the prediction of congestion, the network decreases the congestion window size. With more number of incoming requests to an intermediate relay, the congestion window gradually decreases following a smell index rule. Thus, causes a variation in throughput of the overall network. Also, more number of congestion predictions causes a low increase in throughput. But this increase is still much more than the existing protocols. The comparison for throughput with existing algorithms is shown in Figure 4.16. With consistent increase in the number of channels and enhanced throughput, packet delay and jitter are lower in the OFFRP. Lower jitters allow better stability of the network. The comparative charts for packet delays and jitters are shown in Figure 4.17 and Figure 4.18, respectively.

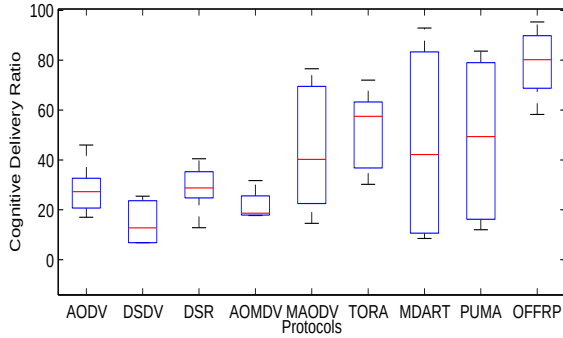


Figure 4.19: Variation : Cognitive Delivery Ratio

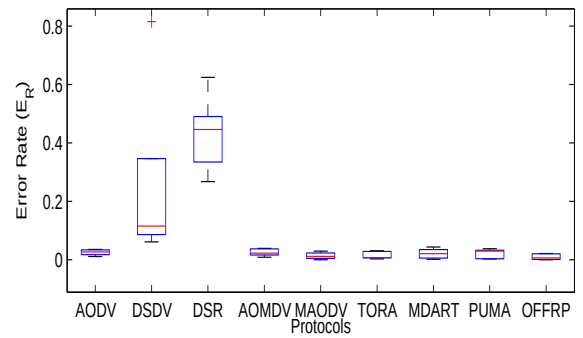


Figure 4.20: Variation : Error Rate (E_R)

4.3.4 Variation For Multiple Simulations

As stated in the simulation section, protocols were analyzed for 100 simulation runs. A protocol with lower variation offers better network quality for the whole transmission pro-

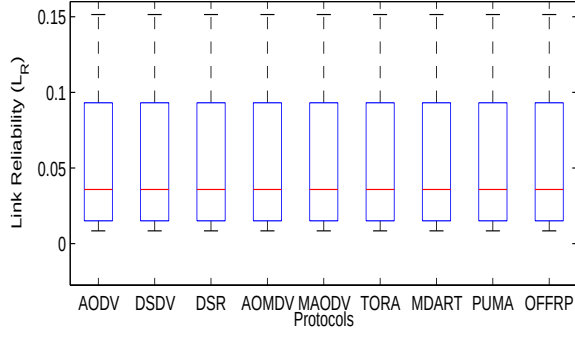


Figure 4.21: Variation : Link Reliability (L_R)

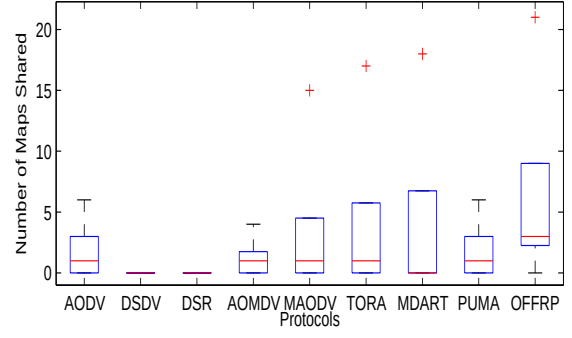


Figure 4.22: Variation : Maps shared

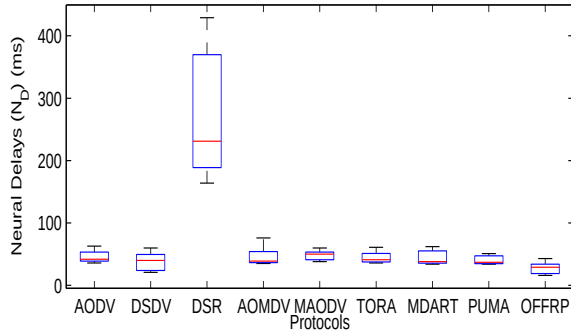


Figure 4.23: Variation : Neural Delays (N_D)

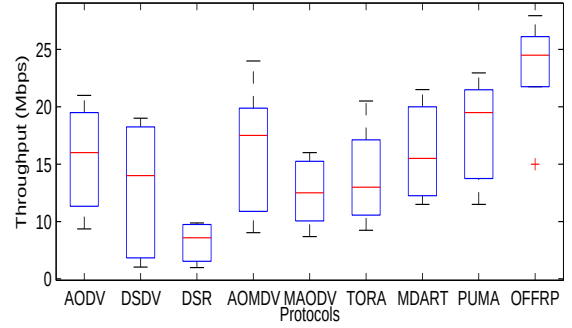


Figure 4.24: Variation : Network Throughput

cess. Mean value of the cognitive delivery ratio was traced higher for OFFRP. However, variation for multiple simulations was lower for AOMDV. OFFRP provided the equivalent deviation between minimum and maximum delivery ratio as shown in Figure 4.19. The error rate was lower for OFFRP and variation was also least for the OFFRP. With lower variation over OFFRP, congestion prediction was recorded better for the proposed protocol with enhanced stability to the network as shown in Figure 4.20. Link reliability being similar for all the protocol does not vary at all. The variation box plot for link reliability is shown in Figure 4.21. As discussed, the network was configured to share the maximum possible number of maps between the nodes. A protocol with more number of maps is considered to be efficient as compared to other protocols. With initial slow start and low transfer rate, the proposed protocol gradually increased its transmission rate and more number of maps were finally shared between the nodes in the network operating over the proposed protocol, thus, offering higher value for mean number of shared maps. The plot for variation in the number of maps shared is presented in Figure 4.22. Delay

variation shows that the overall delays over the existing framework by the proposed protocol is minimum as shown in Figure 4.23. Lower variation and high throughput allow formation of an efficient network with high delivery ratio. Analysis proved that the proposed protocol had more number of consistent connections that allowed higher transfer rate and higher throughput of the overall network as shown in Figure 4.24.

4.4 Conclusion

Networks operating in coordination with each other require efficient routing mechanism that can also provide congestion free paths. Efficient routing protocol can alooof any chances of congestion during path selection itself. Selection of optimized path further enhances the performance of distributed ad hoc networks. In this chapter, a new nature inspired routing protocol using Fruit Fly Optimization is proposed. The proposed routing protocol (OFFRP) provides a congestion free routing between the two simultaneously operating ad hoc networks. The proposed protocol utilizes the smell index property of fruit fly, and uses fruit fly factor as objective function to find an optimized route between the source and destination. The path selected using this approach is stable and continuously maintainable using the feedback mechanism of the fruit flies. The proposed OFFRP is evaluated with existing state-of-art routing protocols and TCP variants using standard network simulations specially configured to support aerial ad hoc formations. Results show that the OFFRP is capable of providing efficient connectivity and relaying in Multi-UAV guided ad hoc networks. This chapter accomplishes the another part of objective-2 which includes the designing of a routing protocol that can perform efficiently over the proposed cooperative framework.

Chapter 5

Service Provisioning Over UAVs Coordinated Networks^{†‡}

UAVs network in collaboration with another network such as ground network can form a fault tolerant and highly reliable system that can cross over the limitations of a single UAV system. Data in the form of cognitive allow formation of guidance system with higher accuracy. Collaboration amongst different network in cooperative mode is the paramount task. This collaboration requires certain platforms that provide common modules for hybridization. These platforms can either be hardware oriented controllers or can be service based software frameworks. Service based architectures allow mutual coordination of network operating simultaneously. Service based frameworks use local modules for operation and activity controls. These local modules are pre-defined and incorporated as service units for control of various hardware operating on network nodes. Service based autopilot modules and mission planning can increase the potential of UAVs as it provides a data level abstraction [236]. For network applications, data driven service architectures are also useful and can provide software facilities similar to enterprise level solution [237]. Service oriented network frameworks can solve various networking issues involved in formation of cooperative network between aerial and ground vehicles [238] [239]. Further, utility and accuracy of the service architecture can be enhanced using cross layer designs [240] [241] [242] [243], or by formation of a prediction and estimation model as a frame of reference for existing layers [244].

[†]Vishal Sharma and Rajesh Kumar, “An Opportunistic Cross Layer Design for Efficient Service Dissemination over Flying Ad Hoc Networks (FANETs)”, in: Proceedings of 2nd IEEE ICECS, February 26-27, 2015, Coimbatore, India, pp. 1551-1557.

[‡]Vishal Sharma and Rajesh Kumar, “3-Tier Neural Model For Service Provisioning Over Collaborative Flying Ad Hoc Networks”, Neural Computing and Applications, Springer- SCIE Indexed,[Revision], [I.F. 1.569].

This chapter presents the facility of service provisioning using the proposed cooperative framework over the UAVs coordinated networks. This chapter includes an overview of the new middleware for UAVs coordinated network. Then, it presents a service dissemination strategy using cross layer formation over this middleware. In the later part, a fully functional 3-tier neural based service model developed on the backbone of this middleware is presented.

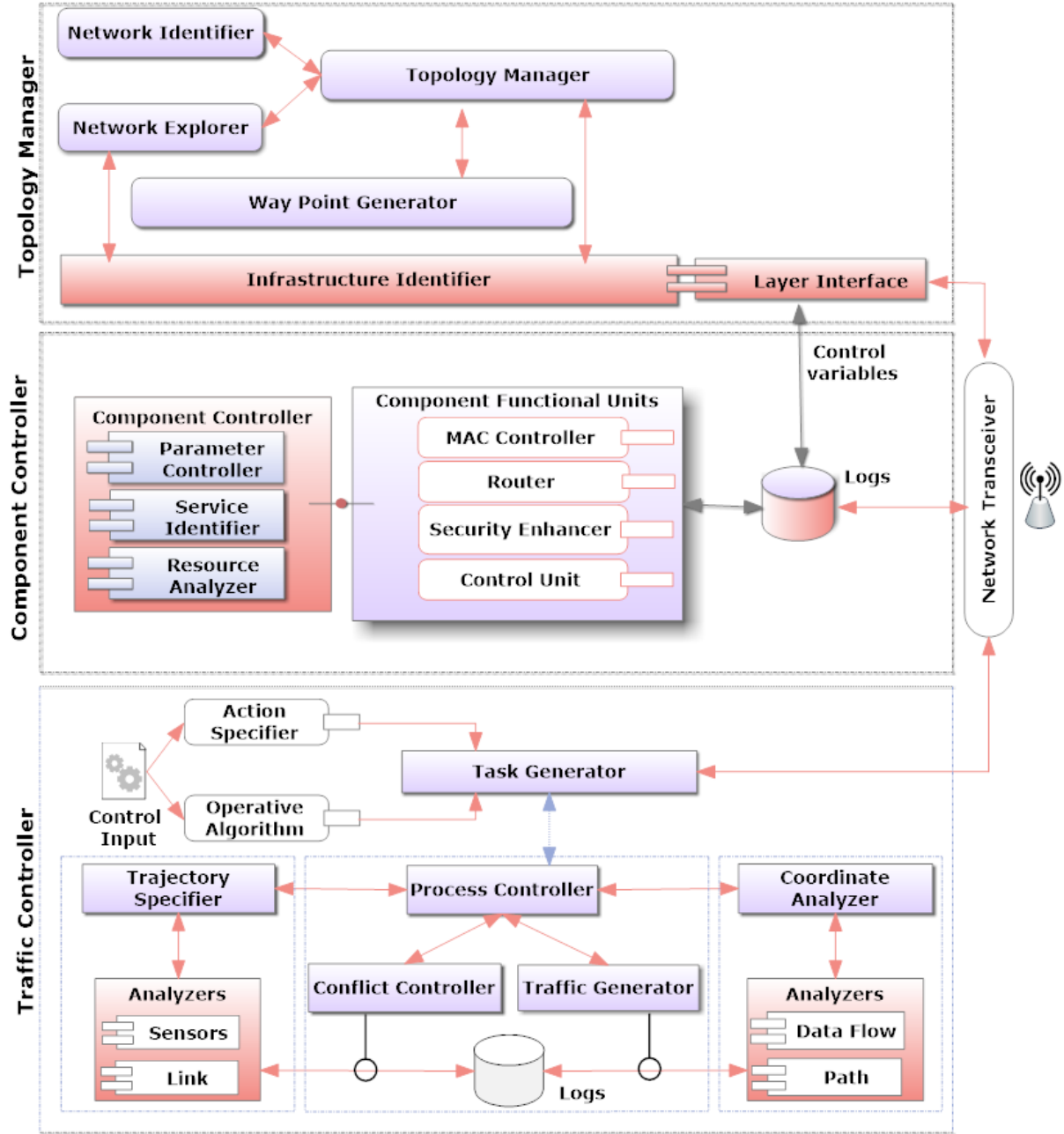


Figure 5.1: Cross Layer Design For Service Dissemination

5.1 Service Middleware For UAVs Guided Networks

The initial middleware for UAVs coordinated networks comprises of three layers, namely, topology manager, component controller, and traffic controller, as shown in Figure 5.1. Topology manager further utilizes the services of network identifier that identifies the current network and nodes available for connectivity, network explorer that finds the other possible networks for connectivity. Both the network identifier and explorer are used by the topology manager to set the waypoints using the waypoint generator. This waypoint generator utilizes the knowledge of underlying infrastructure before fixing the final trajectory of aerial vehicles. Component controller is the main communicating layer of the middleware. It comprises of all the network components that are responsible for service coordinations and network formations. This layer comprises of all functional units of a TCP suite. Traffic controller is the third layer, and is the major role player in task assignments and performance. It uses control inputs that specify the action and a control algorithm that will handle the tasks. The task generator further interacts with the process controller to regulate the conflict free traffic with respect to the specified trajectory. This middleware provides an initial solution for interaction of various aspects of a UAV coordinated networks. However, it is still not capable enough to regulate the services efficiently over the coordinated network. For this, the layers of this middleware are referred using a cross-layer formation that can provide a continuous connectivity irrespective of situations and conditions. The cross layer design incorporated over this middleware is presented in the next section.

5.2 Service Dissemination Using Cross Layer Design

Opportunistic network operates for efficient handling of services to enhance the network lifetime and performance. Opportunistic relaying over flying ad hoc networks, provides reliable service distribution amongst the network nodes such that an enhanced connectivity is established between the aerial and the ground nodes. The network operates in collaboration with the pre-existing ground network that coordinates the operations of

the aerial networks. The application suggested for the proposed service operating over the flying network is to provide efficient task distribution amongst the network nodes in the form of services based on the computational resources of each node. The proposed model is a cross layer approach. The operational layers of the network model for guided UAV ad hoc network is cross-layered on the basis of certain utility parameters to handle the service distribution. Tasks involving aerial networks are configured into series of services. These services are handled using data handling algorithm which uses resources available as its decision metric for dissemination. The proposed model uses the service coordination layered structure for service dissemination.

5.2.1 System Model For Service Dissemination

The mathematical formation for service dissemination operates by forming the service model which is evaluated over the neural schema that handles the input, updates, and output of the flying ad hoc networks. The model operates over the network connectivity time, probabilistic associativity between the aerial and the ground network, services for a number of tasks and the resources available. For opportunistic cross layer formation over flying ad hoc network, probability of connectivity $P_c^{k,t}$ at time t for k^{th} node will be given as:

$$P_c^{k,t} = \frac{\sum_{i=1}^{R \in (CH_{co})} (D_c^k)_t}{\sum_{i=1}^{T_{Aso}} \sum_{j=1}^{CH_{co}} (N_C \in (A, G))}, \quad (5.1)$$

where CH_{co} is the number of channels available for connectivity between the aerial and ground network, D_C^K is the average connectivity of the k^{th} node, T_{aso} is the total associativity of the network, N_C is the total network connectivity at time t . For a network operating over data sharing, service dissemination time S_i for i^{th} service will be computed as:

$$S_i = \sum_{k=1}^{CH_{co}} (T_{sg}) - T_{req}, \quad (5.2)$$

where T_{sg} is the time of generation of the service, known as first response time, and T_{req} is the request time for a particular service. For a network operating in coordination with another network, services are to be handled based on the number of resources available that can efficiently distribute the requested services without any loss over the mutual

network. For aerial network, resources available R_H^A are given as functional dependency such that:

$$R_H^A \longrightarrow f(N_{CH}, N_{CO})_A. \quad (5.3)$$

Similarly, for ground network, resources available R_H^G are given as:

$$R_H^G \longrightarrow f(CH_{co}, N_{CO})_G, \quad (5.4)$$

where CH_{co} , N_{CO} are the number of channels and corridors available for network connectivity. Now, resource handler R_H^M for the mutually coordinated network will be computed as:

$$R_H^M = \text{Min}(R_H^A, R_H^G). \quad (5.5)$$

The network service density ratio S_{DR} will be computed as:

$$S_{DR} = \frac{S_{req}}{CH_{ser}}, \quad (5.6)$$

where S_{req} are the number of service requests, and CH_{ser} is the number of channels available to handle service requests. Now, traffic density ratio T_{DR} will be given as:

$$T_{DR} = \frac{S_{DR}}{\sum_{i=1}^N \sum_{j=1}^{N_{CO}} S_{AG}}. \quad (5.7)$$

Here, S_{AG} are the services operating between the aerial and the ground network. The parameterized dependency of the cross layer formation is subjected to neural schema for analysis of its input, output and updates. The neural model can be used depending upon the amount and flow regulations of the network. In the proposed model, the output I/O at time t is mapped as:

$$I/O = \frac{W_S^t + D_C^{AG}(S_i)}{T_{DR} + ||S_{DR}||r_{min}^{t-1}}, \quad (5.8)$$

where

$$r_{min} = \text{Min}(R' \in (N_A, N_G)), \quad (5.9)$$

and

$$W_S^t = r_{min}^t f(R_H^A, P_C^t, R_H^G) + D_C^{AG}. \quad (5.10)$$

Algorithm 5.1 Opportunistic Cross Layer Formation In FANETs

Require: N_A, N_G, S

Ensure: $Min(T_{aso}) = Avg(D_C^A, D_C^G)$

```
while  $t \leq total\_time$  do
  while  $service\_handled \neq complete$  do
    set  $CO \leftarrow corridors$ 
    set  $CH \leftarrow channels$ 
    map(CO, CH)
     $j \leftarrow count(map(CH, CO))$ 
    initialize  $i$ 
    while  $i \neq j$  do
      set  $map\_counter = true$ 
      if  $map\_network \neq complete$  then
        count  $S_n \leftarrow services$ 
         $R_H^M = Min(R_H^A, R_H^G)$ 
         $P_c^{k,t} = \frac{\sum_{i=1}^{R \in (CH_{co})} (D_c^k)_t}{\sum_{i=1}^{T_{aso}} \sum_{j=1}^{CH_{co}} (N_C \in (A, G))}$ 
         $S_i = \sum_{k=1}^{CH_{co}} (T_{sg}) - T_{req}$ 
        if  $S_{DR} \ \&\& \ T_{DR} \geq 0.5$  then
          continue
          counter=counter+1
        else
           $map\_counter = false$ 
          reset
        end if
      else
        reset
      end if
       $i = i + 1$ 
    end while
     $service\_handled = S + 1$ 
  end while
   $t = t + count(timer\_tick)$ 
end while
```

5.2.2 Opportunistic Cross Layer Formation

The proposed model provides opportunistic network formation by modifying the functioning of the service layers of a UAV network to form a cross layer design over the initial middleware presented earlier in Section 5.1. The cross layer allows hybridization of parameters that allows decision making for service handlers easy and efficient. Each layer provides a driving parameter value which is subjected to the neural schema that disintegrates the service requests and efficiently distributes them amongst network nodes using data handling algorithm. The data handling algorithm uses flow control methodologies to manage the network transfer rate, thus, providing an effective solution for service dissemination over FANETs. Algorithmic selection is based on the neural schema that balances the network in case network stabilization is disturbed. The state of stabilization

is decided by the neural network itself based on the number of parameters and learning rate of the network. Procedure for managing the flow and forming an opportunistic relaying network using cross layer design over FANETs is shown in Algorithm 5.1. This cross layer formation, although provides support for service dissemination over flying networks, but lacks the provisioning of coordination with other ad hoc networks. Thus, utilizing the initial middleware and cross layer formations, a new fully functional service dissemination model is developed for collaborative flying ad hoc networks. The complete details of formation, operations, and utility of this new service model is presented in the next section.

5.3 3-Tier Neural Model For Efficient Service Dissemination

The network formation between the aerial and ground units is supported by the cooperative network framework that provides connectivity interface between these two distributed operating networks [166] [245] [246]. The framework also provides support for neural interfacing of service model that allows efficient service-oriented corridor selection for better connectivity and multiple cognitive map formation over a single connection. This single channel is shared as multiple connections using multiple cooperative services which converts the single corridor into multiple virtual service channel, thus, improving transmissions. The network model considered for aerial-ground ad hoc network formation is shown in Figure 5.2. Service based model allows efficient connectivity between the networks that are operating in coordination with each other. For dynamic synchronization between the two ad hoc networks, service based arguments can be used to allow network coordination and guided connectivity. These service arguments can control the functionality of both intra as well as inter network facilities that are required to be coordinated to form a guidance unit. For argumentative coordination and service control, a 3-tier service oriented neural model is proposed that operates similar to a service oriented architecture, which includes neural layers and estimators. This will allow formation of highly coordinated 3-tier neural schema. The 3-tier model used for collaborative network

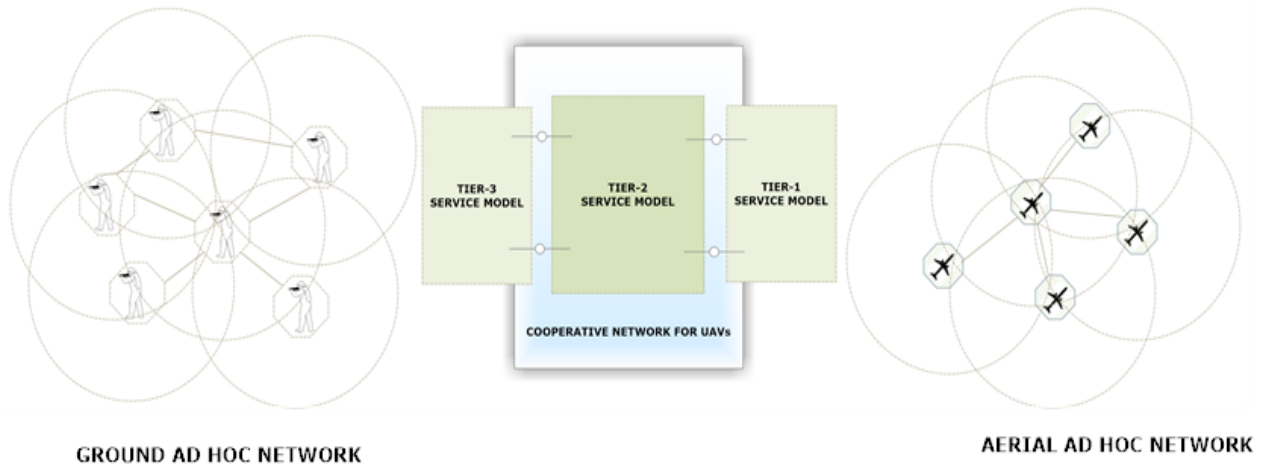


Figure 5.2: Network Model For Aerial-Ground Ad Hoc Formation

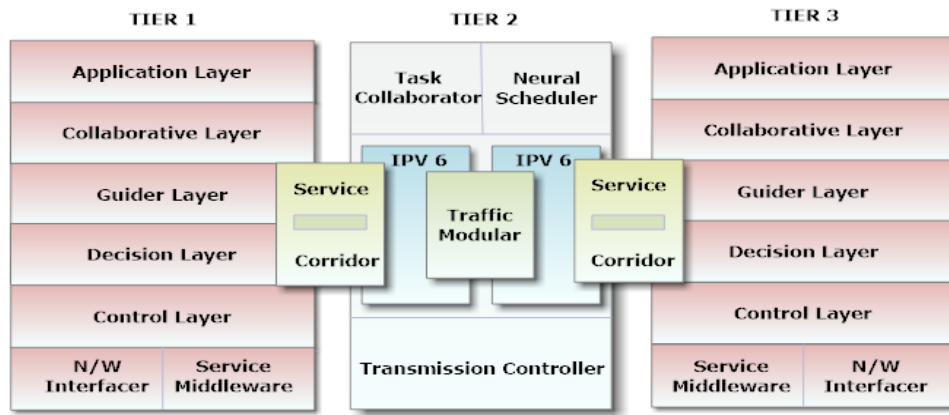


Figure 5.3: 3-Tier Service Neural Oriented Model For UAV Guided Networks

formation between the aerial and the ground ad hoc networks is shown in Figure 5.3. It is a six layer model with bottom layer divided into two components. Tier-1 and tier-3 denotes the layered model operating at aerial ad hoc and ground ad hoc networks, respectively. The inner tier-2 is a layered structure with top most layer divided into two components. The layers included in tier 1 and tier 3 are similar and depending upon their functionalities are termed as:

- (i). Application Layer
- (ii). Collaborative Layer
- (iii). Guider Layer
- (iv). Decision Layer
- (v). Control Layer

(vi). Interfaces

- (a) N/W interfacier
- (b) Service Middleware

The coordinating tier-2 of the service model that acts as an inner control component is divided into following layers:

- (i). Tasks
 - (a) Task Collaborator
 - (b) Neural Scheduler
- (ii). Traffic Modular
- (iii). Transmission Controller

The column based division of model into a series of tiers allows identification and controlling of coordinating unit so as to form an enhanced guidance system using multiple ad hoc networks. Connectivity between the tier-1 and tier-3 to tier-2 is provided through the service corridors based on the nodes selected for inter-network transmissions. The detailed functioning and components of this 3-tier model is provided below:

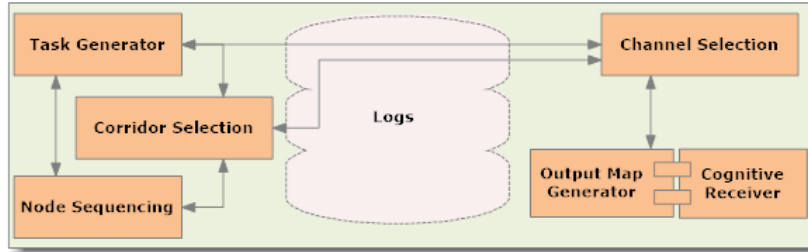


Figure 5.4: Application Layer

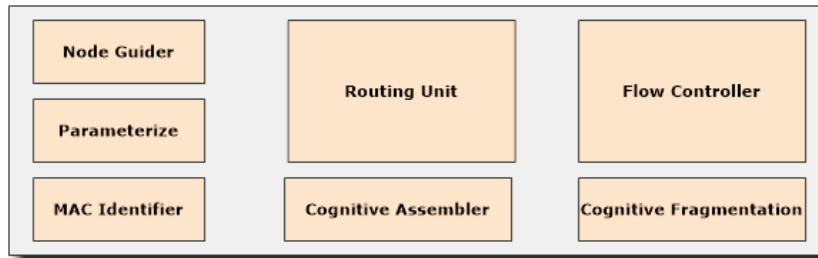


Figure 5.5: Guider Layer

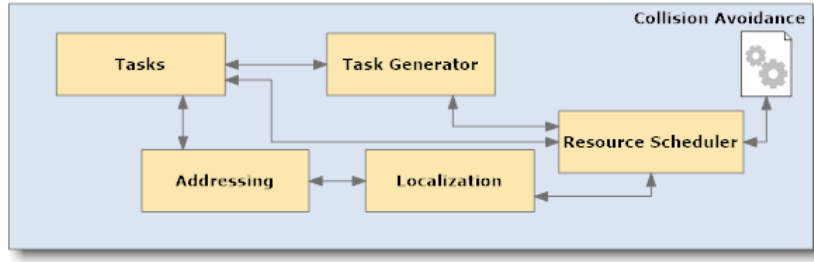


Figure 5.6: Collaborative Layer

5.3.1 Components of Tier-1 and Tier-3

Tier-1 and tier-3 are the coordinating units for mutually operating networks. Tier-1 controls the functionality of the aerial ad hoc network, and tier-3 manages the activities of ground ad hoc network. The layers in these are divided into a series of components provided on the basis of their functionality and utility.

- (i). **Application Layer :** It is the top most layer of both the tiers with facility for user interaction through service generation. The component classification of the application layer is shown in Figure 5.4. The main components of this layer are task generator, channel selector, corridor selector, node sequencer, cognitive receiver and map generator. This layer is responsible for all the visualization formed using map transfers between the two networks. Selection of corridor nodes can be algorithmic or can be statically configured during mission planning. The layer also provides log maintenance to prevent any cognitive regeneration and data capturing during a session loss. Components of the guider layer are shown in Figure 5.5.

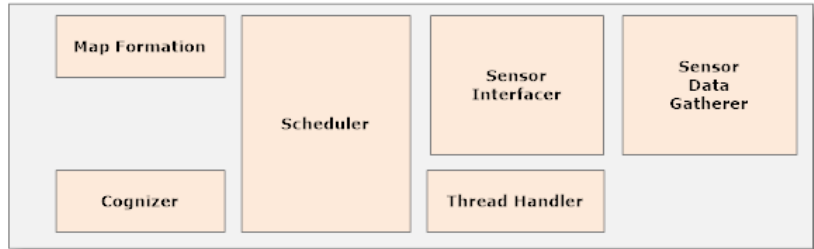


Figure 5.7: Decision Layer

- (ii). **Collaborative Layer :** This layer is pivot in the formation of collaborative structure using autonomous vehicles. Various collision avoidance parameters for the both ad

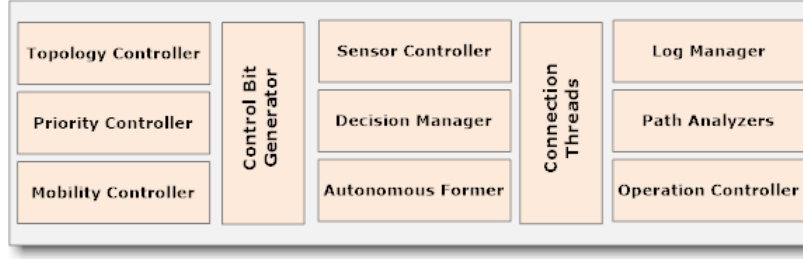


Figure 5.8: Control Layer

hoc networks are handled by this layer using control algorithms. The layer uses control inputs in the form of distributive algorithms that controls the maneuverabilities and the motion of the network nodes. Topology management is also carried out using the localization facility of this layer. Node addressing and collaborative task generation and mission planning are also performed on this layer. The connectivity diagram for the collaborative layer components is shown in Figure 5.6. Resource scheduler in this layer provides a facility to self-configure the framework, depending upon the mission requirements. This scheduler is based on the services provided through topology manager as arguments from its adjacent layers, number of tasks to be handled and node localization.

- (iii). Guider Layer : This layer is responsible for all the coordinated moves a node has to perform in order to remain active during the transmission process. This layer contains an assembler and fragmented component that performs computations over cognitive data so as to provide effective transmission by controlling the payload of the transmitted packets. The routing components are also part of this layer. All the routing protocols are controlled by this component. Apart from this, node tracking facility using MAC addressing is also key functionality of this layer. Parametrization and cognitive flow control support are also provided by this layer.
- (iv). Decision Layer : This layer is also termed as data layer or the map layer of the neural guided model. The layer acts as an interface between the guider layer and network control layer as shown in Figure 5.7. Final map formation based on the search and tracking algorithms is performed by this layer. The layer also contains a sensitive component termed as cognizer that identifies the cognitive bits from the complete packet header and generates a series of cognizer frames. These cognizer

frames are further operated over by cognitive scheduler. This layer also handles the multi-threaded model operations. Various onboard sensors and off board data transfers are controlled by components of this layer. Multiple sensor hybridization is the key functionality of this layer.

- (v). **Control Layer** : It is the most significant layer of the neural model. All the onboard and off board operations are handled by components of this layer. The layer provides control inputs to every unit of the network used as transceiver. The control bit generator operates in coordination with the thread generator to allow distributed parallel processing of multiple service-corridors. This reduces the transmission time and also improves the overall complexity of the system. The main components of this layer are: topology controller, priority controller, mobility controller, sensor controller, decision manager, autonomous former, log manager, path analyzers and operation controller. This layer is customizable and number of components can be varied depending upon the requirement of the mission and supported units available over aerial and ground vehicles. The default components of control layer are shown in Figure 5.8.

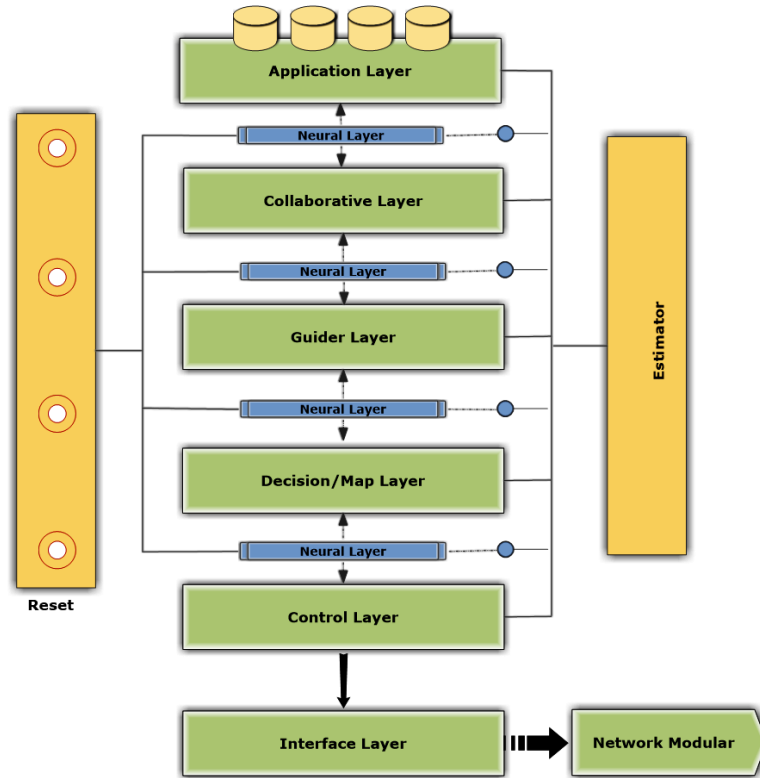


Figure 5.9: Neural Interfaced Layered View

- (vi). Interfaces : Rather than a layer, it is more of structural unit comprising of network interface and service middleware as two simultaneously operating unit. Network interface is used for intra-network, inter-network and transmission process, whereas service middleware provides a platform for collaborative operations of neural based network services. Service middleware are also customizable units of the neural interface that can be configured depending upon the type of services and components of the model used for coordination.

5.3.2 Neural Schema For Service Layer Integration

The layered structure of the 3-tier architecture is supported by neural interfaces between each layer that takes control inputs from the underlying layers and then generates the updated control sequences for the above layers. The neural interfacing is provided for all the layers except for interaction between the interfaces and the control layer. The detailed descriptive view of the neural interfaced layered structure is shown in Figure 5.9. Each neural interface is provided with a reset point that is based on the neural conditions. The reset conditions can be provided separately for all the layers or can be set as single conditions based on common reset points. Neural operability of the layered structure is based on the service identifiers, the number of nodes in aerial and ground ad hoc network and the coordinating neurons selected for collaborative transmission on each layer. The neural units are also mapped using an estimator that helps in the prediction process regarding network transmissions depending upon the logs maintained at certain time intervals.

Let S_N be the services generated for N number of nodes. N_e , N_c be the number of neurons configured for the particular layer interface and a number of control variables, respectively. For efficient connectivity and generation of control sequence for interfaces, $N_e > N$. Let W be the weight matrix defined over services, nodes and neurons such that:

$$W \leftarrow f(S_N, N, N_e) \quad \forall N_e > N, \quad (5.11)$$

where $f(S_N, N, N_e)$ is dependent upon the connectivity of independent and dependent

variables. Connectivity C_S , C_N , C_{N_e} for service variables, nodes and neurons will be given as:

$$C_S = S_N \frac{(S_N - 1)}{2}, \quad (5.12)$$

$$C_N = N \frac{(N - 1)}{2}, \quad (5.13)$$

$$C_{N_e} = N_e \frac{(N_e - 1)}{2}. \quad (5.14)$$

Now, the average number of node-neural connections are calculated as:

$$C_{avg} = \frac{\frac{N(N-1)}{2} + \frac{N_e(N_e-1)}{2}}{2}. \quad (5.15)$$

The average connectivity also defines the connectivity threshold for the network. The number of services to be generated at time Δt for i^{th} control input is computed as:

$$S_i^g = \frac{N S_N N_c}{2 C_{avg}} (N_c - 1). \quad (5.16)$$

Now, total number of services generated for whole connectivity time T_c for control inputs i varying from 1 to C_{avg} is given by:

$$TS = \sum_{t=0}^{T_c} \left[\left(\sum_{i=1}^{C_{avg}} S_i^g \right) \times \alpha_t \right], \quad (5.17)$$

where α_t is the connectivity constant at time t such that $0 < \alpha_t < 1$. The control and the weight matrix for calculating neural distance is given as:

$$Z = [X \ 1]_{n \times m} \text{ and } W = [TS \ 1]_{n \times m}, \quad (5.18)$$

where n are the number of control variables and m are the number of services generated. Now,

$$D = ||\sqrt{(W)^2 - (Z)^2} + W_{factor}||_{normalize}, \quad (5.19)$$

where W_{factor} is the weighing factor computed as:

$$W_{factor} = \frac{N_c S_N}{2} (S_N - 1) (N_c - 1). \quad (5.20)$$

Therefore, the output generated from the neural interface will be given as:

$$\eta_i = \left[\frac{W_i + D_i}{N N_e + ||W_{i-1} + Z_{i-1}||} \right]. \quad (5.21)$$

The above derived output can be applied to both intra as well as inter-network communication. These outputs for control variables form the input of the estimator that carries prediction of various parameters configured for connected layer. Thus, provides better connectivity and improved cognitive controlled transmission.

5.3.3 Neural Estimator

Estimator operates in collaboration with layers of tier-1 and tier-3. The estimator operates on the values passed to it from various neural interfaces and predicts the values of the network parameters to provide effective transmission in collaborative mode. For prediction, previous values, upgraded index and network-residue are used to predict the values in the next step. In generic, estimator operates with the following arguments:

$$val(i) = val(i - 1) + updaton + error correcton. \quad (5.22)$$

The updates require neural input formulated from the output generated by each interface. Error correction is provided in terms of residue so as to match the required value. Thus, for any service input at time t , value of predicted parameter P_t will be computed as:

$$P_t = P_{t-1} + O_{updated} + R_{t-1}, \quad (5.23)$$

where

$$O_{updated} = \frac{P_{t-1} - \eta_{t-1}}{||W_{t-1} + P_{t-1}||}, \quad (5.24)$$

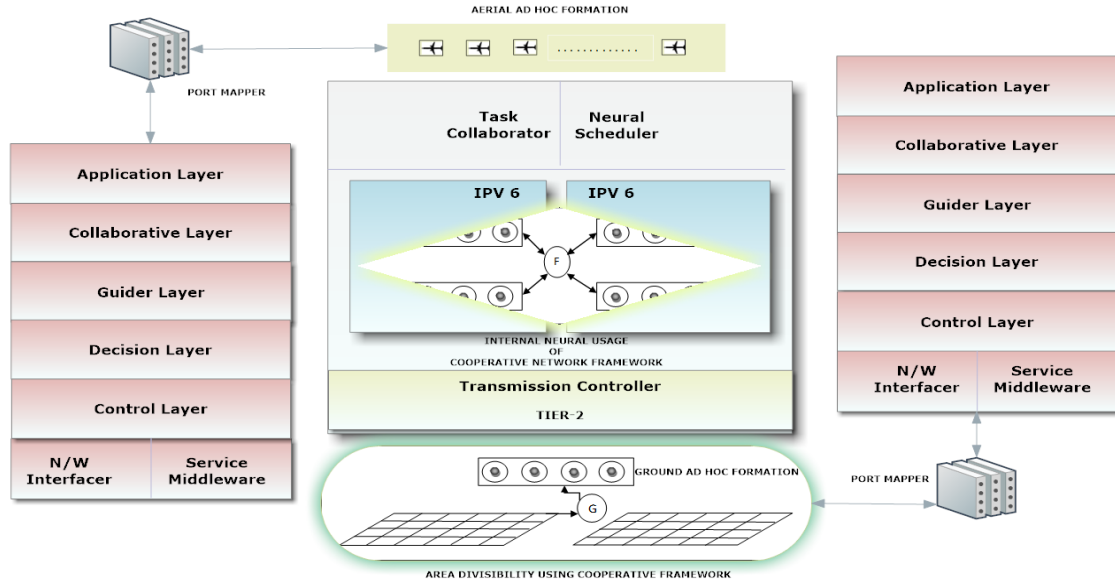


Figure 5.10: Integration of Service Model With Cooperative Network Framework

and R_{t-1} is the residue at time $t - 1$ defined over Gauss-Markov model. For residue calculation, an error is considered to be normally distributed such that $E \simeq N(\mu, \vartheta^2)$, where μ is defined over $P_0, P_1, P_2 \dots P_{t-1}$ and ϑ^2 be the variance defined as:

$$\vartheta = \sqrt{\frac{1}{N_c} \sum_{j=0}^{N_c} (P_{t-1} - P_{t-2})^2}. \quad (5.25)$$

5.3.4 Constraint Over Neural Model

The estimator formed using neural schema that integrates the multiple layers of the service architecture is controlled to go beyond certain limits by defining constraints over this model. For a collaborative network, the estimation rate over the configured network should be less than the certain threshold value TH_v which can be computed as:

$$TH_v = \sum_{i=1}^S \left(\sum_{j=1}^{S_T} \left(\frac{\sum_{k=1}^{C_c} \left(\frac{(n \times m)(n_{ini} \times m_{ini})}{\eta_i} \right)}{\sum_{k=1}^{C_c} \frac{(n \times m)^2}{\eta_i}} \right) \right), \quad (5.26)$$

where S are the number of services currently supported, S_T is the service time for the selected service over C_c number of connected corridors such that error over this constraint is computed as:

$$Error_r = \sum_{i=1}^S \left(\sum_{k=1}^N (TH_v) \right) \eta_i + \epsilon_t'', \quad (5.27)$$

where

$$\epsilon_t'' = ((\vartheta + E) - \overline{P}). \quad (5.28)$$

5.3.5 System Fault Tolerance

The proposed neural model operates towards integration of services with the layered formation of the cooperative network. The corridors selected for transferring data are based upon the availability of connectivity and services between the two units. However, there are certain instance in network operations when such connectivity is not available or lost due to network dynamics. Therefore, a system must be capable of handling such failure causing issues. Fault tolerance for the proposed system is defined as the ability of the system to withstand the uncertain network failures or errors induced in the form of service abstraction during network operations. A network model is required to have a high tolerance to sustain against dynamic network conditions which are not in its favour. Failures in a network can be in the form of node failure, service capturing, session loss, or incorrect neural mapping. These issues can also lead to noise in the network model that can cause devastating effects such as overfitting and negative learning. In the proposed model, these issues are handled by identifying the level of tolerance, and then reassuring neural services to continue with network operations. For a network with neural interfaces, fault tolerance is high if the transmission rate during unfavourable conditions is higher or equal to the minimum required rate. Also, there should be least difference between the $O_{updated}$ and $O_{previous}$. Fault tolerance is measured by inducing certain parameter failure and then, recording output data generated by the system to analyze its behaviour. In the proposed model, a cost based approach was used to analyze the fault-tolerance. This cost is termed as the hindrance cost H_c . Lower the value of H_c , more is the fault-tolerance.

In a network, fault tolerance F_t is given as:

$$F_t \propto \frac{1}{H_c}, \quad (5.29)$$

$$F_t = \frac{\epsilon_r}{H_c}, \text{ such that } \epsilon_r \in Error_r, \quad (5.30)$$

here, ϵ_r defines the error rate during the recovery phase, and cost function H_c is given as:

$$H_c \longleftarrow f(N_e^{act}, N_f, S_i^{act}, Error_r^c, m, S_{aq}), \quad (5.31)$$

where N_e^{act} is the number of active neurons, N_f is the network failures, S_i^{act} is the active services, $Error_r^c$ is the current error rate, m is the number of services and S_{aq} is the accuracy in service mapping. Further, this cost-function H_c can be generalized as:

$$H_c = \frac{\gamma TS}{S_i^{act}} \left(\frac{1}{N_e^{act}} + \frac{1}{S_{aq}} \right) + \frac{\beta}{m} (N_f \times Error_r^c), \quad (5.32)$$

here, γ and β are the hindrance constants where γ defines the service-loss and β defines connection-loss such that $0 \leq (\gamma, \beta) \leq 1$. For the system, fault-tolerance is analyzed by considering the threshold values for generating the hindrance cost.

5.3.6 Integration of Proposed Model With Cooperative Framework

Service Model proposed uses the cooperative network model to allow service based selection of network corridors. These services allow formation of multiple virtual network connections over the single connected corridors. The service model uses a neural interface that operates over the similar selection criteria of the existing model. This neural interfacing interacts with the aerial and ground ad hoc networks over the tier-2 of the proposed service model. Network model formed with the integration of service model with the existing cooperative network framework is shown in Figure 5.10. The internal neural model in this figure is the one proposed as the cooperative network framework

Algorithm 5.2 Algorithm For Selection of Best Possible Corridor

Require: $N, N_e, S_N, N_c \mid N_e > N$
Ensure: Mapped(N) $\mid N \in (S_N, C_N)$
 update_rate $\leftarrow O_{updated}$
while map_formation $\neq$ true **do**
 initialize $\leftarrow$ cooperative_network
 while i $\leq$ N **do**
 formulate_layer $\leftarrow$ connect(S_N, N, N_e)
 generate_neural_interface $\leftarrow \eta_i$
 $TH_v = \sum_{i=1}^S \left(\sum_{j=1}^{S_T} \left(\frac{\sum_{k=1}^{C_c} \left(\frac{(n \times m)(n_{ini} \times m_{ini})}{\eta_i} \right)}{\sum_{k=1}^{C_c} \frac{(n \times m)^2}{\eta_i}} \right) \right)$
 if update_rate $\geq TH_v$ **then**
 $P_{state} \leftarrow$ service_state
 $C_{state} \leftarrow$ cooperative_state
 if Prob(P_{state}) $\geq$ Prob(C_{state}) **then**
 connect()
 else
 reset $\leftarrow$ cooperative_framework(reset(localization))
 end if
 else
 re-initiate
 end if
 corridor $\leftarrow$ collaborative_scheduler()
 X $\leftarrow$ connect(corridor)
 allocate_memory(X)
 transfer()
 i=i+1
 end while
 if map_formation_t=complete **then**
 exit
 else
 continue
 end if
 update_rate(t)=deduce(update_rate)
end while

for multi-UAV guided networks in the Chapter 3 (Figure 3.3 and Figure 3.4). Now, the predictable system state P_{state} over the coordinated network will be computed as:

$$P_{state} = \begin{bmatrix} C_{state} \\ SN_{state} \end{bmatrix} + \epsilon'', \quad (5.33)$$

where C_{state} is the cooperative model state defined over the local reference frame [166] such that:

$$C_{state} = \begin{bmatrix} HS R_U^t \\ Min_R \end{bmatrix}, \quad (5.34)$$

where $_{HS}R_U^t$ is the rate of change evaluated from Ref. [166] and Min_R is the minimum rate of change w.r.t. local reference frame. Also, SN_{state} is the service neural state defined w.r.t. service model such that:

$$SN_{state} = \begin{bmatrix} \eta_i \\ TH_v \end{bmatrix}. \quad (5.35)$$

The integration of service model allows selection of efficient corridors for transmission and enhanced connectivity with lesser memory consumption. The best possible corridor selection is performed using Algorithm 5.2 on the basis of the system equations.

Table 5.1: Parameter Configurations For Service Provisioning Analysis

PARAMETER	VALUE	DESCRIPTION
Area	5000x5000 sq. m	Coverage Area
Ground Nodes	20, 30, 40, 50	Nodes on Flat Grid
UAVs	5, 8, 10	UAVs for guidance
Mac Layer	802.11 b	IEEE Standard for wireless
Propagation Radio Model(Ground)	Two Ray Ground	Topological Model
Propagation Radio Model(Aerial)	Two Ray Ground	Topological Model
Mobility Model(Ground)	Random waypoint	Node movement model
Mobility Model(Aerial)	Fixed waypoint	Node movement model
Simulation Runs	1000s	Number of analysis runs
Service Interval	1s	Analysis Interval
Threads	1-10	Simultaneous handlers
Collaborative Task	Search and Track	Scheduler Task
Minimum Service Variables	5	Minimum service handlers
Maximum Service Variables	20	Maximum service handlers
Minimum Control Variables	1	Minimum service controllers
Maximum Control Variables	2 × Configurable Parameters	Maximum service controllers

5.4 Performance Evaluation

The proposed 3-tier service-oriented neural model was simulated using neural API created over Matlab. The analyses were carried using a mathematical simulator designed for implementation of the proposed model coded in Matlab, and NS-2 was used to generate traffic. The model was tested for its spatial and temporal dependencies. Services were generated for onboard, intra as well as inter-network analysis. 5, 8 and 10 UAVs were used that maneuvered over an area of 5000x5000 sq. m consisting of 20, 30, 40 and 50 nodes. 50 simulation runs were performed to record the average data for various parameters considered for service model. Each simulation was set to 1000 seconds with an interval of 1 second for recording and generation of vehicle services. Number of threads

varied from 1 to 10. Search and tracking of the area were taken as primary task and fed into the scheduler that performed parallel analysis to generate the service connectivity matrix. The details of the remaining parameters configured for analysis of the proposed model are shown in Table 5.1. Simulations allow analysis of each of the UAV's behaviour and confidence level attained during connectivity with ground nodes. Tier-1 and tier-3 were operated on two distinct systems. One of them act as the simulation platform for ground nodes on flat grid and another for aerial nodes. Tier-2 was integrated with both the systems so as to generate service corridor for cognitive transfers. Each of the service corridors was analyzed using a service analyzer that maintains the logs for data transferred, and also keep track of service exchanges between the network. The standard results obtained for service oriented model that remained constant for whole simulation are shown in Table 5.2, and the detailed results recorded for services operating over the cooperative framework is shown in Table 5.3. For training and validation, neural components were configured for responsiveness, validation, and testing range. The neural modelling derived for the proposed service model was evaluated using a similar pattern as that of the cooperative network framework as shown in Figure 5.11a and Figure 5.11b.

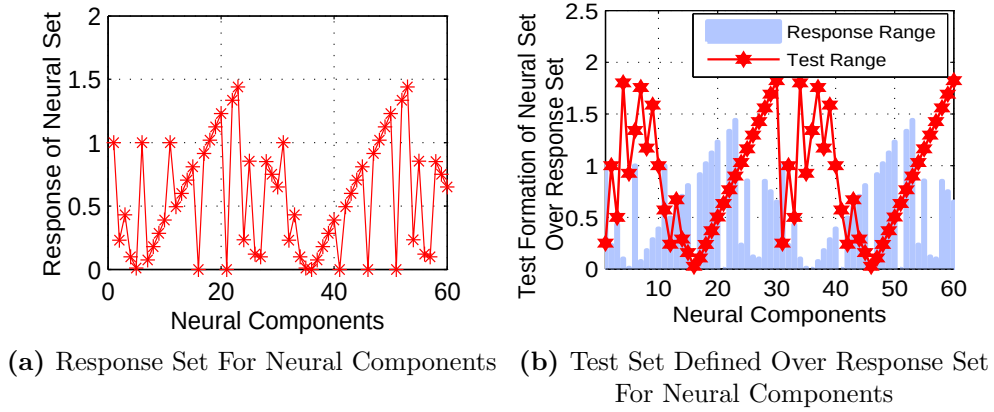


Figure 5.11: Confidence Interval Evaluation of Service Time.

Table 5.2: Service Parameter Configurations

Parameters	Nodes			
	20	30	40	50
Service Variables	5	10	15	20
Neurons	40	60	80	100
Service Connectivity	10	45	105	190
Neural Connectivity	780	1770	3160	4950

Table 5.3: Results For Services Operating Over Neural Model

Aerial Vehicles	Ground Nodes	M_S	S_{ER}	Req_L	Res_L	$S_D(ms)$	$Pack_{Ex}$
5	20	2	0.17033	563	4110	175.1677	10758
	30	2	0.1422	563	4135	222.9176	10772
	40	8	0.11407	563	4160	270.6675	10775
	50	14	0.08594	563	4185	318.4174	10780
8	20	5	0.14	672	5240	366.1673	11342
	30	8	0.1305	672	5255	413.9172	11345
	40	14	0.1121	672	5260	461.6671	11358
	50	18	0.0615	672	5295	509.417	11575
10	20	6	0.133	852	6046	573	11732
	30	12	0.122	852	6052	567	11746
	40	15	0.111	852	6055	558.5	11755
	50	20	0.01	852	6060	416.667	11760

Tabular Notations: M_S represents number of maps shared, S_{ER} is the service error rate, Req_L is the request content length in bytes, Res_L is the response length in bytes, S_D is the service duration in milliseconds, and $Pack_{ex}$ are the number of bytes exchanged during service connectivity.

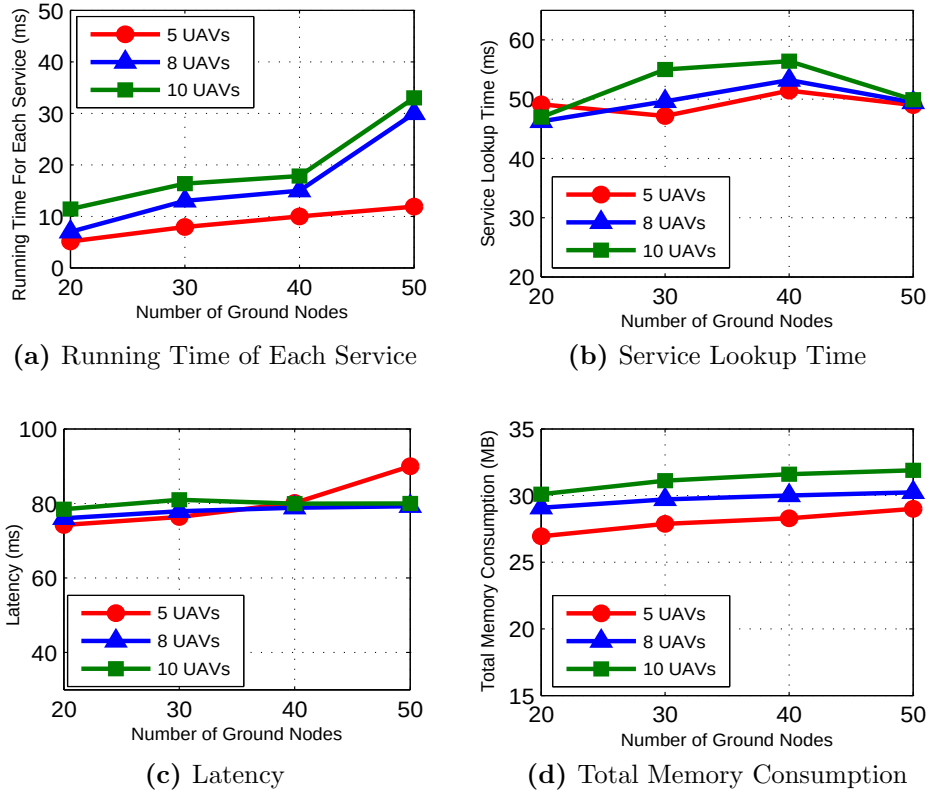


Figure 5.12: Time Consumption and Memory Requirement Analysis of Service Model.

5.4.1 Analysis of Proposed 3-Tier Service Model

Figure 5.12a shows the comparative analysis for running time required by each service in a network of 20, 30, 40 and 50 nodes maneuvered by 5, 8 and 10 UAVs. More the number of nodes, more is the time taken by each service to initialize, thus, increasing the overall running time of services in inter-network communications. For a network comprising of nodes leaving and joining at definite rate, it is mandatory to identify the services

before starting the ad hoc communication. This search time is termed as the service lookup time. Service lookup time is dependent upon the network scalability and node movement. Lesser nodes in pre-defined area requires a simple service selection mechanism, thus, lowering the average lookup time, whereas in a network with more number of nodes require selection algorithms and schedulers that increase the overall lookup time as shown in Figure 5.12b. Latency is the other measure of performance of mission critical systems. Unlike traditional networks, guided networks depend upon a timely receipt of data. Latency also measures the service response time. Lower the latency, more reliable is the cognitive transfers and hence, efficient is the map formation. A highly scalable and adaptable network has lower latency as more number of nodes provide more service corridor for transmission. The comparative analysis of service latency for the guided network is shown in Figure 5.12c. Spatial dependencies of a network can be determined from the memory consumption of total process running during inter-network communication. Memory consumption is directly related to the number of nodes, area to be traversed, a number of cognitive transfers, and the number of computational units operating onboard. A highly scalable network requires more memory i.e. a network with dense populations of nodes consumes more memory as compared to sparsely populated network. The comparison of memory consumption with variable number of aerial vehicles is shown in Figure 5.12d.

Another aspect of analysis of service-oriented model is the confidence interval of the service parameters that defines the range suitable for performance of the network. The confidence interval allows selection of service neural points about the service curves that equally disintegrates all the parameters for enhanced connectivity. Figure 5.13a shows the confidence interval for running time of each service. The confidence interval increases with increment in network aerial nodes, whereas confidence interval for service lookup time does not show much variation w.r.t. increment in number of UAVs as shown in Figure 5.13b. Confidence interval for latency decreases with increase in number of aerial nodes, thus, showing network effectiveness with increased scalability as shown in Figure 5.14a. Average memory consumption for onboard operation remains same in guided network. However, despite of the increase in the number of nodes, confidence variation

does not show much variation for onboard operation on both aerial as well as ground nodes as shown in Figure 5.14b. Thus, results show that 3-tier service oriented model is capable of handling transmission in a guided network with efficient delivery of cognitive as services. Further, service parameters considered for model formation are easily configurable, thus, making the proposed model a reliable medium in formation of aerial guided ad hoc network.

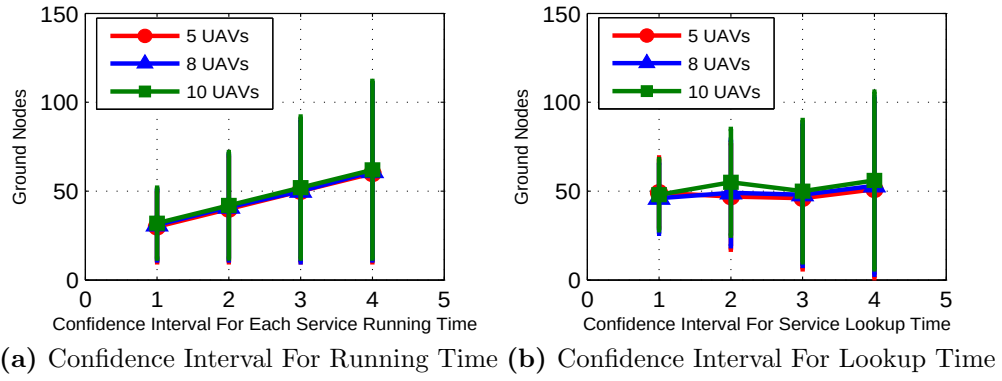


Figure 5.13: Confidence Interval Evaluation of Service Time.

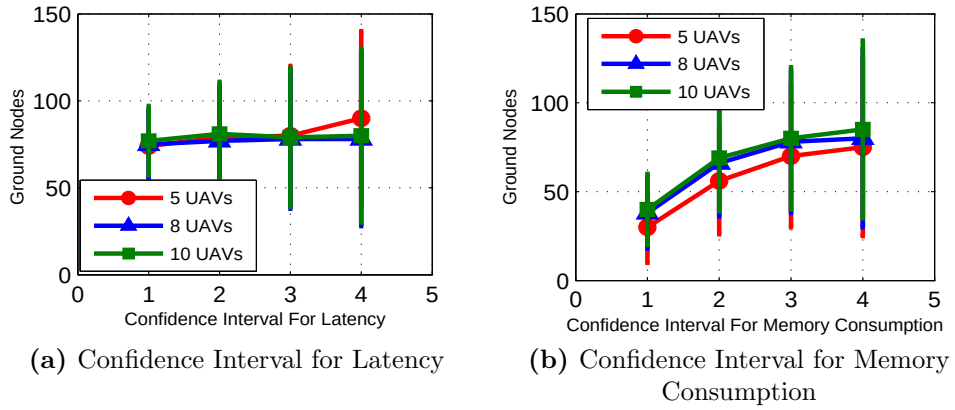


Figure 5.14: Confidence Interval Evaluation of Service Usability.

5.4.2 Analysis of Cooperative Network Formed Using Proposed Model

Figure 5.15a shows the comparative plot for the cognitive formation rate over the cooperative network controlled using 3-tier service generator. More number of nodes offer

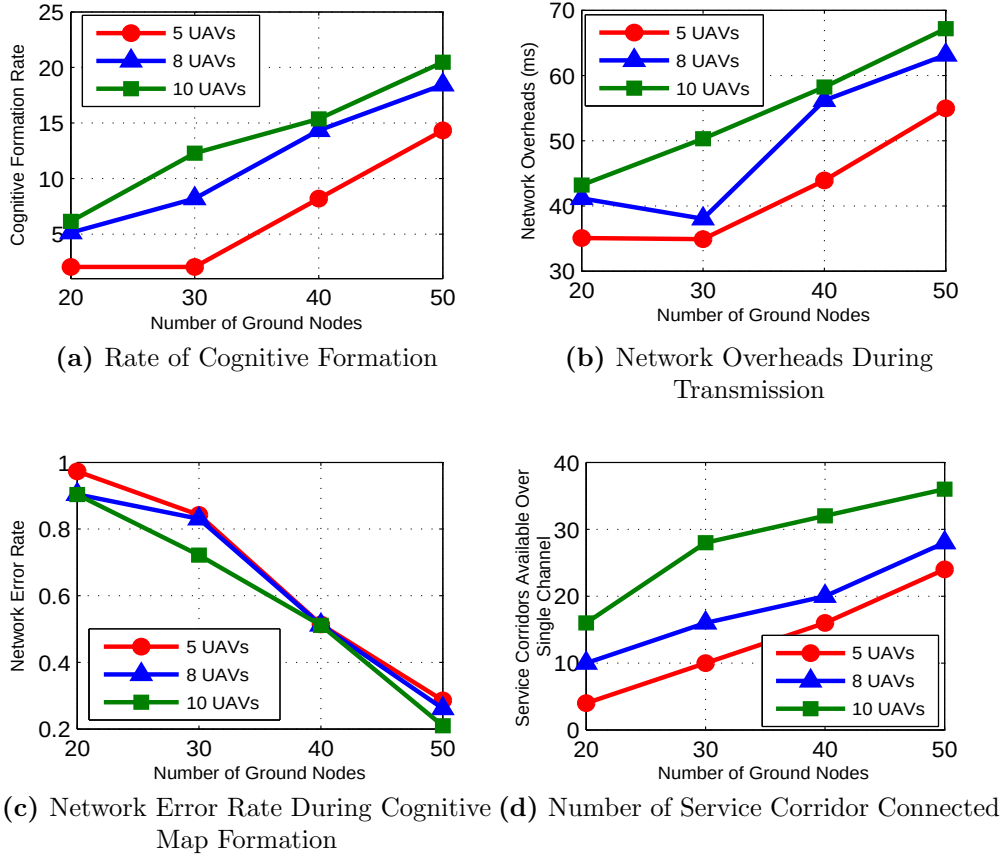


Figure 5.15: Network Parameter Analysis For 3-Tier Cooperative Framework Collaboration.

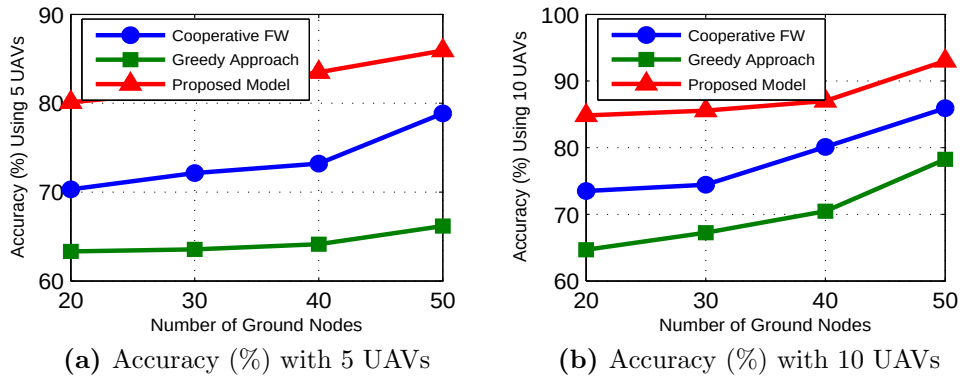


Figure 5.16: Comparison For Accuracy (%) by Proposed Model vs. Existing Cooperative Framework and Greedy Approach.

more corridors for connectivity, also number of service connections increase with increase in network nodes, thus, improving the rate of map formation. During network operations, network overheads might hinder the performance of the map formation, and can also affect network to extreme by causing deadlocks during transmission. Network over-

Table 5.4: Percentage Enhancement In Cooperative Framework Using Proposed Service Model

Aerial Vehicles	Ground Nodes	$R_{cog}(+)$	$C_{for}(+)$	$O_{ver}(-)$	$R_{time}(+)$	$N_{error}(-)$
5	20	24.18	25	10.31	10.97	28.99
	30	33.23	30	10.91	22.88	36.51
	40	27.96	18.75	7.78	21.4	35.18
	50	15.07	20.83	5.377	29.92	35.28
8	20	33.45	30	9.92	18.11	34.57
	30	33.18	36.25	11	20.18	34.75
	40	13.27	30	4.84	30.05	39.73
	50	16.79	33.57	6.53	36.79	34.244
10	20	32.63	37.5	10.91	20.99	33.83
	30	12.478	42.85	4.73	25.72	38.38
	40	22.22	31.25	8.17	33.722	23.07
	50	17.38	22.22	7.08	36.25	34.79

Tabular Notations: R_{cog} represents number of cognitive received, C_{for} is the number of cooperative formations, O_{ver} are the number of overheads, R_{time} is the service running time, and N_{error} are the network errors during service provisioning. Plus (+) and minus (-) denotes increase and decrease in parameter values, respectively.

heads are liable to occur as the ideal network cannot be achieved due to certain network losses that are bound to affect the performance. However, the lower value of network overheads can certainly improve the performance. Figure 5.15b shows that the network overheads increase with the increase in number of network nodes, however, the range of these overheads is quite within the limits such that network performance is not affected. The comparison for network error rate versus a number of UAV maneuvering over ground nodes is shown in Figure 5.15c. If network is based on the artificial intelligence schema, the network error rate should decrease with time as it provides strengthening of the network due to improved learning rate. Justifying this statement, the plot shows decrease in number of network error rate with an increase in the network nodes and more cognitive transfers over the similar network. 3-tier service collaboration with the cooperative network framework provides multiple virtual service corridors over the single transmitting channel. This adds up to the performance with enhanced cognitive transmissions and quick response to unexpected network substitutions as shown in Figure 5.15d. Analysis were also recorded in terms of percentage enhancement of various parameters operating over the initially proposed cooperative network framework [166] using the proposed service neural model as shown in Table 5.4. Further, the proposed service model was tested for its endurance defined as fault tolerance against the service failures. The proposed model was analyzed for its accuracy in mapping services amongst the network nodes, ability to identify faults, hindrance cost during failures, and fault tolerance in comparison with initially proposed cooperative framework [166] and the greedy approach in service mapping. Accuracy is defined as the ratio of successfully mapped services to the total service requests generated during network operations. The comparison was traced

using 5 and 10 UAVs maneuvering over ground nodes for collaborative service generation and corridor selection. Analysis shows that the proposed model offered better accuracy in comparison with existing cooperative framework and greedy approach in service selection. Proposed framework shows average percentage improvement of 13.16% and 21.57% in comparison with cooperative framework and greedy approach, respectively. The plot for accuracy comparison are shown in Figure 5.16a and Figure 5.16b. For comparison

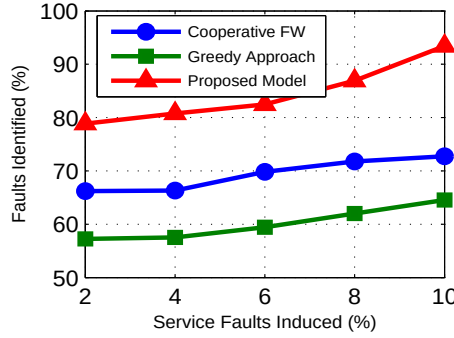


Figure 5.17: Comparison For Identification of Faults by Proposed Model vs. Existing Cooperative Framework and Greedy Approach.

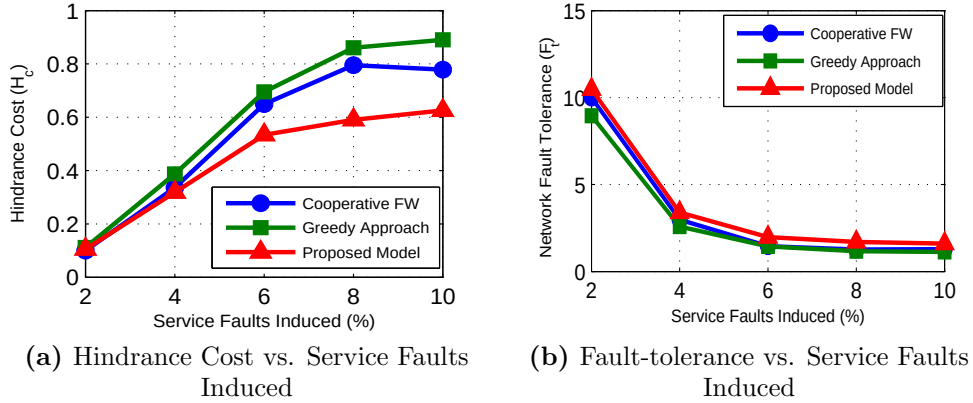


Figure 5.18: Comparison For Hindrance Cost and Fault-tolerance Offered by Proposed Model vs. Existing Cooperative Framework and Greedy Approach.

of accuracy in identification of faults, faults were induced over the service configurations shown in Table 5.2. The percentage variation in faults ranged between 2% to 10%. These faults were generated when the network was transmitting at maximum transfer rates. Analysis shows that the proposed model is capable of identifying 17.7% and 28.5% more errors than the cooperative framework and the greedy approach, respectively. The comparative plot for percentage variation in identification of service faults is shown in

Figure 5.17. Analyses were also traced for network hindrance cost on the occurrence of failures. Hindrance cost was then evaluated on error rate ϵ_r for computing the final cost based fault-tolerance of the network. During network analysis, value of cost constants γ and β was kept fixed at 0.5. With same percentage of fault and using formulation defined in equation (5.30) and (5.32), fault tolerance was computed over normalized hindrance cost H_c for service coordination in proposed, cooperative and greedy approaches. Analysis shows that the proposed model is capable of providing more fault-tolerance against uncertain node and service failures than the existing cooperative framework and greedy approach. The comparative plots for hindrance cost and average fault-tolerance attained by the proposed model vs existing framework and greedy approach with respect to the percentage of faults induced in the services are shown in Figure 5.18a, and Figure 5.18b, respectively. From performance evaluation, it can be analyzed that the proposed service-oriented 3-tier neural model is capable of providing continuous, robust and fault tolerant connectivity between the variedly configured nodes of networks operating in coordination with each other.

5.5 Conclusion

This chapter presented the service provisioning over UAVs coordinated networks. In the first part, a general middleware for service coordination over aerial vehicles is discussed. This middleware limits its functionality to only command and control procedures. To extend its utility, a cross layer formation over this middleware is proposed that allows service dissemination over collaborative flying networks. However, this service based cross layer design allowed efficient service dissemination, but does not provide a continuous support for service coordination between the aerial ad hoc formations and the ground ad hoc formations. For this, a service oriented 3-tier neural model is proposed that allows formation of cooperative network between aerial and ground ad hoc network. The proposed model utilizes the layered structure mapped by neural interface. The proposed model provides service based abstraction and control of various components to form a guided system. It offers data transfers with higher accuracy and lower latency.

Chapter 6

QoS Scheduling For End Users In Ground and UAV Ad Hoc Networks^{†‡}

Quality of service aims at providing heavy traffic to end users. This allows the facility of multimedia applications to be utilized without any communication delay [247] [248]. This chapter proposes a new Quality of Service (QoS) provisioning over ground ad hoc networks-MANETs, and the UAVs ad hoc networks. The proposed QoS scheduling is developed in two parts. The first part presents an estimation-based queue scheduling to provide QoS to end users in mobile ad hoc networks (MANETs). Second part utilizes this estimation strategy to form a UAV assisted queue scheduling approach.

In the initial approach for ground ad hoc networks, integration of Quaternion-based Kalman Filter is performed that predicts the required bandwidth and the network delays with higher accuracy. It is shown that bandwidth can be optimized, but it is not possible to aloof delays in the network. Thus, while implementing such admission control procedures, estimation process allows control over delays and sustain them from going beyond certain threshold value. The proposed model defines a new set of traffic conditions with novel mathematical modeling, which provides support for queue scheduling. A weight based strategy is used to select the appropriate queue for efficient transmission and guaranteed QoS to end users.

[†]Vishal Sharma and Rajesh Kumar, “Estimation-Based Queue Scheduling Model To Improve QoS For End Users In MANETs”, Computing and Informatics- SCIE Indexed [accepted], [I.F. 0.504].

[‡]Vishal Sharma, and Rajesh Kumar, “UAVs Assisted Queue Scheduling in Ground Ad Hoc Networks”, International Journal of Ad Hoc and Ubiquitous Computing, Inderscience- SCIE Indexed,[Under Review], [I.F. 0.554].

6.1 Quaternion Filter Based Network Formation

An efficient network can be described as combination of models that have the sensing capabilities regarding the data flow over the network. Sensing capabilities to a network can be provided through estimation process. For a model ensuring QoS to end users, it is required to estimate the traffic flow and the actual bandwidth requirement. The gap between the actual and allocated resource can cause an increase in network overheads thus, lowering the QoS to end users. Therefore, an efficient network model is required that can observe and predict the network conditions and then take decisions for the correct allocation of resources such as the required bandwidth for queue transmission. This estimation of network state can be achieved using Quaternion Kalman Filter. Quaternion filter analyzes the prior and posterior network states to form the prediction equation. This prediction equation allows to evaluate the next required resource. Therefore, it is required to form system equation and then perform traffic prediction before implementing the QoS scheduling Model. The equation formulated to represent the system behaviour are derived using existing Quaternion Algebra over Kalman Filter [224] [223] [249] and integrating it with network parameters and standard network equations. However, no approach exists in literature that has highlighted such intensive application of Quaternion Kalman Filter over a mobile ad hoc network for queue scheduling predictions.

Let $\overline{w}_1$, $\overline{w}_2$, $\overline{w}_3$ be the weighing factors for link prediction, traffic prediction and queue prediction, respectively, that are defined over quaternion [224] as:

$$\overline{w}_1 = X_1M + X_2L + X_3Rd + X_4, \quad (6.1)$$

$$\overline{w}_2 = Y_1N_p + Y_2N_{rt} + Y_3T_{pkl} + Y_4, \quad (6.2)$$

$$\overline{w}_3 = Z_1QMS + Z_2TQS + Z_3WPU + Z_4, \quad (6.3)$$

where $X_1, X_2, X_3, Y_1, Y_2, Y_3, Z_1, Z_2, Z_3$ are the hyper-imaginary numbers [224] for bandwidth and delay estimators, M is node mobility, L is Link state, Rd is the relaying

distance between any two nodes, N_p is the number of presentation units, N_{rt} is the number of retransmission, T_{pkl} is the total transmission time, QMS is the memory slots occupied in queue, TQS is total available memory slots in queue, WPU is total weight of presentation unit and X_4, Y_4, Z_4 are the scalar that defines the measurable quantities. Quaternion defined in the equations(6.1, 6.2, 6.3) allow application of vector spaces with complex as well as real numbers over 4-dimensional parameters defined over the network. It also provides the values that can be easily fed as an input to other analysis system such as Gauss-Markov model or Euler model. From definition of quaternion [224], $\overline{w_1}$, $\overline{w_2}$, $\overline{w_3}$ can be expressed as:

$$\begin{aligned}\overline{w_1} &= \begin{bmatrix} w_1 \\ X_4 \end{bmatrix} \\ &= \begin{bmatrix} M & L & Rd & X_4 \end{bmatrix}^T,\end{aligned}\tag{6.4}$$

$$\begin{aligned}\overline{w_2} &= \begin{bmatrix} w_2 \\ Y_4 \end{bmatrix} \\ &= \begin{bmatrix} N_p & N_{rt} & T_{pkl} & Y_4 \end{bmatrix}^T,\end{aligned}\tag{6.5}$$

$$\begin{aligned}\overline{w_3} &= \begin{bmatrix} w_3 \\ Z_4 \end{bmatrix} \\ &= \begin{bmatrix} MQS & TQS & WPU & Z_4 \end{bmatrix}^T.\end{aligned}\tag{6.6}$$

Let $\overline{w}$ be the overall network prediction weighing factor quaternion defined over $\overline{w_1}$, $\overline{w_2}$, $\overline{w_3}$ as:

$$\overline{w} = A_1\overline{w_1} + A_2\overline{w_2} + A_3\overline{w_3} + A_4,\tag{6.7}$$

such that,

$$\begin{aligned}\overline{w} &= \begin{bmatrix} \mathbf{w} \\ A_4 \end{bmatrix} \\ &= \begin{bmatrix} \overline{w_1} & \overline{w_2} & \overline{w_3} & A_4 \end{bmatrix}^T, \end{aligned} \quad (6.8)$$

Quaternion defined in equation(6.7) provides with the aggregated equation of multiple parameters defined as an individual entity over single quaternion. This provides with a single solution that can be used for further analysis using less computational time. Let α be the difference in queue rate due to mobility for each node (assuming α to be constant and same for each node), $\mathbf{w}$ is defined using Quaternion [224] as:

$$\mathbf{w} = \begin{bmatrix} k'_x \sin(\frac{\alpha}{2}) & k''_x \sin(\frac{\alpha}{2}) & k'''_x \sin(\frac{\alpha}{2}) & 1 \\ k'_y \sin(\frac{\alpha}{2}) & k''_y \sin(\frac{\alpha}{2}) & k'''_y \sin(\frac{\alpha}{2}) & 1 \\ k'_z \sin(\frac{\alpha}{2}) & k''_z \sin(\frac{\alpha}{2}) & k'''_z \sin(\frac{\alpha}{2}) & 1 \end{bmatrix}. \quad (6.9)$$

Note that for MANETs, topology is flat grid i.e. the axis of movement will be defined for (x,y) only with $z = 0$. Thus, $\mathbf{w}$ becomes:

$$\begin{aligned}\mathbf{w} &= \begin{bmatrix} k'_x \sin(\frac{\alpha}{2}) & k''_x \sin(\frac{\alpha}{2}) & k'''_x \sin(\frac{\alpha}{2}) & 1 \\ k'_y \sin(\frac{\alpha}{2}) & k''_y \sin(\frac{\alpha}{2}) & k'''_y \sin(\frac{\alpha}{2}) & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\ &= \begin{bmatrix} k' & k'' & k''' & 1 \end{bmatrix}, \end{aligned} \quad (6.10)$$

where k', k'', k''' are the unit vectors along the axis of movement defined for $\overline{w_1}, \overline{w_2}, \overline{w_3}$ [224]. This can be noted that in actual mobile ad hoc networks, there might be situations where no LOS (Line of Sight) is available due to blocks of terrestrial objects such as buildings. Thus, it is required to include the area-based state equations and also form prediction equation for geographical set up of area. However, in this case, pre-defined geographical constraints are provided thus, neglecting any flaw due to geographical bounds. Also, for an efficient network, interference model is required to address the activity involved due to wireless signals. However, the model considered operates at Layer 2 of ad

hoc network, thus neglecting any parameters of underlying Layer 1. Also, interference is not considered due to reason that separate channels are used for transmission analysis that do not allow any conflict between signals that may arise if transmitted over the same channel. It is to be noted that a network model is never fully efficient, thus, for prediction equations, it is necessary to model network errors for providing effective QoS to end users. Let β_m be the measured bandwidth requirement for queue transmission such that

$$\beta_m = \beta_r + R + N_e + E_c, \quad (6.11)$$

where β_r is the actual bandwidth allocated, R is the routing overheads, N_e is an average network error and E_c an is error due to queue crashing. Queue crashing is defined as failure in queue formation and transmission due to the unambiguous behaviour of the priority queue selector. Note that R , N_e and E_c follows a Gaussian distribution. These variables can be estimated using regression models such as Gauss-Markov Model. Routing overheads R can be estimated using the theory of least squares, i.e. R for i^{th} queue of j^{th} node is defined as:

$$R_{j,i} = \lambda_0 + \lambda_1 t_{j,i1} + \lambda_2 t_{j,i2} + \dots + \lambda_n t_{j,in} + \epsilon_{j,i}, \quad (6.12)$$

where λ denotes the average network error rate at active time slot $t_{j,in}$ for j^{th} node and ϵ_i is error on the i^{th} queue observation. Also, ϵ is absolutely random and is normally distributive with mean 0 and variance σ^2 . Application of least squares over Gauss-Markov model allows conjecture for various parameters as number of equations formulated for the system overpowers the number of real time values available in the underlying system. Note that the $\lambda_0 + \lambda_1 + \lambda_2 + \dots + \lambda_n$ are sum of sets of finite but undefined value since error rate can be affected by random and any of the network parameters. In general, these can be expressed as matrices by taking the expectation of the above equation as:

$$E[R_j] = R' T', \quad (6.13)$$

where R' is matrix of network errors and T' is matrix of network time slot. Applying Gaussian-Markov model, R' can be expressed over $f(R_1, R_2, \dots, R_n)$ which is an estimator

function. Sum of unknown parameters $\lambda_1 + \lambda_2 + \dots + \lambda_n$ can be expressed as estimator function $h(\lambda_1, \lambda_2, \dots, \lambda_n)$ such that

$$E[f(R_1, R_2 \dots R_n)] = h(\lambda_1, \lambda_2, \dots, \lambda_n), \quad (6.14)$$

where $h(\lambda_1, \lambda_2, \dots, \lambda_n)$ can also be evaluated using linear estimator as:

$$h(\lambda_1, \lambda_2 \dots \lambda_n) = L' R', \quad (6.15)$$

where L' is the link speed matrix at time slot t_i . Thus, the whole system of multiple quaternion can be expressed as:

$$y(t) = \begin{bmatrix} \overline{w}_1(t) & \overline{w}_2(t) & \overline{w}_3(t) \\ N_e(t) & R(t) & E_c(t) \end{bmatrix}, \quad (6.16)$$

using the definition of quaternion derivation and error model [224], the state governing equation is given as:

$${}_G \overline{\mathbf{w}}'(t) = \frac{1}{2} \Omega y(t) \beta_r {}_G \overline{\mathbf{w}}(t), \quad (6.17)$$

where Ω is the vector product matrix [224]. It is to be noted that from actual definition, only global frame is considered. This is specified by the reason that GPS is used for node positioning and no local positioning reference frame is used in this case. Let $\beta_t, \beta_{t-1}, \dots, \beta_1$ be the vector that denotes the bandwidth allocation for time $t, t-1, t-2, \dots, 1$ and $D_t, D_{t-1}, \dots, D_1$ be the vector that denotes end to end delay for the same time slot. Applying Kalman Filter [223], following equations are formed:

$$\beta_t = G_t \overline{w}_1(t) + O_t, \quad (6.18)$$

$$D_t = H_t \overline{w}_2(t) + O'_t, \quad (6.19)$$

where

$$\overline{w}_1(t) = G'_t \overline{w}_1(t-1) + S_t, \quad (6.20)$$

$$\bar{w}_2(t) = H'_t \bar{w}_2(t-1) + S'_t. \quad (6.21)$$

Here, G_t, G'_t, H_t, H'_t denote the set of known parameters which can be obtained from values defining changes in network models over time. These values can be considered constant over the time. O_t, O'_t are the observation errors and S_t, S'_t are the system errors that provides values for any kind of network delays. Equations (6.18-6.21) provide state equations over which Kalman Filter can be applied for estimations. It can be noted that these equations are generalized from standard equation of line with slope k i.e. $y=mx+k$. Thus, using quaternion and then feeding the results into linear estimation system would simplify the prediction process. These predictions can be carried out using simple prediction theory approaches such as Bayes' formulations for probability [223] [250]. Therefore, using the similar approach, following state equations can be used for traffic and network state prediction.

$$P(\bar{w}_1(t)|\beta_t, \beta_{t-1}) = \frac{P(N_{avg}|\bar{w}_1(t), \beta_{t-1}) \times P(\bar{w}_1(t)|\beta_{t-1})}{P_{total}(N_{avg}|\bar{w}_1(t), \beta_{t-1})}, \quad (6.22)$$

similarly,

$$P(\bar{w}_2(t)|D_t, D_{t-1}) = \frac{P(R_{avg}|\bar{w}_2(t), D_{t-1}) \times P(\bar{w}_2(t)|D_{t-1})}{P_{total}(R_{avg}|\bar{w}_2(t), D_{t-1})}. \quad (6.23)$$

For overall network estimation, state knowledge equation is given by:

$$P({}_G\bar{\mathbf{w}}'(t)|F_t, F_{t-1}) = \frac{P(S_{avg}|{}_G\bar{\mathbf{w}}'(t), F_{t-1}) \times P({}_G\bar{\mathbf{w}}'(t)|F_{t-1})}{P_{total}(S_{avg}|{}_G\bar{\mathbf{w}}'(t), F_{t-1})}, \quad (6.24)$$

where

$$F_t = M_t(\beta_t, D_t){}_G\bar{\mathbf{w}}'(t) + K_t(O_t, O'_t), \quad (6.25)$$

and

$${}_G\bar{\mathbf{w}}'(t) = M'_t(G_t, G'_t){}_G\bar{\mathbf{w}}'(t-1) + K'_t(S_t, S'_t). \quad (6.26)$$

N_{avg} , R_{avg} and S_{avg} are the average network error, average routing overheads and average system error, respectively. The equations (6.24-6.26) allow Kalman Filter estimator to be integrated with queue scheduler for efficient allocation of QoS to end users.

Let $X_{\beta(t)}$, $X_{D(t)}$ denote the observation matrix for bandwidth and delay estimation, respectively. Over the network with a global frame of reference, $X_{\beta(t)}$, $X_{D(t)}$ will be defined using [224] as:

$$X_{\beta(t)} = [\Pi \lfloor_G \bar{\mathbf{w}}'(t) f_0(\beta_t) \times \rfloor L'], \quad (6.27)$$

and

$$X_{D(t)} = [\Pi \lfloor_G \bar{\mathbf{w}}'(t) f_0(D_t) \times \rfloor L'], \quad (6.28)$$

where f_0 denotes the estimator function. The noise error N_0 is defined as:

$$N_0 = (\Omega y(t) \beta_r) - P_e + Avg(N_{eavg}, R_{eavg}, S_{eavg}), \quad (6.29)$$

where P_e is the process error. Let P be the covariance matrix over $y(t)$ and $P'_{\beta(t)}$ and $P'_{D(t)}$ be the covariance over N_0 such that:

$$P'_{\beta(t)} = X_{\beta(t)} P X_{\beta(t)}^T + N_0, \quad (6.30)$$

and

$$P'_{D(t)} = X_{D(t)} P X_{D(t)}^T + N_0. \quad (6.31)$$

Hence, using Indirect Kalman Filter [249], Kalman gain K_β and K_D will be given as:

$$K_\beta = P X_{\beta(t)}^T P'_{\beta(t)}, \quad (6.32)$$

and

$$K_D = P X_{D(t)}^T P'_{D(t)}. \quad (6.33)$$

Therefore, using property of Kalman Filter [224] [249], correction $C_\beta(e)$ and $C_D(e)$ for bandwidth and delay, respectively, will be defined as:

$$C_\beta(e) = K_\beta N_0, \quad (6.34)$$

and

$$C_D(e) = K_D N_0. \quad (6.35)$$

6.2 Proposed Model

In the proposed model, admission control and scheduling process were integrated with estimation-based Quaternion Kalman Filters that served the purpose of successful allocation of quality of services to end users, thus, improving the network quality and sustaining high link speed. The proposed model is based upon two major factors; number of presentations units (PU) and maximum number of handover possible in the network. The technique used in the proposed model divides the nodes into groups of transceiver units, each with defined number of channels available for transmission. The presentation units were divided into three categories: High Priority, Regular, and Normal PUs. The packet priority was noticed on type of presentation unit. The receiver at each device arranged the packets on the basis of PUs and feed them into the transmitting queue that was maintained for each device which was using it either for direct transmission or relaying it for multi hop transmission. A timely window was deployed that controlled the entry of packets after a fixed interval; a weight was assigned to each queue based upon priority and size. The queue with more number of priority packets was assigned a more negative weight. For this, the variables of Kalman Filters provide the state observation values which were used to predict the network behaviour before queue selection with higher accuracy. Thus, the queue with lowest weight was transferred first followed by another similar process between the next intervals using network prediction chart. The interval can be managed on the basis of bandwidth allocated or available at that instance of transmission time. Another major concern was the number of handovers which includes the time to switch between nodes or devices when a particular node goes beyond the range and another node is used for relaying purpose. This has to be maintained to know actual bandwidth available as change in the path may cause variation in bandwidth available for allocation to a particular channel used by transmitter to transmit data of particular queue. Also, it is required to compute overall delays that may degrade the performance at a later stage. Thus, packets with high priority get transmitted at once followed by the regular packets. With extremely high bandwidth, the interval between packet transmissions decreases, hence, lowering the amount of memory units to be used for maintaining

queue. This gives advantages during the peak transmission of presentation units. The proposed model, as shown in Figure 6.1, can ensure following key points:

- (i). It serves queue with the most number of priority packets as well as with lower weight.
- (ii). Bandwidth allocation is dynamic and therefore, no need to reserve bandwidth, hence, improving the quality of each presentation unit arriving at the end user.
- (iii). Each queue is handled with set transmission criteria. This maintains the link quality; and reliability of the network also stabilizes.
- (iv). It reduces the delay effect due to pre-calculation of queue size and estimation of transmission slot available on particular nodes.
- (v). The estimation process provides the self regulation and optimization in bandwidth selection and delay control.

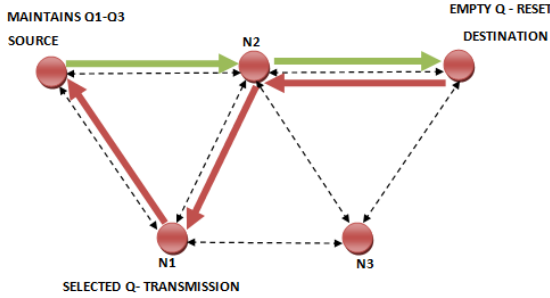


Figure 6.1: Queue Transmission

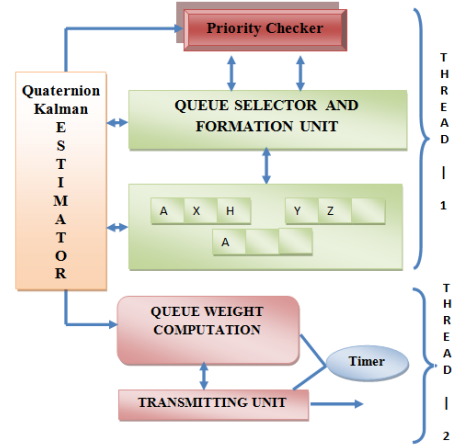


Figure 6.2: Operational View of Proposed Scheme

For analysis three queues, namely, Q1, Q2, and Q3 for transmission of packets over the network were considered. The configuration of these queues is shown in Table 6.1. The weight is assigned on the basis of packet priority. Q1 gets transmitted first, followed by Q2 and Q3. From Table 6.1, it is clear that final weight calculated is used to compare the transmission criteria for queues, however, if the weight would have been same for any of the queues, then, a number of high priority packets would have been considered for deciding the transmission order. The pseudo code for queue formation and transmission is given in Algorithm 6.1. This algorithm is handled by two thread operations, one for queue formation and updating of the table while the other handles queue selection and

Table 6.1: Queue Configuration (*the transmission Limit is defined in terms of PU transferred in unit time slot)

Parameters	Q1	Q2	Q3
Number of Memory Slots	8	8	8
Extended Memory Slots	5	5	5
Transmission Limit*	8	8	8
Memory Slots Occupied	7	8	5
Weight for High Priority Bit	-3	-3	-3
Weight for Regular Bit	-2	-2	-2
Weight for Normal Bit	-1	-1	-1
Number of High Priority PU	3	1	1
Number of Normal PU	3	4	4
Number of Regular PU	1	3	0
Weight Calculated	-14	-13	-7

Algorithm 6.1 Queue Formation and Transmission Algorithm

Require: $k \geq 0 \vee N$

while $i < k$ **do**

Priority_Checker (PU) $\leftarrow$ *Kalman_Filter_Estimate*(PU)

Maintain PriorityTable(*Weight*)

Select Queue with empty slots

if *SelectedQueue* > 1 **then**

Compare Queue with lower weight

InsertPU $\leftarrow$ *Kalman_Filter.QueueEstimate*(*LowerWeightQ*)

UpdatePriorityTable(*Weight*)

TransmissionScenario $\leftarrow$ *Kalman_Filter_Bandwidth*(PU)

CheckNetworkState $\leftarrow$ *Kalman_Filter_NetworkStatus*(PU)

Timecount (T) $= t + \delta t$, where T is total time including reset time t

Transmit Queue

end if

end while

Reset

transmission process, as shown in Figure 6.2. Thus, the overall scenario does not lead to any overheads that might have occurred due to single thread handling all the operations serially. In general, parallel run of the algorithm saves the time required for queue generation and selection of packets for transmission. The algorithm works by computing a number of presentation units to be transferred that are fed into Quaternion Kalman Filter-based priority checker module which computes the priority of these packets based on configuration as shown in Table 6.1. This whole process is carried out in $O(k)$ time, where k denotes the number of presentation units. The Queue selection for lower weight and insertion of presentation unit are done in $O(1)$ time. Thus, for thread handling the queue formation, the overall complexity remains to be $O(k)$. For other thread, operating simultaneously, the complexity depends upon the number of presentation units fetched and transmitted over the network. Also, if more such units are used on a single node the complexity becomes $O(n * k)$, where n denotes number of such units on single node. This can be controlled by running multiple threads for alternative units. But, this may lead to more computation and higher consumption of processing energy.

6.3 Bandwidth and Delay Analysis

In the proposed model, queue scheduling has been optimized at the node end, but there is always issue when transmission is to be considered due to mobile nodes that may vary the transmission process due to various parameters such as delay and bandwidth. Delays can never be removed completely from a real time application, however, delays can be lowered to such a level that these can be neglected. The bandwidth offered to a node also controls the amount of delay in the network that might have occurred due to low performance at node because of various computations being performed simultaneously.

6.3.1 Bandwidth Evaluation

Bandwidth depends upon the number of channels used per node in the overall network. For real time applications, the number of channels may vary from node to node. The overall network can be classified by the number of channel variants in the network structure. For N number of hops, if the same number of channels are indulged, then the overall channels used will be computed as $N \times C$, where C denotes number of channels used by a node. The numbers of channels to be used are pre-determined and pre-configured. Also, $N \times C_i = N$ i.e. $C_i = 1$ where i varies from 1 to N, if single channel is used on each node. Thus, bandwidth will depend upon the minimum rate that each node can follow to transmit presentation units and number of channels used for transmission process. Therefore, the required network bandwidth will be calculated as:

$$\beta = \frac{Min_{rate} + \frac{20\% of RoutingOverheads}{LossTime}}{No.ofchannelused}, \quad (6.36)$$

$$= \frac{Min_{rate} + \frac{20\% of R_H}{L_T}}{\sum_{i=1}^N C_i}, \quad (6.37)$$

$$= \frac{Min_{rate} + \frac{20\% of R_H}{L_T}}{N}, for C_i = 1, \forall i. \quad (6.38)$$

Using equation(6.34), the estimated bandwidth will be given as:

$$\beta = \frac{Min_{rate} + \frac{20\%of R_H}{L_T}}{N} + C_\beta(e). \quad (6.39)$$

The approach will be better for network with smaller data applications to be handled between the hops, however, in an extensive network, hybrid nodes are present. The set of these nodes differs in configuration, and the type of data that can be transmitted over them depends upon the number of channels used and link capacity of each node. The minimum rate of transmission will vary as per the transmission policy of the network. Here, transmission policy, implies whether all hops are transmitting at the same rate or variable rate. For same rate, the number of channels used per node will remain same. Therefore, the power requirement for computation will remain same for all hops. For networks operating on multimedia type of applications, constant rate for each hop can be used. For them, the minimum transmission rate will be selected as follows:

$$Min_{rate} = \min(\sum_{k=1}^N R_k),$$

where R_k denotes achievable rate for each node. Min_{rate} is the minimum rate feasible to each node in the network for transmission process [7].

6.3.2 Delay Evaluation

Delays in a network are that time slot during which the transmission rate is lower than the certain threshold value defined for successful transmission to the receiver. A delay in a network cannot be aloof completely, however, for enhanced performance, delays should be controlled under certain threshold value. Thus, it is required to identify those factors that will extensively affect the network in terms of performance failing which may cause various network overheads. Considering the same network model with N number of nodes, the mobility can be denoted as $m_1, m_2, \dots, m_k \forall k \in N$, where k denotes the active number of connections. Also, $d_1, d_2, \dots, d_k \forall d \in D$, is the distance covered by each node in defined range where D is the end to end distance. Active number of connections

are considered keeping in mind that the network delays will be considered only for those nodes which are configured to be active during scenario generation of the network. Let, $T_{arrival}$ be the time taken by node to arrive in range, I_t be the network idle time, D_p be the Propagation Delays, and T_{pkl} be the Transmission Delays. Hence,

$$Delays_{(t)} = T_{arrival} + I_t + D_p + T_{pkl}, \quad (6.40)$$

$$D_p = \frac{N_{rt}(N-1)N_p}{\beta}, \quad (6.41)$$

$$T_{arrival} = \sum_{i=1}^k \frac{d_i}{m_i}. \quad (6.42)$$

Therefore,

$$Delays_{(t)} = \sum_{i=1}^k \frac{d_i}{m_i} + I_t + \frac{N_{rt}(N-1)N_p}{\beta} + T_{pkl}. \quad (6.43)$$

Equation (6.41) computes delays related to transmission process. It is the idle time for which no transmission occurs in the network. This can be computed as an average wait time of the network structure. Average wait time can be defined as the number of cycles that were utilized without any data transfer. Also, T_{pkl} is the time loss due to packet loss. It can also include various losses such as fading loss, vegetation loss. For a controlled network, i.e. network in which mobility of node is strictly defined and is manageable, all active nodes cover equal distance during the transmission process. For such network scenario, node approaching time $\sum_{i=1}^k \frac{d_i}{m_i}$ changes to $\frac{d_{avg}}{m_{avg}}$. Therefore, in general for this type of mobile node network, the delays will be defined as:

$$Delays_{(t)} = \frac{d_{avg}}{m_{avg}} + I_t + \frac{N_{rt}(N-1)N_p}{\beta} + T_{pkl}. \quad (6.44)$$

Using Equation(6.35), estimated end to end delay will be given as:

$$Delays_{(t)} = \frac{d_{avg}}{m_{avg}} + I_t + \frac{N_{rt}(N-1)N_p}{\beta} + T_{pkl} + C_D(e). \quad (6.45)$$

Here, N_{rt} is the number of retransmissions carried when a packet is dropped or when no ACK is received from the receiving unit. The model proposed here is capable of handling packet admission as well as scheduling tasks. Hence, avoiding any congestion in the network. It ensures QoS to end users with improved efficiency over traffic differentiation services. The working of the algorithm shows that bandwidth is optimized and is not wasted as it would have been during the pre-reservation techniques. The slots left idle during the transmission process are also utilized. Packet categorization and resource allocation improved the network survivability assuring the successful transmission of packets up to the end receiver.

6.3.3 Network Throughput

In a mobile ad hoc network, network throughput varies due to continuous link switching between the nodes. Throughput will improve only if the round trip latency is decreased or maintained at a certain limited value. This can be computed either by pre-deciding the limit of maximum threshold value for round trip latency or by decreasing the transmission delays. If ACK is not a concern for protocol chosen for transmission, i.e three-way-handshake is not used for transmission process, latency need not to be doubled as no round trips are involved. Thus, the transmission process is of concern in throughput evaluation. In the proposed model, ACK is not a bounded requirement for calculations. Therefore, evaluation will depend upon the delays evaluated for the overall network. Here, the boundary conditions for throughput calculations remain same for queue formation and transmission. The dependent factors include latency in terms of node selection or transmission handovers over the link, the type and number of channels incorporated over the links active at transmission time. Thus, network throughput can be evaluated for two types of scenarios.

6.3.3.1 Common Channel-Based Network

In this scenario, a common channel-based network model is considered, which operates on a single type of channel with same active frequency range over each node. Also,

the number of channels used on each node are same. For such type of network model, throughput (Th) can be computed as:

$$Th = \frac{N_p}{N_{latency}}, \quad (6.46)$$

where $N_{latency}$ is the network latency computed as:

$$N_{latency} = \frac{(N_p - N_{rt})}{link_{capacity}} + Cmp_{delays}. \quad (6.47)$$

Here, Cmp_{delays} are the computational delays for overall transmission process, $link_{capacity}$ is the maximum traffic that the network can support and it is computed as:

$$link_{capacity} = \frac{Min_{rate}}{Number_of_channels_used}, \quad (6.48)$$

where Cmp_{delays} are bound to occur even if the network is performing to its capacity. The value of Cmp_{delays} ranges from 8 ms to 10 ms. $N_{latency}$ is the transmission delay that is computed over the link capacity and the number of bits to be transferred between the particular link. The throughput computed will be defined only for single link active at the time of transmission process. Also, for network that rely on ACK for error detection and correction, the $N_{latency}$ will be twice for round trip. Since, the network model is similar, i.e. each node has same active channel frequency with equal number of channel usage, thus, it is efficient to compute overall throughput of the network as $N_{latency}$ will be same for each link that becomes active during transmission. Therefore, for overall network,

$$N_{latency} = \frac{(N_p - N_{rt})k}{link_{capacity}} + Cmp_{delays}, \forall k \in N, \quad (6.49)$$

$$link_{capacity} = \sum_{i=1}^k \frac{Min_{rate}}{C_i}. \quad (6.50)$$

The performance of each link in such type of network remains same. However, it is noticed that for such type of network model, even if the achievable rate for some nodes is larger than the minimum achievable rate, still transmission is carried out at minimum rate. This decreases the performance of the network.

6.3.3.2 Variable Channel-Based Network

In this scenario, variable channel-based network model is considered. This chooses the transmission rate depending upon the amount of available transmission energy rather than operating at a similar rate. The power consumption for such type of network is more, but the performance is better than those operating at a similar rate. $N_{latency}$ for such type of network will vary from link to link, and will be computed as:

$$N_{latency} = \frac{\sum_{i=1}^k (N_{p,i} - N_{rt,i})}{link_{capacity}} + Cmp_{delays}, \forall k \in N, \quad (6.51)$$

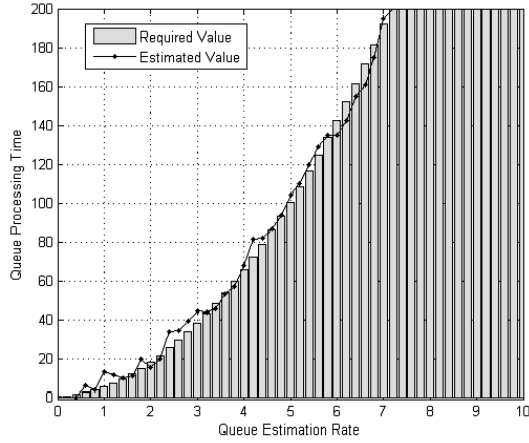
where

$$\begin{aligned} link_{capacity} &= \frac{\frac{1}{R_1} + \frac{1}{R_2} + \dots + \frac{1}{R_k}}{C_1 + C_2 + \dots + C_k}, \\ &= \sum_{i=1}^k \frac{1}{C_i}. \end{aligned} \quad (6.52)$$

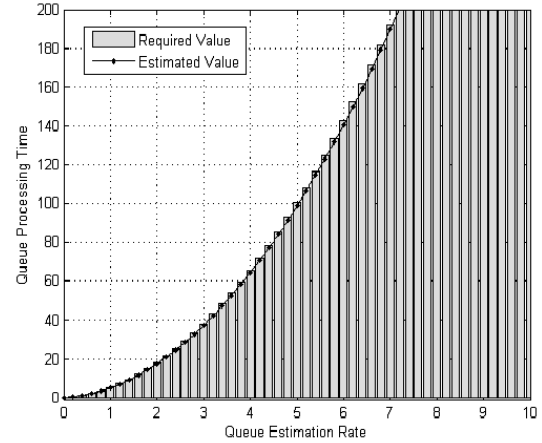
With the use of variable channels for transmission process, the power consumption of the network increases, but it can be optimized either by reducing the packet size, or by decreasing the number of retransmissions N_{rt} . This will allow, the node to perform to maximum capacity, but it may also decrease the fault tolerance of the network. Thus, energy always remains a constraint for the network when QoS is to be improved.

Table 6.2: Parameter Configurations For QoS Analysis

PARAMETER	VALUE	DESCRIPTION
Dimensions	1500X1500 sq. m.	Area for Node Placement
Number of Nodes	5,15,25	Node sets for evaluation
Simulation Time	1000 s	Total evaluation time
Traffic Type	CBR-VBR	Traffic over TCP
Number of Connections	4,10,14	Active Number of connections
Packet Size	1024 bytes	Average Packet Size
Mac Layer	802.11 b	IEEE Standard for wireless
Buffer Size	5000,10000,15000	Buffer Capacity of each Node
Propagation Radio Model	Two Ray Ground	Topological Propagation Model
Maximum Speed	30 m/s	Node mobility
Minimum Speed	10 m/s	Node mobility
Pause Time	2 s	Halt time for Node
Interval Time To send	2 p /s	Time to wait before sending
Number of Users	20	Active users across each Node
Transmission Range	250 m	Transmission Distance of Node
Node Mobility Range	0-2.5 m	Node Steps during movement
Max. Channel Capacity	8	Queue capacity of each channel
Initial Bandwidth	256 kbps	Initial Transmission rate

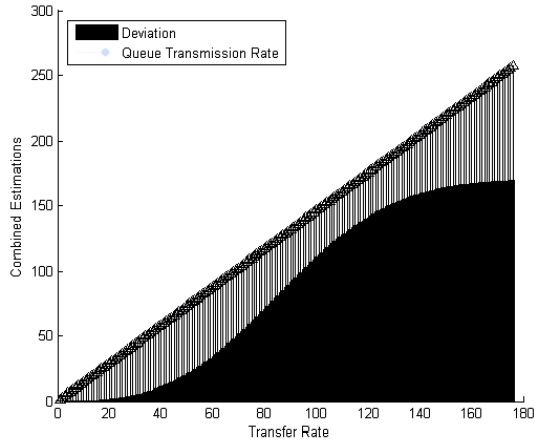


(a) Queue Estimation without Kalman Filter

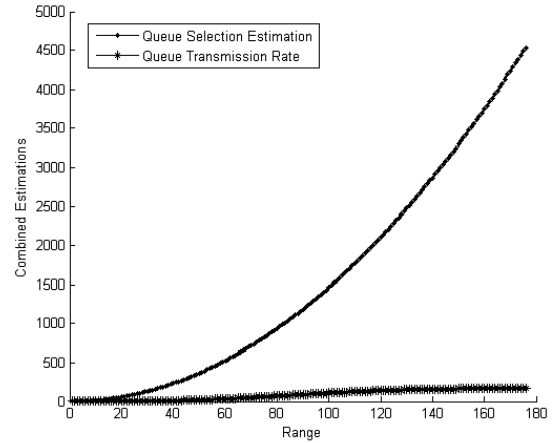


(b) Queue Estimation Using Quaternion Based Kalman Filter

Figure 6.3: Queue Estimation Comparison



(a) Deviation during Estimations

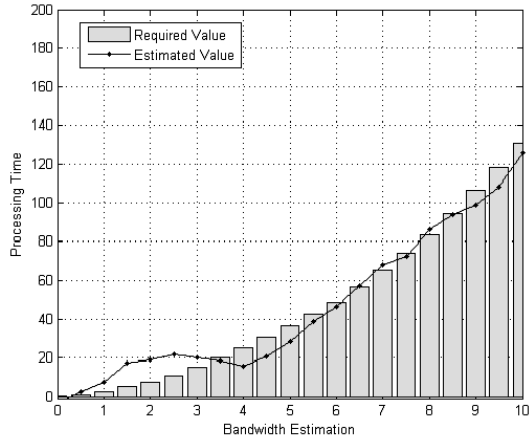


(b) Estimation Improvement during Queue Transmission

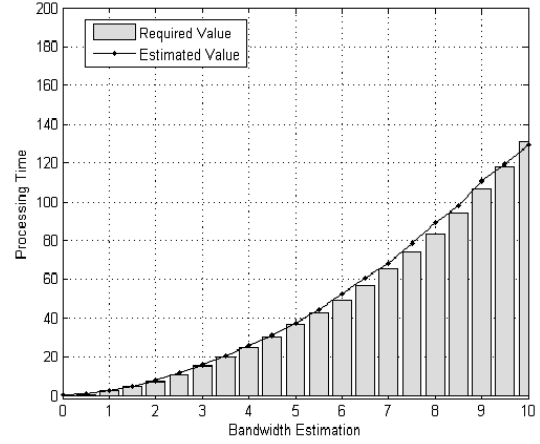
Figure 6.4: Statistical Analysis of Queue Estimation using Quaternion Kalman Filter

6.4 Estimation Analysis Over Ground Network

The simulation for proposed model was performed using *NS-2.35* simulator and the result analysis were evaluated using MATLAB. The configuration of the system model, as shown in Table 6.2, consists of a maximum of 25 nodes with 14 active numbers of connections. With packet size of 1024 bytes, a total of 50 simulation-runs are made to verify the results and analyze the behavior of the network model. During the first

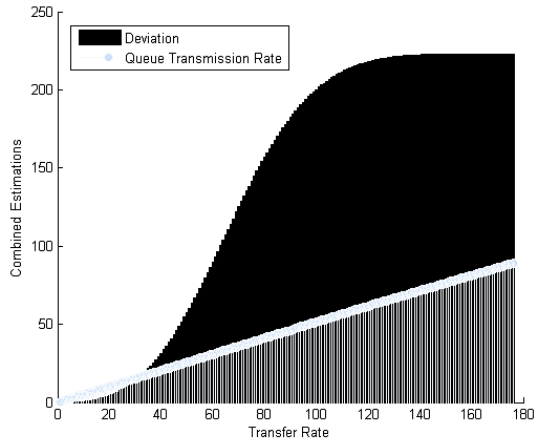


(a) Bandwidth Estimation without Kalman Filter

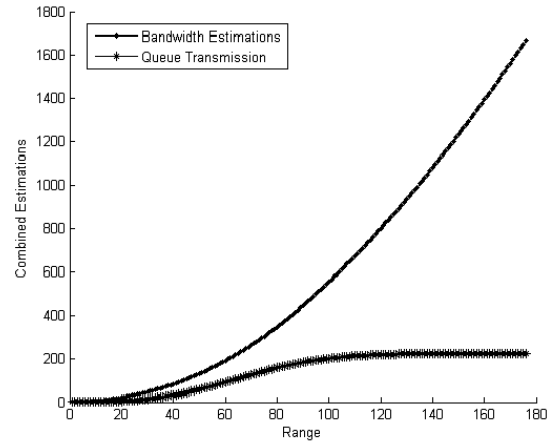


(b) Bandwidth Estimation Using Quaternion Based Kalman Filter

Figure 6.5: Bandwidth Estimation Comparison



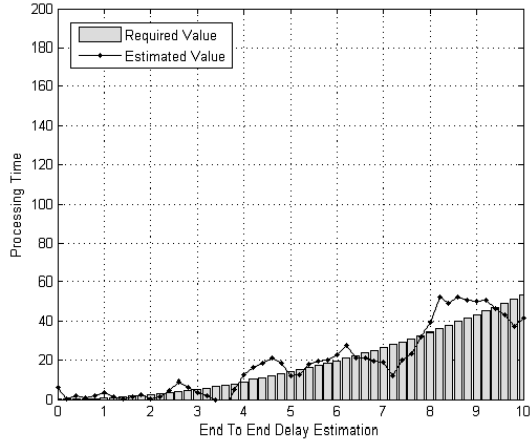
(a) Deviation during Estimations



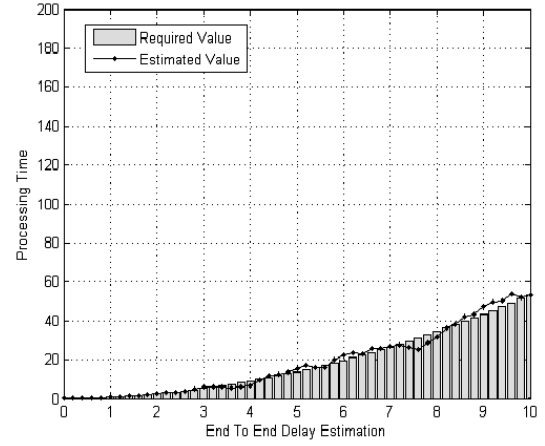
(b) Estimation Improvement during Queue Transmission

Figure 6.6: Statistical Analysis of Bandwidth Estimation using Quaternion Kalman Filter

phase of simulations, queue model was evaluated using MATLAB to check the estimation accuracy of Quaternion-based Kalman Filter. The results were evaluated for queue estimation, bandwidth estimation and end to end delay estimation over the ad hoc network. Figure 6.3a, Figure 6.3b and Figure 6.4 provide the graphical analysis of queue estimation. Kalman Filter gives more accurate results for queue selection. Similarly, evaluations were carried out for bandwidth and end to end delay as shown in Figure 6.5a to Figure 6.8. The results for estimations were realized and verified by transforming the

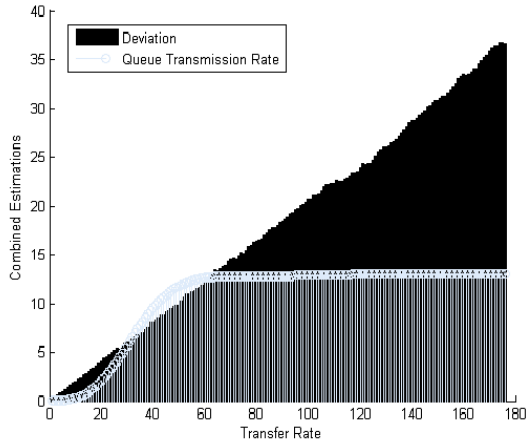


(a) Delay Estimation without Kalman Filter

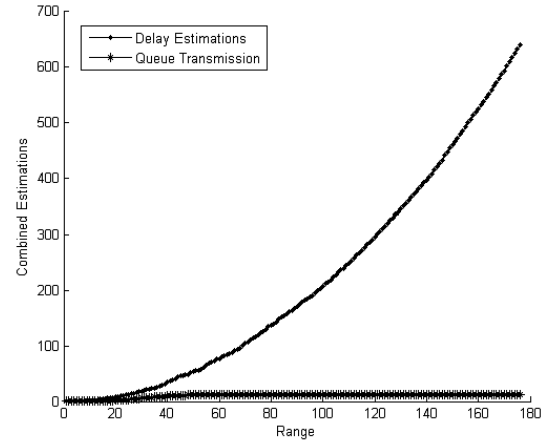


(b) Delay Estimation Using Quaternion Based Kalman Filter

Figure 6.7: Delay Estimation Comparison



(a) Deviation during Estimations



(b) Estimation Improvement during Queue Transmission

Figure 6.8: Statistical Analysis of End to End Delay Estimation using Kalman Filter

existing Kalman Filter [223] into the Quaternion Kalman-based estimator. Bandwidth and delay predictions are difficult to estimate as dynamic changes in the network directly affect these parameters. Despite of that, Quaternion-based Kalman Filter was able to estimate these values with less error correction.

6.5 UAV Assisted QoS Provisioning For Ground Ad Hoc Networks

In the extension to this scheduler, a similar model is presented for UAVs assisted ground ad hoc networks. This scheduler is further designed in two parts. The first part decides the appropriate allocation of UAVs to a particular area based on the demand patterns and traffic requirements. This allocation is done on the basis of the scheduling cost strategy. Each area is assigned a scheduling cost based on its traffic requirements. The second part utilizes this position accuracy to perform queue scheduling using a Satisfied Importance Analysis (SIA) approach. This approach utilizes a queue ranking table and a dominance table to form a final queue transmission strategy.

6.5.1 System Model

The network comprises of n number of UAVs available for relays over m number of ground nodes. The problem is to appropriately allocate UAVs to a particular set of ground nodes providing guaranteed QoS to end users. Let Q_1 , Q_2 , and Q_3 be the three quaternions defined for traffic state of area, UAV capacity, and network transmission state, respectively. Using the definition of quaternion [224], these quaternions are defined as:

$$Q_1 = a_1 T_u + a_2 \lambda_s + a_3 \gamma_s + A, \quad (6.53)$$

$$Q_2 = b_1 T_r + b_2 U_h + b_3 N_c + J, \quad (6.54)$$

$$Q_3 = c_1 T_q + \frac{c_2}{\mu_p} + c_3 S_q + M, \quad (6.55)$$

where a_1 , a_2 , a_3 , b_1 , b_2 , b_3 , c_1 , c_2 , and c_3 are the weights assigned to their respective metrics. T_u are the total users, λ_s is the service distribution, γ_s is the mean size of each PU, T_r is the transmission rate, U_h is the users handled by each UAV, N_c is the degree of connectivity, T_q is the total number of queues, $\frac{1}{\mu_p}$ is the packet arrival rate of each queue,

and S_q is the queue size. A , J , M are the scalars representing area, radio range of UAV, and memory available, respectively. Now, the final quaternion Q_f for the whole network is given by:

$$Q_f = d_1 Q_1 + d_2 Q_2 + d_3 Q_3 + C_p, \quad (6.56)$$

where d_1 , d_2 , d_3 are the weights following rules of hyper-imaginary numbers, and C_p defines the maximum capacity of the network. Now, the complete quaternion system S_q can be expressed as:

$$S_q = \begin{bmatrix} Q_1 & Q_2 & Q_3 \\ N_e & R^\delta & E_q \end{bmatrix}, \quad (6.57)$$

where δ is the path loss exponent, N_e is the network error, and E_q is the queuing error. For estimation, the network governing equation $\overline{Q_g}$ is calculated as:

$$\overline{Q_g} = \frac{1}{2 \mu_p} \Omega S_q Q_f. \quad (6.58)$$

Here, Ω is the vector product matrix [224] used for collision avoidance between UAVs. Now, the estimation equation for a node at location x in the area A is computed using Bayesian probabilistic model [250] as:

$$P(\overline{Q_g}(t)|f_t, f_{t-1}) = \frac{P(\epsilon_s|\overline{Q_g}(t), f_{t-1}) \times P(\overline{Q_g}(t)|f_{t-1})}{\int_A (\epsilon_s|\overline{Q_g}(t), f_{t-1}) dA}, \quad (6.59)$$

where ϵ_s is the system error, f_t and f_{t-1} are the estimator functions operating over bandwidth and delays at time t and $t - 1$. Now, updates for f over the link matrix M_L is computed as:

$$U_{(t)} = [\Pi[\overline{Q_g}(t) f \times] M_L]. \quad (6.60)$$

Now, P_{Q_g} over updates will be computed as:

$$P_{Q_g} = U_{(t)} C_m U_{(t)}^T + \epsilon_n, \quad (6.61)$$

where C_m is covariance matrix derived over system state, ϵ_n is the noise error. Therefore, Kalman gain K_{Q_g} will be given by

$$K_{Q_g} = C_m U_{(t)} P_{Q_g}, \quad (6.62)$$

and correction in cost function for a particular area is given by:

$$C_{Q_g} = K_{Q_g} \epsilon_n. \quad (6.63)$$

The proposed model operates in two parts. Firstly, it uses a scheduling cost function to find the requirement of each area. Then, it uses the calculated cost to allocate UAVs to appropriate areas, thus, solving the issue of UAVs positioning. Secondly, it performs queue scheduling based on the importance level and priority using SIA [251] [252].

6.5.2 Scheduling Cost Method For UAVs Allocation

The proposed model uses the correction C_{Q_g} computed in the system model to adjust its scheduling cost S_{cf} . This scheduling cost depends upon the queue size S_q , packet arrival rate $\frac{1}{\mu_p}$, intermediate nodes/relays I_r , unhandled ground node distribution λ_u , and weight of each queue W . For each of the high priority packet, a negative weight of -1 is allocated to the queue. Queue with more number of priority packets posses more negative weight. Scheduling cost helps in assigning UAVs to appropriate areas with a high density of ground nodes, and with higher transmission rate requirements for efficient QoS. The scheduling cost function S_{cf} for each area division Div is given by:

$$S_{cf}^{Div} = \min \left(\frac{e^{\frac{1}{\mu_p}} S_q I_r}{\lambda_u^{Div} |W|} + C_{Q_g} \right). \quad (6.64)$$

Higher the cost function, more are the delays, and more is the requirement of UAVs. Area A is divided into sub areas comprising of nodes with direct connection to one another. Further, UAVs maneuvering computes S_{cf} for each area, and use it to form a binary search tree (BST). All UAVs are not in direct contact with every subdivided area, this will cause variation in number of relays required to provide a connectivity. Thus,

scheduling cost will be different for all UAVs. The steps involved in formation of final set of area to be maneuvered by a UAV are given in Algorithm 6.2. This algorithm allows

Algorithm 6.2 Scheduling Cost Based UAVs Allocation

Require: n =UAVs, K =Area subdivisions

Ensure: $n \geq K$

```

i=1
while (i ≤ n) do
  compute  $S_{cf}^K$ 
  form  $BST_i \leftarrow S_{cf}^K$ 
  share  $S_{cf}^K$ 
  if (BST == received) then
    p = 1
    while (p ≤ K) do
      if ( $\forall x \neq i, S_{cf}^p(x) < S_{cf}^p(i)$ ) then
        delete  $S_{cf}^p(i)$ 
      else if ( $\forall x \neq i, S_{cf}^p(x) == S_{cf}^p(i)$ ) then
        if ( $\forall x \neq i, I_r^p(x) < R_e^p(i)$ ) then
          delete  $S_{cf}^p(i)$ 
        else
          retain  $S_{cf}^p(i)$ 
        end if
      end if
    else
      continue
    end if
    p=p+1
  end while
else
  send request
end if
i=i+1
end while
proceed to threshold check

```

initial allocation of each UAV to a particular area depending upon the value of scheduling cost. If all the conditions of threshold are met, the estimator queue scheduler is used for queue transmission. There might be situations when reallocation or rearrangement of topology is required. This is performed on the basis of threshold comparison for area scheduling cost function as shown in Algorithm 6.3. This algorithm allocates and checks the minimum threshold satisfying conditions for each UAV. Reallocation or deploying of more number of UAVs is the major deciding factor of this algorithm. The procedure accounts for minimum number of iterations required to lower the scheduling cost of each area.

Algorithm 6.3 Threshold Check and Reallocation

Require: $S_{th} = 1.7 \times 10^{-2}$, n , $reallocation = false$

Ensure: n allocated to K

```
i = 1
while (i ≤ n) do
   $A_i[] = \{y: y \text{ is areas allocated to } i^{th} \text{ UAV}\}$  &&  $count(A_i[]) \leq K$ 
  j = 1
  while (j ≤ count( $A_i[]$ )) do
    recompute  $S_{cf}^j(i)$ 
    if ( $S_{cf}^j(i) \geq S_{th}$ ) then
      mark  $A_i[j]$ 
    else
      continue
    end if
    j = j+1
  end while
  i = i+1
end while
if ( $UAV(A) == NULL$ ) then
  either reallocate or assign new UAVs
  if ( $reallocation == true$ ) then
    Check  $uav\_capacity \leftarrow Q_2$ 
    assign area to UAVs with  $uav\_capacity \geq avg(Q_2)$ 
  else
    deploy
  end if
end if
iterate till  $S_{cf}^j \leq S_{th}, \forall j \in A$ 
```

6.5.3 Queue Scheduling

Admission control procedures and the congestion control approaches can provide a possible solution for handling delays in a network operating under dynamic conditions. However, in networks comprising of variedly configured nodes, these approaches cannot be readily applied because of difference in configurations of each node. For such network, a base level modifications are required. Thus, defining a new queue scheduling approach provide a possible solution equivalent to both congestion control as well as admission control procedures. The proposed queue scheduling is derived using the Satisfied Importance Value (SIA) that operates over the satisfaction value V_s , and the standardized importance rating V_{is} . These two components help in defining the governing rules for the selection of appropriate queue in order to provide QoS to end users in UAVs assisted ad hoc networks. For the initial setup, a rating table is prepared for each of the queues based on the metrics shown in Table 6.3. Using the ratings defined in Table 6.3, dominance of each metric is computed on a scale of 0 to 10. Higher the value, more is the dominance. The dominance is computed based on the network type and average rating

Table 6.3: Queue Rating (X*:-Variable Entities)

Type	Entities	Weight	Rating (R_T)
High Priority Packets	X*	-3	-3 X
Regular Priority Packets	X*	-2	-2 X
Normal Packets	X*	-1	-1 X
Ground Relays	X*	-2	-2 X
Slots	8	-1	-8
Aerial Relays	X*	-1	-1 X
Time to Live (TTL)	X*	-1	-1 X
Round Trip Time (RTT)	X*	-2	-2 X
Bandwidth Requirement	X*	-3	-3 X

Table 6.4: Parameter Dominance

Type	D_R	$L_S = D_R \times R_T $
High Priority Packets	9	$9 \times -3 X $
Regular Priority Packets	8	$8 \times -2 X $
Normal Packets	7	$7 \times -1 X $
Ground Relays	8	$8 \times -2 X $
Slots	5	$5 \times -8 $
Aerial Relays	8	$8 \times -1 X $
Time to Live (TTL)	6	$6 \times -1 X $
Round Trip Time (RTT)	7	$7 \times -2 X $
Bandwidth Requirement	9	$9 \times -3 X $

Algorithm 6.4 SIA Based Queue Scheduling

Require: Q=number of Queues, V_s , V_{is} **Ensure:** n allocated to K, Algorithm 6.2 and Algorithm 6.3 holds

```

i = 1
retries = 0
queues[Q] = sort( $V_s$ , descend)
queueis[Q] = sort( $V_{is}$ , descend)
while (i ≤ Q) do
    transmission_queue[ ] = NULL
    pending_queue[ ] = NULL
    if (queues[i] > 0 && queueis[i] > 0) then
        transmission_queue[ ] = i, appropriate
    else if (queues[i] > 0 && queueis[i] < 0) then
        transmission_queue[ ] = i, requires reanalysis

    else
        pending_queue[ ] = i
    end if
    i=i+1
end while
if (pending_queue[ ] != NULL) then
    Call Algorithm 6.3, 6.4
    if (pending_queue[ ] != NULL) then
        Call Algorithm 6.2, 6.3, 6.4
    else
        continue
    end if
else
    continue
end if

```

of each parameter considered for scheduling. The degree of relevance D_R , and the level of satisfaction L_S are shown in Table 6.4. This table helps in defining the rules for SIA to select a queue optimally with guaranteed QoS to end users. Parameters V_s and V_{is} for each of the metrics are computed using this dominance using [251] [252] as:

$$V_s = \frac{L_T - \text{average of initial } L_S}{\text{standard deviation of initial } L_S}, \quad (6.65)$$

and

$$V_{is} = \frac{D_T - \text{average of initial } D_R}{\text{standard deviation of initial } D_R}, \quad (6.66)$$

where

$$L_T = \frac{Sum(L_S)}{Number\ of\ Queues}, \quad (6.67)$$

and

$$D_T = \frac{Sum(D_R)}{Number\ of\ Queues}. \quad (6.68)$$

Further, V_s and V_{is} are governed by set of rules for selection of appropriate queue. This selection process guarantees QoS to end users with relatively less consumption of bandwidth and less delays. The steps for selection of queue in order of relative priority for efficient scheduling is provided in Algorithm 6.4.

6.5.4 Theoretical Analysis

Algorithm 6.2 provides the solution to UAVs allocation based on the traffic demand in order to provide a base for efficient queue scheduling. The algorithm takes into account the n number of UAVs available to maneuver over an area with K subdivisions. Further, it computes the cost function S_{cf}^K , and constructs a BST of area subdivisions for each of the UAVs. For a single UAV, the complexity of deciding the location in the worst case will be $O(K \log K)$. For the overall network, this complexity increases to $O(n K \log K)$. However, in the best case, this complexity becomes $O(n (\log K)^2)$. For reallocation, the complexity in the worst case is $O(n)$, and for best case it is $O(\log n)$. In the best case, number of UAVs to be reallocated will certainly be less than n , however, in worst case, all the UAVs will be reallocated that will increase the complexity of the network. For scheduling, Algorithm 6.4 iterates for Q number of queues. In the best case, the complexity will be $O(Q)$ as all the queues will be directly transmitted without further computation. However, in the worst case, each queue might get process but remains in the pending queue list because of non-satisfaction of conditions marked over V_s and V_{is} . Thus, in the worst case, without considering the subroutine calls to Algorithm 6.2 and Algorithm 6.3, the complexity of queue scheduling in UAVs assisted networks becomes $O(Q^2)$.

Table 6.5: Parameter Configurations For UAV Assisted QoS Provisioning

Parameter	Value	Description
A	10000x10000 sq. m.	Simulation Area
T_u	30-100	ground nodes
n	1-15	UAVs
M	512 Mb	Memory
P_s	1024 bytes	Packet Size
λ_s	30	Packet Distribution
T_r	256 kbps	Initial Transmission Rate
T_q	3	Total Queues
S_q	5000-15000 bytes	Queue size
C_p	54 Mbps	Maximum Capacity
$\frac{\lambda_p}{\mu_p}$	256 kbps	Offered Rate
δ	4	Path loss Exponent
R	0.8 km	Radio Range
S_{th}	1.7×10^{-2}	Scheduling Cost Threshold
Simulation Time	1000 sec	Total Analysis Time
Simulation Runs	100	Number of Analysis

6.6 Performance Evaluation

The proposed scheduling cost-based UAV positioning was analyzed for an environment comprising of ground nodes moving in an area of 10000×10000 sq.m. in set of 4 clusters. Each cluster comprised of 30 to 100 mobile nodes acting as network users T_u . Number of UAVs used for relaying varied from 1 to 15. Other configurations for simulation study are provided in Table 6.5. Initially, the proposed approach was analyzed for positional accuracy and processing time taken to allocate UAVs to a particular area. With more demands and less UAVs, it becomes complex to attain an optimal solution. Thus, more number of UAVs might be required in that case. Results show that with optimal number of UAVs, high position accuracy is attained in mapping ground nodes with high QoS experience to end users. Figure 6.9 shows that with the increase in number of ground users and relatively lesser number of UAVs, position accuracy suffers; and converges towards a higher value if the number of UAVs are increased. Position accuracy is reflected in terms of processing time also; this is the time taken by the network to stabilize with all UAVs at correct location. The plot for processing time with varying number of ground nodes and UAVs is shown in Figure 6.10. Four existing QoS approaches, namely, Proportional Fairness (PF) [158], McLaughlin [155], $M|M+|1$ [164], and Weighted Fairness Queue (WFQ) [157] were used for comparison with the proposed approach. UAVs provide a relay support to existing ground ad hoc networks with the provisioning of efficient data transmission and improved quality of service (QoS). The

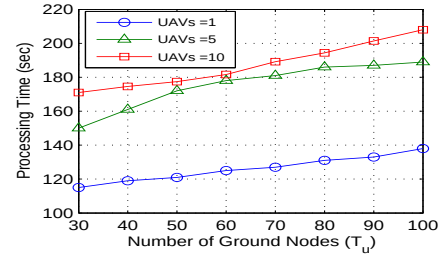
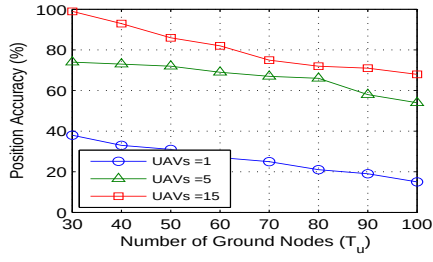


Figure 6.9: Position Accuracy vs. Ground Nodes **Figure 6.10:** Processing Time vs. Ground Nodes

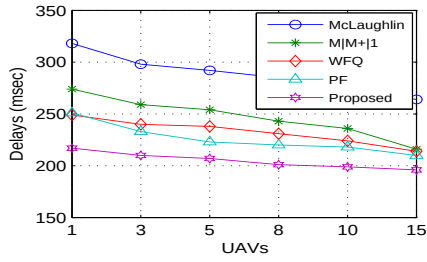


Figure 6.11: Delays vs. UAVs

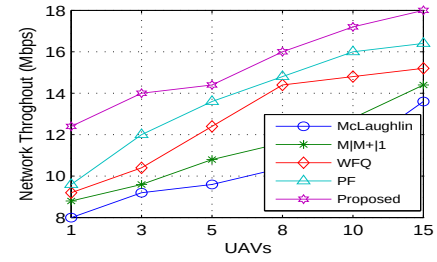


Figure 6.12: Network Throughput vs. UAVs

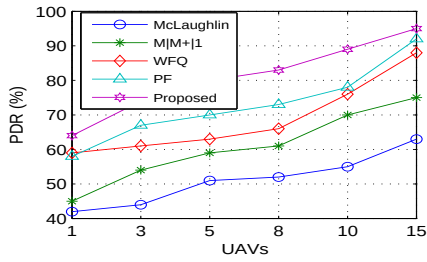


Figure 6.13: Packet Delivery Ratio (PDR) vs. UAVs

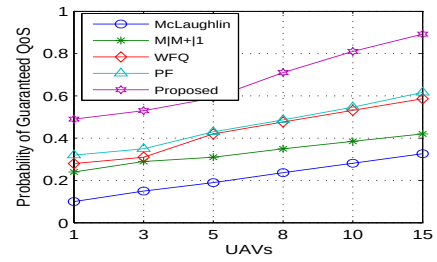


Figure 6.14: Probability of Guaranteed QoS vs. UAVs

proposed scheduling cost based UAV positing was analyzed for delays that are liable to occur in a network with temporary connections. Results show that the proposed model caused 45.5% less delays than the McLaughlin model, 23.25% less delays than the $M|M+|1$, 16% less delays than the WFQ, and 12.5% less delays than the PF model. The plot for delay comparison with varying number of UAVs is presented in Figure 6.11. With lesser delays, UAVs are capable of providing continuous connectivity between the end users on the ground. Facility of bandwidth and delay estimation, defined in the system model, provides higher data rate with improved QoS. The proposed model was analyzed for network throughput in comparison with the existing approaches. Results show that the network throughput for the proposed model was recorded 36.9%, 24.3%, 19.6%, and 12.3% higher than the McLaughlin model, $M|M+|1$, WFQ, and PF model, respectively. The

plot for network throughput comparison is shown in Figure 6.12. Further, the network formation was analyzed for packet delivery ratio (PDR) between the end users. Results in Figure 6.13 show that the proposed model offered PDR 36.6%, 25.8%, 17.2%, and 11% higher than the McLaughlin model, $M|M+1$, WFQ, and PF model, respectively. A network that guarantees QoS to end users is more reliable and efficient. QoS to end users is marked by lower delays, and high data rates. The scheduling cost approach allows accurate positioning of UAVs over the appropriate areas providing high data rates. A probability of guaranteed QoS was computed over the ground network maneuvered by UAVs that analyzed the users served with higher data rates throughout the transmission. Proposed model provided better QoS to end users with efficient allocation of UAVs to appropriate areas. Figure 6.14 shows that the proposed model guaranteed 66%, 50%, 35%, and 31% better QoS than the McLaughlin model, $M|M+1$, WFQ, and PF model, respectively. With lesser delays, higher data rate, and higher guaranteed QoS to end users, proposed model caused 66.29%, 42.29%, 38.82%, and 29.31% less errors in scheduling queue over the UAVs assisted ad hoc formations than the McLaughlin model, $M|M+1$, WFQ, and PF model, respectively.

6.7 Conclusion

Estimation-based queue scheduling proposed in this chapter is capable of handling large traffic and ensures QoS to end users. The transmission schema was integrated with Quaternion-based Kalman Filter that estimated the queue selection, performed traffic prediction, and delay estimations with higher accuracy. The proposed model offered relatively high data transfer rate with delays controlled within a certain range. The accuracy, with which prediction is performed, allows ad hoc networks to be used under crucial circumstances. Further, a UAV assisted quality of service provisioning is presented. This approach uses Satisfied Importance Analysis (SIA) based queue scheduling approach in combination with the UAVs positioning to guarantee QoS to end users. Less delays and high probability of guaranteed QoS was recorded for the proposed model in comparison with the existing scheduling approaches.

Chapter 7

Stabilization and Topological Broadcasting Over Networked UAVs^{†‡}

This chapter presents the novel strategies for stabilization against failures over networked UAVs. It also provides a solution to topological broadcasting issues in collaborative Ground-Flying ad hoc networks.

For stabilization against failures over networked UAVs, a novel self-healing neural model is presented. This model utilizes the concept of dummy neurons, and back propagation to form a self-stabilizing network. Self-healing cost function is used to maintain the stabilization. The neural model is tested for the mean square error and gradient using Levenberg-Marquardt (LM), BFGS Quasi-Newton (BFG), and Resilient Backpropagation (RP) algorithms. The proposed approach is capable of handling uncertain network failures efficiently.

In the next part, broadcast storm issue over the coordinated network formation between the ground ad hoc and the aerial ad hoc units is considered. A topological approach is proposed based on the parameter sensibility which helps in the formation of logical proximity graphs. These proximity graphs are then used to identify the efficient links between the source and destination that can provide reliable communications without causing any redundancy during broadcasting.

[†]Vishal Sharma, Rajesh Kumar and Prashant Singh Rana, “A Self-Healing Neural Model For Stabilization Against Network Failures Over Networked UAVs”, IEEE Communication Letters- SCIE Indexed, 19(11), 2015, pp. 2013-2016, [I.F. 1.268].

[‡]Vishal Sharma, Rajesh Kumar and Neeraj Kumar Rathore, “Topological Broadcasting Using Parameter Sensitivity-Based Logical Proximity Graphs in Coordinated Ground-Flying Ad Hoc Networks”, Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications (JoWUA)-Scopus Indexed, 6(3), 2015, pp. 54-72.

7.1 Self-Healing Model For Stabilization Against Failures Over Networked UAVs

The proposed model operates towards the formation of network with maximum connectivity, enhanced stability and high convergence so as to provide continuous connectivity with enough transmission rate during uncertain failure of network nodes. For the network, set $U = \{U_1, U_2 \dots, U_n\}$ comprising of n number of UAV that maneuvered over a predefined area. Initially, each UAV posses similar configurations and resources. Set X maintains a count for number of UAV failed during network operations such that $X \subset U$. Ideal state, $X = \phi$, is difficult to attain in case of failures. The model needs to learn, heal and provide a decision for continuity of network operations. This is done by the formation of a healing function h_f . This healing function is the controlling end of all the network operations in the proposed neural model. For a stabilized network formation, h_f is derived for each UAV as well as for the overall network. For single UAV operation, h_f is subjected to control parameters as input, whereas for overall network, h_f is subjected to UAVs as input. For the considered model, network state at time t w.r.t. U and X is given as a function of time as:

$$S_t = f(U(t), X(t)), \quad (7.1)$$

and the initial state is given as:

$$S_o = f(U_o, X_o), \quad X_o = \phi, \quad (7.2)$$

Now, at time $t + 1$, network state is given as a function of previous state and healing function such that:

$$S_{t+1} = S_o + h_f(f(U(t), X(t))) \quad (7.3)$$

$$= S_o + h_f(S_t). \quad (7.4)$$

Healing function h_f will either increase or decrease with an exponential deviation of weight defined over it. Thus, this objective function will be computed over both deviation in total dependencies of U , X as well as over partial dependencies defined w.r.t. active connections and weights. The healing function at time t is computed as:

$$h_f^t = \min \left(\sum_{i=1}^n e^{\sqrt{\frac{1}{c_T} \sum_i^{c_T} (W_i^t - W_m)^2}} + \sqrt{\frac{1}{n^t} \sum_{j=1}^{n^t} (cx^t)^2} \right), \quad (7.5)$$

where c_T is the total connections for n UAVs, W defines the weight, W_m is the average weight, n^t being a number of UAV active at time t , and cx^t is the number of faulty connections at time t . Equation (7.4) represents the state equation for the overall network, which is a functional dependency over objective function h_f . The whole network is governed by certain constraints which are to be satisfied in order to realize a stabilized and a fully functional network.

- **Constraint 1:** Let R_{min}^{TH} be the minimum transfer rate required to maintain a stabilized connectivity between the network nodes. Now, at time t ,

$$R_{min}^t \geq R_{min}^{TH}. \quad (7.6)$$

- **Constraint 2:** As given in equation (7.4), $h_f \rightarrow f(U, X)$ is a bivariate objective function. Thus, at time t ,

$$h_f(f(\alpha, \gamma)) \rightarrow \min(S_t, S_o). \quad (7.7)$$

Here, α and γ are the boundaries of the system where α denotes the instance for which function is to be evaluated and γ is the instance of previous state.

- **Constraint 3:** A system more close to its initial state is categorized as a better system. Thus, for a sustainable network, with X approaching maxima M_X at time t ,

$$\lim_{X \rightarrow M_X} h_f(f(U(t), X(t))) \leq h_f^{min}, \quad (7.8)$$

where h_f^{min} is the healing function attained over the minimum number of active

UAVs (A_{min}) required for stabilized run of the overall network. Alternatively, equation (7.8) can be summarized as:

$$M_X \leq (n - A_{min}). \quad (7.9)$$

- **Constraint 4:** Constraint 3 subjects that the deviation of the overall network at any instance from the initial state should be minimum. Thus, it is important to analyze the deviation of network connectivity from the initial state of required configurations. Now, for an enhanced connectivity, let y % of UAVs always be active with C_{min} number of active connections, such that deviation σ^{TH} for k number of iterations is computed as,

$$\sigma^{TH} = \sqrt{\frac{1}{k} \sum_{i=1}^k \left(\frac{y \times n}{100} - C_{min} \right)^2}. \quad (7.10)$$

For, 50% connectivity (taken as threshold in the proposed model), equation (7.10) will be given as:

$$\sigma^{TH} = \frac{1}{2} \sqrt{\frac{1}{k} \sum_{i=1}^k (n - 2 C_{min})^2}. \quad (7.11)$$

Now, the deviation of the system σ for a given state at time t will be given by:

$$\sigma^{St} = \sqrt{\frac{1}{k} \sum_{i=1}^k (C_{act}^t - C_{min})^2}, \quad (7.12)$$

where C_{act}^t are the active connections at time t . For a stabilized and continuous network formation,

$$\sigma^{St} \leq \sigma^{TH}. \quad (7.13)$$

7.1.1 Proposed Self-Healing Neural Model

The proposed model operates over number of active connections mandatory to be maintained for continuous transmission. The model when subjected to a network fault, identi-

fies the connection loss, regulates its learning and feedback rate, and finally converges to heal the overall network. This healing is controlled by the healing function governed by the constraints explained in Section 7.1. A subjective requirement of the proposed model is presented in Figure 7.1. The model comprises of n number of UAV initially configured

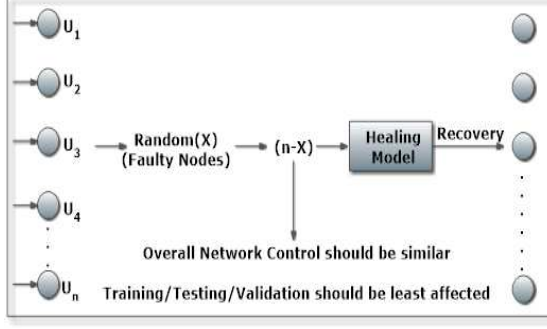


Figure 7.1: Operational View of Proposed Model

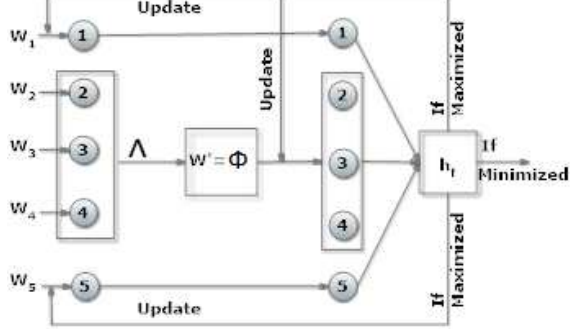


Figure 7.2: Operational View of Healing Phase

to cooperate and attain a task collectively. Further, the system is affected by randomized failures denoted by X . The remaining $(n - X)$ UAVs are subjected to healing model which takes a decision if the system can be used with the loss of X nodes or requires remodelling or retraction of the complete network. Further, the model takes into account the healing range of the whole network, and if applicable, heals by recovering the whole network by formation of dummy-neurons. This approach of decision, modification and recovery is termed as “Decide and Recover” approach. The proposed approach aims at balancing the number of connections required for continuous network operations. This implies that the model aims at maintaining C_{min} above the prescribed threshold value with transfer rate R_{min} greater than the threshold value. Thus, the proposed strategy aims at balancing the neural structure without affecting its learning and feedback rate. The model initializes the state and marks it with values defining the total number of connections between the input and the output state. In case of multiple layers, two layers are taken into account at a time. Therefore, this approach can be treated as bivariate learning and healing model. Further, the previous state acts as an input to the current state with new healing function derived on the basis of active connections at the current time instance. If the active connections are way beyond the threshold value, the network operations continue and no healing is performed. This saves computational iterations as the network can be continued even with lost nodes. However, if the number of connections decreases

below the required threshold value, then the healing function is subjected to converging iterations which decreases the considered objective function h_f until it converges to a safe state. With each iteration, a dummy input is created to recover the overall network. When the dummy inputs for a dead neuron are equal to the number of inputs for that neuron in the initial state, the neuron is introduced in the state. This neuron acts as a dummy neuron. When the sum of dummy neuron and active neurons become equal to the total number of neurons considered initially, iterations stop and model is analyzed for its convergence. If the number of connections available at this moment are enough to sustain the network, the model is healed, else the network recomputes its healing function and iterates further to stabilize as shown in Figure 7.3. Also, it can be analyzed that during

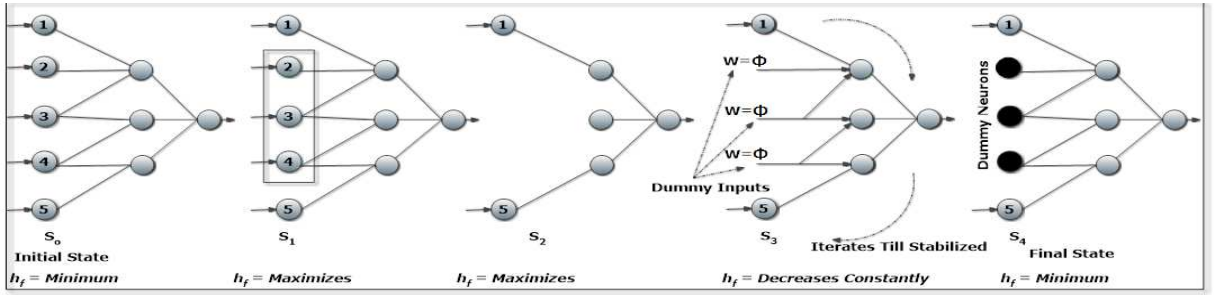


Figure 7.3: State Wise Functionality of Proposed Self-Healing Neural Model

state S_1 , three neurons fail, thus increases the value for h_f . This increase in objective function continues till state S_2 where the whole setup is left with only two neurons in the input layer. This maximizes the h_f and hinders the network operations. After this, healing model comes into play and iterates the whole neural setup by decreasing the h_f continuously till it converges to a minima. This convergence is carried by assigning values to dummy neurons one at a time. Finally, a state with dummy neurons with similar number of connections as that of the initial state is obtained which is least affected in terms of its learning and feedback. However, the weight at this state for certain neurons might remain ϕ . Therefore, the proposed model should not be preferred for networks involving task assignment with similar distribution over each node. However, networks comprising of nodes with heavy task handling capabilities, the proposed self-healing model can prove effective to a large extent. The internal working of the healing phase of the proposed model is shown in Figure 7.2. Until the whole network is recovered, dummy connections are not made active. Also, dummy will be updated only if σ^{TH} holds.

7.1.2 Performance Evaluation

In this section, the performance analysis of the proposed self-healing approach for co-operative UAVs is presented. The analyses were traced over both benchmark training algorithms as well as over individual operations of the proposed approach. During simulations, a three layer neural, as discussed in Section 7.1.1, was formed with a number of neurons equal to the number of UAV, and Poisson distribution was used to generate the fault test cases. For events that are uncertain and dynamic, Poisson distribution accounts for better computation of probabilistic data as compared to other distributions. The number of inputs for each neuron were 3 and 4. However, these inputs can vary depending upon the number of network parameters. Various other parameters configured to analyze the performance of the self-healing approach are given in Table 7.1. Analyses

Table 7.1: Parameter Configurations For Self-Healing Model

Parameters	Value
UAVs	10-25
Neurons (Input Layer)	10-25
Neurons (Hidden Layer)	$\frac{\text{Input layer}}{3}$
Neurons (Output Layer)	1
Inputs	3,4
Faults	Poisson Distributed
Average Epochs	2-15
Poisson λ (mean)	No. of Inputs=3,4
Poisson k	No. of Failures=X
Initial Learning Rate	0.05
Gross Learning Increment	0.5
Overfitting	Handled by retraining
Training Dataset	http://bit.ly/selfhealingnn
Maximum Iterations	1000
Simulation Runs	1000

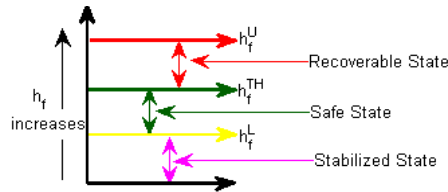


Figure 7.4: Healing Function h_f Range And State Categorization

were traced for variation in self-healing function h_f over different scenarios. The whole model was subjected to four state analysis diagram, where h_f^U denotes the upper range above which the system crashes, h_f^L denotes lower range below which the system is in stabilize state, and h_f^{TH} denotes threshold state below which system is in safe state, as shown in Figure 7.4. Table 7.2 presents the value of h_f traced for 10 UAVs with variable

number of faults over 3 and 4 inputs. It also presents the required dummy connections to attain stabilize and safe state. Further, it presents the variation in the value of h_f attained during stabilized and safe state for the considered model. The recovery of the network with variation in the number of inputs does not get much affected. However, with an increase in the number of inputs, more iterations will be required to overcome the failures. Further, with the variation in the number of inputs, the number of dummy neurons required to stabilize the network also increases. However, this increase is negligible in terms of the stabilization attained against the failures. From Table 7.2, it can

Table 7.2: Self-Healing Model Analysis For 10 UAVs

UAVs	Inputs	faults	h_f^{Fault}	h_f^{safe}	$h_f^{stabilize}$	Dummy Required (Safe State)	Dummy Required (Stabilize State)	h_f^{TH}	h_f^L	h_f^U
10	3	3	2.8926	2.0771	1.5923	3	6	2.2186	1.6924	4.4372
		4	3.7011	2.4716	1.6756	4	8	2.6534	2.0032	4.9432
		5	4.5488	2.7394	1.9639	7	10	2.7953	2.0685	5.5906
		6	5.2452	2.5517	2.013	8	12	2.7526	2.0398	5.5052
		7	5.5869	2.8867	2.0766	10	13	2.948	2.1722	5.896
10	4	3	4.1346	3.036	2.258	4	7	2.5686	2.0484	4.8897
		4	4.9431	3.188	2.3868	5	9	2.8216	2.3592	5.3957
		5	5.7908	3.3404	2.518	8	13	3.948	2.4245	6.0431
		6	6.4872	3.4921	2.6488	11	15	3.1026	2.3958	5.9577
		7	6.8289	3.6439	2.778	13	17	3.298	2.5282	6.3485

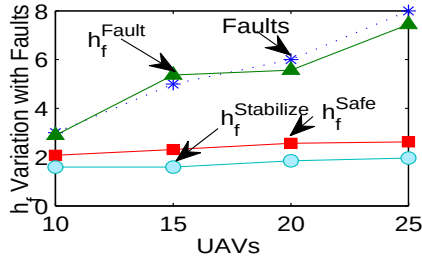


Figure 7.5: Variation of h_f over 30% Faults

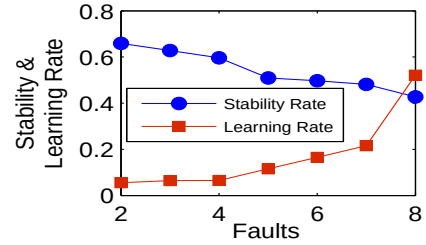


Figure 7.6: Stabilization Rate and Learning Rate Analysis over Variable Faults

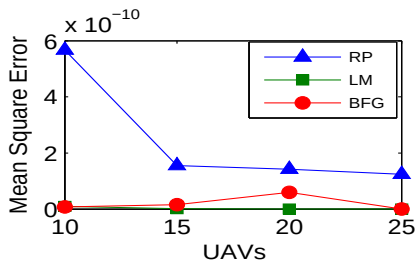


Figure 7.7: Mean Square Error over Benchmark Training Algorithms

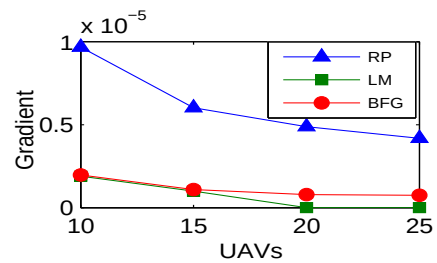


Figure 7.8: Gradient over Benchmark Training Algorithms

be analyzed that with the increase in number of faults, number of dummy connections

requirement increases. Also, h_f increases with increase in number of faults. For continuation in the network operations at time t , $h_f^t \leq h_f^{TH}$, and for stabilization, $h_f^t \leq h_f^L$. 30% node failure was introduced over a network with variable number of UAV. Comparative analysis of variation in h_f over this failure rate is shown in Figure 7.5. With the same failure rate over the increasing number of UAV, h_f increases. But this increase is lower than the threshold defined for the overall network. Thus, the model performs well under substituted failure conditions. Analysis proved that the proposed approach makes network highly stable in case of occurrence of errors. The plot for stability and learning rate with variation in network faults over 10 UAVs is presented in Figure 7.6. With increase in number of faults, networks learning rate increases to cope up with its performance. Lower the number of faults, more is the stabilization. In order to validate the proposed model, its performance was analyzed for the mean square error and gradient over benchmark training algorithms, namely, Levenberg-Marquardt(LM), BFGS Quasi-Newton (BFG), and Resilient Backpropagation (RP) using UAVs-dataset downloadable from- <http://bit.ly/selfhealingnn>. The dataset provides initial maneuvering UAVs, failed UAVs, and detected UAVs in both real and normalized values as input and output to the proposed model for training. Figure 7.7 and Figure 7.8, showed that LM outperform as compared to other training algorithms as its gradient converged gradually with decrease in mean square error. Also, it took a lower number of iterations to converge than the other benchmark training algorithms. Thus, it is found that the proposed model is capable of stabilizing the faulty networks efficiently. Issues like overfitting against wrong detection of failed UAVs was sorted using retraining. However, further novel approaches can be developed for handing overfitting to reduce the learning time of network.

7.2 Topological Broadcasting Over UAV Networks

Aerial nodes operating in continuous connection with the ground nodes have an added advantage of complete availability of line of sight in most of the cases. This line of sight provides strong connectivity between the ground and the aerial nodes. However, there is a disadvantage that same nodes can have line of sight availability to multiple aerial nodes.

Thus, if a routing table is constructed for data transmission, and upon the broadcasting of packets, it is liable that data gets travelled to same nodes in replication following multiple paths. A suitable approach is required that can keep check on this broadcasting and can ensure that data is not replicated as this will prevent overconsumption of network resources and will ultimately increase the network lifetime and performance. Thus, an efficient approach is required to form a topological arrangement of nodes in both the networks such that lower overheads due to data replication, least buffer overflows, and continuous streaming are obtained between these coordinated networks. Using this ideology, parameter sensitivity based logical proximity graphs are used to form an efficient topological arrangement that eradicates the occurrence of redundancy over the whole network. The key issues resolved by the proposed work are:

- Solution to broadcast storm in coordinated ad hoc network formed between aerial and ground nodes.
- Check on network partitioning.
- Continuous connectivity with improved fault tolerance of the network.

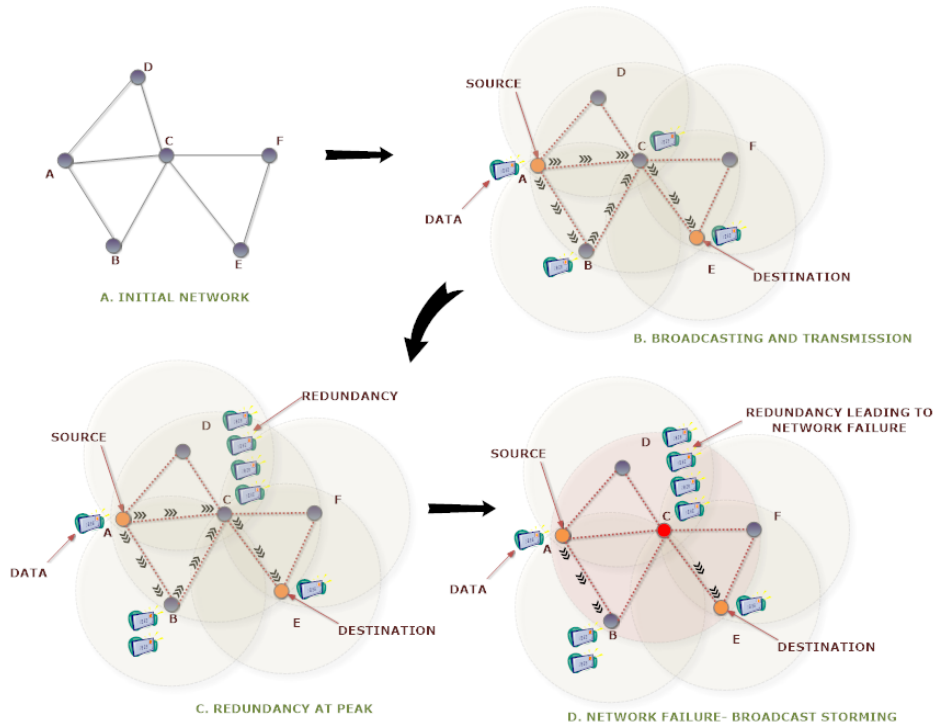


Figure 7.9: Broadcast Storm Issue

7.2.1 Proposed Solution To Broadcast Storm

A network formulated using an existing topological arrangement between the nodes of two different ad hoc units lead to problem of broadcast storming. One of the reasons of such problem with these networks is the availability of multiple data channels that can be used for reliable transmission. The problem deals with the prevention of replicated data in the whole network by controlling the topological arrangements of these networks and limiting the incoming/outgoing connections from a node. Broadcast storming mostly occurs during the rehabilitation phase of a network. A broken link on reactivation attains same data which are already passed to the connected node which causes multiple copies of the same data. This data is again processed and causes wastage of network resources and also occupies the node by preventing them from receiving other incoming requests. A sample diagram illustrating the problem of broadcast storm is shown in Figure 7.9.

The proposed approach integrates the proximity graphs with the broadcasting principle of the existing ad hoc networks. These graphs are formed using the concept of parameter sensitivity that helps in lowering the overloaded connections. Traditionally, proximity is defined over some geometrical constraints while forming graphs. But, in this approach logical proximity is considered while forming connections between the nodes. The whole model operates in three parts.

- Derivation of parameter sensitivity for the network.
- Formation of logical proximity graphs using the parameter sensitivity.
- Integration to form the initial route finding algorithm between the source and the destination.

7.2.2 Network Parameter Sensitivity

Sensitivity in this work is considered as the crispness in the value which will provide the optimized results for the whole network. Parameter sensitivity refers to the selection of best possible parameters for the identification of optimized link between the source and

destination. Two types of parameter sensitivity are observed in the coordinating network formed between the aerial and the ground nodes. One is termed as the intra-parameter sensitivity, which is observed between the nodes of the same network, and the other is the inter-parameter sensitivity, which is observed between the nodes of two different networks. Formulation for parameter sensitivity is similar for both the types of network formations. Primarily, here, inter-sensitivity considered as this type of formation is new in ad hoc family, and also, it requires crisp resolution to such problems. Parameter sensitivity aims at providing efficient network formation with continuous connectivity between the network nodes based on the optimally selected network nodes. In a network comprising of hybrid nodes, let $P_S^{i,j}$ be the parameter sensitivity between the nodes i and j . For connectivity between the nodes, $P_S^{i,j}$ should be maximum. This parameter sensitivity, $P_S^{i,j}$ operates on the set $S = \{x : x \in \text{networkconfigurationandproperties}\}$ of network parameter that are to be maximized for continuous transmission. For each element in the set S , each element should attain its maxima when network operates at its full capacity i.e.

$$\lim_{t \rightarrow t'} x^i \rightarrow \max, \forall x \in S, \quad (7.14)$$

where t' refers to the time stamp where the network state acquires maximum transmission. This implies that when the network operates at full capacity, the dependent parameters also attain maxima. For a network with variable parameters, $P_S^{i,j}$ is defined as:

$$P_S^{i,j} \rightarrow f(i, j, S), \quad (7.15)$$

where $f(i, j, S)$ is the connectivity function derived on source i , destination j , and the set of network parameters considered for optimization. Set S used in the parameter sensitivity comprises of following elements.

$$S = \{D^{i,j}, L_A^{i,j}, L_S^{i,j}, L_R^{i,j}\}, \quad (7.16)$$

where $D^{i,j}$ is the geometrical distance, $L_A^{i,j}$ is the link availability, $L_S^{i,j}$ is the attained link speed, and $L_R^{i,j}$ is the approximated link reliability between the nodes i and j . Distance D is further dependent upon two variables: LOS which is a boolean for availability of line

of sight; false value makes $D=0$, for true boolean value of LoS , D is computed over R_B which defines the geometrical radio range between the nodes. R_B is computed as:

$$R_B = \frac{c}{\sqrt{RSS}}, \quad (7.17)$$

where c defines the signal intercepts defined over noise. Its value can also be computed from shannon's noise capacity or can be directly replaced by the network pathloss. RSS is the received signal strength which is computed using existing formula [253]:

$$RSS = R_{min}^P - A_p \eta \log_{10}\left(\frac{D_{max}}{B_r}\right), \quad (7.18)$$

where R_{min}^P is the minimum power rating attained by the network, A_p is the average pathloss, η is the actual pathloss, D_{max} is the maximum range of the radio signals, and B_r is the broadcast radius of the node. Link availability L_A is the maximum capacity a channel can offer and is derived over corridors available and probability of packets lost and received during initial handshakes. L_R is computed over successful delivery of packets as:

$$L_R = \frac{PDR \times C_{min}}{C_{max}}, \quad (7.19)$$

where PDR is the packet delivery ratio over the link, C_{min} is the number of requests handled, and C_{max} is the maximum requests a channel can handle. Now, network parameter sensitivity P_S which operates over maximum optimized values of all the parameters on which the performance of the network depends, will be given as:

$$P_S = \frac{R_B}{\beta} \left[\Theta_1 \frac{L_S}{L_A} + \Theta_2 \frac{1}{D} \right]. \quad (7.20)$$

Here, Θ_1 and Θ_2 are the transmission constants where Θ_1 operates over link metrics and Θ_2 operates over the physical constraints such that $0 \leq \Theta_1 \leq 1$ and $\Theta_1 \leq \Theta_2 \leq 1$.

For every communicating unit or connection, parameter sensitivity needs to attain the maxima throughout the connectivity. In the next step, this parameter sensitivity is used to form the logical proximity graphs that form the basis for the topological arrangement which eradicates the issue of broadcast storm in such networks.

7.2.3 Formation of Logical Proximity Graphs

Proximity is derived over geometrical constraints in normal graphs. However, in the proposed approach, logical proximity is considered which is derived over the parameter sensitivity of the various network parameters over which the network depends. Logical proximity refers to the identification of the best/optimized link between the network nodes. For a regular routing, any of the existing algorithm can be used to find the optimized path between the two nodes. Such graphs are formed on the basis of the acknowledgements and the handshakes. HELLO packets are broadcasted, and then used to find the path between the source and destination either using reactive or proactive approach. But such routing guarantees current state connectivity. All the packets are broadcasted by the omni-directional antennas. Nodes in connection with each other continues to receive data till the existence of the link. An alternate path is selected for transmission in case of link failures. No account of previously attained data is maintained in that case which leads to problem of redundancy in such networks which increases the overall overheads.

Let $G = (V, E)$ be the graph with V number of vertices and E number of edges corresponding to the route such that E accounts for all the edges between the source and destination. Graph G is optimized by P_S defined over each link such that new graph G_P is formed as logical proximity over G . Logical proximity graphs are different than the traditional proximity graphs. Rather than using geometrical considerations for the link between the nodes, logical proximity graphs use parameter sensitivity value and weight over the link between the nodes. For the graph $G_P = (V_P, E_P)$, $V_P \in V$ and $E_P \in E$, and $G_P \subset G$. This means that the final proximity graph will contain only those edges which satisfy the sensitivity over the particular edge between the two nodes. As a functional dependency, proximity graph can be defined as: $G_P = P(G)$ where P denotes the graph function operating over V and E . In the initial phase, existing network is used and HELLO based parameter values are obtained by the sender using acknowledgements. A proximity is computed over each link of pre-existing graph such that:

$$P_R = W \times P_S, \quad (7.21)$$

where W defines the link weight between the two nodes and is computed as:

$$W = \frac{C_H^M}{I}, \quad (7.22)$$

where C_H^M is the maximum number of channels available. I defines the interconnections currently active on the node. It can be noted that in a regular graph, more number of interconnections are preferred. But, in the proposed approach only required number of interconnections are used in order to eradicate the replicated connections. Also, for a non-receipt of acknowledgments or negative acknowledgments, $W = 0$, and for a radio range $r \gg Th_r$, $P_S = 0$ where Th_r is the threshold upto which the connection holds. Now, based on the threshold proximity P_R^{TH} which is attained over the average expected value of the sensitivity of each parameter, form a sub graph with new vertices and new edges. These edges are determined as proximity links L_P between the two potential network nodes P_P . Thus, by definition, G_P can also be defined as:

$$G_P = (P_P, L_P). \quad (7.23)$$

Higher the value of L_P , better is the quality of the link. In a network, all the edges which obeys the threshold value of proximity, are treated as active links, thus, for a fully connected graph G with all links in the active state, $G_P \subseteq G \ \forall \ L_p \in E$. If in a network, single proximity link is available between the nodes, then the graph is directly used for route formation, and data is transferred. But, this happens only in best cases. In rest of the scenarios, it is likely that nodes have multiple proximity links and some of the nodes might be isolated leading to network partitioning. In that case, the proximity-sensitivity routing (PSR) algorithm is used to identify the next nodes and the actual governing links. This PSR algorithm identifies the multiple proximity links and selects one possibility by overcoming the network partitioning problem (if any) in the final proximity graph.

7.2.3.1 Proximity-Sensitivity Routing (PSR)

Selection of optimal links after the formation of final proximity graphs is not as simple as picking the best value. This may lead to full network failure or multiple partitioning which

makes the network nodes unreachable. This selection is, however, based on the proper rules and sustaining rate of each of the links available as a path between the network nodes. In such cases, node heading and direction are also taken into consideration. PSR does not provide a full routing between the source and destination, rather it only provides a selection of possible optimized routes between the nodes without isolating any node. The various steps involved in PSR are as follows:

1. Compute initial proximity which is always defined for the incoming requests. Now as the wireless ad hoc networks operate over bi-directional links, count of degree is traced for each of the nodes.
2. Now, pre-analyze the whole network based on the P_R^{TH} . Virtually eradicate single link from each node and check if a node gets isolated or not.
3. If a node gets isolated, hold the link, mark it as conquered and give a counter to it. Otherwise, prompt for neighbouring nodes of source in the next HELLO message.
4. Eliminate a receiver's connection from the nodes that are directly related to the source. Again, check for isolation and proceed further.
5. Now, perform the similar operation towards the other nodes that act as relay and forms a direct connection with the destination. Steps 4 and 5 will provide with active proximity links that are directly connected to the source and destination.
6. Repeat the whole process considering other nodes as source and destination till all the cycles in the network graph are removed and each node possesses a single path towards the other node.
7. In case of similar indexing with the links, apart from the source and destination, hold the links with greater proximity value.
8. Finalize the graph, From this, it can be traced that a proximity based optimized route is available between the network nodes that will decrease any chances of redundancy and thus, will lower the network overheads.

7.2.3.2 Managing Network Partitioning and Congestion Window

In a network comprising of two variedly operating ad hoc units, chances of network partitioning are more as compared to traditional ad hoc networks. Nodes that do not obey

Algorithm 7.1 Proximity Sensitivity Based Route Selection

Require: *source, destination, $G = (V, E)$*
Ensure: *radio_range = set, links = active*
 Initialize $G = (V, E)$
 Establish link(G)
 interconnections_counter = link_count(G)
 multiple_interconnections = true, $i = 1$
while (multiple_interconnections != false) **do**
 broadcast $\rightarrow$ "HELLO" messages
 $S \leftarrow \{D^{i,j}, L_A^{i,j}, L_S^{i,j}, L_R^{i,j}\}$
 Compute $\leftarrow P_S$
 Assign $\leftarrow W$
 $j = 1$
 while ($j \leq V$) **do**
 Compute $\leftarrow P_R$
 if ($P_R \geq P_R^{TH}$) **then**
 Save $\leftarrow$ link (Source, j)
 if (interconnections(j) > 1) **then**
 Remove $\leftarrow$ duplicate_links(Source, j)
 else
 continue
 end if
 else
 Check partitioning
 if (partitioning = true) **then**
 re-initialize $\leftarrow$ configurations(j)
 decrement $\leftarrow$ cwnd
 Adjust $\leftarrow W$
 else
 continue
 end if
 end if
 $j = j + 1$
 end while
 decrement $\leftarrow$ interconnections_counter
 $i = i + 1$
end while

the policies of the logical proximity graphs may get isolated thus leading to network partitioning. Two approaches are used to recover such nodes. One of the approaches is rebroadcasting to allow reconfiguration of nodes that will adjust the weight of the link which will make its connection active. Other is limiting the increase of the congestion window to conquer the link between the isolated node and other network nodes. The congestion window is increased on the basis of standard TCP-protocol used in the initial configurations of the network. However, in order to overcome the isolated nodes, congestion window size *cwnd* is decreased as:

$$cwnd = cwnd - \frac{RTT}{P_S}, \text{ such that } P_S \geq P_S^{TH}. \quad (7.24)$$

This would reactivate the links, and no nodes will remain isolated. However, this solution can lead to over consumption of network resources due to multiple retransmissions. This may lead to node failure due to faster utilization of resources than the normal state which may cause the network to sigh off. These issues can be targeted as futuristic research issues to provide a more optimal solution in the proposed PSR algorithm. Algo-

rithm 7.1 presents the steps used in eliminating the links that causes redundancy in the network.

7.2.4 Performance Evaluation

The proposed PSR algorithm provides a possible remedy against the network redundancy that may lead to multiple failures and network partitioning. The proposed algorithm utilizes a new concept of parameter sensitivity to derive the logical proximity graphs that directly gives the efficient route between the source and destination with minimum number of sub-interconnections which are major causes of network redundancy. The proposed approach was analyzed using network simulations configured over NS-2 and *Matlab*TM. The coordination between these networks was controlled using existing cooperative framework [166]. The number of ground nodes varied from 10 to 100 and number of aerial nodes were kept between 5 to 20. Mobility of ground nodes was varied between 5 ms to 30 ms to analyze the effect of movement over broadcasting. The random waypoint topology model was used for ground nodes. Aerial nodes were configured pre-hand and were given fixed waypoints. The whole system was deployed for its comparison with the existing

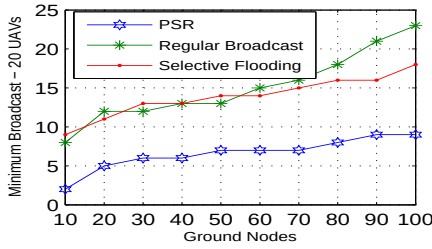


Figure 7.10: Broadcasts vs. Ground Nodes

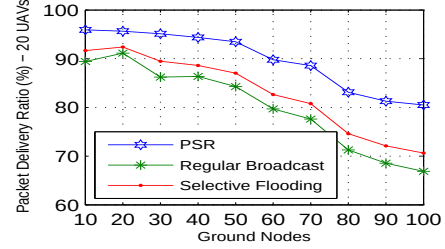


Figure 7.11: PDR vs. Ground Nodes

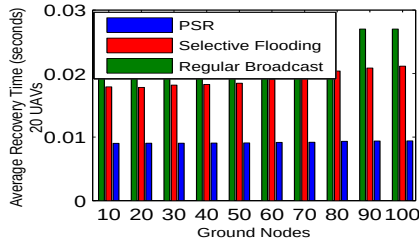


Figure 7.12: Link Recovery Time vs. Ground Nodes

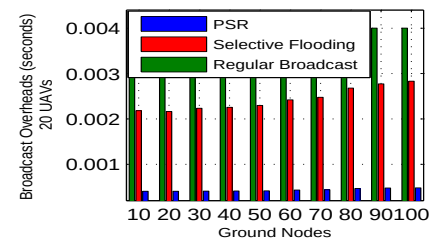


Figure 7.13: Broadcast Overheads vs. Ground Nodes

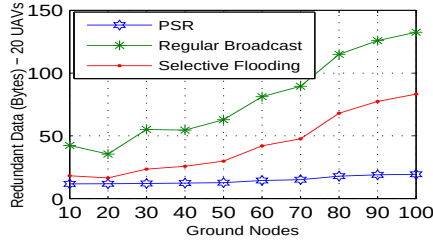


Figure 7.14: Redundancy vs. Ground Nodes

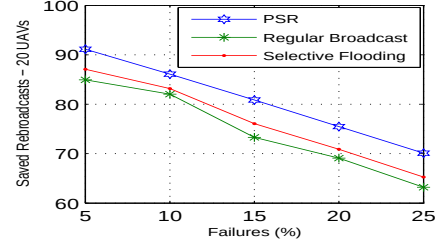


Figure 7.15: Saved Rebroadcasts vs. Failures (%)

selective flooding and regular broadcasting approach. Selective flooding refers to data broadcasting, whenever required on the basis of link selection. Selective flooding is a self generated scenario created using traditional flooding [185]. Regular broadcast is the normal approach of HELLO exchanges which are performed whenever a node searches for a path or when a link breaks. 20 UAVs were maneuvered over the ground nodes ranging from 10 to 100. Traces were generated to analyze the network performance based on the broadcasting and redundancies. Lower requirement of broadcasting refers to the efficient network formation. Proposed PSR algorithm required 53.67% less broadcasts than the selective flooding approach and 56.29% less broadcasts than the regular broadcasting approach. The plot for broadcast comparisons is shown in Figure 7.10. The network was then analyzed for packet delivery ratio. Higher value of delivery ratio denotes efficient network formation. PSR provided 7.5% more delivery than the selective flooding approach and 10.72% more than the regular broadcasting approach as shown in Figure 7.11. Link failures hinder the performance of the network. A network with failed links has to recover the faulty links. This recovery time directly affects the overall transmission and also adds upto the total broadcast overheads. These overheads should be least and well controlled as their value beyond a certain limit can stop the overall functioning of the network. The proposed PSR took 51.98% less time in recovering from faulty links in comparison with selective flooding and 66.03% less time in comparison with the regular broadcasting approach. The comparative plot for link recovery time in contrast with variation in number of nodes is shown in Figure 7.12. Also, PSR caused 67.44% less overheads than the selective flooding and 80.2% less overheads than the regular broadcasting as shown in Figure 7.13. Solutions aiming at the broadcast storm problem must be capable of providing lower redundancy in terms of data floating in the network during rebroadcasts sessions. The proposed PSR algorithm is capable of handling this situation efficiently as

it caused 66.17% less redundancy than the selective flooding and 81.6% than the regular broadcasting approach. The comparative plot for redundancy caused over the network is shown in Figure 7.14. One of the most important metric for analysis of solutions proposed for broadcast storms is saved rebroadcasts. An approach that saves more rebroadcasts is efficient in handling data redundancies, and it saves network resources from over consumption, thus, increases the network lifetime. Saved rebroadcasts are derived over the ratio of difference between the actual receiver-transmitter links to the total receiver links. PSR saved 5.25% broadcasts in comparison with selective flooding and 7.7% in comparison with regular broadcasting. The analysis plot for saved rebroadcasts comparison is shown in Figure 7.15.

7.3 Conclusion

In this chapter, a problem of network stabilization over cooperative UAVs is considered. A self-healing model is proposed which helps in attaining a stabilized state by autonomously regulating its learning rate by introducing the concept of dummy connections. The proposed model was validated using benchmark training algorithms and was able to stabilize the faulty network. In the future, work can be done in forming alternate methods for minimizing the self-healing function.

Broadcast storm is one of the reasons that depletes the network resources drastically. Broadcast storm arising from data redundancy due to multiple broadcasts towards the same link requires more resources. However, these resources are used for processing of the same data. An efficient approach based on the parameter sensitivity is proposed in this chapter. The proposed approach forms a logical proximity graph operating over the derived sensitivity and link weights. These graphs allow selection of appropriate links for routing and does not cause any network partitioning. The proposed algorithm that is used for finding routes under this approach is termed as proximity-sensitivity routing algorithm (PSR). Analysis proved that the proposed approach is capable of handling broadcast storm issue effectively over the coordinated network formed between the ground nodes and the aerial nodes.

Chapter 8

Security Enhancement of Cooperative Network Framework[†]

In this chapter, a network framework is presented for secure transmission between the aerial and the ground nodes. The network involves two ad hoc units: one operating using aerial vehicles, and other operating using ground nodes. These two networks are connected using the proposed security framework which is highly robust and fault tolerant. The framework provides security in terms of localization, corridor privacy and cognitive transfers. The mathematical modelling of the system is formulated using existing Gauss-Markov model that provides parameterizations of network components that are further evaluated over the α - β - γ filter [254]. The filter allows prediction of the network components using the network equations. These estimations allow selection of secure relays that integrates teredo tunneling [255] [256] for secure channel formation.

8.1 Proposed Framework For Secure Network Formation

Multi-UAV guided networks involve hybrid parameters that interact in correspondence to each other while operating for data/cognitive transfers. A cooperative network, thus formed, allows easy repair, arrangement and modifications based on the certain time stamps. The network formulated using diverse nodes is similar to the formation of an ambient network with intelligent behaviour. A distributed network operating as a secure-

[†]Vishal Sharma and Rajesh Kumar, “Teredo Tunneling Based Secure Transmission Between UAVs and Ground Ad Hoc Networks”, International Journal of Communication Systems, Wiley- SCIE Indexed, DOI: 10.1002/dac.3144, [I.F. 1.106].

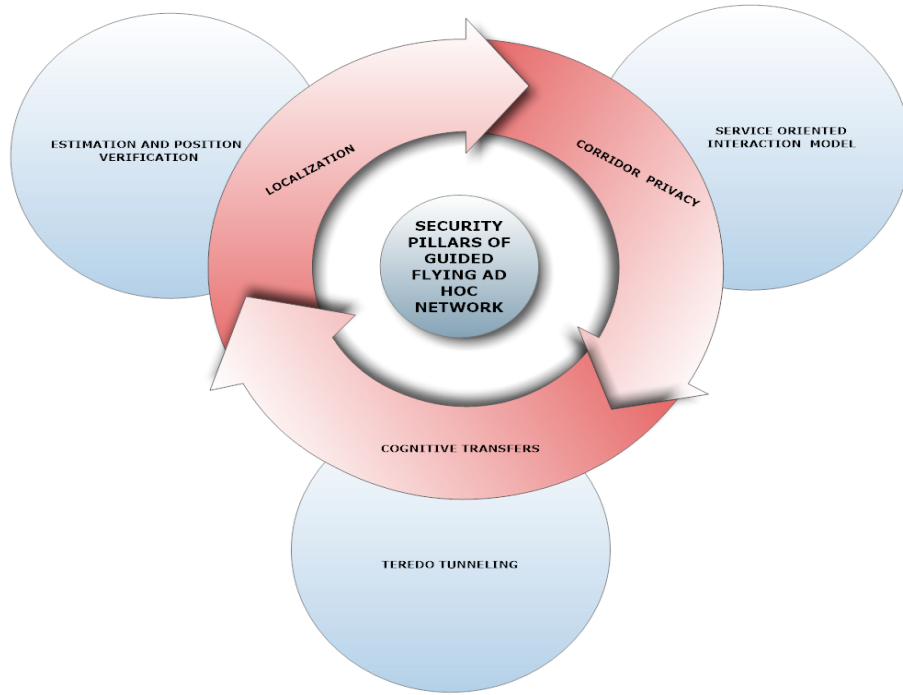


Figure 8.1: Security Pillars For Guided Flying Ad Hoc Network

ambient unit must be aware regarding the state of the network, and the devices based on the data transferred and their localization. This projection and evaluation of the network components is based on the parameterized estimation which is performed at regular interval, and the resultant updating rate is shared amongst all the components to match the performance criteria of the network. The system formulated for secure communication between the aerial and the ground network is based on the aspects of localization, cognitive transfers and corridor privacy. Localization is handled using multi-level secure layer model that handles the layer to layer interaction dealing with coordinates of the network nodes; cognitive transfers in the form of area maps are handled using abstract layer formation using teredo tunneling; and the corridor privacy is handled using agent based system service model that interacts with the existing cooperative network framework [166] to handle the secure neural coordination between the services and the network components, as shown in Figure 8.1. The proposed work is presented in three parts: one is the component model describing the working and the security components of the ambient system formed using UAVs and ground vehicles, second is the system model based on the mathematical modelling derived over Gaussian-Markov based alpha-beta-gamma filter, and third is the teredo tunneling formation.

8.1.1 Component Model For Secure Network Formation

The network framework formulated is divided into two parts: first part presents the interaction model with the underlying cooperative network framework, and the second presents the secure service model for corridor privacy that provides support for teredo tunnelling using system model for secure cognitive transfers between the two distributed networks. The interaction model forms a localization base for cooperative network, and provides a verifier and an estimator schema that binds together the abstracted functioning of the framework, as shown in Figure 8.2. The interaction model uses two service-based secure channel that interacts with the area layer $F1$, ground network layer $F2$ and aerial ad hoc network layer $F3$ of the existing network framework [166]. This interaction is based on the secure neural components that use alternative verification scheme to estimate the cognitive transmitted over the secure channel. The verification procedure is performed by forming a virtual secure server amongst the nodes, whereas the estimation is performed using a virtual dynamic server. Secure neural channel provides connectivity between the layer $F1$, $F2$, and $F3$ and the component verifier. This component verifier is connected to the service oriented architecture (SOA) for aerial-area and ground-area mapping. These SOAs interact with the network checker which forms the input of the estimator that facilitates the teredo formation. A separate log maintainer is used for each of the verifier to allow estimation of network components. Thus, using the model in Figure 8.2, layered data which is generated as series of plain format gets classified based on the channel and components that maps it to a particular module of service model

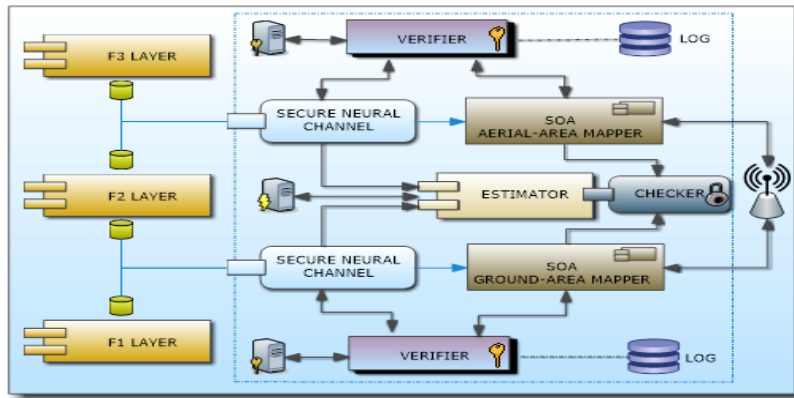


Figure 8.2: Network Interaction Model For Cooperative Network Framework

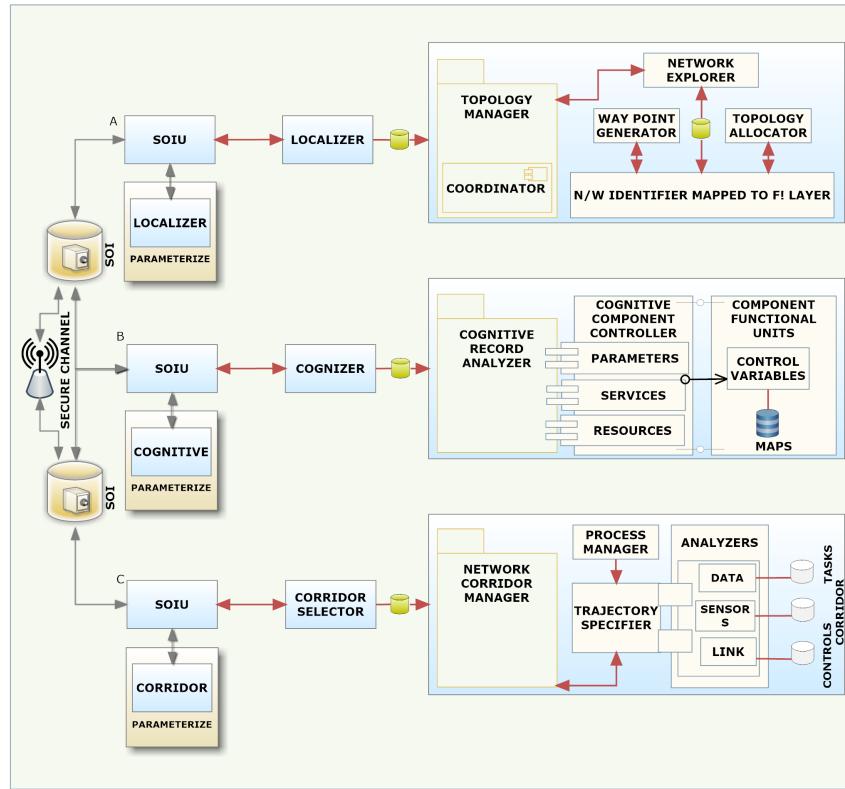


Figure 8.3: Service-Oriented Security Model For Cognitive Sharing

shown in Figure 8.3. Further, a generic key management system is used to check the correspondence over the secured channel. For the considered network model, message authentication is maintained using RCL4 over MD5 that provides 128 bit level security for fast and secure transmission whereas the key management is carried out using RSA mechanism that operates over the verifier module. The major functioning regarding maneuverabilities is derived from the initially proposed cooperative network framework [166] that also provides support for cognitive management and map formation. The interaction model aims at abstracted security applications that focus at network security defined in terms of connectivity and availability. Enhanced network availability provides robust and continuous network mapping for the formation of accurate and fault tolerant system. In the another module, a service-oriented interaction module is formulated that further enhances the security of UAV guided networks by providing cognitive security during certain network operations that might be otherwise prone to various network attacks. This service-oriented secure mapper provides associativity between the network management components and network interacting components as shown in Figure 8.3.

The service model is supported by two service units. One is the service-oriented interaction units (*SOIU*). These SOIUs provide extraction facility from the underlying interaction model. These also handles the task of parametrization by fetching the desired data components from the cognitive that are transferred over the secure channel. The number of SOIUs varies depending upon the security features incorporated in the service model. Another is the service oriented interface (*SOI*). SOIs are the interacting modules that provide the connectivity between the various SOIUs. The operational activity of the SOIs is based on the estimation algorithm formulated over the network configurations and performance. In the proposed service model, three SOIUs are considered that interacts with localizer, data-cognizer and corridor selector. Two SOIs are used to provide inter-connectivity between the three SOIUs. Further, a verification based secure channel is created between the two SOIs to allow sharing of estimated cognitive data as an abstracted functionality. The localizer is mapped to the topology manager of the network that computes waypoints, and finds an optimal path based on the localization data and also ensure the presence of the single coordinated device at a time at given location according to cooperative network framework. Topology manager utilizes the support of non-redundant coordinate generator, waypoint generator, topology allocator, and a network explorer to ensure the correct mapping between the network identifier and the underlying F1 layer of Figure 8.2. The data-cognizer is mapped to cognitive record analyzer that further uses the component controls to transmit the encrypted data over the secure neural lines, and uses certain control variables to manage the cognitive flow preventing any conflicts that may cause network faults similar to attacks. Cognitive record analyzer further controls the cognitive parameters, services, and resources. This layer is also responsible for control of variables that facilitates the map formation between the aerial and the ground nodes. Corridor selector is the third component which is mapped to the service based network corridor manager that provides corridor privacy. Based on the certain data rules and sensor values, it provides abstracted communication over the teredo channel. This component controls the physical system of the network. Tasks are analyzed by the data analyzers, corridors by sensor analyzers, and controls by link analyzers. This service oriented security model acts as a middle man between the network interaction model and the teredo formation.

8.1.2 System Model

The system considered in the formation of secure channel between the ground and the aerial nodes operates using the initially proposed cooperative network framework which handles the conceptual and physical maneuverabilities of the nodes. Also, it provides support for the initial network formation between the aerial vehicles and the ground ad hoc network. Further, the system comprises of a secure framework that provides the estimation support to the network using updates from system equations. The system model presented is the novel approach derived using Gauss-Markov based α - β - γ filter [254]. The filter allows the updates and estimations to be performed correctly over the network based on the one-step back time stamp. The initial parameter calculations allow formation of the system model. The computation of these network parameters and their usability for model derivation is presented below:

Let N_A , N_G be the number of aerial and ground vehicles, respectively. For a network operating in coordination, localization accuracy A_{AL} , A_{GL} for aerial and ground ad hoc network will be given as:

$$A_{AL} = \frac{\sum_{i=1}^{N_A} \left(\sum_{j=1}^{CH_A} (A_{CA})_j - A_{CTA} \right)}{C_{OT}}, \quad (8.1)$$

and

$$A_{GL} = \frac{\sum_{i=1}^{N_G} \left(\sum_{j=1}^{CH_G} (A_{CG})_j - A_{CTG} \right)}{C_{OT}}, \quad (8.2)$$

respectively. Here, CH_A , CH_G are the number of channels supported by aerial and ground nodes, respectively. A_{CA} and A_{CG} are the number of active connections over the supported channels for a particular aerial and ground node, respectively, with A_{CTA} and A_{CTG} be the total connection support for whole aerial and ground network, respectively. C_{OT} is the total connections available for transmission between the aerial and the ground network. In cooperative mode, neural support plays a vital role in handling the system formation. Let N_E be the number of neural nodes supported by the cooperative network

framework, then, the number of accurate neural interactions will be given as:

$$N_{EA} = \left[\frac{N_E(N_E - 1)}{2} - \left(\sum N_{UEA} + \sum N_{UEG} \right) \right], \quad (8.3)$$

where N_{UEA} and N_{UEG} are the unconnected neural components for aerial and ground interactions, respectively. In ideal mode of network formation,

$$\sum N_{UEA} + \sum N_{UEG} = 0. \quad (8.4)$$

Thus, for overall UAV guided cooperative network, relative localization accuracy A_{AGL} will be given as:

$$A_{AGL} = \left[\frac{A_{AL}A_{GL}}{N_{EA}} \right]. \quad (8.5)$$

Equation (8.5), depicts the transmission behaviour between the aerial and ground network as distributed parallel transmission, whereas between aerial-neural and ground-neural as convergent serial transmission. Over the connection accuracy, the number of cognitive transfers can be evaluated in three parts: first for the aerial network, second for ground network and third for cooperative model, which can be given as:

$$C_{TRA} = \frac{\sum_{i=1}^{N_A} (R_A)_i + \sum_{j=1}^{N_E} (R_{ANE})_j}{T_{CA}}, \quad (8.6)$$

$$C_{TRG} = \frac{\sum_{i=1}^{N_G} (R_G)_i + \sum_{j=1}^{N_E} (R_{GNE})_j}{T_{CG}}, \quad (8.7)$$

$$C_{TAG} = \frac{\sum_{i=1}^{N_A} (R_A)_i + \sum_{j=1}^{N_G} (R_G)_j}{T_{CA} + T_{CG}}, \quad (8.8)$$

where R_A , R_G , R_{NE} denotes the minimum cognitive transmission rate for aerial, ground and neural nodes over the network during individual operations, respectively. T_{CG} , T_{CA} are the total cognitive transfers for the aerial and ground network, respectively. For abstracted network formation between the aerial and ground nodes, neural components are not evaluated as they form the part of the abstracted secure teredo tunnel for transmission. Thus, C_{TAG} is evaluated using the components of aerial and ground ad hoc

networks only. A network with highly abstracted neural connectivity is dependent on the activeness of the neural interaction units to provide delay-aloof network coordination. For an abstracted neural formation, neural sensitivity N_{ES} is computed as:

$$N_{ES} = \left[\frac{\sum_{i=1}^{N_A} (A_{TCA} - L_{CA}) + \sum_{i=1}^{N_G} (A_{TCG} - L_{CG})}{M_{AG}} \right], \quad (8.9)$$

where A_{TCA} , A_{TCG} are the total active connections for aerial and ground network, respectively. L_{CA} and L_{CG} are the number of connections lost over M_{AG} aerial-ground mapping for secure ambient cooperative network. Based on this neural sensitivity, there are certain processes that run as active or passive parts of the framework. These process are directly related to the responsiveness of the network model. Thus, for sensitivity formation, network responsiveness N_{RSP} will be given as:

$$N_{RSP} = T'_{N_A} \sum_{i=1}^{N_{AP}} (P_A + S_A) + T'_{N_G} \sum_{i=1}^{N_{GP}} (P_G + S_G), \quad (8.10)$$

where T'_{N_A} and T'_{N_G} are the initial interaction time for aerial and the ground network, respectively, with P_A , P_G be the number of parallel processes for aerial and ground nodes, and S_A , S_G be the number of serial processes for aerial and ground nodes. The network adaptability AD_P in terms of recovered connections is given as:

$$AD_P = \frac{C_A^R + C_G^R}{A_{TCA} + A_{TCG}}, \quad (8.11)$$

where

$$C_A^R = A_{TCA} - F_{TA},$$

and

$$C_G^R = A_{TCG} - F_{TG},$$

here, F_{TA} and F_{TG} are the number of failed connections for aerial and ground corridors, respectively. Over the failed corridors and connections, the system might get subjected to faults that remain unidentified. This may hinder the performance of the whole network leading to cognitive failures. Thus, for network estimation, fault identification inconsis-

tency F_{IN} is required, which can be calculated using the trivial solution as;

$$F_{IN} = \left(1 - \frac{F_{TA} + F_{TG}}{F_{AG}}\right), \quad (8.12)$$

where F_{AG} denotes the total network corridor faults. Lower value of F_{AG} allows more accurate formation of the secure ambient cooperative network. Network accuracy can be attained by prior knowledge of the number of network interactions N_{IT} over the system framework, which can be computed as:

$$N_{IT} = \left(\sum_{i=1}^{N_A} (A \leftrightarrow N)_i + \sum_{j=1}^{N_G} (G \leftrightarrow N)_j - \sum_{k=1}^{N_E} (D_{NE})_k \right), \quad (8.13)$$

such that network repair time N_R will be given as:

$$N_R = T_P - \left(\sum_{i=1}^{(A_{TCA}+A_{TCG})} (T_{RE})_i + T_{ID} + T_{DR} \right), \quad (8.14)$$

where $\leftrightarrow$ symbolizes the network interactions between aerial, ground and neural formation of the network. T_P is the total network performance time, T_{RE} is the connection recovery time, T_{ID} is the network idle time, and T_{DR} is the network drop time for the losses. Corridor privacy is based on the interaction components running secure services to form an interaction model that can enhance the security level of the network components, and also provides the abstraction for system queries during updates, and information gathering process. The service model dependency of the network model is determined in terms of service identification S_{ID} which is a functional dependency as shown below:

$$S_{ID} \longleftarrow f(C_{SN}, N_A, N_G, SOI, SOIU), \quad (8.15)$$

where C_{SN} are the number of components dependent upon the type of service. SOI are the number of service oriented interfaces and $SOIU$ are the number of service oriented interaction units used to identify the service model for the corresponding framework. For

trivial formation, service identification S_{ID} can be computed as:

$$S_{ID} = \left(\sum_{i=1}^{C_{SN}} (N_{SN})_i + \sum_{j=1}^{SOI} (M_{SOI})_j + \sum_{k=1}^{SOIU} (M_{SOIU})_k \right), \quad (8.16)$$

where N_{SN} are the service components on network nodes, M_{SOI} and M_{SOIU} are the mapped services over SOI and $SOIU$, respectively. These service allocated to various network nodes are derived over percentage service P_{SN} classification as:

$$P_{SN} = \left(\frac{\sum_{i=1}^{N_A} (S_A)_i + \sum_{j=1}^{N_G} (S_G)_j + \sum_{k=1}^{N_E} (S_{NE})_k}{S_{ID}} \right) \times 100, \quad (8.17)$$

where S_A , S_G , S_{NE} are the active services for aerial, ground and neural nodes, respectively. Let X and X' be the mapping function for number of mapped network nodes and number of channels supported over these network nodes such that:

$$X \longleftarrow f(N_A, N_G, N_E),$$

and

$$X' \longleftarrow f(CH_A, CH_G).$$

Also, for the network operating to its maximum capacity, A_C^A and A_C^G be the minimum number of active corridors required to sustain the transmission. Now, corridor failures C_{OF} over the secure network formation will be given as:

$$C_{OF} = 1 - \left(\frac{\sum_{i=1}^X \left(\sum_{j=1}^{X'} (A_C^X)_j \right)_i}{\sum_{i=1}^{N_A} \left(\sum_{j=1}^{CH_A} (A_C^A)_j \right)_i + \sum_{i=1}^{N_G} \left(\sum_{j=1}^{CH_G} (A_C^G)_j \right)_i} \right). \quad (8.18)$$

The network's connection accuracy C_{OA} will be a trivial solution and can be computed as a compliment to C_{OF} . For the network under attack, number of forged connections CF_O will be computed as:

$$CF_O = C_{OA} - (N_{CO} + C_{LO}), \quad (8.19)$$

where N_{CO} are the number of non-coherent connections computed using the degree of

connectivity of each of the network node, C_{LO} are the network losses computed trivially as:

$$C_{LO} = \left(\frac{N_{LOSS} + D_{NE} + R_{OD}}{T_{LSP}} \right), \quad (8.20)$$

where R_{OD} are the router drops over the T_{LSP} link speed of the connected channel with N_{LOSS} be the total network loss.

For α - β - γ filter, the parameterizations of the network components is derived over Gauss-Markov model. For the underlying cooperative network, α at time t is derived using network connections based on the radial bias function such that:

$$\alpha_t = \alpha_o \frac{-\sum_{i=1}^X ((Max_{CO})_i - (C_{OS})_i)^2}{2 M^2}, \quad (8.21)$$

where M denotes the number of configurable parameters over the cooperative framework. Now,

$$\beta_t = \beta_o e^{\alpha_o + \alpha_t}, \quad (8.22)$$

and

$$\gamma_t = \frac{\gamma_o}{\alpha_t + e^{-\beta_t}}, \quad (8.23)$$

here, α_o , β_o , γ_o are the initial configurations considered for filter application and can be varied depending upon the state of the network initially considered. These constant provides corrective connectivity based formation for UAV guided network. However, for a network without the constraint of learning and estimation, α_o , β_o , γ_o can be assigned values arbitrarily using randomization approaches.

Let L_C be the cooperative network localization. This localization allows matrix computation for various network nodes to select the secure relay nodes over the teredo channel. The functional dependency of L_C over the network configured above is given as:

$$L_C \longleftarrow f(A_{AGL}, S_{ID}, N_{ES}, N_{RSP}). \quad (8.24)$$

Also,

$$L_C = \begin{pmatrix} L_C^o \\ L_C'' \end{pmatrix}, \quad (8.25)$$

where

$$L_C'' = \begin{bmatrix} A_{AGL} \\ S_{ID} \\ N_{ES} \\ N_{RSP} \end{bmatrix}. \quad (8.26)$$

Using α - β - γ filter, localization will be computed as:

$$L_C^t = \alpha_t L_C^o + \beta_t L_C'' + \gamma_t R_{L_C}^t + \epsilon_L^t, \quad (8.27)$$

where $R_{L_C}^t$ will be computed using Gaussian estimator [257] as:

$$R_{L_C}^t = \left(\frac{\sum_{i=1}^X \frac{L_C^o L_{C,i}^t}{C_{SP,i}}}{\sum_{i=1}^X \frac{(L_{C,i}^o)^2}{C_{SP,i}}} \right) \quad \forall C_{SP} \in (A_{TCA}, A_{TCG}), \quad (8.28)$$

and

$$\epsilon_L^t = L_{C,i}^t \left(\sum_{i=1}^N R_{L_C}^t \left(\frac{\gamma_t}{\alpha_t + \beta_t} \right) \right). \quad (8.29)$$

Let C_{OG} be the cognitive transfers for network operating in distributed mode. For parallel network operations, UAV to ground mapping is considered using serial connections over neural framework such that relative C_{OG} is given as:

$$C_{OG} = \frac{C_{OGA} C_{OGG}}{C_{OGN}}, \quad (8.30)$$

where

$$C_{OGA} \longleftarrow f(N_{RSP}, S_{ID}, CF_O, C_{TRA}). \quad (8.31)$$

Also,

$$C_{OGA} = \begin{pmatrix} C_{OGA}^o \\ C_{OGA}'' \end{pmatrix}, \quad (8.32)$$

where

$$C''_{OGA} = \begin{bmatrix} N_{RSP} \\ S_{ID} \\ CF_O \\ C_{TRA} \end{bmatrix}. \quad (8.33)$$

Using α - β - γ filter, cognitive transfer parametrization will be computed as:

$$C_{OGA}^t = \alpha_t C_{OGA}^o + \beta_t C''_{OGA} + \gamma_t R_{C_{OG}}^t + \epsilon_{C_{OG}}^t. \quad (8.34)$$

Similarly, the equation can be formulated for ground nodes using similar configurations with different values for ground ad hoc network. Using the similar estimator model as that of localization, $R_{C_{OG}}^t$ can be evaluated for time stamp t using estimator [257]. However, the map formation is continuous, thus, all the previous cognitive are taken into account for cognitive parametrization and estimation such that:

$$R_{C_{OG}}^t = \sum_i^{t-1} \left(\frac{\sum_{k=1}^{CH_X} \left(\frac{C_{OGX}^o \times C_{OGX,k}^t}{COR_{min,k}} \right)}{\sum_{k=1}^{CH_X} \frac{(C_{OGX}^o)^2}{COR_{min,k}}} \right), \quad \forall C_{OGX} \in (C_{OGA}, C_{OGG}), \forall X \in (N_A, N_G), \quad (8.35)$$

where $COR_{min,k}$ is the minimum cognitive rate over the k^{th} channel. Now,

$$\epsilon_{C_{OG}}^t = \sum_{k=1}^{CH_X} \left(\sum_{i=1}^Z (R_{C_{OG}}^t) \right)_k C_{OG,k}^t, \quad \forall Z \in (A_{TCA}, A_{TCG}). \quad (8.36)$$

Let C_{OP} be the corridor privacy for the network model based on the service components of the network such that:

$$C_{OP} \longleftarrow f(N_{RSP}, AD_P, N_{IT}, C_{OF}). \quad (8.37)$$

Also,

$$C_{OP} = \begin{pmatrix} C_{OP}^o \\ C_{OP}'' \end{pmatrix}, \quad (8.38)$$

where

$$C_{OP}'' = \begin{bmatrix} \begin{pmatrix} N_{RSP} \\ AD_P \\ N_{IT} \\ C_{OF} \end{pmatrix} \end{bmatrix}. \quad (8.39)$$

Using α - β - γ filter, estimation for corridor privacy will be computed as:

$$C_{OP}^t = \alpha_t C_{OP}^o + \beta_t C_{OP}'' + \gamma_t R_{C_{OP}}^t + \epsilon_{C_{OP}}^t, \quad (8.40)$$

where using estimator [257],

$$R_{C_{OP}}^t = \sum_i^{A_C^Y} \left(\frac{\sum_{k=1}^{CH_X} \left(\frac{C_{OP,k}^o \times C_{OP,k}^t}{CH_{min,k}} \right)}{\sum_{k=1}^{CH_X} \frac{(C_{OP,k}^o)^2}{CH_{min,k}}} \right), \quad \forall X \in (N_A, N_G), \forall Y \in (A_C^A, A_C^G), \quad (8.41)$$

where $CH_{min,k}$ is the minimum channels over the k^{th} corridor, and $\epsilon_{C_{OP}}^t$ will be given as:

$$\epsilon_{C_{OP}}^t = \sum_{k=1}^{A_C^Y} \left(\sum_{i=1}^{CH_X} (R_{C_{OP}}^t) \right)_k C_{OP,k}^t. \quad (8.42)$$

Let η_M' be the estimator factor for the derived secure model such that

$$\eta_M'(t) = \alpha_t L_C^t + \beta_t C_{OG}^t + \gamma_t C_{OP}^t + C_t, \quad (8.43)$$

where C_t is the characteristic constant for estimator factor used as network balancing factor, and $U_{M,i}^{R,t}$ will be given as:

$$U_{M,i}^{R,t} = \left(\frac{\sum_{i=1}^X \frac{\eta_{M,i}' \times \eta_{M,i}^o}{A_{TCA} + A_{TCG}}}{\sum_{i=1}^X \frac{(\eta_{M,i}^o)^2}{\left(\frac{(\gamma_t - 1)(\beta_t - 1)}{\alpha_t} \right)}} \right). \quad (8.44)$$

Therefore, the update correction factor U_m'' for secure relaying will be computed as:

$$U_m'' = \sum_{i=1}^X \left(\left(\sum_{j=1}^Y \left(\sum_{k=1}^{CH_X} (U_{M,i}^{R,t}) \right) (U_{M,o}^{R,t}) \right) E \left(\frac{U_{M,o}^{R,t}}{U_{M,i}^{R,t}} \right) \right). \quad (8.45)$$

Now, the estimation factor $\Delta\eta_M^t$ for parameterized prediction of the secure network model formation with N_t' radial function based constant for noise model will be computed as:

$$\Delta\eta_M^t = U_m'' N_t'. \quad (8.46)$$

The estimation factor can be integrated with the system parameters for whom estimation is to be performed. This allows the formation of secure teredo channel with varied values that enhances the security application of the multi-UAV guided system.

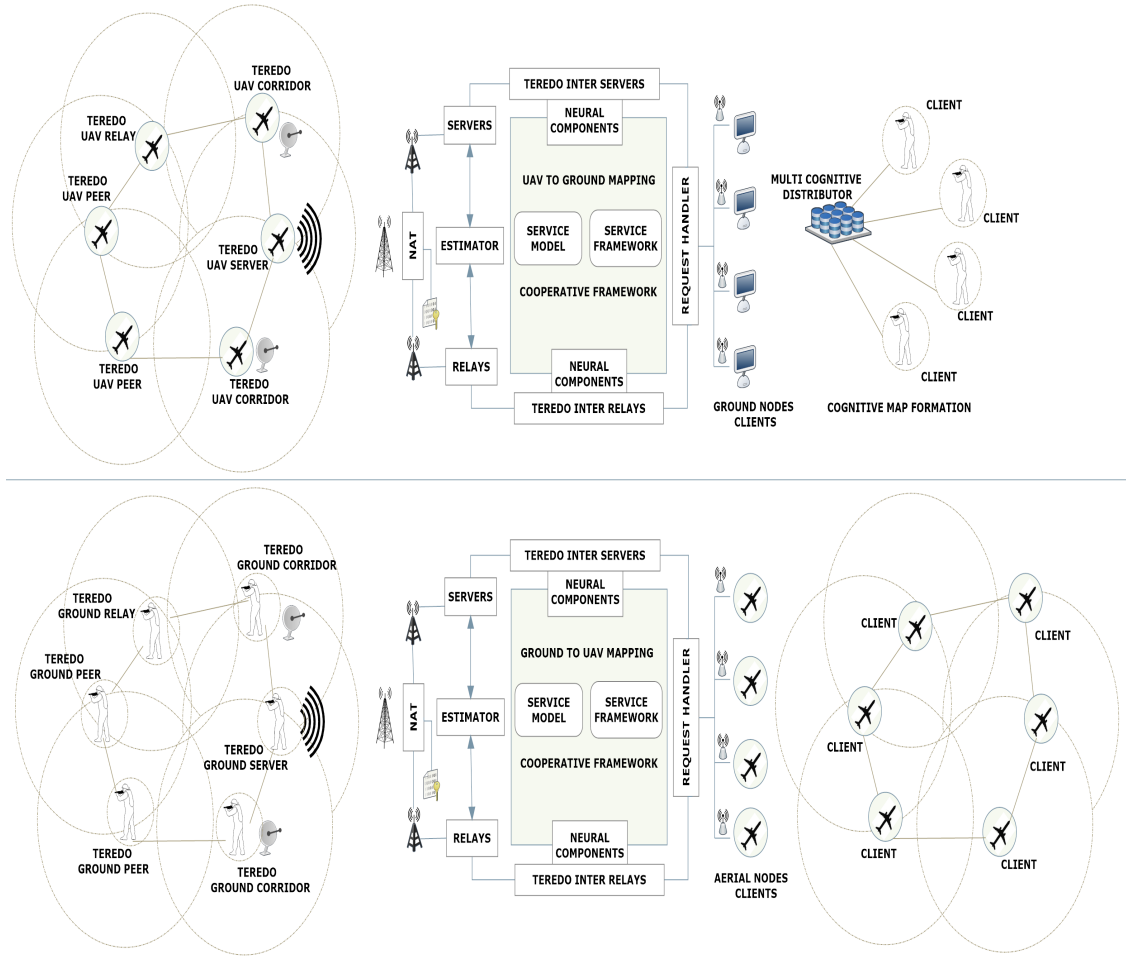


Figure 8.4: Teredo Tunneling For Secure Connection Over Network Framework

8.1.3 Teredo Tunneling For Secure Connections

Another module of the secure framework is the teredo tunneling that accounts for secure cognitive flow over the network. The secure channel is used to provide localization,

Algorithm 8.1 Estimation Algorithm For Secure Network Framework

```

Require: feedback_node() /* Node responsiveness */
           localize() /* Initial node coordinate fixing */
Ensure:  $N_A, N_G \mid N_G > N_A$  /* ground and aerial vehicles */
            $Connectivity(N_A, N_G) \geq Avg\_connectivity(N_A, N_G)$ 
while  $t' \leq total\_time$  do
  set counter = 0
  while  $i \leq X$  do
    if  $i \in (N_A, N_G)$  then
       $U_{M,i}^{R,t} = f(\eta_M'(t'))$ 
       $U_m'' = f(U_{M,i}^{R,t})$ 
       $\Delta\eta_M^{t'} = U_m'' N_{t'}'$ 
      if  $(\Delta\eta_M^{t'} \geq R_{min}(\Delta\eta_M^{t'}))$  then
        node\_accurate  $\leftarrow i$ 
        connectivity\_authenticate( $N_A, N_G$ )  $\leftarrow$  node\_accurate
      else
        unauthenticate( $N_A, N_G$ )
      end if
      boolean = verification( $\Delta\eta_M^{t'}, U_p, T_R$ )
      if boolean = true then
        set  $N_E \leftarrow neural\_nodes(N_G)$ 
        set  $C \leftarrow f(N_A, N_G, N_E)$ 
         $X'' \leftarrow C$ 
      else
        reset
      end if
      i = i + 1
      transmit\_secure\_channel( $X''$ )
    end while
    counter = counter + 1
     $t' = t' + time\_stamp$ 
  end while

```

parameter sharing, optimization of neural components based on the estimated values shared over the secure tunnel, stabilized corridor selection using cognitive over tunnel, and abstracted operating of service units and interaction module. The tunneling used is based on the generic teredo formation. During UAVs to ground mapping, network address translation is performed at the aerial network, whereas during ground to aerial mapping, translation is performed at one of the ground nodes as shown in Figure 8.4. Two teredo formations are carried for secure transmissions: one is inter-teredo formation, and another is intra-teredo formation. Intra-teredo formation is generic, and is carried out similar to teredo formation performed during mapping of IPV6 with IPV4. The intra-teredo formation is supported by selection of teredo servers that helps in identification of teredo relays that maps the network teredo peers to teredo clients for secure transmission over the network with varied protocol configurations. The intra-teredo formation is carried out at aerial ad hoc network and ground ad hoc network to map the nodes amongst themselves for secure transmission. Whereas inter-teredo formation is non-trivial, and is based on the minute multi-model teredo analyzer in between the aerial and the ground network

Algorithm 8.2 Teredo Tunneling: Selection of Relays, Clients, Peers and Servers

```

Require:  $initialize \leftarrow (N_A, N_G)$  /* Node correspondance */
            $N_E, CH_X, services$  /* Supporting network metrics */
Ensure:  $CH_X \geq \frac{(N_A, N_G)}{2}$  /* Channel associativity */
            $t\_interval = 0.5$ 
            $conn\_time = sim\_time$ 
while  $t \leq conn\_time$  do
  while  $i \leq X$  do
    initialize  $\leftarrow (\alpha_o, \beta_o, \gamma_o)$ 
     $\alpha_t = \alpha_o \frac{-\sum_{i=1}^X ((MaxCO)_i - (COS)_i)^2}{2 M^2}$ 
     $\beta_t = \beta_o e^{\alpha_o + \alpha_t}$ 
     $\gamma_t = \frac{\gamma_o}{\alpha_t + e^{-\beta_t}}$ 
     $L_C \leftarrow f(A_{AGL}, S_{ID}, N_{ES}, N_{RSP})$ 
     $COGA \leftarrow f(N_{RSP}, S_{ID}, C_{FO}, C_{TRA})$ 
     $COGG \leftarrow f(N_{RSP}, S_{ID}, C_{FO}, C_{TRG})$ 
     $COG = \frac{COGA COGG}{COGN}$ 
     $COP \leftarrow f(N_{RSP}, AD_P, N_{IT}, C_{OF})$ 
    if  $connection\_state = true$  then
       $max\_ch \leftarrow nodes(CH_i \geq \frac{(N_A, N_G)}{4})$ 
       $count_{ch} \leftarrow (max\_ch)_i$ 
       $teredo\_relays = i$ 
       $deg\_con(i) \leftarrow ini\_mat(teredo\_relays)_i$ 
      if  $node\_state = true$  then
         $max\_node\_request\_handler = node\_max(L_C, COG, COP)$ 
         $teredo\_server = count(max\_node\_request\_handler)$ 
         $path_i \leftarrow inc\_mat(source, sink, teredo\_relays_i)$ 
        if  $boolean(path_i) = false$  then
           $reset \leftarrow inc\_mat$ 
          recompute
        else
          continue
        end if
        if  $path_i \leftarrow contain\_true(teredo\_server) = true$  then
           $route_{t,i} \leftarrow server\_channel(path_i)$ 
        else
           $reset \leftarrow path_i, route_{t,i}$ 
        end if
      end if
      while  $service\_support \leq SOIU$  do
         $P_i^t \leftarrow Max(connections \leftarrow SOIU)$ 
         $P_t'' \leftarrow identify(P_i^t)$ 
         $matrix\_formed_i = cognitive(P_t'')$ 
         $contain\_db \leftarrow matrix\_formed_i$ 
         $service\_support = service\_support + 1$ 
      end while
       $map \leftarrow contain\_db$ 
    end if
     $final\_teredo\_server \leftarrow max(service\_support)$ 
    set  $final\_teredo\_relays$ 
    set  $final\_teredo\_clients$ 
     $i = i + 1$ 
  end while
   $transmit \leftarrow cognitive(final\_teredo\_server, final\_teredo\_relays, final\_teredo\_clients)$ 
   $transfer \leftarrow data(maps)$ 
   $t = t + t\_interval$ 
end while

```

operating over neural frames. This teredo model is a combination of the service model and the service framework that operates over the similar cooperative network framework, and uses neural components to select the network teredo servers and relays. Request handlers are used to identify the certain clients that will be directly operated over the secure channel created using inter-teredo formation. In case of ground nodes as clients,

a distribution mechanism is used to share the relative cognitive information amongst the nodes, whereas in case of UAV ad hoc network, the cognitive transfers are directly performed using request handler that provides the intended data to particular UAVs, thus, lowering any excess amount of load that may cause network faults. This interaction mode, service-modules and teredo formation provide security over aerial and ground vehicles with secure localization, cognitive transfers and corridor privacy. Algorithm 8.1 forms an estimation system using system equations defined for localization, cognitive transfers, corridor privacy, updating metrics, and estimator values. The algorithm also performs the verification and checking procedure before transmission. Algorithm 8.1 provides the base for teredo selection by estimating the network entities connected for cooperative network formation. Further, this algorithm is subjected to filtration process which is evaluated over Gauss-Markov estimation model to select the appropriate teredo servers, relays, clients and peers over the intra-teredo as well as inter-teredo formation. The steps and conditions for teredo formation and node selection is presented in Algorithm 8.2.

Table 8.1: Parameter Configurations For Secure Framework

PARAMETER	VALUE	DESCRIPTION
Area	10000x10000 sq. m	Simulation theater
Ground Nodes	20,30,40,50	ground ad hoc network
UAVs	5,8,10	UAVs for aerial ad hoc network
Traffic Type	CBR/TCP	Data flow type
Propagation Radio Model(Ground)	Two Ray Ground	Topological Propagation Model
Mobility Model(Ground)	Random waypoint	Movement model
Mobility Model(Aerial)	Fixed waypoint	Trajectory model
Maximum Speed(Ground)	20m/s	surface mobility
Minimum Speed(Ground)	10m/s	surface mobility
Maximum Speed(Aerial)	25m/s	UAV mobility
Minimum Speed(Aerial)	20m/s	UAV mobility
Interval Time To send	1.5 p /s	Wait time
Transmission Range	500 m	Transmission Distance of Node
Maximum Data Rate	2.4 Gbps	Maximum Transmission rate
Simulation Time	1000s	Total time for cognitive transfers
Service Request Length	256-1024	Request Content Length
Service Response Length	2000-8000	Response Content Length
Attackers	2%-5%	Number of attackers in aerial and ground network
Simulation Runs	50	Number of analysis runs

8.2 Performance Evaluation

The simulations of the teredo tunneling formation was carried over existing cooperative network framework [166]. The tunneling was created using teredo-simulator coded in C#. Other metrics were evaluated using *Matlab*TM. For channel formation and trajectory ob-

servations, waypoint generator tools were used. The traffic over the network was created using the network simulator integrated with Matlab. The security analyses were recorded using traces operating over Matlab and network simulator simultaneously. For simulations, the number of ground nodes varied from 20 to 50, and number of UAVs varied from 5 to 10, keeping a constraint that the number of UAVs was always less than the number of ground nodes. This was done intentionally to test the network security rigourously. Other parameters configured for analysis of the proposed framework are shown in Table 8.1. The network analyses were carried using the security model formulated over the associated parameters and dependent metrics. The network was traced for its localization, privacy and secure channel formation. Analyses were carried in three parts: firstly, assessment metrics were defined along with the baseline simulations; secondly, teredo formation was analyzed for various metrics; thirdly, a comparative analysis of the proposed model with the baseline and Hubenko architecture [222] is presented.

8.2.1 Assessment Metrics and Baseline

The proposed model was analyzed for its performance over various metrics such as system fairness, packet delivery ratio over varied hops, network throughput, and average delays. The description of these metrics is provided below:

- **System Fairness:** It defines the fair utilization of network resources during transmission process. Fairness F_s is computed using approach defined in Ref. [258] as:

$$F_s = \frac{(\sum_{i=1}^X (R_{COG})_i)^2}{X(\sum_{i=1}^X (R_{COG})_i^2)}, \quad (8.47)$$

where X is the number of mapped nodes, and $(R_{COG})_i$ is the cognitive rate for i^{th} node.

- **Network Throughput:** This accounts for the maximum coverage provided by network while operating on full spectral efficiency. It accounts for the maximum possible transmission rate attained between the aerial and the ground nodes.
- **Data Delivery Rate:** This accounts for successful receipt of data over the con-

nected channels between the mapped nodes. This metric accounts for all the data exchanges between the two variedly operating networks.

- **Delays:** It refers to the time consumed in transferring data from one network to another network. Although delays are bound to occur in networks that operate in an environment with possibility of partitioning, yet this metric provide a justification in terms of performance. A low value of delay would allow the sustaining of network to large extent.

Baseline Simulations: Baseline simulations were performed to analyze the performance of the proposed model without considering the attack cases. Initially, for baseline simulations, an area of 10000 x 10000 sq.m. was maneuvered by 10 UAVs over the ground nodes ranging between 20 to 50. CBR/TCP traffic type was used for both aerial and ground nodes. Two ray ground propagational model was used for topology generation of ground nodes, whereas for aerial nodes, pre-defined waypoints were generated over the same area without any redundancy. The minimum data rate of physical layer was fixed at 2 Mbps with maximum upto 2.4 Gbps. Each node was capable of transmitting at a range within 500m with speed varying between 10m/s to 20m/s for ground vehicles, and 20m/s to 25m/s for aerial vehicles. Total 100 flows were analyzed over a simulation time of 1000 seconds with a pause time of 1.5 seconds per packet. Baseline simulations were initially compared with the attackers varying between 2% and 5%. Simulations show that the system fairness for the baseline was 23.3% and 43.7% higher than the 2% and 5% attackers model, respectively, as shown in Figure 8.5. Also, baseline offered 25.3% and 19.2% better data delivery rate than the attacker model with 2% and 5% attackers, respectively. The comparative plot for data delivery rate with varying number of attackers and baseline is presented in Figure 8.6. Further, baseline provided 27.5% and 33.2% better throughput than the attacker model with 2% and 5% attackers, respectively. Figure 8.7 shows the comparative plot for baseline as compared to varying number of attackers. Average delays for the baseline was recorded 18.5% and 42.1% less than the attacker model with 2% and 5% attackers, respectively, as shown in Figure 8.8.

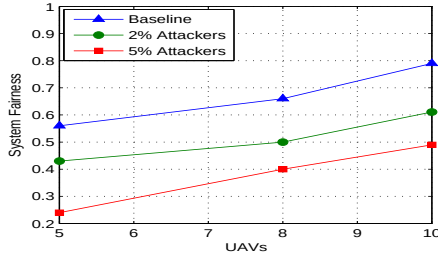


Figure 8.5: System Fairness vs. UAVs

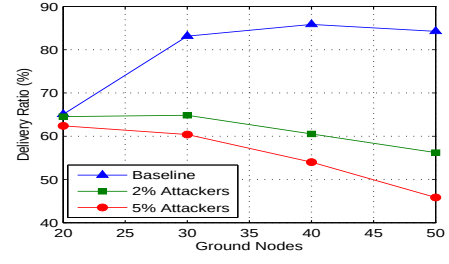


Figure 8.6: Data Delivery Ratio vs. Nodes

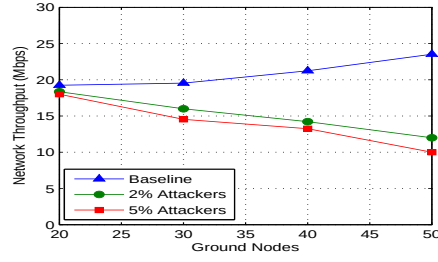


Figure 8.7: Network Throughput vs. Nodes

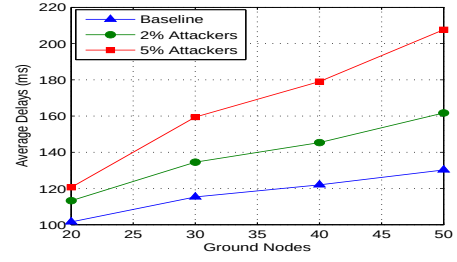


Figure 8.8: Average Delays vs. Nodes

8.2.2 Teredo Formation Analysis

Network connection accuracy determines the state of the network during intrusion. Higher value of connection accuracy is attained over the network that are robust and respond well to uncertain intrusions. With increasing number of network nodes and decreasing number of intruders, network connection accuracy increases. Also, scalability of the network is directly related to network connection accuracy. The comparative graph for connection accuracy with network scalability over ground ad hoc network is shown in Figure 8.9. A network with higher cognitive transfer ratio even during the intrusions are more secure and fault tolerant. Such networks do not get much affected by intrusion identification procedures. Figure 8.10 shows the plot for cognitive delivery ratio during intrusions. With correspond to the cognitive delivery ratio, there are certain cognitive that failed to get delivered within the time stamp. The larger value of such cognitive hinders the process of data transfer, and also lowers the interaction rate of the network components. Figure 8.11 presents the cognitive failures over UAVs vs ground nodes with varying number of attackers. Figure 8.12 shows the fault identification time of the proposed framework with variable number of nodes. Higher value of identification time denotes the slow operation of the algorithm to identify faults. For the proposed framework, the fault identification

time is way lower than the service threshold value. This proves that despite of increasing number of intruders, the proposed framework is capable of providing secure channels between the source and the destination. In a network operating in distributed mode,

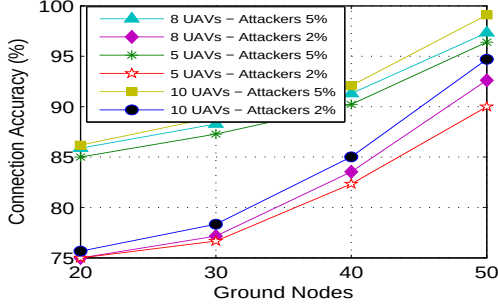


Figure 8.9: Connection Accuracy vs. Nodes

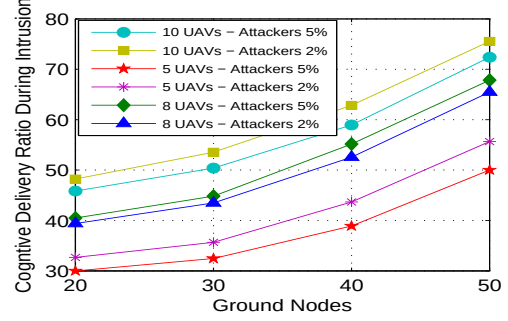


Figure 8.10: Cognitive Delivery Ratio vs. Nodes

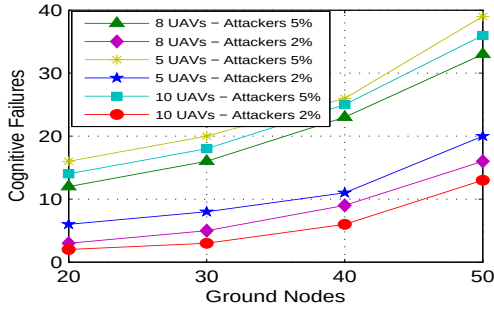


Figure 8.11: Cognitive Failures vs. Nodes

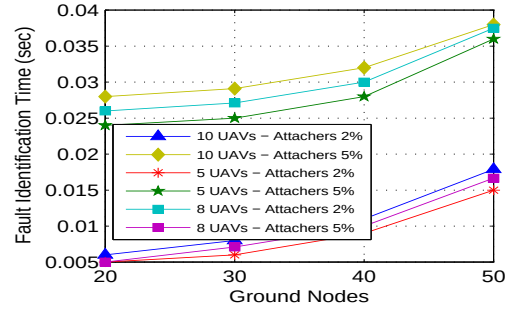


Figure 8.12: Fault Identification Time vs. Nodes

performance can also be evaluated over number of forged connections. These are the connections that are compromised and still being used for transmission process. The overall network connections in comparison with forged connections over 2% attackers are 25.67 %, 31.8%, and 41.5% higher with 5, 8, and 10 UAVs, respectively. For 5% attackers, the overall connections in comparison with forged connections are 19.1%, 22.9%, and 34% higher with 5, 8, and 10 UAVs, respectively. With more number of UAVs, more number of alternative paths are available, thus, connection forging has less impact on the proposed approach with increasing number of UAVs. The comparative analysis of the forged connections is shown in Figure 8.13. Another parameter for performance evaluation is cognitive overheads. Overheads hinder the network performance and also affect the transmission rate. Overheads increase with the increment in number of network attackers as shown in Figure 8.14. Over the UAVs operating in the aerial ad hoc network, it is important to identify the packet faults to form a secure and robust guidance system. The percentage of packet faults identified over the aerial ad hoc network

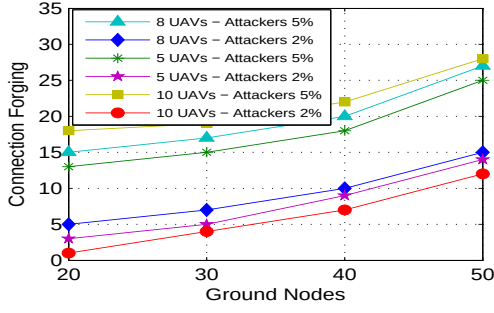


Figure 8.13: Connection Forging vs. Nodes

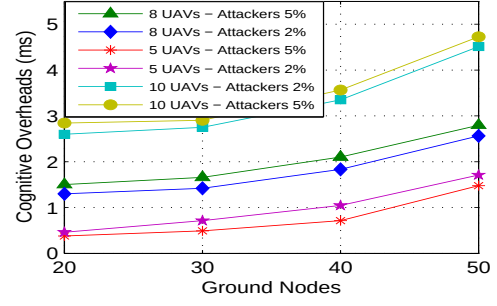


Figure 8.14: Cognitive Overheads vs. Nodes

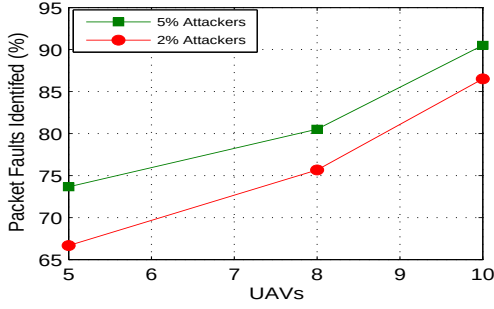


Figure 8.15: Packet Faults Identified vs. UAVs

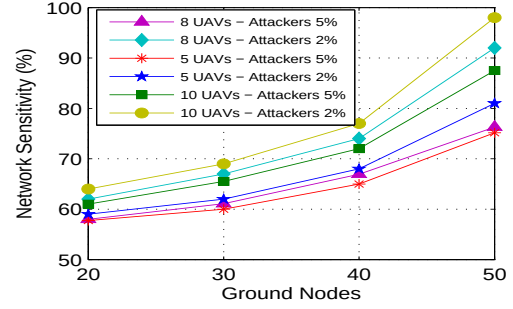


Figure 8.16: Network Sensitivity vs. Nodes

is shown in Figure 8.15. A network with higher connectivity time would allow more number of cognitive transfers, thus, would allow efficient formation of guidance system. Network connectivity can be analyzed over the network sensitivity. Network with higher sensitivity, offers relatively more connection time, thus, allowing more cognitive delivery ratio over the secure channel. Figure 8.16 shows the comparative plot for network sensitivity with varied number of attackers for UAVs operating over the ground ad hoc network. Services are mandatory to form a secure corridor channel over the proposed security framework. The service oriented interaction units and interfaces operate over these services to interact with the underlying neural model to provide a base for the criteria formation in the selection process of teredo servers and relays. Service identification time increases with increase in the number of nodes in correspondence to increase in number of network attackers, as more number of iterations would be required to select the best possible corridor for service interactions. Figure 8.17 presents the comparative analysis of the service identification time for UAVs while interacting with ground ad hoc network nodes. Another aspect of framework security testing is the situational classification time. This is the time taken by the network framework to identify the attacker model that caused intrusion over the network. Lower value of situational classification time denotes

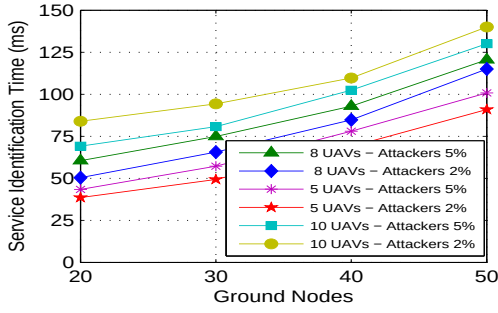


Figure 8.17: Identification Time vs. Nodes

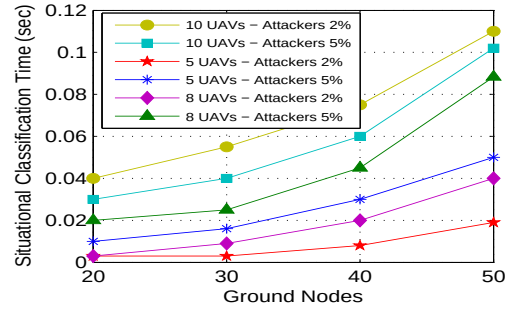


Figure 8.18: Classification Time vs. Nodes

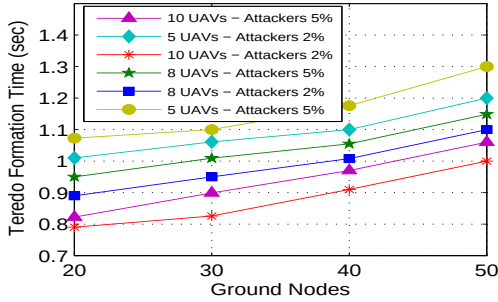


Figure 8.19: Teredo Formation Time vs. Nodes

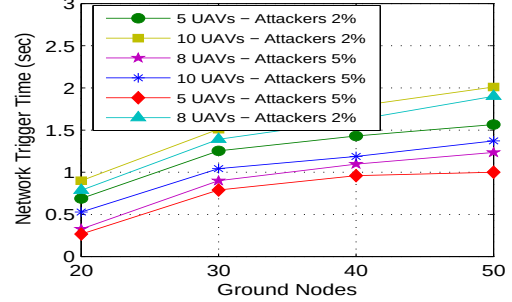


Figure 8.20: Network Trigger Time vs. Nodes

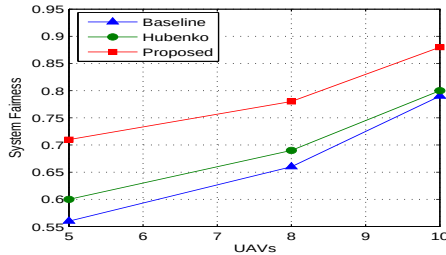
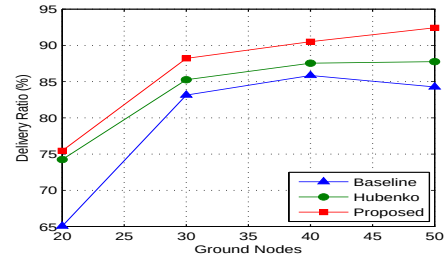
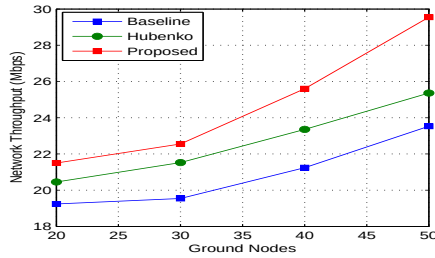
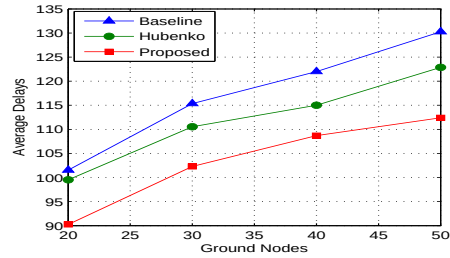
high responsiveness of the network. In the proposed framework, the situational classification time comparison graph is shown in Figure 8.18. Another important evaluation parameter is the teredo formation time. This time is the initial time required by the network to form a secure teredo tunnel and it includes time after reset in case of network intrusions. Figure 8.19 presents the comparative plot for the teredo formation, including both intra as well as the inter teredo formation amongst same network nodes and different network nodes, respectively. For intrusion identification, network framework should have the capability to respond within a certain time limit to allow accurate identification of faults, and also to provide timely remedies to failures. Figure 8.20 presents the network triggering time during the first identification of network intrusions and anomalies. The percentage variation of average values recorded for security parameters over which the framework was tested, with respect to attacker percentage, is shown in Table 8.2.

8.2.3 Comparative Study

For comparative analysis, the proposed teredo approach was compared with the baseline simulations, and the Hubenko architecture [222]. Same metrics i.e. system fairness, data

Table 8.2: Percentage Variation of Parameters with Attackers

Parameter	Variation For 2% Attackers	Variation For 5% Attackers
Cognitive Delivery Rate during Intrusion	21.23%	21.6%
Situational Classification Time	42.93%	37.57%
Network Trigger Time	16.43%	21.9%
Fault Identification Time	19.93%	21.5%
Corridor Failures	29%	28.03%
Total Faults	26.5%	26.16%
Packet Faults Identified	77.28%	82.57%
Connection Forging	39.53%	21.46%
Connection Accuracy	7.33%	9.93%
Cognitive Overheads	20.68%	15.66%
Network Sensitivity	11.46%	11.23%
Teredo Formation Time	4.47%	9.66%
Service Identification Time	17.46%	16.02%

**Figure 8.21:** System Fairness vs. UAVs**Figure 8.22:** Data Delivery Ratio vs. Nodes**Figure 8.23:** Network Throughput vs. Nodes**Figure 8.24:** Average Delays vs. Nodes

delivery rate, network throughput, and average delays were used to compare the performance of the proposed approach. Analyses were traced to plot the efficient behaviour of the proposed model in comparison with the baseline, and the standard UAVs communication secure architecture-Hubenko. The comparative analysis were carried using 10 UAVs maneuvering ground nodes ranging between 20 to 50. Results show that the proposed model provided a fair use of network resources because of accurate selection of peers and relays such that its system fairness was 12.9% higher than the Hubenko architecture, and 16.25% higher than the baseline simulations. Figure 8.21 shows the comparative plot for system fairness between the proposed, baseline, and Hubenko architecture. Also, the proposed teredo approach offered 10% higher data delivery ratio than the Hubenko architecture, and 12% higher than the baseline simulations. This is because the teredo

tunnelling allowed pre-hand selection of an appropriate set of relays, and channels for secure transmission between the aerial and the ground nodes. The comparative plot for data delivery ratio is given in Figure 8.22. With accurate mapping of nodes, proper channel allocation and correct peer network formation, the proposed model offered 8%, and 15.7% higher throughput than the Hubenko architecture, and the baseline simulations, respectively. The comparative plot for network throughput is shown in Figure 8.23. Delays in a network are bound to occur. In networks operating with varied set of nodes, delays occur because of service switching between the network nodes. Delays cannot be eradicated completely. However, controlling them beyond a certain limit can help attaining a stable, and a robust network. Results in Figure 8.24 show that the proposed model caused 7.9% less delays than the Hubenko architecture, and 10.3% less delays than the baseline simulations. Results show that the proposed network framework is capable of providing effective security solutions in terms of node localizations, corridor privacy, and secure cognitive transfers in a multi-UAV guided ad hoc networks.

8.3 Conclusion

UAVs oriented networks are highly used in both civilian as well as military activities, thus, security is always a foremost issue required to be resolved before their deployment. In this chapter, a service oriented secure network framework is proposed that provides secure tunnel formation using teredo server-relay formation over the initially proposed cooperative network framework. Security, defined in terms of availability and integrity, is achieved using secure corridor channelization and secure cognitive transfers. The analysis shows that the proposed framework withstands the network anomalies and faults, and proves robust and fault-tolerant by preventing any network hindrance during intrusions. The network model uses prediction and estimation procedures to identify its states, and forms a decision system in case of intrusions. This prediction and estimation is performed using Gauss-Markov based α - β - γ filter that provides updates at regular time stamp. The network effectiveness is proved in terms of robustness, scalability and fault-tolerant behaviours of the framework under rigorous attacking conditions.

Chapter 9

Conclusion and Future Scope

This chapter concludes the thesis and presents some suggestions and scope towards which the work presented can be extended. Section 9.1 presents the brief summary of work done in this thesis. It presents a brief ideology of various problems considered and their possible solutions. Section 9.2 presents the overall conclusion of the thesis with a summarization of outputs. Section 9.3 brings out the certain limitations of the work done. Finally, Section 9.4 presents the scope of future work for possible extension of the present work.

9.1 Summary of Work Done

The major contributions of the thesis are:

- (i). A cooperative network framework is developed for efficient transmission between the unmanned aerial vehicles and the ground ad hoc networks.
- (ii). A new routing protocol using the fruit fly optimization algorithm is developed to relay data efficiently over the proposed cooperative network framework.
- (iii). A 3-tier service model is developed to provide efficient, robust and fault tolerant connectivity over the proposed framework.
- (iv). An estimation based queue scheduler is proposed for QoS provisioning to end users in both ground as well as UAVs assisted ad hoc formations.
- (v). A new self-healing neural model is developed to handle the uncertain failures over the networked UAVs. Solution for broadcast storm over collaborative flying ad hoc networks is also presented.
- (vi). Channel security, corridor privacy and cognitive security are provided over cooper-

ative framework using a Teredo Tunneling approach.

9.2 Conclusion

In this thesis, an attempt has been made to collaborate the two simultaneously operating ad hoc networks, namely, UAV ad hoc networks, and ground ad hoc networks. Efficient coordination and service enhancement between these two varied ad hoc units are the major issues resolved in this work. The main contribution of this thesis is done in various phases. The brief summary of each part is as follows:

- (i). In the initial phase, the coordination between the aerial and the ground ad hoc networks is considered as a problem (Chapter 3), and a possible solution is presented by developing a cooperative network framework. This framework unites the aerial and the ground nodes by utilizing the Quaternion Kalman Filter and neural networks, which forms a coordinated network that allows tasks such as search and tracking to be performed efficiently and without redundancy.
- (ii). The cooperative framework developed, provides a base for the formation of the routing layer for improved coordination between the aerial and the ground networks. To fully utilize the functionality of this framework, a new routing protocol is developed using the features of Fruit Fly Optimization (FFO) algorithm (Chapter 4). This FFO is a nature inspired algorithm that provides an objective function based on the smell index. The proposed protocol allowed efficient relaying of data in the form of cognitive between the aerial ad hoc network and the ground ad hoc network.
- (iii). In order to improve the connectivity and data support, a 3-tier service model is developed which provides efficient, robust, and fault tolerant connectivity over the proposed framework (Chapter 5). This model provides a new layered model for improved service dissemination over collaborative flying ad hoc networks. The proposed model provides services based abstraction and control of various components to form a guided system.
- (iv). With the enhancement of service provisioning, quality of service (QoS) is the other major issue resolved in this work (Chapter 6). A new queue schedulers are pro-

posed for both ground as well as aerial ad hoc networks that provide efficient and reliable QoS provisioning between the end users. This QoS provides prediction and estimation for network delay and bandwidth utilization between the end users using the Quaternion-based Kalman Filter and the Satisfied Importance Analysis (SIA) approach.

- (v). A network operating with aerial nodes in dynamic conditions is liable to suffer from uncertain failures. For this, a new self-healing neural model is proposed that provides stabilization against uncertain network failures (Chapter 7). The proposed neural model utilizes the concept of dummy neurons that helps in recovery during failures. Further, issues like broadcast storm are considered for collaborative network formation between the aerial and the ground nodes. A new proximity-sensitivity routing is proposed which utilizes the concept of logical proximity graphs. The proposed approach is capable of handling broadcast storm issues in collaborative flying networks with extremely low redundancy.
- (vi). Security is one of the issues that prevails in all types of network formations. Data security, authentication, and trust aware policies already exist in the literature. But these approaches provide security to end users only. However, security in terms of localization, corridor privacy and cognitive transfers, is one of the major issues for such coordinated formations. For this, a service oriented secure network framework is proposed which is developed on the backbone of initially proposed cooperative network framework. The framework uses prediction and estimation procedures to identify its states, and forms a decision system in case of intrusions.

9.3 Limitations of Work

In this thesis some limitations were identified and listed as follows:

- (i). The major analysis of all the work has been carried out on search and tracking based operations. More sensitive tasks may be applied for further enhancement of the developed cooperative network framework.
- (ii). Real time deployment can be performed to identify the parameters that further

requires enhancement.

- (iii). The parameters configured for analyses follow standard communication rules, however, these parameters can be optimized to further enhance the performance.
- (iv). The neural sets developed, and Quaternion algebra used operates over a limited set of parameters. More set of parameters need to be identified for further improvement in results.

9.4 Scope of Future Work

Research is a continuous process. The work presented in this thesis provides coordination between the aerial and the ground nodes, thus, aiming at the formation of an efficient guidance system between the aerial ad hoc network and the ground ad hoc network. Some of the suggestions in which this work can be expanded are:

- (i). New security measures can be incorporated into the proposed routing protocol. It is expected that incorporation of security measures over this routing protocol will make it further reliable.
- (ii). Various mutual coordinated approaches such as rendezvous, task assignments, and sensor based awareness can be handled through algorithmic solutions over these collaborative networks.
- (iii). UAV failures, UAV injections, mutual peering and channel authentication can be considered as open issues for such coordination network formations.

References

- [1] I. Bekmezci, O. K. Sahingoz, and Ş. Temel, “Flying ad-hoc networks (FANETs): A survey,” *Ad Hoc Networks*, vol. 11, no. 3, pp. 1254–1270, 2013.
- [2] L. Levin, M. Segal, and H. Shpungin, “Cooperative data collection in ad hoc networks,” *Wireless networks*, vol. 19, no. 2, pp. 145–159, 2013.
- [3] K. P. Valavanis, *Advances in unmanned aerial vehicles: state of the art and the road to autonomy*, vol. 33. ISBN: 9781402061141, Springer Science & Business Media, 2008.
- [4] D. Medina, F. Hoffmann, S. Ayaz, and C.-H. Rokitansky, “Feasibility of an aeronautical mobile ad hoc network over the north atlantic corridor,” in *5th Annual Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks*, pp. 109–116, IEEE, 2008.
- [5] L. Chen and W. R. Heinzelman, “Network architecture to support QoS in mobile ad hoc networks,” in *Proceedings of the International Conference on Multimedia and Expo (ICME)*, pp. 1715–1718, 2004.
- [6] A. Fridman, S. Weber, C. Graff, D. E. Breen, K. R. Dandekar, and M. Kam, “OMAN: a mobile ad hoc network design system,” *IEEE Transactions on Mobile Computing*, vol. 11, no. 7, pp. 1179–1191, 2012.
- [7] C. Bae and W. E. Stark, “A tradeoff between energy and bandwidth efficiency in wireless networks,” in *Proceedings of the Military Communications Conference, MILCOM*, pp. 1–7, IEEE, 2007.
- [8] R. Rajaraman, “Topology control and routing in ad hoc networks: A survey,” *ACM SIGACT News*, vol. 33, no. 2, pp. 60–73, 2002.
- [9] P. Santi, “Topology control in wireless ad hoc and sensor networks,” *ACM computing surveys (CSUR)*, vol. 37, no. 2, pp. 164–194, 2005.
- [10] R. Ramanathan and J. Redi, “A brief overview of ad hoc networks: challenges and directions,” *IEEE communications Magazine*, vol. 40, no. 5, pp. 20–22, 2002.

- [11] M. Hauge, M. Brose, J. Sander, and J. Andersson, “Multi-topology routing for QoS support in the CoNSIS convoy MANET,” in *Proceedings of the Military Communications and Information Systems Conference (MCC)*, pp. 1–8, IEEE, 2012.
- [12] J. P. Hansen, S. Hissam, and L. Wrage, “QoS optimization in ad hoc wireless networks through adaptive control of marginal utility,” in *Proceedings of the Wireless Communications and Networking Conference (WCNC)*, pp. 1192–1197, IEEE, 2013.
- [13] P. Goudarzi, “Minimum distortion video transmission over wireless ad hoc networks,” in *Proceedings of the 14th European Wireless Conference*, pp. 1–7, IEEE, 2008.
- [14] M. Tarique, K. E. Tepe, S. Adibi, and S. Erfani, “Survey of multipath routing protocols for mobile ad hoc networks,” *Journal of Network and Computer Applications*, vol. 32, no. 6, pp. 1125–1143, 2009.
- [15] A. Boukerche, B. Turgut, N. Aydin, M. Z. Ahmad, L. Bölöni, and D. Turgut, “Routing protocols in ad hoc networks: A survey,” *Computer Networks*, vol. 55, no. 13, pp. 3032–3080, 2011.
- [16] G. Zhang, K. Yang, P. Liu, and X. Feng, “Incentive mechanism for multiuser cooperative relaying in wireless ad hoc networks: A resource-exchange based approach,” *Wireless personal communications*, vol. 73, no. 3, pp. 697–715, 2013.
- [17] M. Conti and S. Giordano, “Mobile ad hoc networking: milestones, challenges, and new research directions,” *IEEE Communications Magazine*, vol. 52, no. 1, pp. 85–96, 2014.
- [18] R. Sun, J. Yuan, I. You, X. Shan, and Y. Ren, “Energy-aware weighted graph based dynamic topology control algorithm,” *Simulation Modelling Practice and Theory*, vol. 19, no. 8, pp. 1773–1781, 2011.
- [19] H. Zhou, H. Kong, L. Wei, D. Creighton, and S. Nahavandi, “Efficient road detection and tracking for unmanned aerial vehicle,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 1, pp. 297–309, 2015.
- [20] P. F. Bonnefoi, D. Sauveron, and J. H. Park, “MANETS: an exclusive choice between use and security,” *Computing and Informatics*, vol. 27, no. 5, pp. 799–821, 2012.

- [21] S. Samarakoon, M. Bennis, W. Saad, and M. Latva-aho, "Opportunistic sleep mode strategies in wireless small cell networks," in *Proceedings of the International Conference on Communications (ICC)*, pp. 2707–2712, IEEE, 2014.
- [22] C. Bolkcom, "Homeland security: Unmanned aerial vehicles and border surveillance," in *Proceedings of the Library of Congress Washington DC Congressional Research Service*, pp. 1–7, DTIC Document, 2004.
- [23] A. Abdessameud and A. Tayebi, "Global trajectory tracking control of VTOL-UAVs without linear velocity measurements," *Automatica*, vol. 46, no. 6, pp. 1053–1059, 2010.
- [24] D. Perez, I. Maza, F. Caballero, D. Scarlatti, E. Casado, and A. Ollero, "A ground control station for a multi-uav surveillance system," *Journal of Intelligent & Robotic Systems*, vol. 69, no. 1-4, pp. 119–130, 2013.
- [25] I. Bekmezci, M. Ermis, and S. Kaplan, "Connected multi UAV task planning for Flying Ad Hoc Networks," in *Proceedings of the International Black Sea Conference on Communications and Networking (BlackSeaCom)*, pp. 28–32, IEEE, 2014.
- [26] W. Li, X. Cheng, T. Jing, and X. Xing, "Cooperative multi-hop relaying via network formation games in cognitive radio networks," in *Proceedings of INFOCOM*, pp. 971–979, IEEE, 2013.
- [27] J. Li, Y. Zhou, L. Lamont, M. Toulgoat, and C. A. Rabbath, "Packet delay in UAV wireless networks under non-saturated traffic and channel fading conditions," *Wireless personal communications*, vol. 72, no. 2, pp. 1105–1123, 2013.
- [28] D. L. Gu, G. Pei, H. Ly, M. Gerla, B. Zhang, and X. Hong, "UAV aided intelligent routing for ad-hoc wireless network in single-area theater," in *Proceedings of the Wireless Communications and Networking Conference, (WCNC)*, vol. 3, pp. 1220–1225, IEEE, 2000.
- [29] L. Lilien, A. Gupta, and Z. Yang, "Opportunistic resource utilization networksA new paradigm for specialized ad hoc networks," *Computers & electrical engineering*, vol. 36, no. 2, pp. 328–340, 2010.
- [30] L. T. Lilien, L. Ben Othmane, P. Angin, A. DeCarlo, R. M. Salih, and B. Bhargava, "A simulation study of ad hoc networking of UAVs with opportunistic resource utilization networks," *Journal of Network and Computer Applications*, vol. 38, pp. 3–

15, 2014.

- [31] Y. Yang, A. Minai, and M. M. Polycarpou, “Evidential map-building approaches for multi-UAV cooperative search,” in *Proceedings of the American Control Conference*, vol. 1, p. 116, 2005.
- [32] L. Doitsidis, K. P. Valavanis, N. C. Tsourveloudis, and M. Kontitsis, “A framework for fuzzy logic based UAV navigation and control,” in *Proceedings of the International Conference on Robotics and Automation*, vol. 4, pp. 4041–4046, IEEE, 2004.
- [33] I. Rubin, A. Behzad, H.-J. Ju, R. Zhang, X. Huang, Y. Liu, and R. Khalaf, “Ad hoc wireless networks with mobile backbones,” in *Proceedings of the 15th International Symposium on Personal, Indoor and Mobile Radio Communications, PIMRC*, vol. 1, pp. 566–573, IEEE, 2004.
- [34] C. E. Palazzi, C. Roseti, M. Luglio, M. Gerla, M. Sanadidi, and J. Stepanek, “Satellite coverage in urban areas using Unmanned Airborne Vehicles (UAVs),” in *Proceedings of the 59th Vehicular Technology Conference, VTC-Spring*, vol. 5, pp. 2886–2890, IEEE, 2004.
- [35] C. E. Palazzi, C. Roseti, M. Luglio, M. Gerla, M. Sanadidi, and J. Stepanek, “Enhancing Transport Layer Capability in HAPS–Satellite Integrated Architecture,” *Wireless Personal Communications*, vol. 32, no. 3-4, pp. 339–356, 2005.
- [36] A. Ajami, J. Balmat, J.-P. Gauthier, and T. Maillot, “Path planning and ground control station simulator for uav,” in *Proceedings of the Aerospace Conference*, pp. 1–13, IEEE, 2013.
- [37] Y. Wei, M. B. Blake, and G. R. Madey, “An operation-time simulation framework for UAV swarm configuration and mission planning,” *Procedia Computer Science*, vol. 18, pp. 1949–1958, 2013.
- [38] P. R. Chandler, M. Pachter, D. Swaroop, J. M. Fowler, J. K. Howlett, S. Rasmussen, C. Schumacher, and K. Nygard, “Complexity in UAV cooperative control,” in *Proceedings of the American Control Conference*, vol. 3, pp. 1831–1836, IEEE, 2002.
- [39] M. Flint, M. Polycarpou, and E. Fernandez-Gaucherand, “Cooperative control for multiple autonomous UAV’s searching for targets,” in *Proceedings of the 41st Conference on Decision and Control*, vol. 3, pp. 2823–2828, IEEE, 2002.

- [40] P. Vincent and I. Rubin, “A framework and analysis for cooperative search using UAV swarms,” in *Proceedings of the ACM symposium on Applied computing*, pp. 79–86, ACM, 2004.
- [41] M. M. Polycarpou, Y. Yang, and K. M. Passino, “A cooperative search framework for distributed agents,” in *Proceedings of the IEEE International Symposium on Intelligent Control (ISIC’01)*, pp. 1–6, IEEE, 2001.
- [42] B. Bellur and R. G. Ogier, “A reliable, efficient topology broadcast protocol for dynamic networks,” in *Proceedings of the Eighteenth Annual Joint Conference of the IEEE Computer and Communications Societies, INFOCOM’99*, vol. 1, pp. 178–186, IEEE, 1999.
- [43] L. Merino, F. Caballero, J. Dios, and A. Ollero, “Cooperative fire detection using unmanned aerial vehicles,” in *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 1884–1889, IEEE, 2005.
- [44] M. D. Richards, D. Whitley, J. R. Beveridge, T. Mytkowicz, D. Nguyen, and D. Rome, “Evolving cooperative strategies for UAV teams,” in *Proceedings of the 7th annual conference on Genetic and evolutionary computation*, pp. 1721–1728, ACM, 2005.
- [45] J. P. How, C. Fraser, K. C. Kulling, L. F. Bertuccelli, O. Toupet, L. Brunet, A. Bachrach, and N. Roy, “Increasing autonomy of UAVs,” *Robotics & Automation Magazine, IEEE*, vol. 16, no. 2, pp. 43–51, 2009.
- [46] J. Capitán, L. Merino, F. Caballero, and A. Ollero, “Decentralized Delayed-State Information Filter (DDSIF): A new approach for cooperative decentralized tracking,” *Robotics and Autonomous Systems*, vol. 59, no. 6, pp. 376–388, 2011.
- [47] M. Müller, “Flying Ad-Hoc Networks,” *Institute of Media Informatics Ulm University*, pp. 53–60, 2012.
- [48] H. Wang, W. Wang, J. Huang, D. Huo, and Y. Xu, “Modeling multiple unmanned aerial vehicles placement problem in ad hoc network via quadratic unconstrained binary optimization,” in *Proceedings of the International Conference on Unmanned Aircraft Systems (ICUAS)*, pp. 926–932, IEEE, 2013.
- [49] O. K. Sahingoz, “Networking models in flying Ad-hoc networks (FANETs): Concepts and challenges,” *Journal of Intelligent & Robotic Systems*, vol. 74, no. 1-2,

pp. 513–527, 2014.

- [50] M. Liu, J. Lin, and Y. Yuan, “Research of UAV cooperative reconnaissance with self-organization path planning,” in *Proceedings of the International Conference on Computer, Networks and Communication Engineering*, Atlantis Press, 2013.
- [51] J. Miles, G. Kamath, S. Muknahallipatna, M. Stefanovic, and R. F. Kubichek, “Optimal trajectory determination of a single moving beacon for efficient localization in a mobile ad-hoc network,” *Ad Hoc Networks*, vol. 11, no. 1, pp. 238–256, 2013.
- [52] A. Benini, A. Mancini, and S. Longhi, “An imu/uwb/vision-based extended kalman filter for mini-uav localization in indoor environment using 802.15. 4a wireless sensor network,” *Journal of Intelligent & Robotic Systems*, vol. 70, no. 1-4, pp. 461–476, 2013.
- [53] O. Eroglu and G. Yilmaz, “A terrain referenced UAV localization algorithm using binary search method,” *Journal of Intelligent & Robotic Systems*, vol. 73, no. 1-4, pp. 309–323, 2014.
- [54] J. Schleich, A. Panchapakesan, G. Danoy, and P. Bouvry, “UAV fleet area coverage with network connectivity constraint,” in *Proceedings of the 11th ACM international symposium on Mobility management and wireless access*, pp. 131–138, ACM, 2013.
- [55] S. Hauert, J.-C. Zufferey, and D. Floreano, “Evolved swarming without positioning information: an application in aerial communication relay,” *Autonomous Robots*, vol. 26, no. 1, pp. 21–32, 2009.
- [56] O. Cetin, I. Zagli, and G. Yilmaz, “Establishing obstacle and collision free communication relay for UAVs with artificial potential fields,” *Journal of Intelligent & Robotic Systems*, vol. 69, no. 1-4, pp. 361–372, 2013.
- [57] K. Tsagkaris, G. Nguengang, A. Galani, I. Grida Ben Yahia, M. Ghader, A. Kaloylos, M. Gruber, A. Kousaridas, M. Bouet, and S. Georgoulas, “A survey of autonomic networking architectures: towards a Unified Management Framework,” *International Journal of Network Management*, vol. 23, no. 6, pp. 402–423, 2013.
- [58] R. Sabatini, T. Moore, and C. Hill, “GNSS avionics-based Integrity augmentation for RPAS detect-and-avoid applications,” in *Proceedings of the Fourth Australasian Unmanned Systems Conference, Melbourne*, pp. 1–13.

- [59] R. Sabatini, A. Gardi, and M. A. Richardson, "LIDAR obstacle warning and avoidance system for unmanned aircraft," *International Journal of Mechanical, Aerospace, Industrial and Mechatronics Engineering*, vol. 8, no. 4, pp. 702–713, 2014.
- [60] S. Sharma, C. Sur, A. Shukla, and R. Tiwari, "CBDF Based Cooperative Multi Robot Target Searching and Tracking Using BA," in *Computational Intelligence in Data Mining-Vol. 3*, pp. 373–384, Springer, 2015.
- [61] Y. Ben-Asher, S. Feldman, P. Gurfil, and M. Feldman, "Distributed decision and control for cooperative UAVs using ad hoc communication," *IEEE Transactions on Control Systems Technology*, vol. 16, no. 3, pp. 511–516, 2008.
- [62] J. Hu, L. Xie, K.-Y. Lum, and J. Xu, "Multiagent information fusion and cooperative control in target search," *IEEE Transactions on Control Systems Technology*, vol. 21, no. 4, pp. 1223–1235, 2013.
- [63] R. R. McCune and G. R. Madey, "Agent-based simulation of cooperative hunting with UAVs," in *Proceedings of the Agent-Directed Simulation Symposium*, p. 8, Society for Computer Simulation International, 2013.
- [64] L. Bing, L. Jie, and H. KeWei, "Modeling and flocking consensus analysis for large-scale UAV swarms," *Mathematical Problems in Engineering*, vol. 2013, <http://dx.doi.org/10.1155/2013/368369>, pp. 1–9.
- [65] B. S. Morse, C. H. Engh, and M. A. Goodrich, "UAV video coverage quality maps and prioritized indexing for wilderness search and rescue," in *Proceedings of the 5th international conference on Human-robot interaction*, pp. 227–234, IEEE Press, 2010.
- [66] Y. Altshuler, V. Yanovsky, I. A. Wagner, and A. M. Bruckstein, "Efficient cooperative search of smart targets using uav swarms," *Robotica*, vol. 26, no. 04, pp. 551–557, 2008.
- [67] D. Shen, G. Chen, J. B. Cruz Jr, and E. Blasch, "A game theoretic data fusion aided path planning approach for cooperative UAV ISR," in *Proceedings of Aerospace Conference*, pp. 1–9, IEEE, 2008.
- [68] T. Yamasaki, H. Sakaida, K. Enomoto, H. Takano, and Y. Baba, "Robust trajectory-tracking method for UAV guidance using proportional navigation," in *In-*

- ternational Conference on Control, Automation and Systems, ICCAS'07*, pp. 1404–1409, IEEE, 2007.
- [69] D. Niculescu and B. Nath, “DV based positioning in ad hoc networks,” *Telecommunication Systems*, vol. 22, no. 1-4, pp. 267–280, 2003.
 - [70] M. Iordanakis, D. Yannis, K. Karras, G. Bogdos, G. Dilintas, M. Amirfeiz, G. Colangelo, and S. Baiotti, “Ad-hoc routing protocol for aeronautical mobile ad-hoc networks,” in *Proceedings of the Fifth International Symposium on Communication Systems, Networks and Digital Signal Processing (CSNDSP)*, pp. 1–5, Citeseer, 2006.
 - [71] C. Zang and S. Zang, “Mobility prediction clustering algorithm for UAV networking,” in *Proceedings of the GLOBECOM Workshops (GC Wkshps)*, pp. 1158–1161, IEEE, 2011.
 - [72] L. Lin, Q. Sun, J. Li, and F. Yang, “A novel geographic position mobility oriented routing strategy for UAVs,” *Journal of Computational Information Systems*, vol. 8, no. 2, pp. 709–716, 2012.
 - [73] H. Wang, D. Huo, and B. Alidaee, “Position Unmanned Aerial Vehicles in the Mobile Ad Hoc Network,” *Journal of Intelligent & Robotic Systems*, vol. 74, no. 1-2, pp. 455–464, 2014.
 - [74] B. Bellur, M. Lewis, and F. Templin, “An ad-hoc network for teams of autonomous vehicles,” in *Proceedings of the First Annual Symposium on Autonomous Intelligence Networks and Systems*, pp. 1–6, Citeseer, 2002.
 - [75] P. Zhan, D. Casbeer, and A. L. Swindlehurst, “A centralized control algorithm for target tracking with UAVs,” in *Proceedings of the 39th IEEE Asilomar Conference*, vol. 13, pp. 109–113, IEEE Piscataway, NJ, 2005.
 - [76] A. Zhu and S. X. Yang, “A neural network approach to dynamic task assignment of multirobots,” *IEEE Transactions on Neural Networks*, vol. 17, no. 5, pp. 1278–1287, 2006.
 - [77] L. F. Bertuccelli and J. P. How, “Search for dynamic targets with uncertain probability maps,” in *Proceedings of the American Control Conference*, pp. 6–pp, IEEE, 2006.

- [78] Y. Yang, M. M. Polycarpou, and A. A. Minai, "Multi-UAV cooperative search using an opportunistic learning method," *Journal of Dynamic Systems, Measurement, and Control*, vol. 129, no. 5, pp. 716–728, 2007.
- [79] J. G. Manathara, P. Sujit, and R. W. Beard, "Multiple UAV coalitions for a search and prosecute mission," *Journal of Intelligent & Robotic Systems*, vol. 62, no. 1, pp. 125–158, 2011.
- [80] G. R. Madey, M. B. Blake, C. Poellabauer, H. Lu, R. R. McCune, and Y. Wei, "Applying dddas principles to command, control and mission planning for uav swarms," *Procedia Computer Science*, vol. 9, pp. 1177–1186, 2012.
- [81] P. Cevik, I. Kocaman, A. S. Akgul, and B. Akca, "The small and silent force multiplier: a swarm UAVelectronic attack," *Journal of Intelligent & Robotic Systems*, vol. 70, no. 1-4, pp. 595–608, 2013.
- [82] H. Oh, S. Kim, H.-S. Shin, A. Tsourdos, and B. A. White, "Behaviour recognition of ground vehicle using airborne monitoring of unmanned aerial vehicles," *International Journal of Systems Science*, vol. 45, no. 12, pp. 2499–2514, 2014.
- [83] A. Jensen and Y. Chen, "Tracking tagged fish with swarming unmanned aerial vehicles using fractional order potential fields and Kalman filtering," in *Proceedings of International Conference on Unmanned Aircraft Systems (ICUAS)*, pp. 1144–1149, IEEE, 2013.
- [84] M.-H. Kim, H. Baik, and S. Lee, "Response Threshold Model Based UAV Search Planning and Task Allocation," *Journal of Intelligent & Robotic Systems*, vol. 75, no. 3-4, pp. 625–640, 2014.
- [85] H. J. Ortiz-Pena, M. Karwan, M. Sudit, M. Hirsch, and R. Nagi, "A multi-perspective optimization approach to UAV resource management for littoral surveillance," in *Proceedings of the 16th International Conference on Information Fusion (FUSION)*, pp. 492–498, IEEE, 2013.
- [86] I. Jawhar, N. Mohamed, J. Al-Jaroodi, and S. Zhang, "A framework for using unmanned aerial vehicles for data collection in linear wireless sensor networks," *Journal of Intelligent & Robotic Systems*, vol. 74, no. 1-2, pp. 437–453, 2014.
- [87] S. Malek, Y. Bazi, N. Alajlan, H. AlHichri, and F. Melgani, "Efficient framework for palm tree detection in UAV images," *IEEE Journal of Selected Topics in Applied*

Earth Observations and Remote Sensing, vol. 7, no. 12, pp. 4692–4703, 2014.

- [88] J. J. Lugo and A. Zell, “Framework for autonomous on-board navigation with the AR. Drone,” *Journal of Intelligent & Robotic Systems*, vol. 73, no. 1-4, pp. 401–412, 2014.
- [89] J. Wan, L. Zhong, and F. Zhang, “Cooperative Localization of Multi-UAVs via Dynamic Nonparametric Belief Propagation under GPS Signal Loss Condition,” *International Journal of Distributed Sensor Networks*, vol. 2014, <http://dx.doi.org/10.1155/2014/562380>, pp. 1–10, 2014.
- [90] X.-S. Yang, *Nature-inspired metaheuristic algorithms*. ISBN: 9781905986286, Luniver press, 2010.
- [91] F. Dressler and O. B. Akan, “A survey on bio-inspired networking,” *Computer Networks*, vol. 54, no. 6, pp. 881–900, 2010.
- [92] B. Kim, J. Yang, and I. You, “A survey of NETLMM in all-IP-based wireless networks,” in *Proceedings of the International Conference on Mobile Technology, Applications, and Systems*, pp. 60–65, ACM, 2008.
- [93] M. Dorigo and M. Birattari, “Ant colony optimization,” in *Encyclopedia of machine learning*, pp. 36–39, Springer, 2010.
- [94] D. Karaboga, S. Okdem, and C. Ozturk, “Cluster based wireless sensor network routing using artificial bee colony algorithm,” *Wireless Networks*, vol. 18, no. 7, pp. 847–860, 2012.
- [95] J. C. Bansal, H. Sharma, and S. S. Jadon, “Artificial bee colony algorithm: a survey,” *International Journal of Advanced Intelligence Paradigms*, vol. 5, no. 1-2, pp. 123–159, 2013.
- [96] J. Kennedy, “Particle swarm optimization,” in *Encyclopedia of Machine Learning*, pp. 760–766, Springer, 2010.
- [97] X.-S. Yang, “Firefly algorithms for multimodal optimization,” in *Stochastic algorithms: foundations and applications*, pp. 169–178, Springer, 2009.
- [98] X.-S. Yang and S. Deb, “Cuckoo search via Lévy flights,” in *Proceedings of the World Congress on Nature & Biologically Inspired Computing, NaBIC*, pp. 210–214, IEEE, 2009.

- [99] X.-S. Yang and A. Hossein Gandomi, “Bat algorithm: a novel approach for global engineering optimization,” *Engineering Computations*, vol. 29, no. 5, pp. 464–483, 2012.
- [100] W.-T. Pan, “A new fruit fly optimization algorithm: taking the financial distress model as an example,” *Knowledge-Based Systems*, vol. 26, pp. 69–74, 2012.
- [101] G. Di Caro and M. Dorigo, “AntNet: Distributed stigmergetic control for communications networks,” *Journal of Artificial Intelligence Research*, pp. 317–365, 1998.
- [102] D. Camara and A. A. Loureiro, “A gps/ant-like routing algorithm for ad hoc networks,” in *Proceedings of the Wireless Communications and Networking Conference*, vol. 3, pp. 1232–1236, IEEE, 2000.
- [103] S. Marwaha, C. K. Tham, and D. Srinivasan, “Mobile agents based routing protocol for mobile ad hoc networks,” in *Proceedings of the Global Telecommunications Conference, GLOBECOM’02*, vol. 1, pp. 163–167, IEEE, 2002.
- [104] J. S. Baras and H. Mehta, “Dynamic adaptive routing in manets,” in *Proceedings of Annual ARL CTA Symposium*, pp. 1–5.
- [105] Z. Yuan-Yuan and H. Yan-Xiang, “Ant routing algorithm for mobile ad-hoc networks based on adaptive improvement,” in *Proceedings of the International Conference on Wireless Communications, Networking and Mobile Computing*, vol. 2, pp. 678–681, IEEE, 2005.
- [106] G. Di Caro, F. Ducatelle, L. M. Gambardella, and M. Dorigo, “AntHocNet: an adaptive nature-inspired algorithm for routing in mobile ad hoc networks,” *European Transactions on Telecommunications*, vol. 16, no. 5, pp. 443–455, 2005.
- [107] M. Ahmed, A. Boudhir, and M. Bouhorma, “New Routing Algorithm Based on ACO Approach for Lifetime Optimization in Wireless Sensor Networks,” *International Journal of Networks and Systems*, vol. 1, no. 2, 2012.
- [108] E. Osagie, P. Thulasiraman, and R. K. Thulasiram, “PACONET: imProved ant colony optimization routing algorithm for mobile ad hoc networks,” in *Proceedings of the 22nd International Conference on Advanced Information Networking and Applications*, pp. 204–211, IEEE, 2008.
- [109] J. Wang, E. Osagie, P. Thulasiraman, and R. K. Thulasiram, “HOPNET: A hybrid

- ant colony optimization routing algorithm for mobile ad hoc network,” *Ad Hoc Networks*, vol. 7, no. 4, pp. 690–705, 2009.
- [110] A. M. A. Elmoniem, H. M. Ibrahim, M. H. Mohamed, and A.-R. Hedar, “Ant Colony and Load Balancing Optimizations for AODV Routing Protocol,” *Int J Sens Netw Data Commun*, vol. 1, no. 1, pp. 1–14, 2011.
 - [111] G. Singh, N. Kumar, and A. K. Verma, “ANTALG: An Innovative ACO based Routing Algorithm for MANETs,” *Journal of Network and Computer Applications*, vol. 45, pp. 151–167, 2014.
 - [112] H. F. Wedde, M. Farooq, and Y. Zhang, “BeeHive: An efficient fault-tolerant routing algorithm inspired by honey bee behavior,” in *Ant colony optimization and swarm intelligence*, pp. 83–94, Springer, 2004.
 - [113] H.F. Wedde, M. Farooq, T. Pannenbaecker, B. Vogel, C. Mueller, J. Meth, and R. Jeruschkat, “BeeAdHoc: an energy efficient routing algorithm for mobile ad hoc networks inspired by bee behavior,” in *Proceedings of the 7th annual conference on Genetic and evolutionary computation*, pp. 153–160, ACM, 2005.
 - [114] V. Jacobson, “Berkeley tcp evolution from 4.3-tahoe to 4.3-reno,” in *Proceedings of the 18th Internet Engineering Task Force, University of British Columbia, Vancouver, BC*, p. 365, 1990.
 - [115] S. Floyd, A. Gurtov, and T. Henderson, “The NewReno modification to TCP’s fast recovery algorithm,” <https://tools.ietf.org/html/rfc3782>, 2004, [last accessed: December’15].
 - [116] L. S. Brakmo and L. L. Peterson, “TCP Vegas: End to end congestion avoidance on a global Internet,” *IEEE Journal on Selected Areas in Communications*, vol. 13, no. 8, pp. 1465–1480, 1995.
 - [117] S. Mascolo, C. Casetti, M. Gerla, M. Y. Sanadidi, and R. Wang, “TCP westwood: Bandwidth estimation for enhanced transport over wireless links,” in *Proceedings of the 7th annual international conference on Mobile computing and networking*, pp. 287–297, ACM, 2001.
 - [118] C. Fu, *TCP veno: end-to-end congestion control over heterogeneous networks*. Doctoral dissertation, The Chinese University of Hong Kong, 2001.

- [119] D. Leith and R. Shorten, “H-TCP: TCP for high-speed and long-distance networks,” in *Proceedings of PFLDnet*, vol. Feb’16, pp. 1–16, 2004.
- [120] L. Xu, K. Harfoush, and I. Rhee, “Binary increase congestion control (BIC) for fast long-distance networks,” in *Proceedings of the Twenty-third Annual Joint Conference of the IEEE Computer and Communications Societies, INFOCOM*, vol. 4, pp. 2514–2524, IEEE, 2004.
- [121] S. Ha, I. Rhee, and L. Xu, “CUBIC: a new TCP-friendly high-speed TCP variant,” *ACM SIGOPS Operating Systems Review*, vol. 42, no. 5, pp. 64–74, 2008.
- [122] C. Caini and R. Firrincieli, “TCP Hybla: a TCP enhancement for heterogeneous networks,” *International journal of satellite communications and networking*, vol. 22, no. 5, pp. 547–566, 2004.
- [123] T. Kelly, “Scalable TCP: Improving performance in highspeed wide area networks,” *ACM SIGCOMM computer communication Review*, vol. 33, no. 2, pp. 83–91, 2003.
- [124] S. Liu, T. Başar, and R. Srikant, “TCP-Illinois: A loss-and delay-based congestion control algorithm for high-speed networks,” *Performance Evaluation*, vol. 65, no. 6, pp. 417–440, 2008.
- [125] A. Baiocchi, A. P. Castellani, and F. Vacirca, “YeAH-TCP: yet another highspeed TCP,” in *Proceedings of the PFLDnet*, vol. 7, pp. 37–42, 2007.
- [126] R. De Amicis, G. Conti, S. Piffer, and F. Prandi, “Service oriented computing for Ambient Intelligence to support management of transport infrastructures,” *Journal of Ambient Intelligence and Humanized Computing*, vol. 2, no. 3, pp. 201–211, 2011.
- [127] D. J. Cook, J. C. Augusto, and V. R. Jakkula, “Ambient intelligence: Technologies, applications, and opportunities,” *Pervasive and Mobile Computing*, vol. 5, no. 4, pp. 277–298, 2009.
- [128] K. Kim, “A model-driven workflow fragmentation framework for collaborative workflow architectures and systems,” *Journal of Network and Computer Applications*, vol. 35, no. 1, pp. 97–110, 2012.
- [129] J. Choi, Y. Cho, J. Choi, and J. Choi, “A layered middleware architecture for automated robot services,” *International Journal of Distributed Sensor Networks*, vol. Article ID 201063, <http://dx.doi.org/10.1155/2014/201063>, pp. 1–10, 2014.

- [130] J. L. Paunicka, D. E. Corman, and B. R. Mendel, “A CORBA-based middleware solution for UAVs,” in *Proceedings of the Fourth International Symposium on Object-Oriented Real-Time Distributed Computing (ISORC)*, pp. 261–267, IEEE, 2001.
- [131] J. L. Paunicka, B. R. Mendel, and D. E. Corman, “The OCP—an open middleware solution for embedded systems,” in *Proceedings of the American Control Conference*, vol. 5, pp. 3445–3450, IEEE, 2001.
- [132] J. Floch, S. Hallsteinsen, E. Stav, F. Eliassen, K. Lund, and E. Gjorven, “Using architecture models for runtime adaptability,” *IEEE Software*, vol. 23, no. 2, pp. 62–70, 2006.
- [133] E. Pastor, J. Lopez, and P. Royo, “A hardware/software architecture for UAV payload and mission control,” in *Proceedings of the 25th Digital Avionics Systems Conference*, pp. 1–8, IEEE, 2006.
- [134] J. Kramer and J. Magee, “Self-managed systems: an architectural challenge,” in *Proceedings of Future of Software Engineering (FOSE)*, pp. 259–268, IEEE, 2007.
- [135] N. Suri, M. Marcon, R. Quitadamo, M. Rebeschini, M. Arguedas, S. Stabellini, M. Tortonesi, and C. Stefanelli, “An adaptive and efficient peer-to-peer service-oriented architecture for manet environments with agile computing,” in *Proceedings of the Network Operations and Management Symposium Workshops*, pp. 364–371, IEEE, 2008.
- [136] N. Suri, “Dynamic service-oriented architectures for tactical edge networks,” in *Proceedings of the 4th Workshop on Emerging Web Services Technology*, pp. 3–10, ACM, 2009.
- [137] J. López, P. Royo, E. Pastor, C. Barrado, and E. Santamaria, “A middleware architecture for unmanned aircraft avionics,” in *Proceedings of the ACM/IFIP/USENIX international conference on Middleware companion*, pp. 24–29, ACM, 2007.
- [138] E. W. Frew and T. X. Brown, “Networking issues for small unmanned aircraft systems,” *Journal of Intelligent and Robotic Systems*, vol. 54, no. 1-3, pp. 21–37, 2009.
- [139] N. Peña, D. Scarlatti, and A. Ollero, “UAVs integration in the SWIM based architecture for ATM,” *Journal of Intelligent and Robotic Systems*, vol. 54, no. 1-3, pp. 39–59, 2009.

- [140] C. W. Heisey, A. G. Hendrickson, B. J. Chludzinski, R. E. Cole, M. Ford, L. Herbek, M. Ljungberg, Z. Magdum, D. Marquis, and A. Mezhirov, “A Reference Software Architecture to Support Unmanned Aircraft Integration in the National Airspace System,” *Journal of Intelligent & Robotic Systems*, vol. 69, no. 1-4, pp. 41–55, 2013.
- [141] N. Mohamed, J. Al-Jaroodi, I. Jawhar, and S. Lazarova-Molnar, “Middleware requirements for collaborative unmanned aerial vehicles,” in *Proceedings of the International Conference on Unmanned Aircraft Systems (ICUAS)*, pp. 1051–1060, IEEE, 2013.
- [142] E. Marconato, D. Rodrigues, A. Chaves, R. Ramdhany, K. Branco, and G. Coulson, “Unmanned Aerial Vehicles & Service-Oriented Architecture: LARISSA and Knowledge Based Frameworks First Study,” in *Proceedings of the Tenth International Conference on Autonomic and Autonomous Systems (ICAS)*, pp. 5–10, 2014.
- [143] N. Dimitriou, L. Sarakis, D. Loukatos, G. Kormentzas, and C. Skianis, “Vertical handover (VHO) framework for future collaborative wireless networks,” *International Journal of Network Management*, vol. 21, no. 6, pp. 548–564, 2011.
- [144] Z. Deng, C. Ma, and M. Zhu, “A reconfigurable flight control system architecture for Small Unmanned Aerial Vehicles,” in *Proceedings of the International Systems Conference (SysCon)*, pp. 1–4, IEEE, 2012.
- [145] N. Mohamed, J. Al-Jaroodi, I. Jawhar, and S. Lazarova-Molnar, “A Service-Oriented Middleware for Building Collaborative UAVs,” *Journal of Intelligent & Robotic Systems*, vol. 74, no. 1-2, pp. 309–321, 2014.
- [146] G. Chen, E. Blasch, D. Shen, H. Chen, and K. Pham, “Services oriented architecture (SOA)-based persistent ISR simulation system,” in *Proceedings of the SPIE Defense, Security, and Sensing*, pp. 76941D–76941D, International Society for Optics and Photonics, 2010.
- [147] S. Lee, S. Rho, and J. H. Park, “Multimedia contents adaptation by modality conversion with user preference in wireless network,” *Journal of Network and Computer Applications*, vol. 37, pp. 25–32, 2014.
- [148] P. Pattanayak and P. Kumar, “A computationally efficient genetic algorithm for MIMO broadcast scheduling,” *Applied Soft Computing*, vol. 37, pp. 545–553, 2015.
- [149] A. Bagwari, R. Jee, P. Joshi, and S. Bisht, “Performance of AODV Routing Pro-

- TOCOL with increasing the MANET Nodes and its effects on QoS of Mobile Ad hoc Networks,” in *Proceedings of the International Conference on Communication Systems and Network Technologies (CSNT)*, pp. 320–324, IEEE, 2012.
- [150] I. Lahyani, N. Khabou, and M. Jmaiel, “Qos monitoring and analysis approach for publish/subscribe systems deployed on manet,” in *Proceedings of the 20th Euromicro International Conference on Parallel, Distributed and Network-Based Processing (PDP)*, pp. 120–124, IEEE, 2012.
 - [151] Y. Zhang, C. Dai, and M. Song, “A novel QOS guarantee mechanism in IEEE 802.16 mesh networks,” *Computing and informatics*, vol. 29, no. 4, pp. 521–536, 2012.
 - [152] H. Jeong, J. Lim, J. Park, and H. Choo, “Forwarding Based Data Parallel Hand-off for Real-Time QoS in Mobile IPv6 Networks,” *Computational Science-ICCS*, pp. 266–273, 2004.
 - [153] J. F. Lee, M. C. Chen, and Y. Sun, “WF 2 QM: Worst-case fair weighted fair queueing with maximum rate control,” *Computer Networks*, vol. 51, no. 6, pp. 1403–1420, 2007.
 - [154] R. Geng, L. Guo, and X. Wang, “A new adaptive MAC protocol with QoS support based on IEEE 802.11 in ad hoc networks,” *Computers & Electrical Engineering*, vol. 38, no. 3, pp. 582–590, 2012.
 - [155] K. McLaughlin, S. Sezer, H. Blume, X. Yang, F. Kupzog, and T. Noll, “A scalable packet sorting circuit for high-speed WFQ packet scheduling,” *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 16, no. 7, pp. 781–791, 2008.
 - [156] M. Shreedhar and G. Varghese, “Efficient fair queuing using deficit round-robin,” *IEEE/ACM Transactions on Networking*, vol. 4, no. 3, pp. 375–385, 1996.
 - [157] G. Varaprasad and R. Wahidabanu, “Quality of service model for multimedia applications in a mobile ad hoc network,” *IEEE Potentials*, vol. 30, no. 2, pp. 44–47, 2011.
 - [158] Y. Sun, C. Phillips, S. Wang, and J. Bai, “An improved QoS awareness scheduling scheme for CR Mobile Ad hoc Networks,” in *Wireless Telecommunications Symposium (WTS)*, pp. 1–6, IEEE, 2013.

- [159] Y. Peng, Y. Yu, L. Guo, D. Jiang, and Q. Gai, "An efficient joint channel assignment and QoS routing protocol for IEEE 802.11 multi-radio multi-channel wireless mesh networks," *Journal of Network and Computer Applications*, vol. 36, no. 2, pp. 843–857, 2013.
- [160] P. Sondi, D. Gantsou, and S. Lecomte, "Design guidelines for quality of service support in Optimized Link State Routing-based mobile ad hoc networks," *Ad Hoc Networks*, vol. 11, no. 1, pp. 298–323, 2013.
- [161] J.-L. Kuo, C.-H. Shih, C.-Y. Ho, and Y.-C. Chen, "A cross-layer approach for real-time multimedia streaming on wireless peer-to-peer ad hoc network," *Ad Hoc Networks*, vol. 11, no. 1, pp. 339–354, 2013.
- [162] S. H. Bouk, I. Sasase, S. H. Ahmed, and N. Javaid, "Gateway discovery algorithm based on multiple QoS path parameters between mobile node and gateway node," *Journal of Communications and Networks*, vol. 14, no. 4, pp. 434–442, 2012.
- [163] F. R. Yu, H. Tang, S. Bu, and D. Zheng, "Security and quality of service (QoS) co-design in cooperative mobile ad hoc networks," *EURASIP Journal on Wireless Communications and Networking*, vol. 2013, no. 1, pp. 1–14, 2013.
- [164] Y. Wen, B. Jin, K. Li, A. M. Cheng, and W. Fang, "A Queueing Theory Based Approach to QoS-Driven Adaptation for Service Discovery over MANETs," in *Proceedings of the 15th International Conference on Computational Science and Engineering (CSE)*, pp. 594–601, IEEE, 2012.
- [165] Y. Yang, X. Qiu, L. Meng, and K. Long, "Task coalition formation and self-adjustment in the wireless sensor networks," *International Journal of Communication Systems*, vol. 27, no. 10, pp. 2241–2254, 2014.
- [166] V. Sharma and R. Kumar, "A Cooperative Network Framework for Multi-UAV Guided Ground Ad Hoc Networks," *Journal of Intelligent & Robotic Systems*, vol. 77, no. 3-4, pp. 629–652, 2015.
- [167] J.-S. R. Jang, "Self-learning fuzzy controllers based on temporal backpropagation," *IEEE Transaction on Neural Networks*, vol. 3, no. 5, pp. 714–723, 1992.
- [168] D. H. Nguyen and B. Widrow, "Neural networks for self-learning control systems," *IEEE Control Systems Magazine*, vol. 10, no. 3, pp. 18–23, 1990.

- [169] Z. Ni, H. He, and J. Wen, "Adaptive learning in tracking control based on the dual critic network design," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 24, no. 6, pp. 913–928, 2013.
- [170] H. Zhang, X. Xiao, and O. Hasegawa, "A Load-Balancing Self-Organizing Incremental Neural Network," *IEEE Transaction on Neural Networks and Learning Systems*, vol. 25, no. 6, pp. 1096–1105, 2014.
- [171] G. Chen and Y.-D. Song, "Cooperative Tracking Control of Nonlinear Multiagent Systems Using Self-Structuring Neural Networks," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 25, no. 8, pp. 1496–1507, 2014.
- [172] Q. Wei, D. Liu, and X. Yang, "Infinite horizon self-learning optimal control of nonaffine discrete-time nonlinear systems," *IEEE Transaction on Neural Networks and Learning Systems*, vol. 26, no. 4, pp. 866–879, 2015.
- [173] D. Liu, Y. Zhang, and H. Zhang, "A self-learning call admission control scheme for CDMA cellular networks," *IEEE Transaction on Neural Networks*, vol. 16, no. 5, pp. 1219–1228, 2005.
- [174] M. Eriksson, M. Armendariz, O. O. Vasilenko, A. Saleem, and L. Nordstrom, "Multiagent-Based Distribution Automation Solution for Self-Healing Grids," *IEEE Transaction on Industrial Electronics*, vol. 62, no. 4, pp. 2620–2628, 2015.
- [175] O. K. Tonguz, N. Wisitpongphan, J. S. Parikh, F. Bai, P. Mudalige, and V. K. Sadekar, "On the broadcast storm problem in ad hoc wireless networks," in *Proceedings of the 3rd International Conference on Broadband Communications, Networks and Systems*, pp. 1–11, IEEE, 2006.
- [176] Y.-C. Tseng, S.-Y. Ni, and E.-Y. Shih, "Adaptive approaches to relieving broadcast storms in a wireless multihop mobile ad hoc network," *IEEE Transactions on Computers*, vol. 52, no. 5, pp. 545–557, 2003.
- [177] N. Wisitpongphan, O. Tonguz, J. S. Parikh, P. Mudalige, F. Bai, and V. Sadekar, "Broadcast storm mitigation techniques in vehicular ad hoc networks," *IEEE Wireless Communications*, vol. 14, no. 6, pp. 84–94, 2007.
- [178] A. M. Vegni and E. Natalizio, "Forwarder smart selection protocol for limitation of broadcast storm problem," *Journal of Network and Computer Applications*, vol. 47, pp. 61–71, 2015.

- [179] Z. J. Haas, J. Y. Halpern, and L. Li, "Gossip-based ad hoc routing," *IEEE/ACM Transactions on Networking (ToN)*, vol. 14, no. 3, pp. 479–491, 2006.
- [180] A. Wegener, H. Hellbrück, S. Fischer, C. Schmidt, and S. Fekete, "AutoCast: An adaptive data dissemination protocol for traffic information systems," in *Proceedings of the 66th Vehicular Technology Conference VTC-Fall*, pp. 1947–1951, IEEE, 2007.
- [181] A. Khelil, P. J. Marrón, C. Becker, and K. Rothermel, "Hypergossiping: A generalized broadcast strategy for mobile ad hoc networks," *Ad Hoc Networks*, vol. 5, no. 5, pp. 531–546, 2007.
- [182] C.-C. Shen, Z. Huang, and C. Jaikaeo, "Directional broadcast for mobile ad hoc networks with percolation theory," *IEEE Transactions on Mobile Computing*, vol. 5, no. 4, pp. 317–332, 2006.
- [183] X. M. Zhang, E. B. Wang, J. J. Xia, and D. K. Sung, "A neighbor coverage-based probabilistic rebroadcast for reducing routing overhead in mobile ad hoc networks," *IEEE Transactions on Mobile Computing*, vol. 12, no. 3, pp. 424–433, 2013.
- [184] O. K. Tonguz, N. Wisitpongphan, and F. Bai, "DV-CAST: A distributed vehicular broadcast protocol for vehicular ad hoc networks," *IEEE Wireless Communications*, vol. 17, no. 2, pp. 47–57, 2010.
- [185] Y. Song and J. Xie, "BRACER: A distributed broadcast protocol in multi-hop cognitive radio ad hoc networks with collision avoidance," *IEEE Transactions on Mobile Computing*, vol. 14, no. 3, pp. 509–524, 2015.
- [186] W. Peng and X. Lu, "AHBP: An efficient broadcast protocol for mobile ad hoc networks," *Journal of computer science and technology*, vol. 16, no. 2, pp. 114–125, 2001.
- [187] A. Touzene and A. Alkathiri, "Performance analysis of an extended grid based broadcast algorithm in mobile ad-hoc networks," *Wireless Networks*, vol. 21, no. 2, pp. 659–672, 2015.
- [188] Y.-C. Tseng, S.-Y. Ni, Y.-S. Chen, and J.-P. Sheu, "The broadcast storm problem in a mobile ad hoc network," *Wireless networks*, vol. 8, no. 2-3, pp. 153–167, 2002.
- [189] W. Peng and X.-C. Lu, "On the reduction of broadcast redundancy in mobile ad

- hoc networks,” in *Proceedings of the 1st ACM international symposium on Mobile ad hoc networking & computing*, pp. 129–130, IEEE Press, 2000.
- [190] D. Reina, S. Toral, P. Johnson, and F. Barrero, “Improving discovery phase of reactive ad hoc routing protocols using Jaccard distance,” *The Journal of Supercomputing*, vol. 67, no. 1, pp. 131–152, 2014.
 - [191] D. Reina, S. Toral, P. Jonhson, and F. Barrero, “Hybrid flooding scheme for mobile ad hoc networks,” *IEEE Communications Letters*, vol. 17, no. 3, pp. 592–595, 2013.
 - [192] A. Mostafa, A. M. Vegni, and D. P. Agrawal, “A probabilistic routing by using multi-hop retransmission forecast with packet collision-aware constraints in vehicular networks,” *Ad Hoc Networks*, vol. 14, pp. 118–129, 2014.
 - [193] T.-C. Huang, S.-C. Chen, and L. Tang, “Energy-Aware Gossip Routing for Mobile Ad Hoc Networks,” in *Proceedings of the 13th International Conference on High Performance Computing and Communications (HPCC)*, pp. 955–959, IEEE, 2011.
 - [194] I. S. Lysiuk and Z. J. Haas, “Controlled gossiping in ad hoc networks,” in *Proceedings of the Wireless Communications and Networking Conference (WCNC)*, pp. 1–6, IEEE, 2010.
 - [195] T. Acharya and G. Paul, “Maximum lifetime broadcast communications in cooperative multihop wireless ad hoc networks: Centralized and distributed approaches,” *Ad Hoc Networks*, vol. 11, no. 6, pp. 1667–1682, 2013.
 - [196] T. Booth and K. Andersson, “Network Security of Internet Services: Eliminate DDoS Reflection Amplification Attacks,” *Journal of Internet Services and Information Security (JISIS)*, vol. 5, no. 3, pp. 58–79, 2015.
 - [197] A. Cizmar, J. Papaj, and L. Dobos, “Security and QoS integration model for MANETS,” *Computing and Informatics*, vol. 31, no. 5, pp. 1025–1044, 2012.
 - [198] Z. Qian, C. Chen, I. You, and S. Lu, “ACSP: A novel security protocol against counting attack for UHF RFID systems,” *Computers & Mathematics with Applications*, vol. 63, no. 2, pp. 492–500, 2012.
 - [199] I. Maza, F. Caballero, J. Capitan, J. Martinez-de Dios, and A. Ollero, “A distributed architecture for a robotic platform with aerial sensor transportation and self-deployment capabilities,” *Journal of Field Robotics*, vol. 28, no. 3, pp. 303–328,

2011.

- [200] I. Maza, F. Caballero, J. Capitán, J. Martínez-de Dios, and A. Ollero, “Experimental results in multi-UAV coordination for disaster management and civil security applications,” *Journal of intelligent & robotic systems*, vol. 61, no. 1-4, pp. 563–585, 2011.
- [201] Q. Wang, J. Wang, J. Yu, M. Yu, and Y. Zhang, “Trust-aware query routing in P2P social networks,” *International Journal of Communication Systems*, vol. 25, no. 10, pp. 1260–1280, 2012.
- [202] H.-C. Chen and H.-Y. Chuang, “A three-party encrypted key exchange protocol with protected password authentication,” in *Proceedings of the International Conference on Frontier Computing: Theory, Technologies and Applications*, pp. 275–280, IET, 2010.
- [203] R. S. Yokoyama, B. Y. L. Kimura, and E. dos Santos Moreira, “An architecture for secure positioning in a UAV swarm using RSSI-based distance estimation,” *ACM SIGAPP Applied Computing Review*, vol. 14, no. 2, pp. 36–44, 2014.
- [204] D. Rodrigues, A. Chaves, K. R. Branco, R. Ramdhany, and G. Coulson, “Knowledge based framework: a case study on fast, reliable, and secure web services in UAVs,” in *Proceedings of 11th International Conference on Information Technology: New Generations (ITNG)*, pp. 103–108, IEEE, 2014.
- [205] Z. Birnbaum, A. Dolgikh, V. Skormin, E. O’Brien, and D. Muller, “Unmanned Aerial Vehicle security using Recursive parameter estimation,” in *International Conference on Unmanned Aircraft Systems (ICUAS)*, pp. 692–702, IEEE, 2014.
- [206] E. Mammadov and W. Gueaieb, “Long-range communication framework for multi-agent autonomous UAVs,” in *International Conference on Unmanned Aircraft Systems (ICUAS)*, pp. 395–403, IEEE, 2014.
- [207] O. K. Sahingoz, “Large scale wireless sensor networks with multi-level dynamic key management scheme,” *Journal of Systems Architecture*, vol. 59, no. 9, pp. 801–807, 2013.
- [208] G. C. Hadjichristofi, L. A. DaSilva, S. F. Midkiff, U. Lee, and W. De Sousa, “Routing, security, resource management, and monitoring in ad hoc networks: Implementation and integration,” *Computer Networks*, vol. 55, no. 1, pp. 282–299, 2011.

- [209] M. Friedewald, E. Vildjiounaite, Y. Punie, and D. Wright, “Privacy, identity and security in ambient intelligence: a scenario analysis,” *Telematics and Informatics*, vol. 24, no. 1, pp. 15–29, 2007.
- [210] J. Kong, X. Hong, and M. Gerla, “An identity-free and on-demand routing scheme against anonymity threats in mobile ad hoc networks,” *Mobile Computing, IEEE Transactions on*, vol. 6, no. 8, pp. 888–902, 2007.
- [211] P. Paruchuri, M. Tambe, F. Ordóñez, and S. Kraus, “Security in multiagent systems by policy randomization,” in *Proceedings of the fifth international joint conference on Autonomous agents and multiagent systems*, pp. 273–280, ACM, 2006.
- [212] P. Paruchuri, J. P. Pearce, M. Tambe, F. Ordonez, and S. Kraus, “An efficient heuristic approach for security against multiple adversaries,” in *Proceedings of the 6th international joint conference on Autonomous agents and multiagent systems*, pp. 311–318, ACM, 2007.
- [213] J. Kong, H. Luo, K. Xu, D. L. Gu, M. Gerla, and S. Lu, “Adaptive security for multilevel ad hoc networks,” *Wireless Communications and Mobile Computing*, vol. 2, no. 5, pp. 533–547, 2002.
- [214] P. Goyal, N. Goyal, R. S. Karwa, and M. John, “Phase-Wise Clustering of Time Series Gene Expression Data,” in *Proceedings of the 10th International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom)*, pp. 1668–1674, IEEE, 2011.
- [215] S.-W. Kim and S.-W. Seo, “Cooperative unmanned autonomous vehicle control for spatially secure group communications,” *IEEE Journal on Selected Areas in Communications*, vol. 30, no. 5, pp. 870–882, 2012.
- [216] P. Freeman, R. Pandita, N. Srivastava, and G. J. Balas, “Model-based and data-driven fault detection performance for a small UAV,” *IEEE/ASME Transactions on Mechatronics*, vol. 18, no. 4, pp. 1300–1309, 2013.
- [217] F. Luo, C. Jiang, J. Du, J. Yuan, Y. Ren, S. Yu, and M. Guizani, “A Distributed Gateway Selection Algorithm for UAV Networks,” *IEEE Transactions on Emerging Topics in Computing*, vol. 3, no. 1, pp. 22–33, 2015.
- [218] A. Naghizadeh, S. Berenjian, E. Meamari, and R. E. Atani, “Structural-based tunneling: preserving mutual anonymity for circular P2P networks,” *International*

Journal of Communication Systems, 2015.

- [219] W. Aye and M. U. Siddiqi, “Key management for secure multicast over IPv6 wireless networks,” *EURASIP Journal on wireless communications and networking*, vol. 2006, no. 2, pp. 75–75, 2006.
- [220] F. A. G. Muzzi, P. R. de Mello Cardoso, D. F. Pigatto, and K. R. L. J. C. Branco, “Using Botnets to provide security for safety critical embedded systems-a case study focused on UAVs,” in *Proceedings of the Journal of Physics: Conference Series*, vol. 633(1), p. 012053, IOP Publishing, 2015.
- [221] Y. Li, W. Niu, P. Li, J. Ma, and Y. Shen, “Secure Networking Protocol with Identity Protection for Cooperation of Unmanned Platforms,” in *Proceedings of the Tenth International Conference on Computational Intelligence and Security (CIS)*, pp. 603–607, IEEE, 2014.
- [222] A. N. Phillips, B. E. Mullins, R. A. Raines, and R. O. Baldwin, “A secure group communication architecture for autonomous unmanned aerial vehicles,” *Security and Communication Networks*, vol. 2, no. 1, pp. 55–69, 2009.
- [223] R. J. Meinhold and N. D. Singpurwalla, “Understanding the Kalman filter,” *The American Statistician*, vol. 37, no. 2, pp. 123–127, 1983.
- [224] N. Trawny and S. I. Roumeliotis, “Indirect Kalman filter for 3D attitude estimation,” *University of Minnesota, Dept. of Comp. Sci. & Eng., Tech. Rep.*, vol. 2, pp. 1–25, 2005.
- [225] G. A. Carpenter and S. Grossberg, “ART 2: Self-organization of stable category recognition codes for analog input patterns,” *Applied optics*, vol. 26, no. 23, pp. 4919–4930, 1987.
- [226] L. Breiman, “Random forests,” *Machine learning*, vol. 45, no. 1, pp. 5–32, 2001.
- [227] C. E. Perkins and P. Bhagwat, “Highly dynamic destination-sequenced distance-vector routing (DSDV) for mobile computers,” in *ACM SIGCOMM computer communication review*, vol. 24(4), pp. 234–244, ACM, 1994.
- [228] C. Perkins, E. Belding-Royer, and S. Das, “Ad hoc on-demand distance vector (AODV) routing,” in *Proceedings of Mobile Computing Systems and Applications Conference*, pp. 90–100, 1999.

- [229] C. Perkins, E. Belding-Royer, and S. Das, “Ad hoc on-demand distance vector (AODV) routing,” tech. rep., No. RFC 3561, 2003.
- [230] D. B. J. D. A. Maltz and J. Broch, “DSR: The dynamic source routing protocol for multi-hop wireless ad hoc networks,” *Computer Science Department Carnegie Mellon University Pittsburgh, PA*, pp. 15213–3891, 2001.
- [231] Y. Yuan, H. Chen, and M. Jia, “An optimized ad-hoc on-demand multipath distance vector (AOMDV) routing protocol,” in *Proceedings of the Asia-Pacific Conference on Communications*, pp. 569–573, IEEE, 2005.
- [232] M. Caleffi and L. Paura, “M-DART: multi-path dynamic address routing,” *Wireless communications and mobile computing*, vol. 11, no. 3, pp. 392–409, 2011.
- [233] R. Vaishampayan and J. J. Garcia-Luna-Aceves, “Efficient and robust multicast routing in mobile ad hoc networks,” in *Proceedings of the International Conference on Mobile Ad-hoc and Sensor Systems*, pp. 304–313, IEEE, 2004.
- [234] V. D. Park and M. Corson, “A performance comparison of the temporally-ordered routing algorithm and ideal link-state routing,” in *Proceedings of the Third IEEE Symposium on Computers and Communications*, pp. 592–598, IEEE, 1998.
- [235] Y. Zhu and T. Kunz, “MAODV implementation for NS-2.26,” *Systems and Computing Engineering, Carleton University, Technical Report SCE-04-01*, 2004.
- [236] E. Santamaria, P. Royo, J. Lopez, C. Barrado, E. Pastor, and X. Prats, “Increasing UAV capabilities through autopilot and flight plan abstraction,” in *Proceedings of the 26th Conference on the Digital Avionics Systems*, pp. 5.B.1–5.B.10, IEEE, 2007.
- [237] W. Tan, Y. Fan, M. Zhou, and Z. Tian, “Data-driven service composition in enterprise SOA solutions: a Petri net approach,” *IEEE Transactions on Automation Science and Engineering*, vol. 7, no. 3, pp. 686–694, 2010.
- [238] J. Al-Jaroodi and N. Mohamed, “Service-oriented middleware: a survey,” *Journal of Network and Computer Applications*, vol. 35, no. 1, pp. 211–220, 2012.
- [239] D. Rodrigues, C. JJB, B. KRC, and T. JO, “Service-oriented architectures for complex safety-critical embedded systems: A case study on UAVs,” in *Proceedings of the São Carlos, I Brazilian Conference on Critical-Embedded Systems (CBSEC)*, pp. 130–130, 2011.

- [240] M. Conti, G. Maselli, G. Turi, and S. Giordano, "Cross-layering in mobile ad hoc network design," *Computer*, vol. 37, no. 2, pp. 48–51, 2004.
- [241] H. Gossain, T. Joshi, C. Cordeiro, and D. P. Agrawal, "A cross-layer approach for designing directional routing protocol in MANETs," in *Proceedings of the Wireless Communications and Networking Conference*, vol. 4, pp. 1976–1981, IEEE, 2005.
- [242] A. Alhosainy, T. Kunz, L. Li, and P. J. Vigneron, "Cross-layer design gains in MANETs," in *Proceedings of the 13th Annual Mediterranean Ad Hoc Networking Workshop (MED-HOC-NET)*, pp. 8–14, IEEE, 2014.
- [243] G. Adam, C. Bouras, A. Gkamas, V. Kapoulas, and G. Kioumourtzis, "Cross layer design for video streaming in MANETs," *Journal of Networks*, vol. 9, no. 2, pp. 328–338, 2014.
- [244] L. Chen and W. B. Heinzelman, "QoS-aware routing based on bandwidth estimation for mobile ad hoc networks," *IEEE Journal on Selected Areas in Communications*, vol. 23, no. 3, pp. 561–572, 2005.
- [245] V. Sharma, R. Kumar, and P. Patiala, "Service-Oriented Middleware for Multi-UAV Guided Ad Hoc Networks," *IT CoNvergence PRActive (INPRA)*, vol. 2, no. 3, pp. 24–33, 2014.
- [246] V. Sharma and R. Kumar, "An opportunistic cross layer design for efficient service dissemination over flying ad hoc networks (FANETs)," in *Proceedings of the 2nd International Conference on Electronics and Communication Systems (ICECS)*, pp. 1551–1557, IEEE, 2015.
- [247] A. Patil, A.-H. Esfahanian, L. Xiao, and Y. Liu, "Resource allocation using multiple edge-sharing multicast trees," in *Proceedings of the International Conference on Mobile Adhoc and Sensor Systems*, pp. 8–pp, IEEE, 2005.
- [248] C. Gomathy and S. Shanmugavel, "Supporting QoS in MANET by a fuzzy priority scheduler and performance analysis with multicast routing protocols," *EURASIP journal on wireless communications and networking*, vol. 2005, no. 3, pp. 426–436, 2005.
- [249] R. Faragher, "Understanding the basis of the Kalman filter via a simple and intuitive derivation," *IEEE Signal processing magazine*, vol. 29, no. 5, pp. 128–132, 2012.

- [250] B. A. Olshausen, “Bayesian probability theory,” *The Redwood Center for Theoretical Neuroscience, Helen Wills Neuroscience Institute at the University of California at Berkeley, Berkeley, CA*, pp. 1–6, 2004.
- [251] J. A. Martilla and J. C. James, “Importance-performance analysis,” *The journal of marketing*, vol. 41, no. 1, pp. 77–79, 1977.
- [252] W.-C. Wang, Y.-H. Lin, C.-L. Lin, C.-H. Chung, and M.-T. Lee, “DEMATEL-based model to improve the performance in a matrix organization,” *Expert Systems with Applications*, vol. 39, no. 5, pp. 4978–4986, 2012.
- [253] J. Figueiras and S. Frattasi, *Mobile positioning and tracking: from conventional to cooperative techniques*. John Wiley & Sons, ISBN: 9781119957560, 2011.
- [254] D. Tenne and T. Singh, “Optimal design of α - β - γ (γ) filters,” in *Proceedings of the American Control Conference*, vol. 6, pp. 4348–4352, IEEE, 2000.
- [255] J. Hoagland, “The Teredo Protocol: Tunneling Past Network Security and Other Security Implications, Symantec,” *Symantec Advanced Threat Research*, doi=10.1.1.114.1082, pp. 1–35, 2007.
- [256] S. Zander, L. L. Andrew, G. Armitage, G. Huston, and G. Michaelson, “Investigating the IPv6 teredo tunnelling capability and performance of internet clients,” *ACM SIGCOMM Computer Communication Review*, vol. 42, no. 5, pp. 13–20, 2012.
- [257] A. Ludwig and K. D. Schmidt, “Gauss–Markov loss prediction in a linear model,” in *Casualty Actuarial Society E-Forum*, vol. fall 2010, pp. 1–48, 2010.
- [258] R. Jain, *The art of computer systems performance analysis*. John Wiley & Sons, ISBN: 9788126519057, 2008.

List of Publications

Refereed Journals: Published/Accepted

- 1) Vishal Sharma and Rajesh Kumar, “A Cooperative Network Framework For Multi-UAV Guided Ground Ad Hoc Networks”, Journal of Intelligent Robotic Systems, Springer- SCIE Indexed, 1;77(3-4), 2015, pp. 629-652, [I.F. 1.178].
- 2) Vishal Sharma, Rajesh Kumar and Prashant Singh Rana, “A Self-Healing Neural Model For Stabilization Against Network Failures Over Networked UAVs”, IEEE Communication Letters, IEEE- SCIE Indexed, 19(11), 2015, pp. 2013-2016, [I.F. 1.268].
- 3) Vishal Sharma and Rajesh Kumar, “Estimation-Based Queue Scheduling Model To Improve QoS For End Users In MANETs”, Computing and Informatics- SCIE Indexed [accepted], [I.F. 0.504].
- 4) Vishal Sharma, Ilsun You, Rajesh Kumar and Varun Chauhan, “OFFRP: Optimized Fruit Fly Based Routing Protocol With Congestion Control For UAVs Guided Ad Hoc Networks”, International Journal of Ad Hoc and Ubiquitous Computing, Inderscience- SCIE Indexed [accepted], [I.F. 0.554].
- 5) Vishal Sharma, Rajesh Kumar and Neeraj Kumar Rathore, “Topological Broadcasting Using Parameter Sensitivity-Based Logical Proximity Graphs in Coordinated Ground-Flying Ad Hoc Networks” Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications (JoWUA)- Scopus Indexed, 6(3), 2015, pp. 54-72.
- 6) Vishal Sharma and Rajesh Kumar, “Teredo Tunneling Based Secure Transmission Between UAVs and Ground Ad Hoc Networks”, International Journal of Communication Systems, Wiley- SCIE Indexed, DOI: 10.1002/dac.3144, May 2016, [I.F. 1.106].

Refereed Journals: Under Revision/Review

- 7) Vishal Sharma and Rajesh Kumar, “3-Tier Neural Model For Service Provisioning Over Collaborative Flying Ad Hoc Networks”, Neural Computing and Applications, Springer- SCIE Indexed,[Revision], [I.F. 1.569].
- 8) Vishal Sharma and Rajesh Kumar, “Cooperative Frameworks and Network Models For UAVs Ad Hoc Networks: A Survey”, Concurrency and Computation: Practice and Experience, Wiley- SCIE Indexed,[Revision], [I.F. 0.997].
- 9) Vishal Sharma, and Rajesh Kumar, “UAVs Assisted Queue Scheduling in Ground Ad Hoc Networks”, International Journal of Ad Hoc and Ubiquitous Computing, Inderscience- SCIE Indexed,[Under Review], [I.F. 0.554].

Conferences

- 10) Vishal Sharma and Rajesh Kumar, “An Opportunistic Cross Layer Design for Efficient Service Dissemination over Flying Ad Hoc Networks (FANETs)”, in: Proceedings of 2nd IEEE ICECS, February 26-27, 2015, Coimbatore, India, pp. 1551-1557.

Posters

- 11) Vishal Sharma and Rajesh Kumar, “A Cooperative Network Framework For Multi-UAVs Guided Ground Ad Hoc Networks”, Poster presented at 9th IRISS’15, ACM India Annula Event, February 4-7, 2015, BITS Goa-India [Amongst top 9 Posters in Network Category].
- 12) Vishal Sharma and Rajesh Kumar, “Cross Layer Design For Secure Service Dissemination Over Flying Ad Hoc Networks”, Poster presented at 3rd ACM Security and Privacy Symposium, February 13-14, 2015 at IIIT Delhi, India.