



HAL
open science

Multi-hop routing in a drone network for road surveillance

Nouman Bashir

► **To cite this version:**

Nouman Bashir. Multi-hop routing in a drone network for road surveillance. Systems and Control [cs.SY]. Université Paris-Nord - Paris XIII, 2022. English. NNT : 2022PA131011 . tel-04332249

HAL Id: tel-04332249

<https://theses.hal.science/tel-04332249>

Submitted on 8 Dec 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

THÈSE

pour obtenir le grade de

Docteur de l'Université Sorbonne Paris Nord

Discipline : "Doctorat de sciences pour l'ingénieur"

présentée et soutenue publiquement par

Nouman Bashir

le 31 Mai 2022

Routage multi-sauts dans un réseau de drones pour la surveillance routière

Directeur de thèse : **Dr. Saadi Boudjit**

JURY

Samia Bouzefrane,	Professeur, Laboratoire CEDRIC - CNAM	Rapporteur
Anthony Busson,	Professeur, Université Claude-Bernard-Lyon-I	Rapporteur
Saadi Boudjit,	MCF-HDR, Université Sorbonne Paris Nord	Directeur de thèse
Mohand Yazid Saidi,	MCF, Université Sorbonne Paris Nord	Examineur
Gabriel Dauphin,	MCF, Université Sorbonne Paris Nord	Examineur
Nadjib Ait Saadi,	Professeur, Université Paris-Saclay / UVSQ Campus	Examineur
Nathalie Mitton,	Directeur de recherche, Inria Lille-Nord Europe	Examineur
M. Dias de Amorim,	Directeur de recherche, CNRS	Examineur
Sherali Zeadally,	Professeur, University of Kentucky, USA	Membre invité

I dedicate my research work wholeheartedly

To my parents,
specifically to my father,
who died during my doctoral studies.

They have always remained a great source
of love, support, and inspiration for me.

Résumé

Les systèmes de surveillance routiers existants sont soit trop coûteux, comme les hélicoptères, en raison de leurs coûts d'exploitation et de maintenance élevés, soit inadéquats pour faire face à la nature dynamique du trafic sur les autoroutes, comme les radars routiers, en raison de leur couverture limitée et des contraintes de mobilité. Des drones autonomes appelés communément UAVs (Unmanned Aerial Vehicles) et dotés de capacités de prise de vues efficaces peuvent être une solution d'avenir pour des systèmes de surveillance de nouvelle génération. Les drones peuvent fonctionner de façon collaborative en mode ad hoc pour contrer les ressources énergétiques limitées et les contraintes de portée de transmission associées à l'utilisation d'un seul drone. Cependant, la mobilité indépendante des drones et la difficulté de maintien des liens dans une topologie fortement dynamique nécessitent des solutions de routage adéquates et tolérantes aux pannes. De plus, pour un déploiement d'une flotte de drones dans un environnement urbain avec des incertitudes du système de navigation et des problèmes de sécurité qui en découlent, un processus efficace de planification de trajectoire est nécessaire. Dans cette thèse, nous avons exploré les possibilités de faire collaborer des drones avec d'autres architectures pour proposer des solutions de surveillance routière de bout en bout efficaces. Dans un premier temps, une solution de routage pour une flotte de drones est proposée afin d'améliorer la connectivité du réseau en présence de plusieurs stations de base (BS) au sol. Cette solution est améliorée en utilisant les services d'un réseau véhiculaire ad hoc (VANET) dans le cas de non-disponibilité d'une station de base au sol à proximité du réseau de drones. L'approche ainsi proposée de la récupération anticipée d'une perte de connectivité avec les stations de base au sol, assure une meilleure continuité de transmission de bout en bout en prenant en compte les pertes de liaisons dues à la topologie fortement dynamique du réseau. La deuxième partie présente un travail collaboratif entre des drones et des capteurs sans fil déployés au sol pour améliorer la connectivité et l'efficacité de la surveillance sur une autoroute. Le réseau de capteurs est l'épine dorsale de cette architecture et fournit des services de routage aux drones en plus de les guider dynamiquement vers le point de surveillance où le taux d'infractions est le plus élevé. Ce travail suit une solution de routage inter-couches pour augmenter la durée de vie du réseau de

capteurs sans fil déployé (WSN) en réduisant le taux de collisions des paquets le long d'un chemin. La dernière partie de cette thèse concerne la planification des trajectoires des drones dans un environnement urbain pour assurer respectivement, la connectivité du réseau et la sécurité de la flotte de drones en évitant les collisions.

Mots Clés— Réseaux collaboratifs, routage tolérant aux pannes, surveillance des autoroutes, récupération de perte de lien, routage multi-sauts, planification de chemins, Réseaux de drones, réseaux de véhicules, réseaux de capteurs.



Abstract

Existing road surveillance systems are either too expensive such as helicopters, due to their high operating and maintenance costs, or inadequate to tackle the dynamic nature of traffic on highways such as roadside traffic radars due to their limited coverage and mobility constraints. Unmanned Aerial Vehicles (UAVs) with excellent maneuvering skills have a promising future for next-generation surveillance systems. UAVs work together in an ad-hoc manner to counter limited energy resources and lower transmission range constraints associated with a single UAV. However, multiple UAVs with dynamic topology suffer frequent link failures and require specific fault-tolerant routing solutions. In addition to this, a robust path planning process is inevitable as the deployment of a UAV fleet in an urban environment with navigation system uncertainties raises safety concerns.

To this end, instead of restricting solely to UAVs, this thesis explores UAV's collaboration with other architectures to come up with robust end-to-end surveillance solutions.

At first, a less overhead bearing routing solution is presented for a fleet of UAVs to enhance backhaul connectivity in the presence of multiple Base Stations (BS). This work is enhanced to seek services of a Vehicular Ad hoc Network (VANET) during the non-availability of a nearby UAV to compensate for link failures. The proposed anticipatory network recovery approach ensures a better end-to-end link continuity by taking care of link failures arising due to the permanent displacement of a UAV.

The second part presents a collaborative work of UAV and Wireless Sensor Network (WSN) to enhance connectivity and surveillance efficiency on a highway. WSN is the backbone of the network and provides routing services to the UAV besides dynamically guiding it to target the best hotspot for catching maximum speed violations. This work follows a cross-layer routing solution to increase WSN lifetime by reducing the chances of collisions along a path.

The last part of the thesis introduces connectivity-aware and collision-free UAV path planning in an urban environment to ensure backhaul connectivity and UAV fleet safety, respectively.

Keywords— Collaborative networks, Fault-tolerant routing, highway surveillance, link failure recovery, multi-hop routing, UAVs, UAV's path planning, VANETs, WSN.



Contents

Résumé	v
Abstract	vii
Contents	ix
List of Tables	xv
List of Figures	xvii
Introduction	1
0.1 The necessity of dynamic road surveillance and UAV potentiality	1
0.2 Communication architecture in UAVs networks	2
0.2.1 Single-hop communication	2
0.2.2 Multi-hop communication	3
0.3 Path planning and formation control of a UAV fleet	4
0.4 Thesis purpose and manuscript organization	5
0.4.1 Part I - Routing protocols and failure recovery approaches for a fleet of UAVs	6
0.4.2 Part II - UAV and WSN collaboration to enhance connectivity and surveil-	
lance efficiency	7
0.4.3 Part III - UAVs path planning techniques in an urban environment	7
0.5 List of publications	8
I Routing protocols and failure recovery approaches for a fleet of UAVs	11
1 A routing protocol for a fleet of UAVs under multiple base stations deployment	13
1.1 Introduction	15
1.2 Related work	16

1.3	Proposed proactive UAV routing protocol	18
1.3.1	Broadcasting of DNA messages	18
1.3.2	Route establishment to BS	20
1.4	Performance evaluation	21
1.4.1	Variation in data rate and deployed BSs	22
1.4.1.1	Data Packet Delivery Ratio (PDR)	22
1.4.1.2	Energy Consumed per Packet Delivery (ECPD)	22
1.4.1.3	Average delay incurred by data packets	23
1.4.2	Variation in the number of transmitting UAVs per fleet	24
1.4.2.1	Data Packet Delivery Ratio (PDR)	24
1.4.2.2	Energy Consumed per Packet Delivery (ECPD)	25
1.4.2.3	Average delay incurred by data packets	25
1.5	Conclusion	26
2	A collaboration framework for UAVs and VANETs to increase UAV's network resilience	29
2.1	Introduction	31
2.2	Related work	32
2.3	Proposed collaborative routing of UAVs and VANETs	33
2.3.0.1	Initialization phase and end-node formation	34
2.3.0.2	BAM propagation and mobility detection mechanism	34
2.3.0.3	Path repair mechanism	36
2.3.0.4	Application data transmission procedure	37
2.4	Performance evaluation	38
2.4.1	Energy consumed per packet delivery	39
2.4.2	Data packet delivery ratio	39
2.4.3	Average delay	40
2.5	Conclusion	41
3	A distributed network recovery approach for unmanned aerial vehicular networks	43
3.1	Introduction	45
3.2	Related work	46
3.3	Proposed network recovery approach	47
3.3.1	Illustration of a UAV network breakdown scenario	47
3.3.2	The proactive routing strategy for a fleet of UAVs	48

3.3.3	Anticipatory network recovery algorithm	49
3.4	Performance evaluation	52
3.4.1	Simulation scenario I - under UAV's electronics or mechanical failures . .	52
3.4.2	Simulation scenario II - under UAV's depletion of energy resources . . .	54
3.4.3	Simulation scenario III - under sudden and announced departures of UAVs	55
3.4.4	Average energy consumption due to communication overhead	56
3.5	Conclusion	58
II	UAV and WSN collaboration to enhance connectivity and surveillance efficiency	59
4	A self-governing collaborative architecture of UAV and WSN for dynamic surveillance	61
4.1	Introduction	63
4.2	Related work	65
4.3	UAV-WSN collaborative system's description	67
4.3.1	Wireless sensor network hierarchy	69
4.3.2	Working of the proposed collaborative surveillance system	69
4.3.3	Probabilistic based trajectory control model for UAV	69
4.3.4	Hop number assignment and immediate reporting mechanism	72
4.4	Performance evaluation	73
4.4.1	Simulation scenario I - hotspot location 2.0 km away from BS	74
4.4.2	Simulation scenario II - hotspot location 3.9 km away from BS	76
4.4.3	Simulation scenario III - hotspot location 5.3 km away from BS	77
4.4.4	Simulation scenario IV - with varying hotspot locations	77
4.5	Conclusion	80
5	A MAC-aware routing protocol for wireless sensor networks	81
5.1	Introduction	83
5.2	Related work	84
5.3	Proposed work	85
5.3.1	Problem formulation with possible collision scenarios	86
5.3.2	Network setup phase	86
5.3.3	Packet scheduling interval calculation	87
5.3.4	Routing path selection criteria	89
5.4	Performance evaluation	89

5.4.1	Data packet delivery ratio	90
5.4.2	Energy consumed per packet delivery	91
5.4.3	Average delay	92
5.4.4	Data packet delivery ratio vs. number of nodes	93
5.5	Conclusion	94
 III Path planning techniques for UAVs in an urban environment		95
 6 A UAV path planning approach considering obstacles and environmental uncertainties		97
6.1	Introduction	99
6.2	Related work	100
6.3	Problem statement and research contributions of this chapter	102
6.3.1	Research contributions of this chapter	104
6.4	Proposed obstacle avoidance scheme	105
6.4.1	Optimization problem statement	105
6.4.2	Suboptimal solution with the Dijkstra algorithm	108
6.4.3	Discussing the location of the predefined points	109
6.5	Simulation testbed and experimental procedures	110
6.5.1	Performance metrics	111
6.6	Results and discussion	112
6.6.1	Total distance traveled by the UAV	112
6.6.2	Trajectory tracking error vs. UAV's turning angles	114
6.6.3	Trajectory tracking error vs. UAV's moving speed	115
6.6.4	Computational load comparison	117
6.7	Conclusion	117
 7 A connectivity-aware path planning for a fleet of UAVs		119
7.1	Introduction	121
7.2	Related work	122
7.3	Scope of the proposed work	124
7.3.1	Initialization and application scenarios	124
7.3.2	Problem statement	126
7.4	Proposed connectivity-aware path planning algorithm	128
7.4.1	Kinematics of UAVs along a given path	128

7.4.2	Path planning	134
7.4.2.1	Generating the graph nodes	135
7.4.2.2	Generating edges	135
7.4.2.3	Generating weights	136
7.4.2.4	Generating a valid path	137
7.5	Simulation testbed and experimental procedures	139
7.5.1	Performance metrics	140
7.6	Results and discussion	141
7.6.1	UAVs as a source of data	141
7.6.1.1	UAV fleet moving with varying speed	141
7.6.1.2	UAV fleet moving with fixed speed	143
7.6.2	Ground users as a source of data	145
7.6.2.1	UAV fleet moving with varying speed	146
7.6.2.2	UAV fleet moving with fixed speed	147
7.7	Conclusion	149
Conclusion and future work		151
8	Conclusion and future work	153
8.1	Achieved work	154
8.2	Thesis contributions	155
8.3	Future work	156
A	Appendix for Chapter 6	157
A.1	Sketch of proof of theorem 1 - Consideration of rectangles' borders	157
A.2	Sketch of proof of theorem 2 - Time of flight of a UAV	158
A.3	Sketch of proof of theorem 3 - Safe distance from the obstacles	159
B	Appendix for Chapter 7	161
B.1	Proof of theorem 4 - Inverse mapping fro each UAV	161
B.2	Proof of theorem 5 - Location of each virtual UAV	163
B.3	Proof of theorem 6 - UAV's obstacle avoidance condition	163
B.4	Proof of theorem 7- Avoidance of any mutual UAV collisions	164
B.5	Proof of theorem 8 - All UAVs connectivity with starting or end locations	168

Bibliography

188

List of Tables

1.1	Simulation parameters	22
2.1	Simulation parameters	38
3.1	Fields of Neighbors Table	48
3.2	Simulation parameters	53
4.1	Comparison of various UAV based traffic surveillance schemes for highways . . .	68
4.2	Simulation parameters	74
5.1	Simulation Parameters	91
6.1	Summary of various path planning methods for UAVs	103
6.2	Simulation parameters and environment	111
7.1	Summary of various path planning and routing techniques for UAVs	125
7.2	Simulation parameters	140

List of Figures

1	Single-hop communication architecture	3
2	Multi-hop communication architecture	3
3	Fleet BS connectivity scenario (a) UAV-A connected with BS at t_0 (b) UAV-B connected with BS at t_1	4
4	Multiple BS deployment	4
5	Thesis organization	6
1.1	UAVs with deployment of multiple base stations	18
1.2	DNA Message format	19
1.3	Routing table entries for UAV-4	20
1.4	Data packet delivery ratio	23
1.5	Energy consumed per packet delivery	24
1.6	Average delay incurred by data packets	24
1.7	Data packet delivery ratio [Data rate= 0.1 Mbps, No. of BSs= 9]	25
1.8	Energy consumed per packet delivery [Data rate= 0.1 Mbps, No. of BSs= 9]	26
1.9	Average delay incurred by data packets [Data rate= 0.1 Mbps, No. of BSs= 9]	26
1.10	Gain in terms of PDR and delay	27
1.11	Gain in terms of PDR and energy consumed	27
2.1	Highway Scenario: Roadside BSs, VANETs and UAVs	34
2.2	Repaired path with the help of vehicle V2	37
2.3	Energy Consumed per Packet Delivery - Collaboration Disabled	39
2.4	Energy Consumed per Packet Delivery - Collaboration Enabled	40
2.5	Data Packet Delivery Ratio (Throughput) - Collaboration Disabled	40
2.6	Data Packet Delivery Ratio (Throughput) - Collaboration Enabled	41
2.7	Average Delay - Collaboration Disabled	42
2.8	Average Delay - Collaboration Enabled	42

3.1	Hop no. representation in a UAV network	48
3.2	Number of Packets Received at BS with Time (Exp. I)	53
3.3	Distance travelled by replacing UAVs (Exp. I)	54
3.4	Number of packets received at BS with time (Exp. II)	55
3.5	Distance travelled by replacing UAVs (Exp. II)	55
3.6	Number of packets received at BS with time (Exp. III)	56
3.7	Distance travelled by replacing UAVs (Exp. III)	57
3.8	Average energy consumption of communication overhead	57
4.1	Operation of CHS as a closed-loop control system	67
4.2	The architecture of CHS	70
4.3	Distance between RN-1 and other reporting nodes	71
4.4	Depiction of movement control algorithm	72
4.5	Reported OC values and UAV location [RN-2.0 as a hotspot region]	75
4.6	Violations detected by the UAV [RN-2.0 as a hotspot region]	75
4.7	Reported OC values and UAV location [RN-3.9 as a hotspot region]	76
4.8	Speed violations detected by the UAV [RN-3.9 as a hotspot region]	77
4.9	Reported OC values and UAV location [RN-5.3 as a hotspot region]	78
4.10	Speed violations detected by the UAV [RN-5.3 as a hotspot region]	78
4.11	Reported OC values and UAV location	79
4.12	Speed violations detected by the UAV	79
5.1	A collision scenario at MAC layer	86
5.2	A network part after hop number assignment (HU)	87
5.3	Case I: SI having value equal to T_1	87
5.4	Case II: SI having value equal to T_1+T_2	88
5.5	Case III: SI having value equal to $T_1+T_2+T_3$	88
5.6	Data packet delivery ratio (throughput)	91
5.7	Energy consumed per packet delivery	92
5.8	Average delay	93
5.9	Data Packet delivery ratio vs. number of nodes	93
6.1	Comparison of configuration space using the graph-based approach and our proposed scheme	104

6.2	Illustration of the trajectory designed by the proposed scheme with the uncertainty model	105
6.3	Disk centered on $V(t)$ where the UAV is assumed to be according to assumption 2.	106
6.4	$V(t)$ is on the line joining A and B . The contour surrounding this line delineates the set of all points at a distance below ρ , where the UAV is assumed according to assumption 2.	107
6.5	Four paths joining A and B consistent with assumption 3, shown on the left as positions of $V(t)$ and on the right using a graph structure.	108
6.6	Five loci of predefined points corresponding to, $\frac{3\pi}{2} - \frac{\pi}{5}$, $\frac{3\pi}{2} - \frac{\pi}{10}$, $\frac{3}{2}\pi$, $\frac{3\pi}{2} + \frac{\pi}{10}$, $\frac{3\pi}{2} + \frac{\pi}{5}$, when counting outwards. At the center is the horizontal obstacle of length $l = 1$ and $\rho = 0.1$. The proposed predefined points are C^- , C^+ , D^- , D^+	110
6.7	Total distance traveled by the UAV [maximum grid size 2500×2500]	113
6.8	Total distance traveled by the UAV [grid size 5000×5000]	113
6.9	Trajectories for the schemes compared (a) 4 obstacles, 500×500 grid size (b) 5 obstacles, 500×500 grid size (c) 6 obstacles, 5000×5000 grid size	114
6.10	Simulation scenario with 75° turning angle	114
6.11	Zoomed-in plots with arrow pointing towards vertices with angles (a) 0° (b) 30° (c) 45° (d) 60° (e) 75°	115
6.12	Tracking error for different turning angles	115
6.13	Scenario considered to determine tracking error under different UAV's speed	116
6.14	Tracking error for different UAV's speed	116
6.15	Impact of simulation grid size on execution time [19 obstacles]	117
6.16	Impact of the distance between the start and destination location on execution time [grid size 20000×20000 , 19 obstacles]	118
7.1	Application scenarios (a) Natural disaster scenario. (b) Continuous tracking and monitoring of a rally. (c) Monitoring of public areas in real-time	126
7.2	Line chart of the distance traversed by the n^{th} UAV for the no speed-change scenario.	131
7.3	(a) All possible routes between start and destination point (b) The selected path by the proposed scheme	139
7.4	Minimum distance with simulation time (a) UAV-UAV (b) UAV-Obstacles	141
7.5	UAV's reaction time to speed change requests (a) 4 transmitting UAVs (b) 8 transmitting UAVs	142

7.6	Number of packets received at BS (end-to-end connectivity check)	143
7.7	Data packets delay to reach BS	143
7.8	Minimum UAV-UAV distance with simulation time (a) Fleet speed 2 m/s (b) Fleet speed 5 m/s (c) Fleet speed 10 m/s	144
7.9	Minimum UAV-Obstacle distance with simulation time (a) Fleet speed 2 m/s (b) Fleet speed 5 m/s (c) Fleet speed 10 m/s	144
7.10	Number of packets received at BS (end-to-end connectivity check)	145
7.11	Data packets delay to reach BS	145
7.12	UAV's reaction time to speed change requests (a) UAV-1 as leader (b) UAV-4 as leader (c) UAV-6 as leader (d) UAV-9 as leader	146
7.13	Number of packets received at BS (end-to-end connectivity check for ground users)	147
7.14	Data packets delay to reach BS	147
7.15	Number of packets received at BS (end-to-end connectivity check for ground users)	148
7.16	Data packets delay to reach BS from ground users	148
A.1	Graphical construction illustrating theorem 3. As $\angle(CD, C^-M) = \frac{3\pi}{4}$, M is on the left of a half-plane delineated by line (C^-C^+) located at a distance of ρ of the obstacle CD	159

Abbreviations

APF	Artificial Potential Field
ATSS	Airborne Traffic Surveillance System
BAM	Base station Association Messages
BS	Base station
CC	Cooperative Communication
CHS	Collaborative Highway Surveillance
CSMA	Carrier-Sense Multiple Access
CSMA/CA	CSMA/Collision Avoidance
CTS	Clear To Send
CUV	Collaboration of UAVs with VANETs
CV	Candidature Value
DAG	Directed Acyclic Graph
DNA	Drone and Network Association
DPDR	Data Packet Delivery Ratio
ECD	Energy Consumed per packet
ECPD	Energy Consumed per Packet Delivery
GNSS	Global Navigation Satellite System
GPS	Global Position System
HNs	Helping Nodes
HU	Hop-Update
LEAMU	Life-Enhancing recovery Approach for a Multi-UAVs
L-UAV	Leaving UAV
MANETs	Mobile Ad-hoc Networks
MBNs	Mobile Broadband Networks
MBS	Mobile Base Station
MPR	Multipoint Relay

ND	Neighbors Discovery
OC	Over-speeding Counter
PDR	Packet Delivery Ratio
PRM	Probabilistic RoadMap
PRMBS	Proactive Routing scheme with Multiple ground BS
PSSN	Novel Packet Scheduling Strategy
QoS	Quality-of-Service
RM	Recovery Message
RNs	Reporting Nodes
RREP	Route Reply
RREQ	Route Request Packet
RRT	Rapidly Exploring Trees
RTS	Request To Send
SDD	Shortest Direct Distance
SI	Scheduling Interval
TC	Topology Control
TDMA	Time Division Multiple Access
UAV	Unmanned Aerial Vehicles
VANET	Vehicular Ad hoc Network
WM	Willingness Message
WSN	Wireless Sensor Network



Introduction

0.1 The necessity of dynamic road surveillance and UAV potentiality

Roughly 1.35 million fatalities are reported each year along with most countries losing 3 percent of their gross domestic product, as a result of road accidents [1]. Currently deployed surveillance systems on highways are either uneconomical or show a lack of coverage area and have high response times[2]. Helicopter in tandem with a ground police station is a very costly option with an average of \$157 per hour helicopter (Alouette III) operational cost [3]. Fixing speed detection instruments along the road is not a reasonable option too because people become familiar with the deployed locations and take caution only around those areas [4, 5]. According to the analysis performed by Pan et al. [6], 70.1% and 80.2% of the vehicles before and after the speed monitoring area, respectively, were over-speeding. Moreover, these instruments can also be attacked by people as was done during "yellow vest" protests in France by vandalizing 60 percent roadside cameras [7].

The ability of Unmanned Aerial Vehicles (UAVs) to fly over inaccessible and dangerous areas with great maneuvering skills has become the reason for providing highly feasible and economical surveillance solutions compared to their helicopter counterpart. Due to technological advancements, UAVs have developed into a mature technology, with its applications in several domains. Initially intended for military purposes, UAV-based applications are now expanding into civil and public domains within applications used for monitoring [8, 9], disaster management [10, 11, 12], and connectivity [13, 14, 15, 16]. UAVs such as quadcopters are now becoming common for aerial video monitoring [17, 18], and in many countries, they are playing a leading role in existing surveillance systems on highways [19]. Transportation of medical equipment, lab sample collection, spraying over different areas, and public space monitoring with guidance are novel use cases during the ongoing pandemic of COVID-19 [20, 21].

UAVs are known for their unique aspects of high mobility, dynamic network topology, lower communication range, and limited energy resources. A single UAV with these shortcomings fails

to perform in many applications, and the trend is now changing from a single larger UAV to multiple smaller UAVs[22]. Smaller lightweight UAVs reduce cost and also allow passage from confined areas[23]. Furthermore, UAVs in collaboration mode provide more scalability, enhanced network lifetime, and robustness to failures[24, 25]. Irrespective of the UAVs' deployment context, UAVs need to maintain continuous UAV-UAV and backhaul connectivity to send application data to the Base Station (BS). With the recent arrival of Fifth-Generation Mobile Networks (5G), UAVs can exploit cellular networks but require the network operator to incorporate specific UAV dynamics into their network planning and operation process. Moreover, the lack of cellular architecture in remote areas or natural disaster scenarios encourages us to explore alternate connectivity options.

In this regard, Vehicular Ad-hoc Networks (VANETs) and Mobile Ad-hoc Networks (MANETs) could be a good source of inspiration to employ an ad-hoc routing approach in UAV networks. However, the peculiar behavior of a UAV network prevents the complete as such adoption of existing ad-hoc routing schemes. Before having a robust ad-hoc solution, traditional ad-hoc architectures require a series of modifications in terms of service quality, node mobility, node connectivity, and routing mechanism.

0.2 Communication architecture in UAVs networks

Subject to the communication range of a UAV and scale of a network, communication architecture, at a generalized level, can be divided into a single-hop or multi-hop schemes. Explanation of these approaches are given below.

0.2.1 Single-hop communication

In a single-hop communication, a source UAV sends its data packets to a destination, i.e., a BS, without involving other UAVs. Data packets are sent without any delay if both the UAV and BS find themselves within the communication range of each other. This kind of communication mode is desirable in applications requiring real-time aspects. In a delay-tolerant network, on the other hand, packets are transmitted only when UAV and BS come into the direct communication range of each other. Fig. 1 shows a single-hop communication architecture. The scope of an application using a single UAV becomes limited as the UAV has to stay within the communication range of a BS. Additionally, this approach becomes energy-wise inefficient when the UAV has to transmit at longer distances as energy consumed in wireless transmission is proportional to distance.

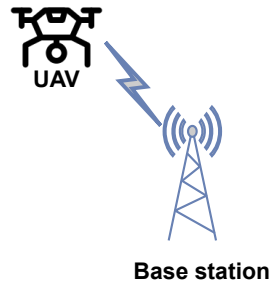


Figure 1: *Single-hop communication architecture*

0.2.2 Multi-hop communication

In multi-hop communication, UAVs transmit their data to BS with the help of other UAVs in a hop by hop manner. A multi-hop approach preserves energy resources as UAVs relay packets to nearby neighbors with low transmitting power. The selection of a next-hop UAV and seamless BS handover during the ongoing transmission becomes the decisive factor for determining the efficiency of a protocol. Fig. 2 shows a multi-hop communication architecture with ground BS deployment.

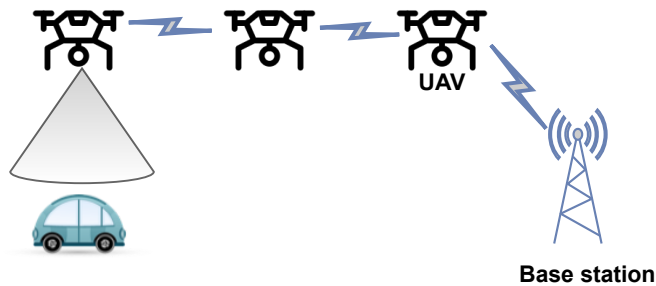


Figure 2: *Multi-hop communication architecture*

To provide continuous backhaul connectivity for having a real-time aspect, at least one UAV within the connected fleet needs to maintain a consistent link with a BS. This condition turns out to be a challenging job for highly dynamic UAV networks. Consider a scenario in Fig. 3a wherein, at instant t_0 , UAV-A maintains connectivity with BS. Similarly, as shown in Fig. 3b at the next instant t_1 , connectivity with BS is now maintained by UAV-B.

As shown in Fig. 4, there comes a situation where the entire fleet losses connectivity with BS, and we are left behind with no option but to have multiple BSs deployment. Frequent changes in topology and UAV-BS associativity require more regular routing table updates resulting in an increased routing overhead. Considering the typical nature of a UAV network, a fault-tolerant and

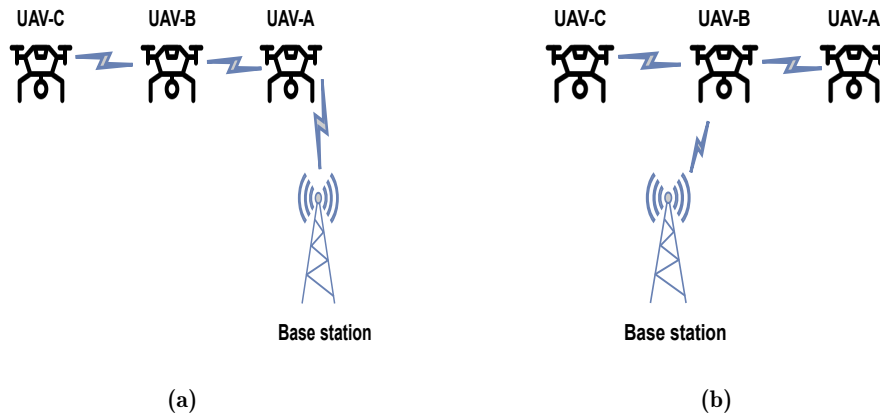


Figure 3: Fleet BS connectivity scenario (a) UAV-A connected with BS at t_0 (b) UAV-B connected with BS at t_1

efficient routing protocol is inevitable to provide reliable continuous backhaul link connectivity for a fleet.

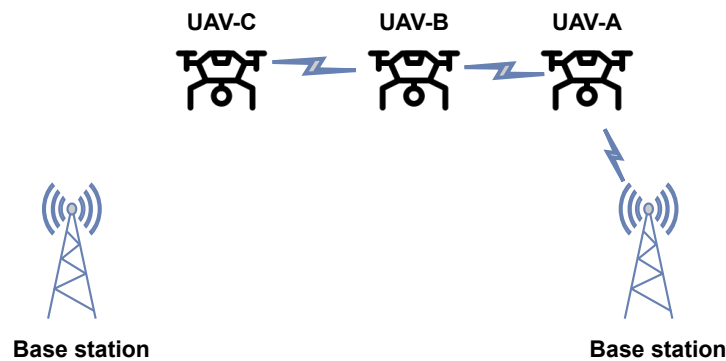


Figure 4: Multiple BS deployment

0.3 Path planning and formation control of a UAV fleet

Regardless of the deployed scenario, it is imperative to have a collision-free path planning to ensure the safety of UAVs and people on the ground, specifically in an urban environment with higher navigational uncertainties. In terms of time-domain, UAV path planning methods fall into online and offline categories[26]. In the first approach, UAVs plan their path in real-time and react to environmental changes, while in the second approach, UAVs are given a path plan before the commencement of a mission. Online methods demand the availability of high computing resources at each UAV or at least at leader UAV followed by dissemination of the designed path to other UAVs. Offline methods, on the other hand, fail to tackle any unforeseen scenarios.

The formation maintenance of a UAV fleet is a challenging task due to the unique dynamics associated with UAVs[27]. Path planning and data routing for a fleet of UAVs are influenced by these characteristics and demand careful consideration. Compared to a single UAV, having a fleet of UAVs in a mission increases the chances of collisions[28, 29], moreover high relative UAVs' mobility results in frequent link failures[30]. Various formation control architectures are available in the literature, such as leader-follower, behavioral-based, and virtual structure[31]. The leader-follower approach is the popular one wherein a leader UAV moves along a predefined trajectory, while other follower UAVs follow the leader and keep a safe relative distance between the leader and other UAVs[32]. In the virtual structure approach, UAVs maintain a reference formation shape by keeping a rigid geometric relation with each other. Every UAV in this method minimizes the error between the defined virtual position and actual position to maintain a reference shape[33]. The behavior-based technique comes up with a predefined behavior for each UAV using a hybrid vector-weighted formation control function[34].

To control the formation of UAVs, a continuous exchange of information among UAVs, such as the current locations and velocities, is necessary[35, 36]. Considering the dynamic nature of UAVs, the absence of state information due to communication problems may lead to severe consequences for the UAVs and people on the ground[37]. Moreover, high communication demand to control a UAV formation with low bandwidth resources of UAVs network renders no space for application requiring high Quality-of-Service (QoS) such as search and rescue operation during a natural disaster scenario.

Therefore, a collision-free path planning for a dynamic UAV network should consider limited resources of UAVs, uncertainties in the system, formation control overhead, and continuous link connectivity to the BS.

0.4 Thesis purpose and manuscript organization

The sparse network topology and limited communication range for UAVs pose many challenges for real-time UAV-based applications. Even though deployment of multiple BSs improves the likelihood of UAV-BS connectivity yet dynamic UAV topology results in frequent loss of UAV-BS associativity and UAV-UAV link failures. Physical breakdown of a UAV may also constitute link failures and require physical replacement by another UAV for the link recovery. These repeated recovery and topology update requests exert pressure on the limited network resources of UAVs. With already loaded network resources and limited computational abilities, it is inevitable to have efficient collision-free path planning with a formation control mechanism that should consider

these limitations.

To this end, this thesis provides solutions to all these issues in three principal parts. The first part presents routing protocols and failure recovery approaches for a fleet of UAVs. Second parts concerns UAV and Wireless Sensor Network (WSN) collaboration to tackle the sparsity of a UAV network and dynamically adapt UAV position with traffic patterns on the highway and capture maximum speed violations. The last part deals with the obstacle-avoidance path planning approaches in an urban environment to reduce computational load and ensure backhaul connectivities.

The thesis organization is presented in Fig. 5.

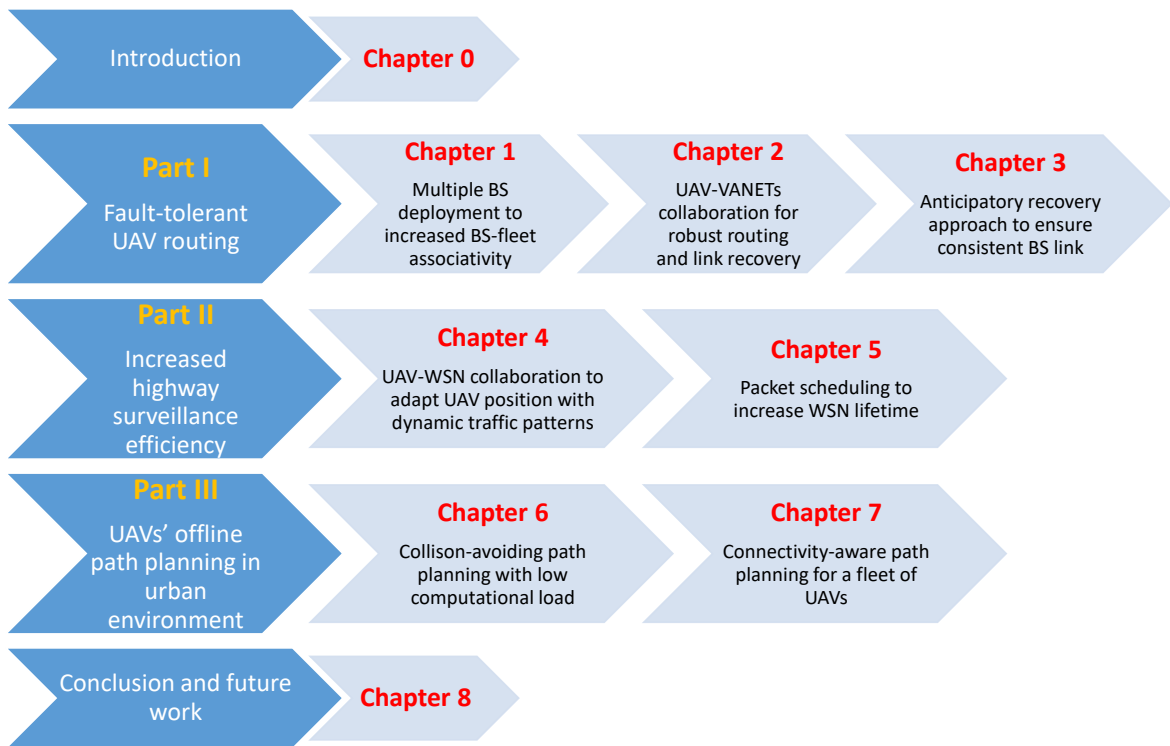


Figure 5: *Thesis organization*

0.4.1 Part I - Routing protocols and failure recovery approaches for a fleet of UAVs

The first part of the thesis proposes fault-tolerant proactive routing protocols ensuring continuous BS connectivity for a fleet of UAVs. Chapter 1 introduces a proactive routing approach to reduce routing overhead and enhance network backhaul connectivity under multiple BS deployment. BSs periodically broadcast Drone and Network Association (DNA) messages, and UAVs relay them in the entire fleet. Data packets take the reverse of the DNA path to reach a respective BS. Contrary to existing protocols, there are no separate hello messages with DNA

playing the role of these messages.

Chapter 2 introduces a collaborative architecture of UAVs and VANETs, called CUV, to deal with frequent link failures in a dynamic UAV's topology. The proposed scheme ensures BS link continuity by taking help from neighboring vehicles in case of the non-availability of a replacer UAV. CUV also deals with hidden node issues and compensates for the loss of control messages by providing the solution at the network layer.

Chapter 3 copes with a network breakdown scenario following the permanent displacement of a UAV from the network. The core of the idea is to select a suitable replacing candidate to avoid further recovery requests and provide smooth link continuity with BS. The proposed approach anticipates the energy depletion of a UAV, initiates a recovery request, and replaces the leaving UAV beforehand. The election campaign runs in the entire network with candidate selection based on distance, remaining energy, neighborhood density, and traffic load factors.

0.4.2 Part II - UAV and WSN collaboration to enhance connectivity and surveillance efficiency

This part proposes a collaborative architecture to provide dynamic highway coverage and addresses its associative issues for UAV-based surveillance systems. Chapter 4 introduces the main idea of UAV-WSN collaboration to adapt UAV position according to varying traffic patterns. WSN is the backbone of the network and provides routing service besides guiding the UAV for the best hotspot to capture maximum speed violations. This approach also facilitates reporting extreme speed violations immediately to BS.

Chapter 5 addresses the limited energy resources in WSNs and proposes a cross-layer solution to reduce packet collisions. Each packet is scheduled after a defined time interval to reduce the chances of collision along a path. This technique offers reduced delays and preserves energy resources due to the absence of packet re-transmissions and repetitive calls to the backoff algorithm.

0.4.3 Part III - UAVs path planning techniques in an urban environment

The last part of the thesis introduces offline UAVs' path planning approaches in an urban environment. Chapter 6 proposes a path planning technique where collision avoidance is mathematically proven under an uncertainty prerequisite, that the UAV follows its requested moving position within some threshold distance. This planning ensures UAV safety even if the underlying control system overshoots beyond its desired value. The proposed approach incurs a low computational load regardless of the size of the deployment area.

Chapter 7 introduces connectivity-aware path planning by dealing simultaneously with the path planning and routing processes for a fleet of UAVs. This approach ensures continuous backhaul connectivity and prevents UAV-UAV and UAV-obstacle collision. Any UAV within the connected fleet can lead the fleet at any desired speed. Moreover, consistent BS-fleet connectivity allows BS to disseminate information in real-time. The fleet line formation acts as a backbone of the network and allows any new UAV to become a part of the existing fleet.

0.5 List of publications

Based on the research work proposed in this thesis, some research articles have been published or submitted for publication to journals and conferences as follows.

Conference Papers:

- 1) N. E. H. Bahloul, N. Bashir, S. Boudjit, D. E. Boubiche. 2019, December. “A Simple Proactive Routing Protocol with Multiple Base Stations Deployment for Unmanned Aerial Vehicular Networks,” In *Proceedings of Global Information Infrastructure and Networking Symposium (GIIS)*, Paris, France, pp. 1–6.
- 2) N. Bashir, S. Boudjit. 2020, January. “An Energy-Efficient Collaborative Scheme for UAVs and VANETs for Dissemination of Real-Time Surveillance Data on Highways,” In *Proceedings of IEEE 17th Annual Consumer Communications & Networking Conference (CCNC)*, Las Vegas, NV, USA, pp. 1–6.
- 3) N. Bashir, S. Boudjit, M. Y. Saidi. 2021, January. “A Distributed Anticipatory Life-Enhancing Recovery Approach for Unmanned Aerial Vehicular Networks,” In *Proceedings of IEEE 18th Annual Consumer Communications & Networking Conference (CCNC)*, Las Vegas, NV, USA, pp. 1–7.
- 4) N. Bashir, S. Boudjit. 2020, October. “A Collision Avoiding Packet Scheduling and Energy-Efficient Routing Technique for Video Wireless Sensor Networks,” In *Proceedings of International Symposium on Networks, Computers and Communications (ISNCC)*, Montreal, QC, Canada, pp. 1–6.

Journal papers:

- 1) N. Bashir, S. Boudjit, S. Zeadally. 2022. “A closed-loop control architecture of UAV and WSN for traffic surveillance on highways,” *Computer Communications*. [Accepted 8 Apr 2022]

-
- 2) N. Bashir, S. Boudjit, G. Dauphin. S. Zeadally. 2022. “An obstacle avoidance approach for UAV path planning,” *IEEE Transactions on Vehicular Technology*. [undergoing revision after first decision]
 - 3) N. Bashir, S. Boudjit, G. Dauphin. 2022. “A Connectivity Aware Path Planning for a fleet of UAVs in an Urban Environment,” *IEEE Transactions on Intelligent Transportation Systems*. [Under review]
 - 4) N. Bashir, Z. H. Abbas, G. Abbas. 2019. “On Demand Cluster Head Formation with Inherent Hierarchical Clustering and Reliable Multipath Routing in Wireless Sensor Networks,” *Adhoc & Sensor Wireless Networks*, vol. 45, no. 1-2, pp. 59–91.

Part I

Routing protocols and failure recovery approaches for a fleet of UAVs

A routing protocol for a fleet of unmanned aerial vehicles under multiple base stations deployment

Abstract

Unmanned Aerial Vehicles (UAVs), aka drones, are prone to frequent link failures owing to their highly dynamic nature and low transmission range. Deployment of multiple ground Base Stations (BSs) increases backhaul connectivity but complicates transmission of control and signaling messages. This chapter provides a Proactive Routing scheme with Multiple ground BS (PRMBS) deployment to enhance backhaul connectivity and reduce network overhead. BSs periodically broadcast control messages, called DNA for Drones and Network Association, in the entire fleet. Unlike other routing protocols, there are no separate hello messages with DNA messages playing the role of these messages. The path for data traffic is formed in a reverse manner to reach the respective BS from which the DNA message was received. Simulations carried out for PRMBS show promising results under multiple BSs deployment¹.

¹N. E. H. Bahloul, N. Bashir, S. Boudjit, D. E. Boubiche. 2019, December. “A Simple Proactive Routing Protocol with Multiple Base Stations Deployment for Unmanned Aerial Vehicular Networks,” In *Proceedings of Global Information Infrastructure and Networking Symposium (GIIS)*, Paris, France, pp. 1–6.

Chapter content

1.1 Introduction	15
1.2 Related work	16
1.3 Proposed proactive UAV routing protocol	18
1.3.1 Broadcasting of DNA messages	18
1.3.2 Route establishment to BS	20
1.4 Performance evaluation	21
1.4.1 Variation in data rate and deployed BSs	22
1.4.1.1 Data Packet Delivery Ratio (PDR)	22
1.4.1.2 Energy Consumed per Packet Delivery (ECPD)	22
1.4.1.3 Average delay incurred by data packets	23
1.4.2 Variation in the number of transmitting UAVs per fleet	24
1.4.2.1 Data Packet Delivery Ratio (PDR)	24
1.4.2.2 Energy Consumed per Packet Delivery (ECPD)	25
1.4.2.3 Average delay incurred by data packets	25
1.5 Conclusion	26

1.1 Introduction

Single point failure, high development, and running time cost render the use of single UAV-based systems unsuitable for many applications [38, 39, 40, 41]. Moreover, a single-UAV-based system fails to meet a real-time aspect required in different applications [42, 43] due to its limited transmission range leading to difficulty in reaching ground BS. Multi-UAVs-based systems have shown to be more cost-effective, scalable, robust to failures, and capable of completing a mission in a quick time [40] [44] [45]. Even though a single small UAV has a limited capability, but multiple small UAVs in collaboration with each other can work beyond the capacity of a single large UAV [40][46].

Applications with multiple UAVs require a routing protocol to transfer data hop-by-hop to reach ground BS and ensure backhaul connectivity. In UAV networks, routing protocols consist of four familiar types, i.e., static, proactive, reactive, and hybrid protocols. Static routing is not advisable for UAV networks unless they offer a static topology by hovering over the same region. Proactive protocols establish a path before the actual demand at the expense of maintaining topological information at each UAV in addition to periodic hello messages. A higher dynamical topology of a UAV network leads to a higher rate of topological information exchange in proactive protocols. In reactive routing, a path is formed on-demand by a source UAV. Reactive routing schemes offer high delays due to the path formation waiting time. The hybrid routing combines reactive and proactive routing and utilizes the better features of both protocols to have a better solution.

Although multi-UAV networks have the upper hand on single UAV-based systems but realizing a routing solution is not a straightforward task due to the constraints imposed by highly dynamic UAV networks. Energy is the most scarce resource of mini airborne vehicles [47], and the insensible use of wireless communication can lead to the depletion of energy in a quick time. Moving speed, dynamic topology, and radio propagation model are among the few differentiating factors due to which routing protocols designed for VANETs and MANETs become unsuitable for UAV networks. Furthermore, the deployment of multiple BSs demands intelligent management over the exchange of control and signaling messages which, on the other hand, can result in overloading of the network.

This chapter presents a simple proactive routing approach for UAV networks, called PRMBS, with multiple deployed side BSs. PRMBS inherits the positive feature of a low end-to-end delay from proactive routing while minimizing overhead at the same time. Drone and Network Association (DNA) messages are periodically broadcast from each BS. DNA messages play a

dual role of hello messages and ensure the associativity of side BSs with each UAV. At each UAV, data packets follow the reverse of the path taken by DNA messages. Since each UAV is associated with multiple BSs, PRMBS offers multiple paths that eventually make this protocol more robust to single path failures.

This chapter is organized as follows. Section 1.2 presents the classical ad hoc routing followed by routing work in UAVs networks. Section 1.3 is comprised of explanation for the proposed routing protocol. In Section 1.4, the proposed PRMBS routing is evaluated. The chapter is concluded in Section 1.5.

1.2 Related work

Considering a few similarities between UAV and MANET networks, many modifications to MANET routing protocols have been proposed for UAV networks but owing to the difference in some basic characteristics, these modifications proved unsuitable for UAV networks [48]. This section, at first, presents some classical routing approaches for ad-hoc networks followed by routing protocols, specifically for UAV networks.

Dynamic Source Routing (DSR) is an example of a reactive approach. In DSR, a path establishment operation is performed when a source node has something to transmit [49]. A path from any source to any destination is feasible in this protocol because of its path maintenance and route discovery nature. Ad hoc On-demand Distance Vector (AODV)[50] is a classical example of reactive routing. Whenever a source node has something to transmit, a route request is generated and flooded in the entire network. The unicast response to this request is made either by the destination node or a node having a path to the destination. Overhead generated by AODV routing is small as each packet contains just the destination address. The delay incurred by data packets becomes large due to frequent link failures and repeated calls to the route discovery process. In some scenarios, too many route requests overload the network and even do not find enough bandwidth for themselves [51].

Optimized Link State Routing (OLSR) protocol is an optimization of the Link-state routing. Topology information is maintained at each node at the expense of regular flooding of link-state messages. Any small change in the topology is accomplished by flooding of link-state update messages. Multipoint Relay (MPR) is introduced in OLSR to reduce the effect of flooding of update messages. MPRs in the network are responsible for the forwarding of control messages. The complete reference to the OLSR protocol is accessible at [52]. OLSR is not a smart solution for highly dynamic UAV networks in which high overhead is obvious due to a high number of

link failures.

An AODV based routing protocol for unammed vehicle networks is presented in [53]. Instead of using Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA), time-slotted ALOHA protocol is embedded along with AODV routing protocol. Formation maintenance demands each node to be peer-aware resulting in a massive load on inner node communication. To reduce the collisions, time-slotted ALOHA protocol assigns time slots to some designated nodes. The pre-assigning of time slots in a highly mobile network is not a good option due to high link failures probability.

In [54], a cross-layer solution, named DOLSR, is presented. Directional antennas are the base for the DOLSR solution while routing is supported by OLSR protocol. This protocol is designed to extend the coverage area along with the attainment of some specific quality-of-services in terms of end-to-end delay, traffic controlling capability, and interference avoidance required by some applications based on UAV networks.

Geographic Position Mobility Oriented Routing (GPMOR) [55] is a geographical routing technique designed specifically for highly dynamic UAV ad-hoc networks. In this protocol, each UAV is supposed to know its geographical position through the use of Global Position System (GPS). Each UAV, after a certain interval, periodically exchanges its position with one-hop neighbors only. Within this interval, each UAV tries to predict the movement of its neighbors to define their new position.

Location-Aware Routing for Opportunistic Delay tolerant (LAROD) [56] is a delay-tolerant geographical-based routing protocol. It tries to get the shortest path to the destination by selecting one or more relays. A node, with the least effort to reach the destination, is the most suitable candidate to be elected as a relay. A UAV having data to transmit broadcasts packets to the potential relay does. One of the randomly selected relay node further broadcasts this packet until the destination is reached. Relay nodes not selected for relaying, simply discard these received packets. If any UAV fails to find a suitable relay, the network will act according to a delay-tolerant principle. Once the destination is reached, a reply is made to prevent redundant transmission for the same packets.

The Extended Hierarchical State Relating (EHSR) [57] is a hybrid hierarchical scheme in which UAVs act as a relay for terrestrial nodes. In EHSR, nodes are grouped according to their elevation. Drones form one group while the other group is formed by terrestrial nodes. Two different routing techniques are utilized for routing within and outside the group. Distance vector routing is used to have communication among different groups while link-state routing is used for intra-group

communication.

1.3 Proposed proactive UAV routing protocol

In literature, most of the work on the development of routing protocols for UAV networks is carried out by considering single deployed BS only. For a sparse network deployment and limited transmission range of a UAV, it is not always possible to have connectivity with BS. With the availability of multiple BSs, dissemination of data traffic becomes easier, but the management of the network becomes challenging at the same time. In this chapter, a simple proactive routing approach with multiple BSs deployment for a UAVs network (a possible network scenario is shown in Fig. 1.1), called PRMBS, is presented to reduce network overhead. This proposed routing mechanism is based on the periodic broadcasting of DNA messages by the ground BSs into the entire network. Moreover, each drone maintains an entry in its routing table for each known BS wherein the next-hop to reach this BS is the previous node from which the DNA message has been received. This section presents the working of the proposed PRMBS routing protocol.

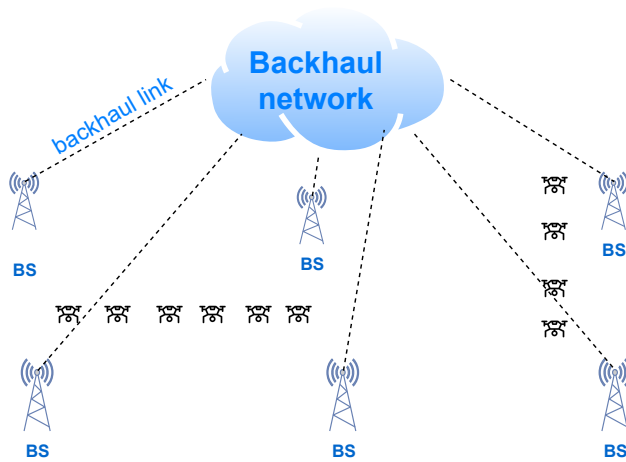


Figure 1.1: UAVs with deployment of multiple base stations

1.3.1 Broadcasting of DNA messages

DNA messages are transmitted periodically soon after the deployment of BSs within the region of interest. Each UAV node maintains a routing table having entries for each known BS in the network. Each entry is constituted of the following fields: BS ID, Relay-node ID, Hop-number, and Link-timeout. The BS ID is the identity of the base station that initiated this DNA message, it could be its IP address for instance. Relay-node ID is the identity of the

UAV from which the DNA message is received, it is the next hop to reach the respective BS. The Hop-number field is the number of hops to reach the respective BS. The Link-timeout is the validity time of this routing entry in the routing table. After the reception of each DNA message, the respective Link-timeout value is updated. Non-reception of DNA message from the same BS-Relay-node combo for a while results in the expiration of respective routing entry.

After every DNA-INTERVAL, each BS broadcasts a new DNA message as a data part of the general IP packet format as depicted in Fig. 1.2. To have a lower overhead, each DNA message is formed to have Relay-node ID and Hop-number fields. The sequence number and source IP address from the IP header of the received message is used to differentiate between different DNA messages. Before transmission, each BS sets the Hop-number field to zero in the DNA message. First time recipient of this message will keep a copy of it in its duplicate table and insert a new record in the routing table. It then rebroadcasts the message with the incremented Hop-number and Relay-node field set to its IP address.

Routing information is updated if another copy of the previously received DNA message is

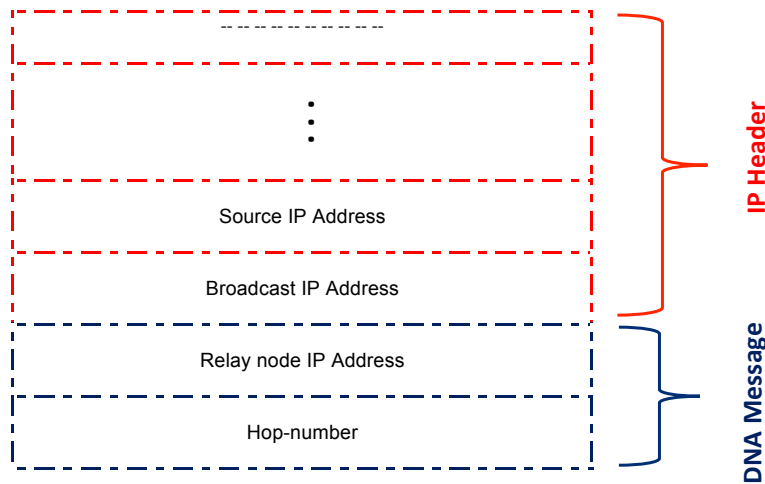


Figure 1.2: DNA Message format

received again but now with a different value in the Relay-node field. In this case, an alternate route entry becomes added to the routing table that leads to the enhanced reliability of the proposed scheme. A routing entry is also updated when a new DNA message from the same BS is received with a different sequence number. Finally, routing entries are removed from the routing table after the expiration of their respective Link-timeout fields.

Fig. 1.3 illustrates routing table entries maintained by UAV-4 in the current scenario for the proposed scheme. The figure shows that the fleet has a connection with the two BSs. UAV-4 can

reach either BS **C** via UAV-5 or **B** via UAV-3. According to the proposed scheme, UAV-4 goes for BS **C** as it offers a lower hop number, i.e., compared to other, i.e., BS **B**.

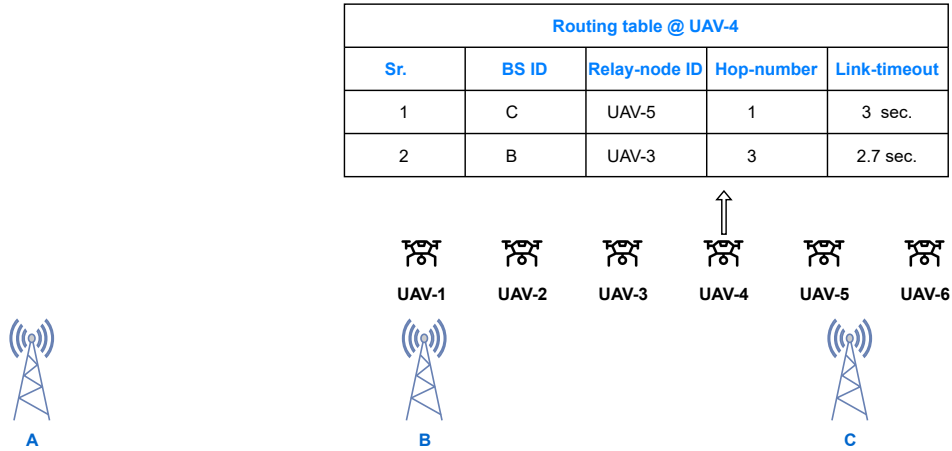


Figure 1.3: Routing table entries for UAV-4

The DNA message reception and forwarding mechanism is explained in Algorithm 1. In Algorithm 1, \mathcal{R} is a binary variable having a true value if the Source IP Address and Relay-node IP address associated with the specific received DNA message, are already available in the routing table. For any UAV, \mathcal{T} is a binary variable having a true value if the received DNA message is not found in the duplicate table.

1.3.2 Route establishment to BS

At each UAV, the next hop to reach a specific BS is the immediate UAV neighbor from which it has received a DNA message with this BS as an initiator of the message. The availability of multiple paths to BSs makes our protocol robust against route failures occurring due to frequent topology changes. As soon as data packets are available to transmit, one of these reverse paths is used to relay the data packet to the closest BS. The next-hop selection is based on the least number of hops to reach BS, and each UAV takes this routing decision independently in a distributive manner. If a link expires due to topology changes or non-reception of DNA messages due to any other reason, the data packets take another available path. The path with the least number of hops to reach BS is selected again soon as it becomes available for routing. During the unavailability of a path to BS, data packets are kept in the buffer and transmitted as soon as a path to any BS becomes available.

Algorithm 1 *DNA message forwarding mechanism*

Input:1: \mathcal{R}, \mathcal{T} **Output:** $DNA(d)$: Broadcast DNA message with random delay d (queue and mac delays)2: **if** $\mathcal{R} = true$ **then**

3: → Update Link-timeout and Hop-number fields in the routing table

4: → Insert header of DNA message in duplicate table5: → Broadcast $DNA(d)$ message6: **else**

7: → Insert new entry in the routing table

8: **if** $\mathcal{T} = true$ **then**9: → Insert header of DNA message in duplicate table10: → Broadcast $DNA(d)$ message11: **end if**12: **end if**

1.4 Performance evaluation

The proposed simple proactive routing scheme is implemented in Network Simulator-2 (ver. 2.35). The considered performance evaluation parameters of the protocol are, i.e., (i) energy consumed per packet delivery (whole network effort in terms of energy), (ii) throughput, and (iii) average delay incurred by data packets during the entire operation. These parameters are evaluated for the variation in the data rate, the number of deployed BSs, and the number of transmitting drones per fleet.

In simulated topology, five fleets of UAVs are considered with thirteen (13) UAVs per fleet. There is no redundancy in the generated data. Base stations that are connected with the backbone infrastructure, are simple static entities having the same transmission and reception capabilities as any other UAV. Each node in the network is equipped with an Omni-directional antenna. For this simulation, Droptail [58] queue type is available at each node with the Two Ray Ground propagation model for wireless communication. The parameters used during the simulation are listed in Table 1.1. The proposed protocol is compared with AODV [50] and the first subsection compares these two protocols for the variation in data rate and the number of deployed BSs. This is followed by the performance evaluation for the variation in the number of transmitting UAVs per fleet.

Table 1.1: *Simulation parameters*

Parameter	Value
Network size	10000 m x 10000 m
UAVs speed	12 m/s
Transmission range of each node	500 m
Data rate	0.1–4 Mbps
Interface queue length	500
Initial energy	50 J
Transmission energy	2324 nJ/bit
Receiving energy	1120 nJ/bit
Simulation time	400 sec
Propagation model	Two Ray Ground
Queue type	Droptail
Antenna type	Omni-directional
MAC protocol	Mac/802_11p

1.4.1 Variation in data rate and deployed BSs

1.4.1.1 Data Packet Delivery Ratio (PDR)

In this chapter, PDR is the ratio of the number of packets received at the backbone network to the number of packets sent by the UAVs. Fig. 1.4 presents the performance of PDR for the variation in data transmission rate and the number of deployed BSs. PRMBS outperforms AODV for the variation in both the parameters. Performance of PRMBS increases with the increase in deployed BSs because it takes this scenario as an opportunity to have multiple paths in a proactive manner. AODV, on the other hand, uses only a single path and frequent failure of the path will result in more RREQs and multiple RREP from different BSs. In AODV, for a higher number of deployed BS, RREQ and RREP engage the network too much that data packets do not find enough bandwidth that result in a lower PDR.

1.4.1.2 Energy Consumed per Packet Delivery (ECPD)

In this chapter, ECPD is a measure of overall network expenses in terms of energy for delivering a single data packet. It is defined as the total energy consumed in drones to the total number of packets received at the backbone network. Fig. 1.5 shows the effect on ECPD for

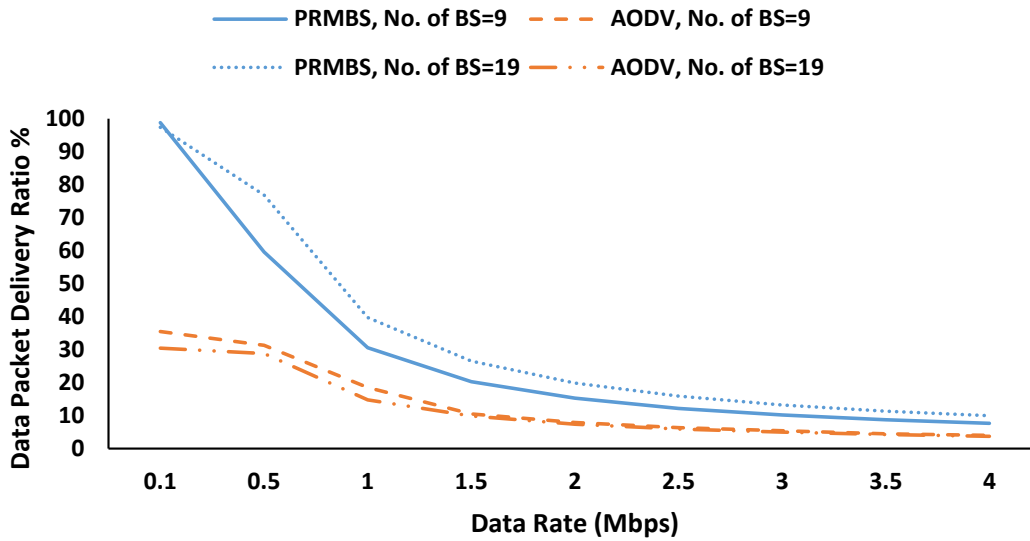


Figure 1.4: Data packet delivery ratio

the variation in data transmission rate as well as the number of deployed BSs. For the scenario of a higher number of deployed BSs (i.e. 19), data packets have to encounter less number of hops to reach the respective BS that results in the less energy consumption as compared to the case having a lower number of deployed BSs. For the fixed number of deployed BSs, the energy consumptions of AODV are greater which is due to the frequent RREQ and RREP messages in addition to the periodic hello messages. Overhead associated with both the protocols dominates at the lower data rate, for example, energy consumed in flooding of DNA messages and periodic hello messages in PRMBS and AODV, respectively, dominates the total energy consumed in transmitting the lower number of packets.

1.4.1.3 Average delay incurred by data packets

The average delay, in this chapter, is the average of the delay faced by all the data packets that have successfully reached the backbone network. This parameter variation against the data rate and number of deployed BS, is presented in Fig. 1.6. Average delay is reduced when the number of BSs is increased in the network because data packets have to traverse less number of hops due to the availability of nearby BS. In AODV, whenever a RERR is received all the data packets are dropped from the queue for the specific destination found in the RERR message. In AODV, a path is established on-demand, and packets are transmitted till the link remains valid and most of the packets get dropped from the buffer due to timeout value or reception of RERR messages. It is due to the same reason that the lower number of packets (as can be inferred from the PDR in Fig. 1.4) sent by AODV are the ones that are sent with a very small delay, as shown

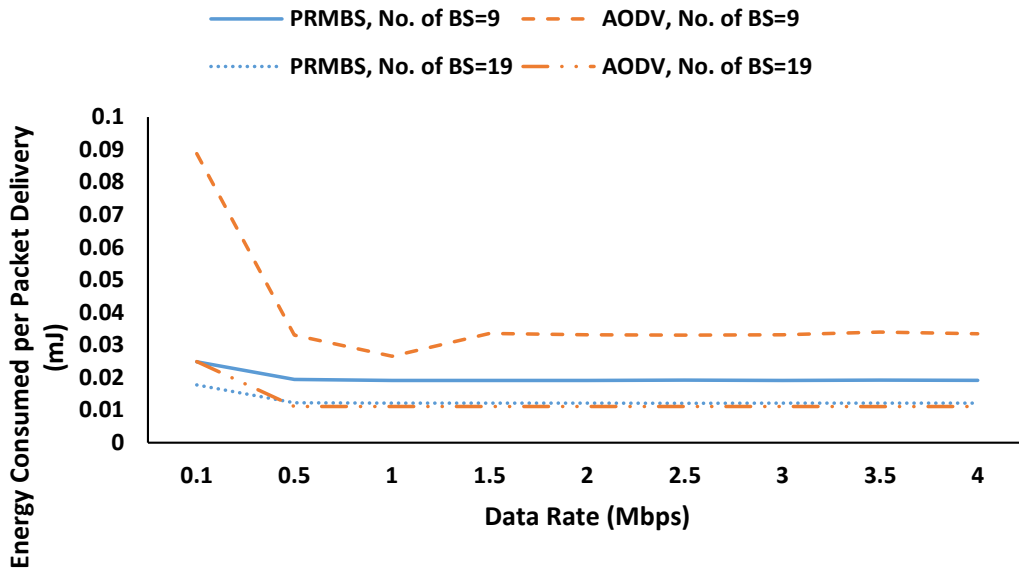


Figure 1.5: Energy consumed per packet delivery

in Fig. 1.6.

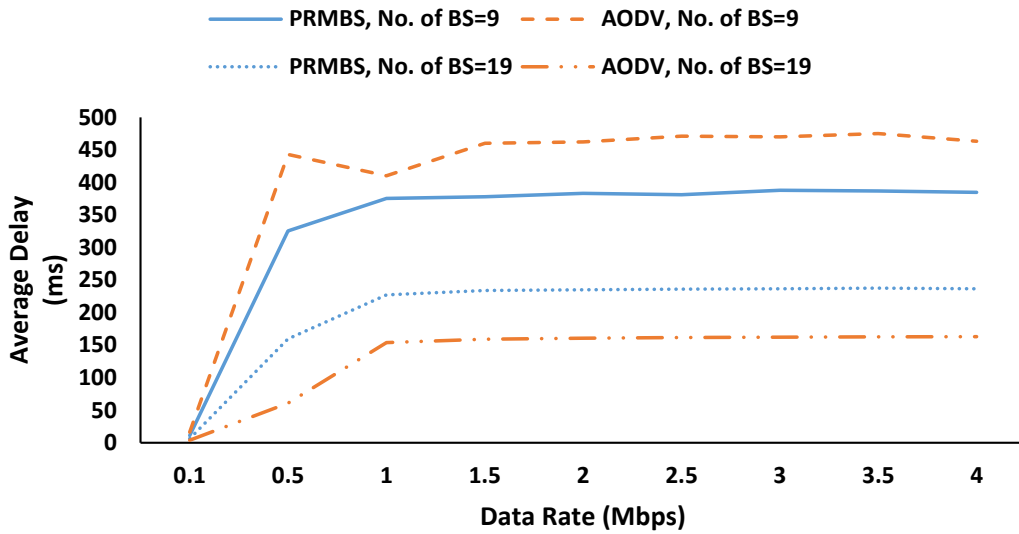


Figure 1.6: Average delay incurred by data packets

1.4.2 Variation in the number of transmitting UAVs per fleet

1.4.2.1 Data Packet Delivery Ratio (PDR)

For the variation in number of transmitting UAVs per fleet, PDR for PRMBS and AODV, is shown in Fig. 1.7. The performance of PRMBS is two times better than AODV for the variation in the number of transmitting nodes per fleet within the data rate of 0.1-Mbps and with 9 (nine) BSs deployed. The availability of alternate paths in PRMBS becomes the reason for the better

performance in terms of PDR. The performance is even more than double for a lower number of transmitting UAVs. In AODV, links are used frequently when the number of transmitting UAVs are increased. On the other hand, less number of UAVs results in a lower burden on these links per unit time. So, rarely used links are removed and has to form a new path frequently which becomes the reason for the lower performance for AODV.

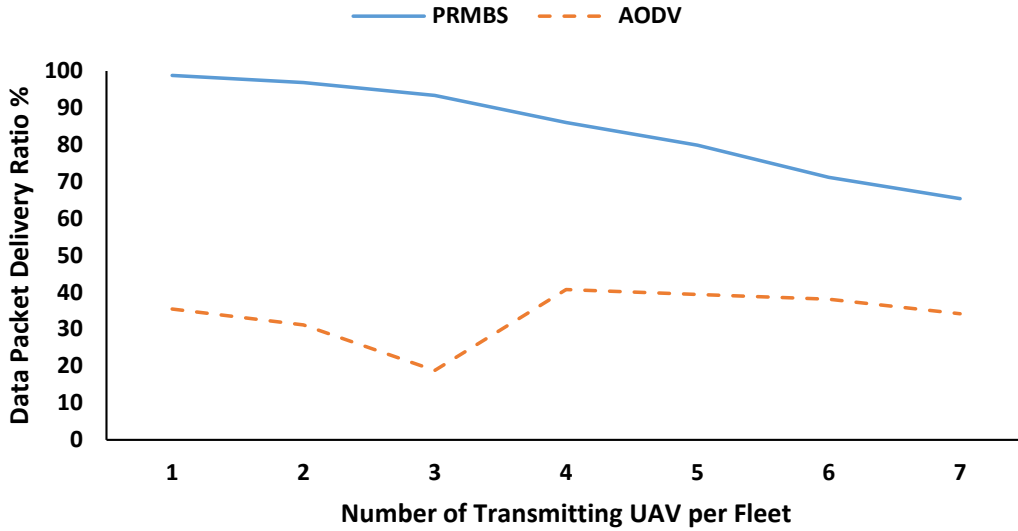


Figure 1.7: Data packet delivery ratio [Data rate= 0.1 Mbps, No. of BSs= 9]

1.4.2.2 Energy Consumed per Packet Delivery (ECPD)

Fig. 1.8 demonstrates the effect on ECPD for variation in data transmission rate as well as the number of deployed BSs. For the PRMBS, the effect of overhead on energy consumption always remains the same irrespective of variation in the number of transmitting UAVs except at the lower number of active UAVs. At a lower number of active UAVs for AODV, though PDR is almost the same (Fig. 1.7), however, overhead in terms of energy is dominated by the frequent RREQs as rarely used paths are removed from the routing table.

1.4.2.3 Average delay incurred by data packets

With the increase in the number of active UAVs, the delay associated with the data packets is also increased for both the protocols. Evaluation of average delay for the successfully received data packets with the variation of active UAVs is shown in Fig. 1.9. In this figure, AODV is having a better delay than PRMBS. So it all depends upon the user demand either he wants to have a lower delay in AODV but having half PDR as compared to PRMBS or two times more PDR than AODV with comparable delay with AODV.

Fig. 1.10 and Fig. 1.11 show gains comparison in terms of PDR to average delay and PDR to

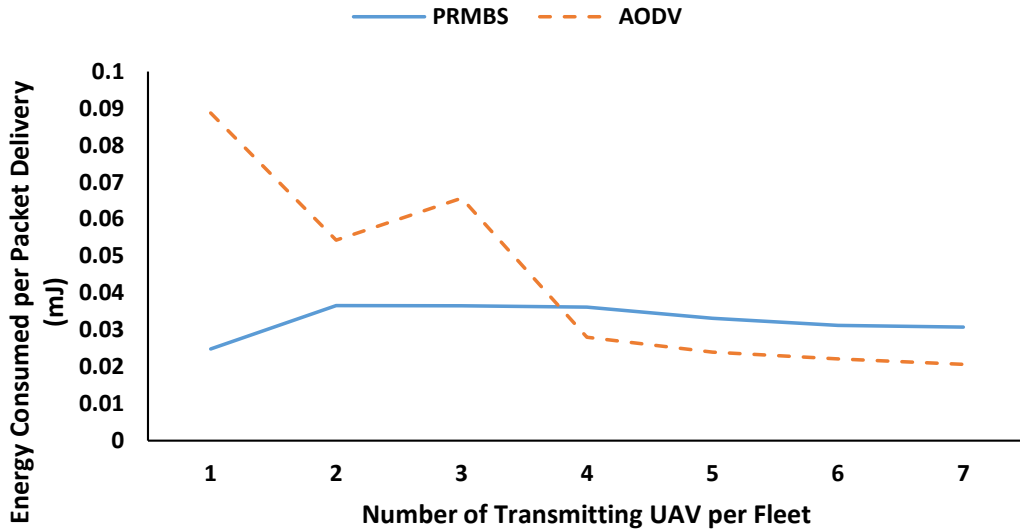


Figure 1.8: Energy consumed per packet delivery [Data rate= 0.1 Mbps, No. of BSs= 9]

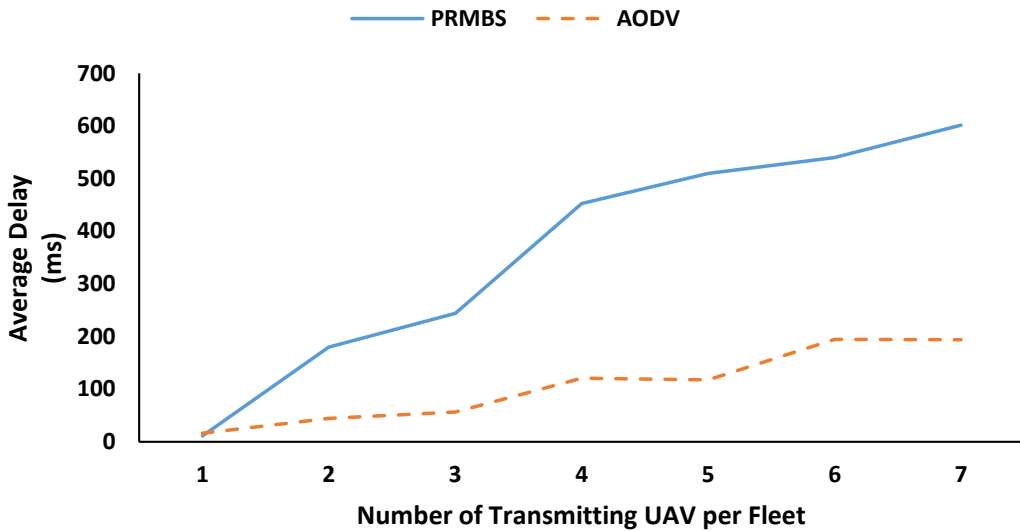


Figure 1.9: Average delay incurred by data packets [Data rate= 0.1 Mbps, No. of BSs= 9]

energy consumption, i.e., ECPD, respectively, for both the protocols. It can be inferred from both these figures that PRMBS gains remain better compared to AODV.

1.5 Conclusion

The proposed PRMBS routing is a simple proactive approach for multiple deployed BSs. Deployment of multiple BSs increases the chances of link establishment to the backbone network. In PRMBS, DNA messages are broadcast periodically that serve the purpose of hello messages as well as association of UAVs with the respective BSs. A large number of route requests, periodic hello packets and frequent route error messages render AODV unsuitable for UAV based

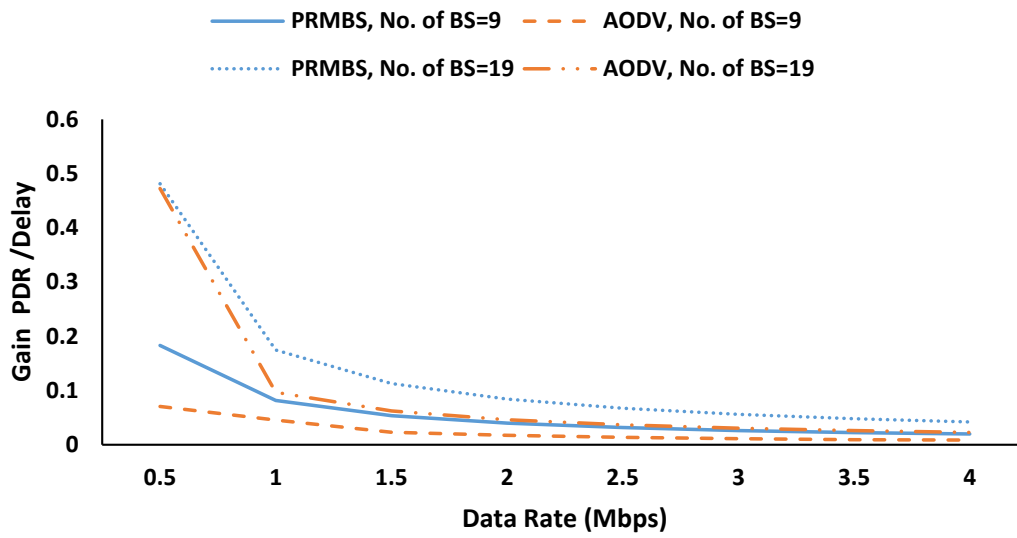


Figure 1.10: Gain in terms of PDR and delay

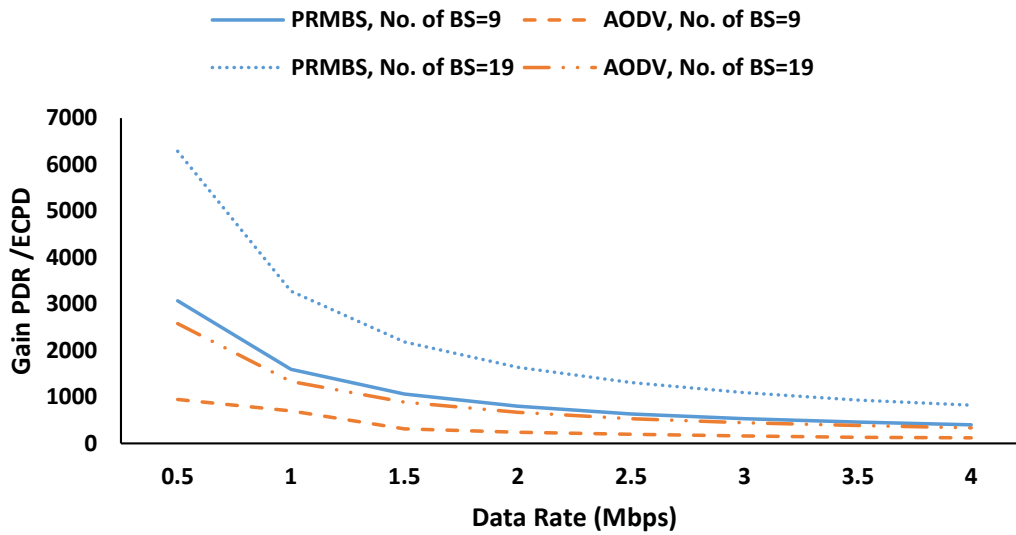


Figure 1.11: Gain in terms of PDR and energy consumed

applications, specifically for multiple base station deployment. In PRMBS, the impact of flooding of DNA messages on the overhead is still very less as compared to AODV signaling messages, which makes the proposed protocol a better option, especially in multiple deployed BSs.

A collaboration framework for UAVs and VANETs to increase UAV's network resilience

Abstract

Keeping in view the real-time aspect of road surveillance, the use of a UAV fleet, with the existing routing protocols, is not a straightforward option. A fleet of drones en route to highways surveillance may face frequent link failures due to its dynamical sparse linear architecture. To compensate for this shortcoming, a collaboration of UAVs with VANETs might be a good choice but as such integration of the two networks could create too much overhead. In this chapter, a proactive energy-efficient and reliable Collaborative scheme between UAVs and VANETs, termed CUV, is presented. To avoid flooding in the entire network, Base station Association Messages (BAM), in a unicast manner, are forwarded till the end-nodes. Although BAM is received by each node which ensures connectivity with the BS but only certain nodes are allowed to retransmit. Furthermore, to take care of lost BAM messages, a novel mechanism for link repair inside a BAM interval is introduced. The proposed scheme is simulated in Network-Simulator-2 and compared with modified well-known proactive OLSR and reactive AODV protocols¹.

¹N. Bashir, S. Boudjit. 2020, January. "An Energy-Efficient Collaborative Scheme for UAVs and VANETs for Dissemination of Real-Time Surveillance Data on Highways," In *Proceedings of IEEE 17th Annual Consumer Communications & Networking Conference (CCNC)*, Las Vegas, NV, USA, pp. 1-6.

Chapter content

2.1	Introduction	31
2.2	Related work	32
2.3	Proposed collaborative routing of UAVs and VANETs	33
2.3.0.1	Initialization phase and end-node formation	34
2.3.0.2	BAM propagation and mobility detection mechanism	34
2.3.0.3	Path repair mechanism	36
2.3.0.4	Application data transmission procedure	37
2.4	Performance evaluation	38
2.4.1	Energy consumed per packet delivery	39
2.4.2	Data packet delivery ratio	39
2.4.3	Average delay	40
2.5	Conclusion	41

2.1 Introduction

Due to the unique characteristic of high mobility and maneuvering freedom of a UAV, multi-UAV-based systems are not easy to implement compared to MANETs, VANETs, and static networks [59]. Dynamic topology and limited energy resources in UAV networks result in frequent link failures and demand special consideration for cooperative communication.

Subject to the communication range of each UAV and the scale of a network, the routing protocol can either be a single-hop or multi-hop [60]. In a single-hop protocol, packets are carried from a source node to a destination without the involvement of other UAVs. If BS and UAV are in direct communication range of each other, packets are sent directly without any delay. However, in delay-tolerant network packets are transmitted only when both, i.e., UAV and the BS come in the transmission range of each other. In a multihop scenario, UAVs communicate cooperatively, and packets are guided to BS in a hop by hop manner [61]. In a multihop network, selection for a next-hop UAV and seamless handover during link failures are the key functions on which the efficiency of a routing protocol is mainly dependent on.

A fleet of UAVs deployed for highways surveillance has a very sparse and dynamic topology owing to which network gets partitioned very frequently. Routing entries in the routing table remain valid only for some fixed duration, i.e., link expiry time. There may arise a situation in which a UAV might have a valid routing entry for other UAV even though both UAVs have moved out of the transmission range of each other or one UAV faces a failure. This kind of situation results in packets losses till the path is repaired. We reckon these issues to be crucial and careful consideration is needed before having a reliable routing solution for UAV networks.

This chapter mainly focuses on the UAV network which is deployed for highways surveillance, and proposes a collaborative scheme between UAVs and VANETs to nullify the effect of the sparse topology in UAV networks. The core focus of the proposed scheme is to come up with a reliable and energy-efficient routing design for UAV systems. Transmission of periodic Base Station Association (BAM) messages from each roadside BS ensures connectivity of each UAV with the respective roadside BS. In CUV, the network takes the help of nearby vehicles to repair the broken path during a UAV failure. To address the issues related to the high mobility and hidden nodes within a UAV network, we propose a novel handover mechanism in which a path is recovered inside the BAM interval. To reduce the energy consumption, we forward BAM messages till the midpoint (in terms of the number of hops) between the two road sides BS. Moreover, there are no periodic hello messages and path formation routines in the proposed approach.

This chapter is organized as follows. Section 2.2 presents closely related routing work in UAVs networks. Section 2.3 explains the proposed routing protocol. In Section 2.4, the proposed collaborative routing approach is evaluated. This chapter is concluded in Section 2.5.

2.2 Related work

Data-Centric Routing and Load Carry and Deliver Routing (LCAD) are examples of static routing protocols wherein routing tables are updated after the deployment or at the start of each operation. Data-Centric Routing is a content-based approach in which a node may request (interest) a particular type of data. This interest is diffused in the whole network while maintaining the reverse path. The data-Centric approach works well for the hierarchical approach in which the cluster head takes the responsibility of data transmission to its member nodes[62]. For each session of data transmission, an interest request has to be flooded in the entire network. Due to fast-moving UAVs, the reverse path cannot be guaranteed always. LCAD is a delay-tolerant routing protocol and even though delays in the delivery of data packets are high but these protocols achieve much higher throughput. If the same data has to be transferred to multiple destinations, a very large delay will be incurred by the data packets. This sort of protocol is not suitable for real-time applications of UAVs [63].

Global State Routing (GSR) [64] is a variant of link-state routing in which update packets restrict themselves between the intermediate nodes. In GSR, the size of the update packet is large, and with highly dynamic topology, intermediate nodes keep on changing which results in extra overheads and bandwidth problems. DSDV is developed mainly to solve the routing loop problem [65]. This protocol is table-driven and the Bellman-Ford algorithm with minor adjustments is used for ad-hoc mobile networks. The routing base is updated using more frequent incremental and less frequent full dump updates. The bandwidth requirements for this kind of proactive protocol remains large due to the periodic update mechanism.

Reactive-Greedy-Reactive (RGR) [66] is a combination of reactive and Greedy Geographic Forwarding (GGF) protocols proposed for UAVs. RGR protocol takes into account UAV location information as well as the reactive end-to-end route in the path formation process. This protocol performs reasonably well as compared to existing protocols like AODV, specifically in search missions in terms of data packets delivery and end-to-end delay.

Hybrid routing protocols reduce delays faced by reactive and overhead caused in proactive protocols as a result of periodic update mechanisms. Zone Routing Protocol (ZRP) and Temporally Ordered Routing Algorithm (TORA) fall into this category[67]. ZRP is a concept in which a

network gets divided into different zones [68] with each zone predefined with a specific set of nodes. Communication within a zone is done with a proactive approach. Inter-zone reactive routing takes the charge for out-of-zone destinations. The zone radius is the main deciding factor for the efficiency of ZRP. Deciding the radius of each zone and the number of UAVs in each zone add to the complexity of a UAV network. TORA [69] is a distributed hybrid scheme in which only the information of next-hop nodes is kept at each node. It tries to minimize the reaction to topological changes. It constructs and maintains a Directed Acyclic Graph (DAG) from the source node to the destination node. This protocol is preferred for a dynamic network having more pervasive link failures because it finds new paths very quickly [48]. This protocol may produce invalid routes. The shortest path algorithm is not used by TORA and sometimes longer paths are chosen to reduce the overhead.

OLSR [52] and AODV [50] are examples of proactive and reactive routing protocols, respectively, and their explanation is available in Chapter 1.

2.3 Proposed collaborative routing of UAVs and VANETs

High overhead and path setup delay in proactive and reactive protocols, respectively, render them improbable for UAV networks. This chapter proposes a novel proactive solution to cope up with the challenges of highly mobile UAV networks. CUV is a collaboration of UAVs and VANETs networks and the main targeted application is surveillance of highways. UAVs deployed for surveillance forward their data packets in a hop by hop manner to the roadside BSs. CUV being a collaborative scheme seeks help from the VANETs in case of non-availability of nearby UAVs ². BAM messages are kept within the two consecutive roadsides BSs rather than flooding in the entire network. Instead of waiting for the next BAM message and reacting to any possible topology change, CUV takes action inside the BAM interval and repairs its path before the reception of the next BAM.

In the subsequent subsections, at first, the initialization phase of our scheme will be presented. It will be followed by a BAM propagation and mobility detection mechanism. In the second last subsection, the path repair approach is discussed. The last subsection contains the data transmission procedure.

²This chapter considers BS, UAVs, and vehicles equipped with the same radio technology, i.e., 802.11p. However, UAVs having multiple interfaces like 802.11p and cellular 5G connectivity can act as flying BSs.

2.3.0.1 Initialization phase and end-node formation

After the initial deployment of aerial drones on the highways, the CUV protocol starts its operation. UAVs in direct communication range of roadside BSs start the hop number assignment procedure as soon as they receive the first broadcast BAM message. UAV receiving BAM message directly from the roadside BS sets its hop number to 1 and retransmits the same message to its neighbors with a change in the hop number field. In Fig. 2.1, UAVs **A**, **H**, and **I** set their hop number to 1 as they receive BAM message directly from BSs. UAV **A** is directly linked with BS-1 while **H** and **I** are directly linked with BS-2. UAVs receiving BAM messages from **A**, **H**, and **I** UAVs will set their hop number to 2 and the BAM message is transmitted again. Further forwarding of a BAM message is stopped whenever it is interacted with a node having already assigned hop number from other BS and hop number difference equal or exceeding than one. UAVs satisfying the condition in the previous statement are called end-nodes in the CUV protocol.

BAM messages are not flooded in the entire network rather BAMs from one BS are forwarded in a unicast manner till the respective end-node. In Fig. 2.1, UAV **E** has been assigned a hop number of 4 with respect to BS-2 and also hop number 5 with respect to BS-1. UAV **E** and **D** are the end-nodes with respect to BS-1 and BS-2, respectively. A UAV will also be called an end-node if it receives no BAM from a higher hop number UAV. UAV **J** is an end-node with respect to BS-2 because of the same reason. A path from end-node to the respective BS is formed whenever a UAV selected as an end-node makes a unicast reply to the respective BS. For a UAV network having sparse topology, vehicles become involved and play their part in path formation and forwarding of data.

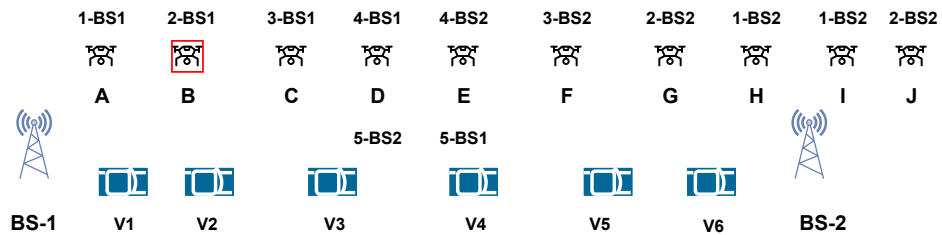


Figure 2.1: Highway Scenario: Roadside BSs, VANETs and UAVs

2.3.0.2 BAM propagation and mobility detection mechanism

After the initialization phase, each roadside BS unicasts periodic BAM updates. Whenever

this BAM is received by the UAV involved along the path, it unicasts this BAM to the next UAV. The same procedure is continued till the respective end-node. All the vehicles receiving these messages update their neighbor's table but do not retransmit these messages. A new hop number is assigned to the UAV, whenever the hop number difference of the received BAM and the receiving UAV exceeds one. All the UAVs, starting from this UAV until the respective end-node, are assigned with new hop numbers. UAV assigned as an end-node remains unchanged unless in the presence of a UAV having an association with other BS. For the latter case, the end-node is shifted from the old UAV to the new one having a link with the other BS. Base station association phenomenon is explained in Algorithm 2 in which \mathcal{N}_s and \mathcal{N}_r are the hop numbers of sender and receiver of BAM messages, respectively. \mathcal{U}_f is a node to whom the BAM message will be forwarded. \mathcal{T} and \mathcal{E} are the binary variables having a true value whenever the BAM message is received at a targeted node having an active path and end-node, respectively.

Whenever a UAV forwards a BAM message, the same message is received by the previous UAV. In Fig. 2.1, if UAV **C** forwards a BAM to UAV **D**, the same message will be received by UAV **B** and this mechanism indirectly validates the existence of a link between **B** and **C** UAVs. When a BAM is not received at the previous UAV, a route repair operation is started and it could be due to one of the two reasons:

- i. actual movement of UAV
- ii. BAM packet drop at the queue due to the ongoing transmission or its collision at MAC level probably due to hidden node

For the first case, In Fig. 2.1, UAV **B** moves out of the transmission range of UAV **A** and it does not receive the BAM message from UAV **A**. The mobility of UAV **B** will be detected by UAV **A** because BAM sent by UAV **A** is not received back from UAV **B**. This scenario will initiate a path repair mechanism at UAV **A**.

The other reason for the initiation of path repair is a drop of timeout BAM packets at the queue because of the ongoing data transmission or collision of BAM at the MAC level due to the presence of a hidden node scenario. For example, UAV **A** forwards BAM to UAV **B** but this packet is either dropped at queue due to ongoing transmission towards BS-1 or its collision occurs. Though UAV **B** is still at its prior position but it will delete link entry for UAV **A**, falsely assuming the movement between itself and UAV **A**. In this way, UAV **A** will initiate a path repair mechanism.

Algorithm 2 *Base station association update mechanism*

Input:

1: $\mathcal{N}_s, \mathcal{N}_r, \mathcal{U}_f, \mathcal{T}, \mathcal{E}$

Output: $Forward(\mathcal{U}_f, d)$: Unicast this BAM to \mathcal{U}_f with random delay d

2: **if** $\mathcal{T} = true$ **then**

3: **if** $\mathcal{N}_r - \mathcal{N}_s > 1$ **then**

4: Change the hop number of current UAV;

5: $Forward(\mathcal{U}_f, d)$

6: **else**

7: **if** $\mathcal{E} = true$ **then**

8: Just update route entry

9: **else**

10: Update route entry;

11: $Forward(\mathcal{U}_f, d)$

12: **end if**

13: **end if**

14: **else**

15: Just update route entry

16: **end if**

2.3.0.3 Path repair mechanism

The path repair mechanism tries to recover the path locally without troubling the entire network. For example, after the detection of mobility at UAV **A**, it broadcasts a path repair request. Upon the reception of this request, either a reply is made from the UAV previously involved along the path or a UAV having a neighbor with hop number two higher than the requesting node's (i.e., UAV **A**) hop number. For example, if a repair starts at UAV **A** due to the BAM packet drop or collision issue, a reply will be made by UAV **B**, on the other hand, for the case of actual mobility of UAV **B**, a reply will be made by Vehicle V2. So, the previous path link A-B-C is repaired locally with a new link of A-V2-C shown in Fig. 2.2.

The path repair mechanism is explained in Algorithm 3 in which \mathcal{R} is a binary variable having a true value if the path repair message is received at a node that still has a valid BS association. For any node, N contains the list of all the neighboring nodes. From the neighbor's list N , $\alpha(n)$ is the hop number of neighbor n , while $\alpha(s)$ is the hop number of node s that sends path repair

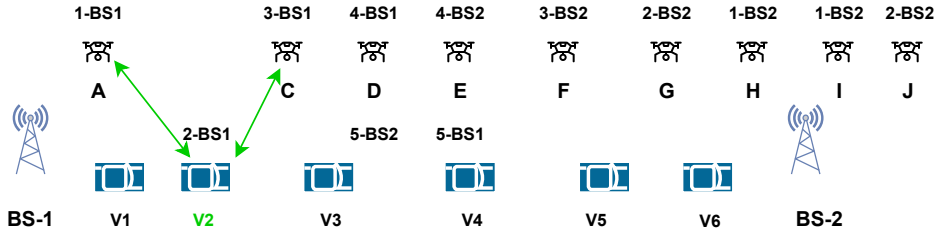


Figure 2.2: Repaired path with the help of vehicle V2

Algorithm 3 Path repair mechanism

Input:

1: \mathcal{R} , N , $\alpha(n)$, $\alpha(s)$

Output: $Reply(s, d)$: Reply to sender node s with random delay d

```

2: if  $\mathcal{R} = true$  then
3:    $Reply(s, 0)$ 
4: else
5:   for all  $N$  do
6:     if  $\alpha(s) + 2 = \alpha(n)$  then
7:        $Reply(s, d)$ 
8:     else
9:       Discard this packet
10:    end if
11:  end for
12: end if

```

message.

2.3.0.4 Application data transmission procedure

Whenever a UAV has data to transmit, it looks for the availability of an active route. As CUV is a proactive approach, most of the time, a path is available before its demand. Data transmitting UAV will forward data packets to a lower hop number node until the roadside BS is reached. Except for end-nodes, all nodes have paths associated with one roadside unit only. In the absence of an active path, data packets are kept in the queue, and a temporary request for a path is made. A reply to this request is made whenever this request is received by a node having an active path.

Table 2.1: *Simulation parameters*

Parameter	Value
Network size	5000 m x 40 m
UAVs Speed	20 m/s
Vehicles Speed	10-25 m/s
Transmission range of each node	100 m
Data rate	0.01–1 Mbps
Interface queue length	100
Initial energy	50 J
Transmission energy	2330 nJ/bit
Receiving energy	1159 nJ/bit
Simulation time	202 sec
MAC protocol (Vehicles and UAVs)	Mac/802_11p

2.4 Performance evaluation

To validate our proposed idea and compare the routing performance with the existing ones, we implemented CUV and the modified versions of OLSR and AODV in Network Simulator-2 (ver. 2.35). The simulator used for this simulation is considered as an efficient simulators for wireless networks [70, 71, 72, 73]. The evaluation of the protocols is based on metrics, i.e., (i) energy consumed per packet delivery, (ii) throughput, and (iii) the average delay incurred by data packets during the entire operation.

In this simulation, a linear sparse topology for the drones,³ having a predetermined path above the highway, is considered. Vehicles moving in both directions are deployed randomly within the highway region. Data is generated by random drones and there is no redundancy in the generated data. All the metrics are evaluated for two scenarios, i.e., UAVs with i) disabled collaborating capability with vehicles and ii) enabled collaborating capability with vehicles. Each node is equipped with an Omni-directional antenna. For this simulation, Droptail [58] queue type is used with the Two Ray Ground propagation model for communication. The parameters used during the simulation are listed in Table 2.1.

³Drones move in the same direction and speed chosen for the simulation is according to one of the commercially available drones [<https://www.dji.com/fr/phantom-4-pro-v2/info#specs>]

2.4.1 Energy consumed per packet delivery

In this chapter, energy consumed per packet delivery (ECPD) is defined as the ratio of total energy consumed in drones to the total number of packets received at the roadside BSs. For the deployed UAV network, Fig. 2.3 and Fig. 2.4 demonstrate the effect of the data transmission rate on ECPD for the scenarios without and with vehicles, respectively. In both the scenarios, the CUV protocol performs better than OLSR and AODV in terms of ECPD. In CUV, BAM messages in a unicast manner are forwarded till the end-nodes and no energy is wasted in path construction become the reason for the best performance among OLSR and AODV. With the inclusion of vehicles (Fig. 2.4), flooding of path requests and overhead generated due to the increase in the size of topology messages in AODV and OLSR, respectively, become the reason for the lower performance as compared to CUV protocol.

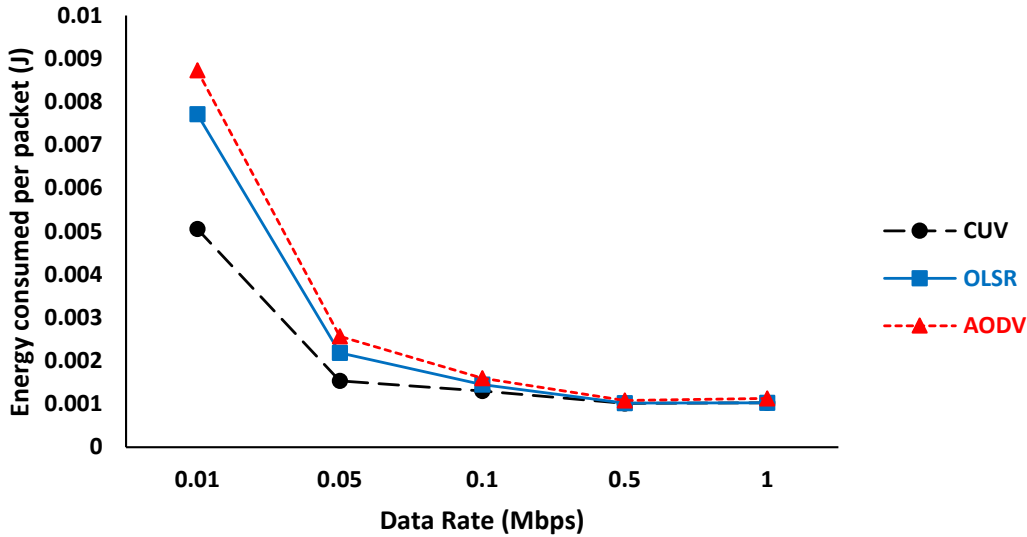


Figure 2.3: Energy Consumed per Packet Delivery - Collaboration Disabled

2.4.2 Data packet delivery ratio

Data packet delivery ratio (DPDR) or throughput is defined as the ratio of the number of packets received at the road sides BSs to the number of packets sent by the UAVs. Fig. 2.5 and Fig. 2.6 present the behavior of CUV, OLSR, and AODV protocols in terms of DPDR with respect to the data transmission rate of UAVs for the scenarios i.e., without and with vehicles, respectively. It can be inferred from Fig. 2.5 and Fig. 2.6 that CUV and AODV have better DPDR than OLSR. AODV being a table-driven reactive protocol adapts well during the mobility scenarios and the same is the reason for the better performance as compared to OLSR. On the other hand, OLSR uses a proactive approach and it is more prone to dynamic networks. The

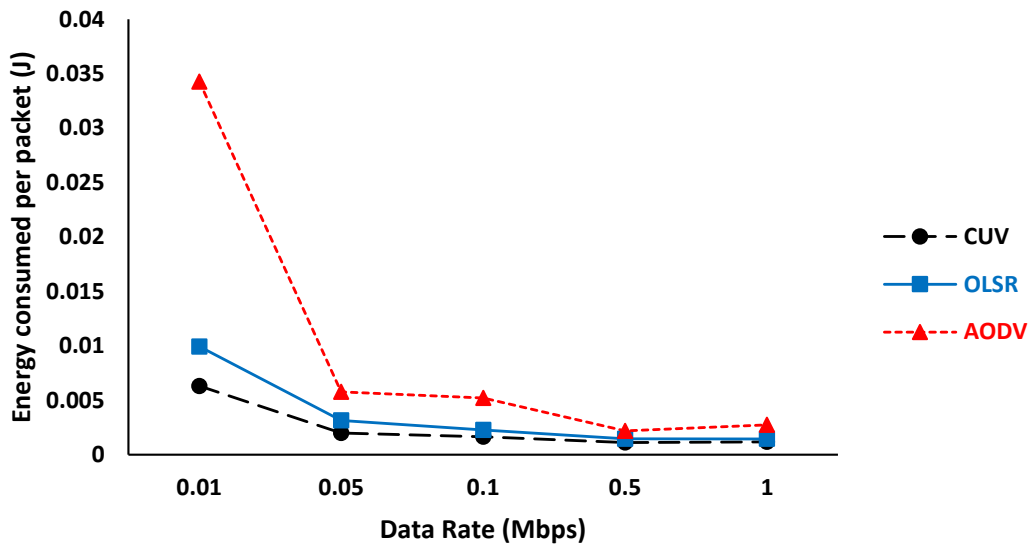


Figure 2.4: Energy Consumed per Packet Delivery - Collaboration Enabled

reason for the better performance of CUV is the resilience to the hidden node issue and detection of node mobility during the BAM message intervals. In Fig. 2.6, the increase in throughput with the addition of vehicles as compared to Fig. 2.5 having only UAVs, signifies the usefulness of our proposed collaborative scheme between UAVs and VANETs.

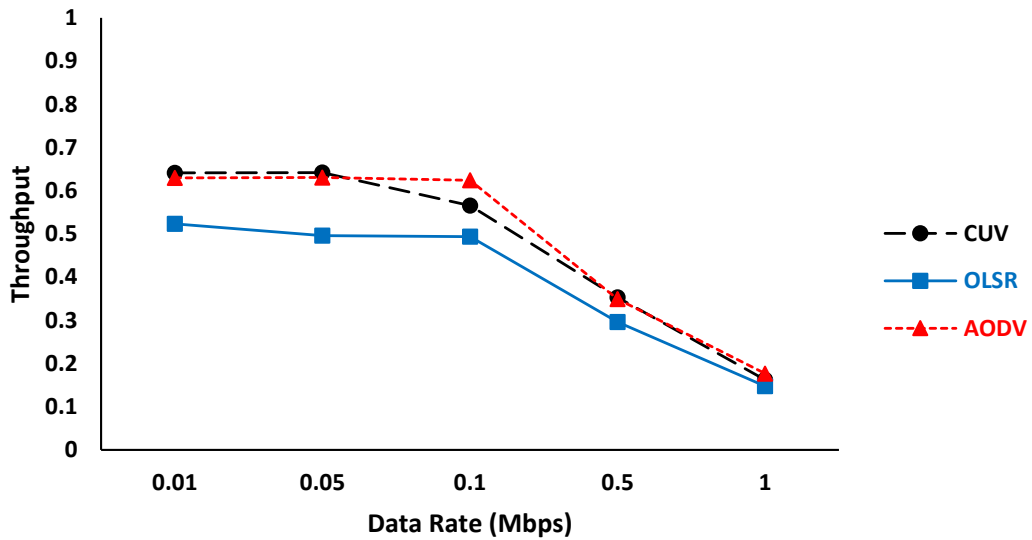


Figure 2.5: Data Packet Delivery Ratio (Throughput) - Collaboration Disabled

2.4.3 Average delay

The average delay is the average of the delay faced by all the data packets to reach roadsides BSs. The average delay against the data rate is presented in Fig. 2.7 and Fig. 2.8 for without

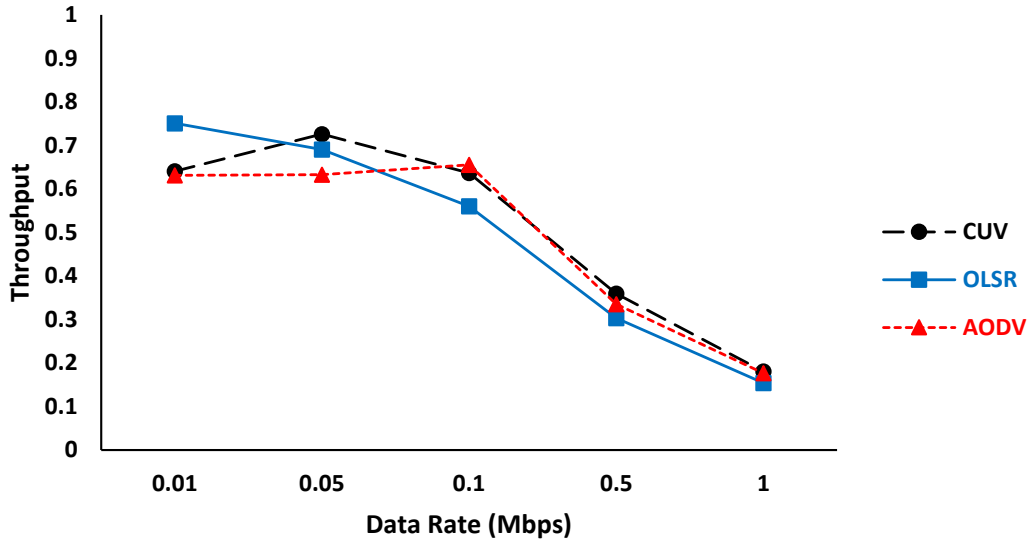


Figure 2.6: *Data Packet Delivery Ratio (Throughput) - Collaboration Enabled*

vehicles and with vehicles, respectively. AODV is a reactive protocol and it takes time to establish a path as compared to proactive protocols. The same effect is reflected in Fig. 2.7 and Fig. 2.8 for both the scenarios as AODV faces the worst average delay by the data packets. The performance of the CUV protocol with regards to the average delay is between the OLSR and AODV protocol. The presence of a repair mechanism engages the packets in the queue until the repairing is completed. Due to the hidden nodes, chances of collision between BAM and data packets are increased with the increase in data rate. Frequent calls to repairing mechanism further add to the wait time of data packet in the queue resulting in an increased delay of CUV protocol. OLSR protocol proactive in nature shows the least delay among AODV and CUV because of having a path prior to its demand.

2.5 Conclusion

The proposed CUV protocol is a novel approach to tackle the issues specifically linked with UAV networks. The CUV is a collaborative scheme for UAVs that seeks help from VANETs during the non-availability of nearby UAVs. In a unicast fashion, BS association messages are kept in between the BSs. For the first time, we tried to nullify the effect of a hidden node problem by providing a solution at the network layer. The results demonstrated that CUV is an energy-efficient and reliable protocol that deals well with the sparse and dynamic topology of UAV networks.

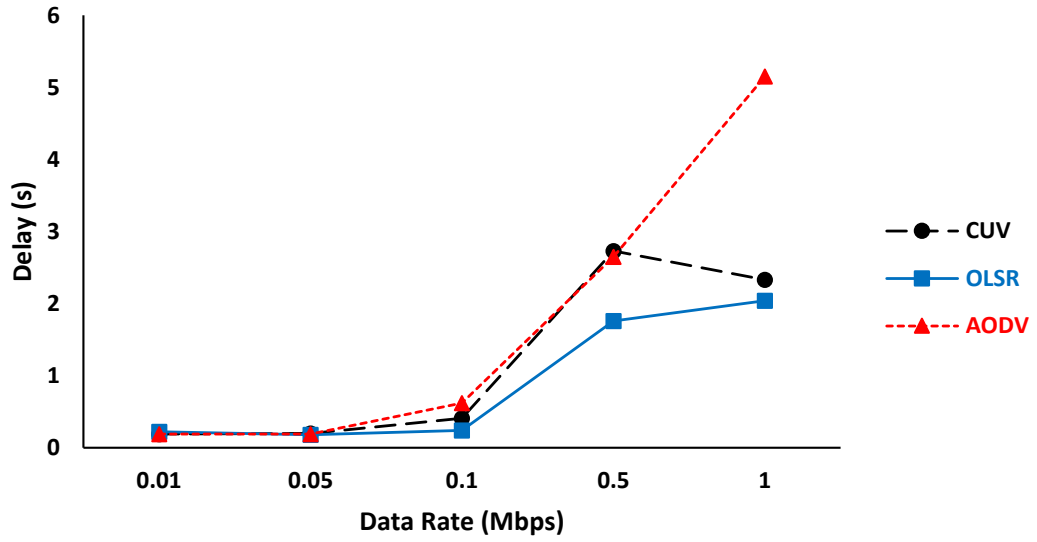


Figure 2.7: Average Delay - Collaboration Disabled

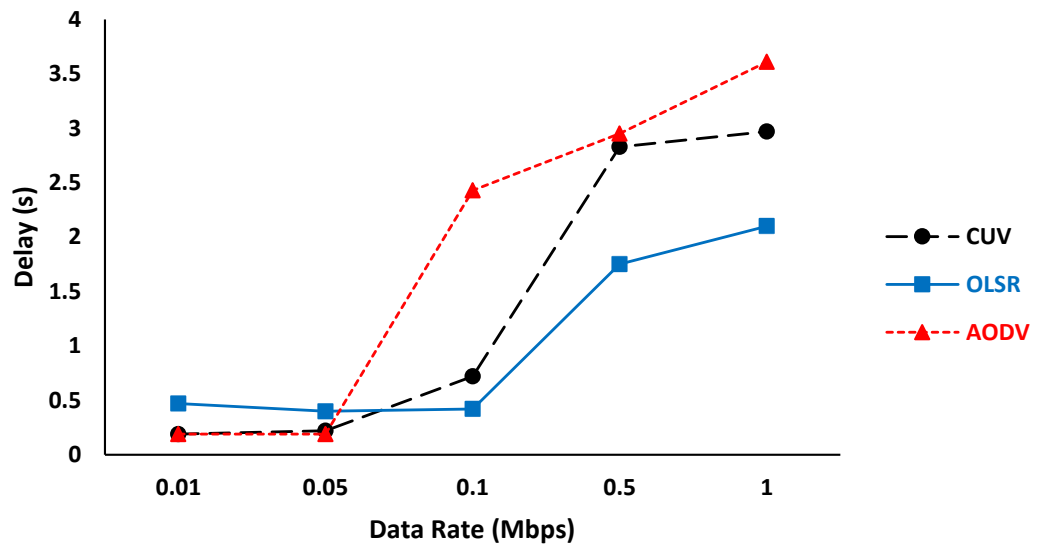


Figure 2.8: Average Delay - Collaboration Enabled

A distributed network recovery approach for unmanned aerial vehicular networks

Abstract

In multi-hop real-time systems, abrupt movement or early depletion of energy resources for some overloaded UAVs may result in the creation of a network hole or even in a breakdown of the whole network. In this chapter, a new Life-Enhancing recovery Approach for a Multi-UAVs (LEAMU) network is proposed that not only provides a routing solution but also serves as a fail-safe method. The crux of the LEAMU is the identification of the best recovering UAV since the selection of an unhealthy UAV will result in more recovery requests afterward. In LEAMU, network hole creation is avoided beforehand through a distributed election of a suitable candidate keeping distance, remaining energy, neighborhood density, and traffic load factors into consideration. The proposed strategy is simulated and has shown to have a promising future for its integration into the existing UAV systems.¹

¹N. Bashir, S. Boudjit, M. Y. Saidi. 2021, January. “A Distributed Anticipatory Life-Enhancing Recovery Approach for Unmanned Aerial Vehicular Networks,” In *Proceedings of IEEE 18th Annual Consumer Communications & Networking Conference (CCNC)*, Las Vegas, NV, USA, pp. 1–7.

Chapter content

3.1	Introduction	45
3.2	Related work	46
3.3	Proposed network recovery approach	47
3.3.1	Illustration of a UAV network breakdown scenario	47
3.3.2	The proactive routing strategy for a fleet of UAVs	48
3.3.3	Anticipatory network recovery algorithm	49
3.4	Performance evaluation	52
3.4.1	Simulation scenario I - under UAV's electronics or mechanical failures	52
3.4.2	Simulation scenario II - under UAV's depletion of energy resources	54
3.4.3	Simulation scenario III - under sudden and announced departures of UAVs	55
3.4.4	Average energy consumption due to communication overhead	56
3.5	Conclusion	58

3.1 Introduction

Despite the several advantages and expansion of its applications in several domains, UAVs having inherent limitations are still not utilized up to their full potential [74]. UAV's inadequate energy resources and short communication range are the major drawbacks that restrict the operational time and coverage area, respectively. The size constraints of a UAV itself prevents the maximum weight a UAV can carry that eventually resulting in shorter flight time [75].

In UAVs, maneuvering operation has the highest impact on energy consumption, leaving behind the communication part at second place [74]. Communication energy becomes the decisive factor considering all UAVs have a flight map with almost the same distance to cover. The communication load on any UAV largely depends upon the transmission distance and overhead generated due to the underlying MAC and routing protocols. The more is the transmission distance, the more will be the required transmitting power. Sometimes, a UAV close to a BS has to transmit its data packets while at the same time acting as a relay for far away nodes that results in more energy consumption as compared to other UAVs.

The development of robust energy-efficient communication or routing protocol is inevitable due to the unique characteristics (e.g., high mobility and limited energy resources) imposed by UAVs. During any ongoing operation, there is no guarantee for all UAVs to be depleted in energy at the same time. For example, in a scenario where a UAV, being a part of an active path, suddenly leaves the network. The leaving reason could be either the fall of energy below the threshold level or a sudden failure of the UAV. A routing protocol, like AODV [50], generates a router error message, and based on the availability of an alternate UAV, the broken path is recovered. A routing protocol fails to tackle a path breakage scenario when a UAV linking two parts of a network, called a cut-vertex UAV, dies out. In this situation, called network hole creation, a routing protocol remains unable to repair the damaged path due to the non-availability of nearby alternate UAVs. So, depending on the network topology, the removal of one or several more UAVs may blackout the entire network.

In this chapter, a new Life-Enhancing recovery Approach for Multi-UAVs (LEAMU) based networks is presented. LEAMU is a proactive routing approach that anticipates a network hole creation and takes necessary measures beforehand to prevent the system from a downfall. A cut-vertex UAV falling in its energy threshold level, called Leaving UAV (L-UAV), initiates a recovery request. A UAV responds to this request by broadcasting a Willingness Message (WM) in the entire network. This message contains distance (D) to L-UAV and its calculated Candidature Value (CV). While calculating CVs, each UAV keeps into its consideration its remaining energy

after joining L-UAV, neighborhood density, and data traffic load factors. The distance between L-UAV and its nearest candidate plus one transmission range distance defines a new election range. A UAV having the highest CV within this range qualifies for the replacement. Depending on the availability of nearby candidates, this election procedure is restricted to one-hop neighbors only. Direct one-hop neighbors take responsibility for the initiation of a recovery request during a sudden failure of UAV.

The remaining chapter is organized as follows. Section 3.2 presents the related work in this field. Section 3.3 presents the core functionality of the proposed strategy. In Section 3.4, the LEAMU scheme is evaluated. The chapter is concluded in Section 3.5.

3.2 Related work

Most of the existing routing protocols for UAVs do not consider path failures occurring as a result of the physical separation of UAVs. This section, at first, presents routing techniques for UAVs deprived of physical path repair capabilities, and this section concludes with some light thrown on connectivity repairing approaches.

Predictive-OLSR (P-OLSR) [76] is an extension of the OLSR protocol [52] that takes advantage of GPS coordinates to predict the link quality in flying ad-hoc networks. Hello message in the original OLSR is modified to share geographical position information among neighbors. Link quality information is included in the Topology Control (TC) messages that are distributed in the entire network. Unlike in OLSR, P-OLSR uses the direction and speed of the neighboring node along with the hop count factor in the route selection process. P-OLSR is suitable for rapidly changing topology and provides better multi-hop communication compared to OLSR.

Boids of Reynolds-AODV (BR-AODV) [77] is AODV based reactive routing protocol designed for UAV networks. In BR-AODV, AODV plays its part in route formation, while Boids of Reynolds ensures connectivity among all the UAVs along a path. This connectivity is maintained until a path is needed, after which UAVs are allowed to follow their predefined map. BR-AODV is designed for dynamically changing networks and avoids generating repetitive route discovery requests in the network.

In [78], the authors present a new fault-tolerant scheme for ad-hoc robotic networks. As robotic movement is controllable so this inherent feature is utilized to move a subset of robots to have a reliable fault-tolerant topology. This technique tries to reduce the effect of cut-vertices through the creation of new edges by utilizing a movement control algorithm. To ensure a biconnected graph, leaf nodes are moved to cut-vertices that result in an overall increase in connectivity and

an entire network's degree of fault-tolerance.

In [79], a Connectivity Recovery algorithm for UAV Networks (C^3 RUN) is presented. C^3 RUN is based on Cooperative Communication (CC) to allow a quick repair of connectivity for UAV networks. In CC, neighboring nodes are the helping nodes that send analogous packets to the destination so that a destination can decode them by combing partial signals. In C^3 RUN, with the use of CC, long-distance communication links are established between different separated parts of the network. The use of CC not only makes it possible to have a quick recovery but also allows nodes to move to better places for the establishment of better CC links.

Most of the classical protocols like AODV and OLSR lack node failure recovery features specifically during cut-vertex node failure or entire network breakdown scenarios. Some literature study is available on failure recovery for UAV networks, with most of them, like C^3 RUN, considering only distance-to-move parameter that can lead to the selection of UAVs having low energy resources. These unwise selections of recovery nodes will result in more recovery requests later. To avoid frequent recovery requests, LEAMU makes a robust decision by considering the most influential factors like energy, neighborhood density, and traffic load factor besides having a distance-to-move factor only.

3.3 Proposed network recovery approach

A highly dynamic UAV network imposes tough challenges for the designing of routing protocols. In this chapter, we propose LEAMU that aims to maintain connectivity by replacing the failing UAVs. LEAMU distinguishes between the sudden failures from the announced departures of UAVs (low battery, etc.). A sudden UAV failure separates the network into two parts that result in the selection of two replacing UAVs. Announced departures, on the other hand, will determine only one best replacing candidate among other UAVs from the entire connected network.

3.3.1 Illustration of a UAV network breakdown scenario

In this subsection, we illustrate the problem we addressed in this chapter through an example. In Fig. 3.1, a multi-UAV network with flying BS is considered. UAV **C-2**, at two hops away from BS, is a cut-vertex node. Any failure of this node will lead to the breakdown of the whole network. UAV **C-2** can affect the network in two ways i.e., (i) occurrence of sudden physical (electronics or mechanical) failure, or (ii) announced departure. LEAMU keeps an eye on the link status of cut-vertex UAVs and tries to resolve any existing or forthcoming failure with the replacement of a suitable candidate.

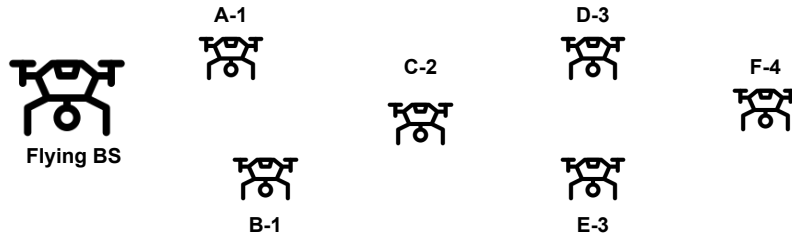


Figure 3.1: Hop no. representation in a UAV network

Table 3.1: Fields of Neighbors Table

Field	Definition
<i>Nbr_ID</i>	The unique address of neighbor
<i>Hop_No</i>	Hop number of neighbor with address <i>Nbr_ID</i>
<i>Link_Status</i>	A flag, having value 1 if <i>Link_Expire_Time</i> field is not expired yet, 0 otherwise
<i>Link_Expire_Time</i>	Expire time of link with neighbor (<i>Nbr_ID</i>)
<i>Nbr_Position</i>	Neighbor's (<i>Nbr_ID</i>) current position

3.3.2 The proactive routing strategy for a fleet of UAVs

In the proposed LEAMU scheme, soon after the network deployment, BS starts to broadcast periodic Heart Beat (HB) messages. These HB messages are similar to hello messages found in most routing protocols and are flooded in the entire network with BS as a root. In LEAMU, at each UAV, the neighbor table maintains the state of each direct neighbor. Table 3.1 presents the fields of Neighbors Table along with their respective definitions. HB message transmission phase is explained in Algorithm 4. In this algorithm, \mathcal{F} is a boolean variable having true value if the immediate sender of the HB message is already available in Table 3.1 of the receiving UAV, and false otherwise. The second boolean variable \mathcal{S} is true when a new sequence number HB message is received at the receiving UAV, and false otherwise. HB message \mathcal{P} contains the immediate sender's address, GPS coordinates, and the hop number information. For any UAV, *Hopno* variable keeps the number of hops value to reach BS.

BS sets its *Hopno* to "0" and broadcasts \mathcal{P} among its one-hop neighbors. Any UAV receiving this message updates *Nbr_ID*, *Hop_No*, *Link_Status*, *Link_Expire_Time*, and *Nbr_Position* fields in the Neighbors Table for the immediate sender of this message. If the

Algorithm 4 *Heart beat message transmission*

Input:1: $\mathcal{F}, \mathcal{S}, \mathcal{P}, Hopno$ **Output:** $Broadcast(\mathcal{P})$: Broadcast this packet \mathcal{P} among one hop neighbors2: **if** $\mathcal{F} = true$ **then**

3: Update neighbor entry in Table 3.1

4: **else**

5: Insert new neighbor entry in Table 3.1

6: **end if**7: **if** $\mathcal{S} = true$ **then**8: Update my own $Hopno$ 9: Update packet fields and $Broadcast(\mathcal{P})$ 10: **end if**

sequence number of the received message matches with the already received HB message, no further action is taken. However, after the reception of HB message with a new sequence number, $Hopno$ is updated with a value one higher than the received one and the HB message is rebroadcast among its one-hop neighbors while updating its fields. This advancement is continued until HB message is received by every UAV in the network. At the end of each HB message transmission phase, each UAV gets updated about its neighbors. The hop number assigned to each UAV by this process is presented in Fig. 3.1 as an example.

3.3.3 Anticipatory network recovery algorithm

Let U be the set of all UAVs in a network. We define a function h that returns, for each UAV u_n ($1 \leq n \leq N$), its hop number, wherein n is a UAV identity and N is the total number of UAVs in a network. We also define Z_n as a set of neighbor UAVs which belong to the u_n 's Table 3.1 entries. A node u_n is in the set of cut-vertex nodes B if its failure may result in a permanent disconnection, for at least one UAV, from the BS.

Any UAV u_n initiates a Recovery Message (RM) if $u_n \in B$ and its energy level falls below a certain threshold value or it detects a sudden failure of its neighboring UAV belonging to set B . RM contains L-UAV's GPS coordinates and its identity as a sequence number. Broadcasting of RM serves two purposes, i.e., (i) notify all UAVs to get ready for participation in replacement election, and (ii) UAVs get recent information about their neighbors. RM transmission procedure is presented in Algorithm 5. In this algorithm, \mathcal{R} becomes true whenever a node gets its first

Algorithm 5 *Recovery message transmission*

Input:

1: $\mathcal{R}, \mathcal{RM}, \mathcal{D}$

Output: $Broadcast(\mathcal{RM})$: Broadcast this packet \mathcal{RM} among one hop neighbors

2: Update neighbors information for Nbr_ID in Table 3.1

3: **if** $\mathcal{R} = false$ **then**

4: **if** $\mathcal{D} = false$ **then**

5: $Broadcast(\mathcal{RM})$

6: **end if**

7: $\mathcal{R} \leftarrow true$

8: **end if**

unique sequence recovery message \mathcal{RM} from any of its neighbors. \mathcal{D} is a variable having true value if a UAV is a direct neighbor of L-UAV and at least have one equal hop number UAV within its neighbors, and false otherwise. After reception of the first \mathcal{RM} , each UAV having false \mathcal{D} value rebroadcasts the received message within its one hop neighbors.

After the RM transmission phase, a UAV u_n with $u_n \notin \mathcal{B}$ shows its willingness to replace L-UAV through broadcasting WM in the whole network. This message is composed of its distance to L-UAV and CV. The Haversine formula, presented in equation 3.1, is used to calculate the great-circle distance D_n from UAV u_n to L-UAV. In this equation, Φ_n and Φ_L are the latitudes (in radian) of u_n and L-UAV, respectively. R is the earth radius, while, ϕ_n and ϕ_L are the longitudes (in radian) of u_n and L-UAV, respectively.

$$D_n = 2R \arcsin \left\{ \sin^2 \left(\frac{\Phi_n - \Phi_L}{2} \right) + \cos(\Phi_n) \cos(\Phi_L) \sin^2 \left(\frac{\phi_n - \phi_L}{2} \right) \right\}^{1/2}, \quad (3.1)$$

For UAV u_n , MT is the time required to go from u_n 's current position to L-UAV's position and once the D_n is calculated, $(MT)_n$ can be calculated using equation 3.2, wherein v is the recovery speed.

$$(MT)_n = \frac{D_n}{v}, \quad (3.2)$$

E_{D_n} is energy that will be utilized by u_n in going from u_n 's current position to intended L-UAV's position. This energy consumption can be computed using energy model presented in [80]. Assuming u_n is moving horizontally with constant speed v . The required thrust T_n is given by

equation 3.3.

$$T_n = \sqrt{(mg)^2 + \left(\frac{D_a A_f v^2 C_d}{2}\right)^2}, \quad (3.3)$$

In equation 3.3, m is the mass of a UAV, g is gravitational acceleration, D_a is the density of air, A_f is the front cross-sectional area of a UAV, v is horizontal recovery speed, and C_d is a drag coefficient. Now, the power P_n required to generate the thrust T_n , is given by equation 3.4.

$$P_n = T_n \times v, \quad (3.4)$$

So, E_{D_n} utilized during the time $(MT)_n$ is given by equation 3.5.

$$E_{D_n} = P_n \times (MT)_n = T_n \times v \times (MT)_n, \quad (3.5)$$

From Eq. 3.2 and Eq. 3.5, equation 3.6 can be derived as

$$E_{D_n} = T_n \times D_n, \quad (3.6)$$

Now, CV at UAV u_n is calculated using equation 3.7 in which $E_{R_c}(n)$ and $E_I(n)$ are the current remaining and initial energy for u_n , respectively. $P_C(n)$ is the number of packets currently transmitted by u_n and P_{Max} is the total number of packets allowed to transmit by any UAV. P_C value is reset after UAV system moves to a new position from the predefined positions set. $N_T(n)$ is the total number of active neighbors for u_n and N_{Max} is the total number of allowed neighbors restricted by topology. In equation 3.7, α , β , and γ are the weight factors.

$$(CV)_n = \alpha \left(\frac{E_{R_c}(n) - E_{D_n}}{E_I(n)} \right) - \beta \left(\frac{P_C(n)}{P_{Max}} \right) + \gamma \left(\frac{N_T(n)}{N_{Max}} \right), \quad (3.7)$$

The highest priority given to α means the election of a reliable candidate with an excessive amount of remaining energy. More preference to β elects a UAV having less involvement in data transmission that ultimately leads to the lesser effect on other mostly used links. A topology is less likely to be affected if a candidate with the highest neighborhood density (i.e., higher γ) is selected. Removing a UAV from a higher neighborhood density also reduces overhearing wastage for the removed UAV.

Soon after the calculations of D_n , $(MT)_n$, and $(CV)_n$, each UAV u_n with $u_n \notin B$ broadcasts WM in the entire network. Each UAV maintains a 3-tuple $(n, (CV)_n, D_n)$ to record WMs received by all other UAVs. In this 3-tuple, n is the address of UAV that originated this WM. All intermediate UAVs receive, record in their 3-tuple, and rebroadcast the received message further. After the exchange of WMs, the distance between L-UAV and the nearest candidate plus one

transmission range becomes a new election range. A candidate having the highest CV within this range qualifies for the selection. The selected candidate moves autonomously to the position of L-UAV with the coordinates already provided by the RM. During the availability of nearby replacing candidates, the election mechanism restricts itself solely to direct neighbors of L-UAV.

3.4 Performance evaluation

To justify the assumptions made by the proposed LEAMU approach and to investigate its performance compared to a connectivity recovery protocol, we simulated LEAMU and C³RUN [79] in Network Simulator-2. The assessment of the protocols is based on three metrics: (i) number of packets received at BS with time to validate the end-to-end link continuity, (ii) distance moved by replacing candidates to reach L-UAV from their current location, and (iii) average communication overhead in terms of energy.

In this simulation, a predefined topology set of UAVs is considered with single ground base station deployment. Simulation with different topology baseline will not affect the validation process as long as the network contains cut-vertex edges. Total 17 UAVs were taken for the obtained results. The performance evaluation contains results for three experiments run under different scenarios. All sudden physical (electronics or mechanical), all energy depletion (announced departures), and the mixture of the first two failures are considered in Experiments I, II, and III, respectively. Values 0.6, 0.2, 0.2 were taken for α , β , and γ , respectively. Each UAV is equipped with an omni-directional antenna. For the simulation of these protocols, *Two Ray Ground propagation* model is used for communication. The remaining simulation parameters are listed in Table 3.2.

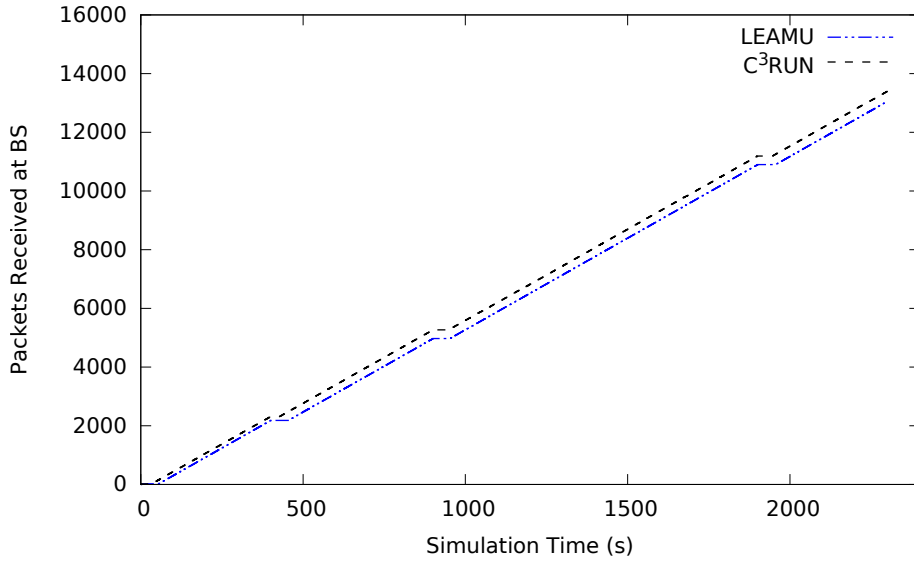
3.4.1 Simulation scenario I - under UAV's electronics or mechanical failures

All the failures in this experiment occur due to the removal of UAVs owing to electronics or mechanical failures. To evaluate the end-to-end link continuity, we plotted the number of packets received at BS against simulation time in Fig. 3.2. Any horizontal line in the plot indicates link breakages while any non-zero slope line indicates data packet reception at BS, and eventually, the continuity of end-to-end link. LEAMU performance in this experiment is comparable to C³RUN as it aims for the best candidate even if it has to select a far away UAV that results in a longer repairing time. Moreover, due to sudden failures of UAV, the anticipatory effect of LEAMU gets nullified. C³RUN, on the other hand, makes a quick recovery because it selects

Table 3.2: *Simulation parameters*

Parameter	Value
Network size	4000 m x 4000 m
Speed of UAV during recovery	15 m/s
Transmission range of each node	500 m
Size of each packet	200 Bytes
Traffic Type	CBR
Simulation time	2300 sec
MAC protocol	MAC/802.11p

the first nearest available candidate. For each specific failure, Fig. 3.3 shows the number of

**Figure 3.2:** *Number of Packets Received at BS with Time (Exp. I)*

replacing UAVs moved along with the total distance travelled for replacement. In this figure, *Failure-1* results in a displacement of only one UAV for both the compared schemes. Replacing UAV travels 302 m and 611 m for C³RUN and LEAMU, respectively. For the case of *Failure-2*, LEAMU's first candidate travels a distance of 618 m and the second candidate travels 872 m, making 1490 m in total. C³RUN, on the other hand, selects its first and second candidates at 290 m and 850 m, respectively, making a total of 1140 m. For *Experiment I* with four registered failures, both schemes result in the movement of six UAVs and a total distance of 3712 m and

3252 m for LEAMU and C³RUN, respectively. Due to the technological growth in the domain of UAVs, these kinds of sudden failures occur very rarely.

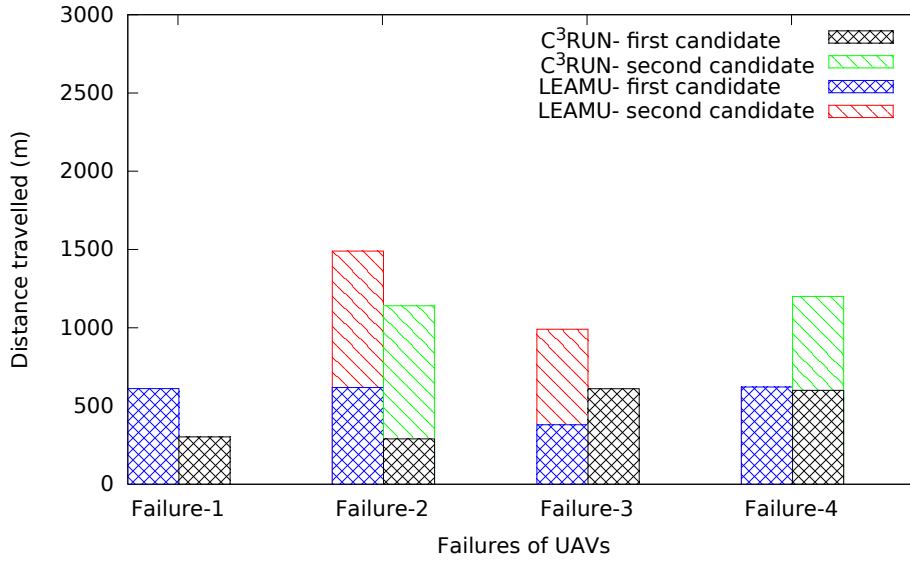


Figure 3.3: Distance travelled by replacing UAVs (Exp. I)

3.4.2 Simulation scenario II - under UAV's depletion of energy resources

This experiment considers UAV's failure arising due to the depletion of energy resources owing to which it leaves the network (announced departure). Fig. 3.4 shows the end-to-end link continuity by plotting the number of packets received at BS against simulation time. LEAMU exploiting its anticipatory feature up to its full potential can be inferred from this figure. LEAMU keeps an eye on the energy resources of all the cut-vertex UAVs and initiates a recovery before they leave the network. Fig. 3.4 verifies that despite having five UAVs failures, LEAMU ensures continuity of the end-to-end link because it anticipates the failure occurrence and takes action in advance. C³RUN, on the other hand, initiates a recovery at the time of failure that results in a discontinuity of the end-to-end link.

Fig. 3.5 shows the distance moved by replacing candidates in *Experiment II*. The positive aspect of LEAMU's anticipatory recovery is the availability of the whole network connectivity during the replacing candidate election mechanism. LEAMU selects only one suitable candidate from the entire network and replaces the leaving UAV before its actual departure. C³RUN finds the network partitioned into two clusters during a failure scenario. It selects one or two replacing UAVs depending upon the availability of candidates in both the clusters. During the five failures in this experiment, replacing UAVs in LEAMU and C³RUN travelled nearly 3800 m distance

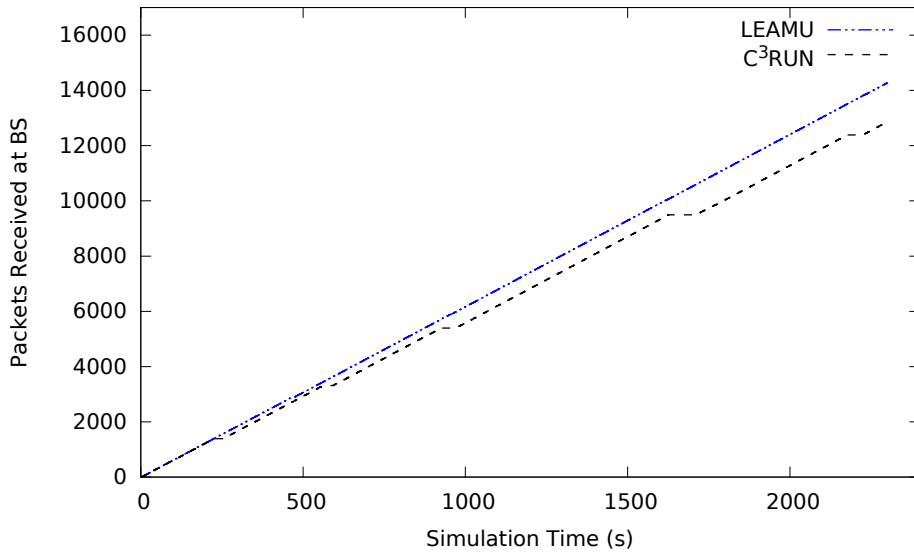


Figure 3.4: Number of packets received at BS with time (*Exp. II*)

with five and six UAV movements, respectively.

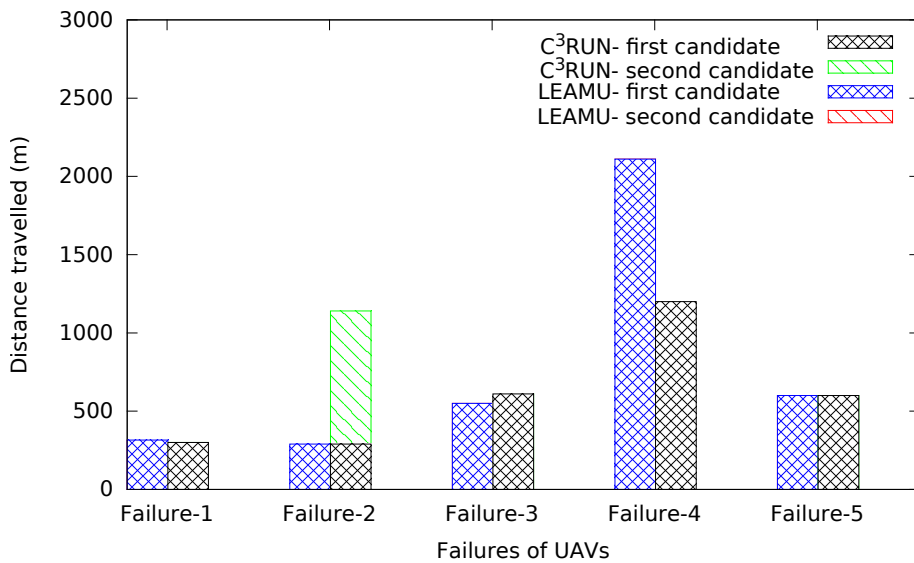


Figure 3.5: Distance travelled by replacing UAVs (*Exp. II*)

3.4.3 Simulation scenario III - under sudden and announced departures of UAVs

To make the simulation replica of a real-world scenario, we mixed two failures, i.e., sudden and announced departures, in this experiment. Fig. 3.6 depicts the end-to-end continuity for this experiment for both the simulated schemes. This figure reveals the effectiveness of the anticipatory part and robustness of candidate selection in LEAMU. Fig. 3.6 shows that LEAMU

has more continuity of the end-to-end link as compared to C³RUN in which a recovery mechanism initiates only with the occurrence of a UAV failure. LEAMU, on the other hand, anticipates the failure of a cut-vertex UAV and brings a replacing UAV near to it before its actual departure. The moment the failing node leaves the network, a replacing UAV takes charge that results in the continuity of the end-to-end link.

Fig. 3.7 presents the total distance travelled by the replacing UAVs. LEAMU and C³RUN travel

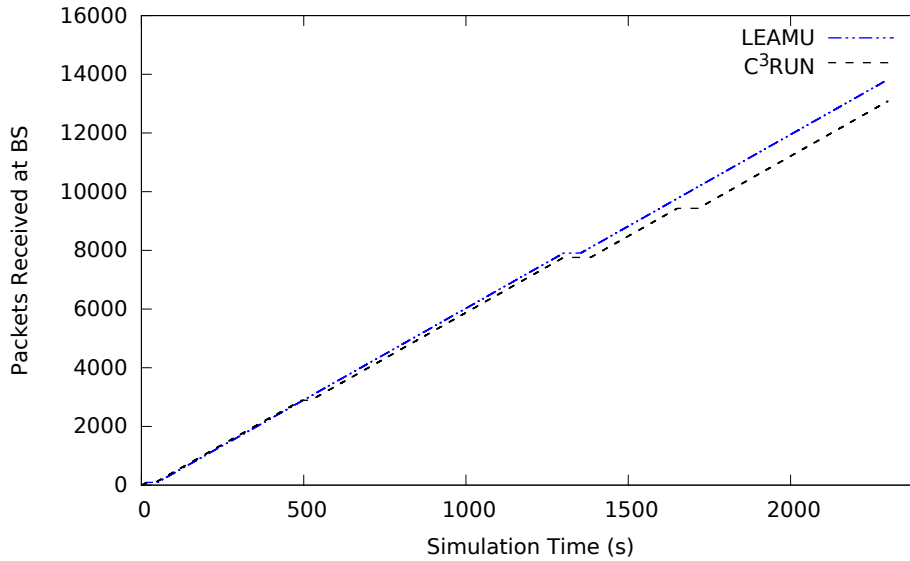


Figure 3.6: Number of packets received at BS with time (*Exp. III*)

2377 m and 4735 m total distance along with 5 and 6 UAV movements, respectively. C³RUN selects the nearest available options without considering their remaining energy resources which results in an another recovery request at a later time. For *Failure-1* in Fig. 3.7, the sudden failure occurrence results in the selection of two replacing candidates by LEAMU. For the following cases, the anticipation feature provides the opportunity to do the election campaign within the entire connected network resulting in the selection of only one robust candidate.

3.4.4 Average energy consumption due to communication overhead

In Fig. 3.8, we plotted average energy consumptions of communication overhead for the three experiments to assess the burden imposed by the simulated recovery schemes on the network. LEAMU scheme turns up to be an energy-efficient solution owing to be having just HB messages with BS as a root. C³RUN, on the other hand, periodically transmits its neighbor's information among its direct neighbors in addition to periodic hello messages. LEAMU shows almost the same energy consumptions expect in *Experiment II* in which all the candidates transmit their

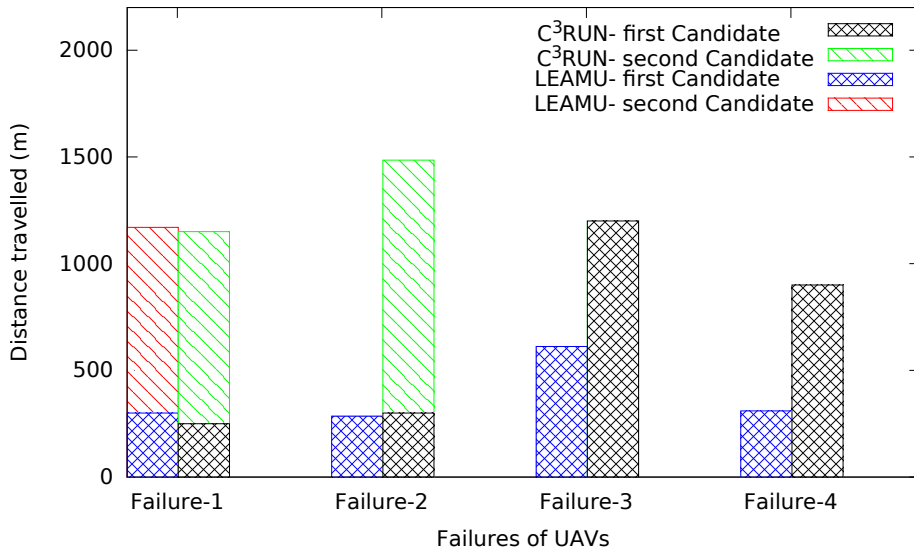


Figure 3.7: Distance travelled by replacing UAVs (*Exp. III*)

CV value in the entire network resulting in a bit higher energy consumption.

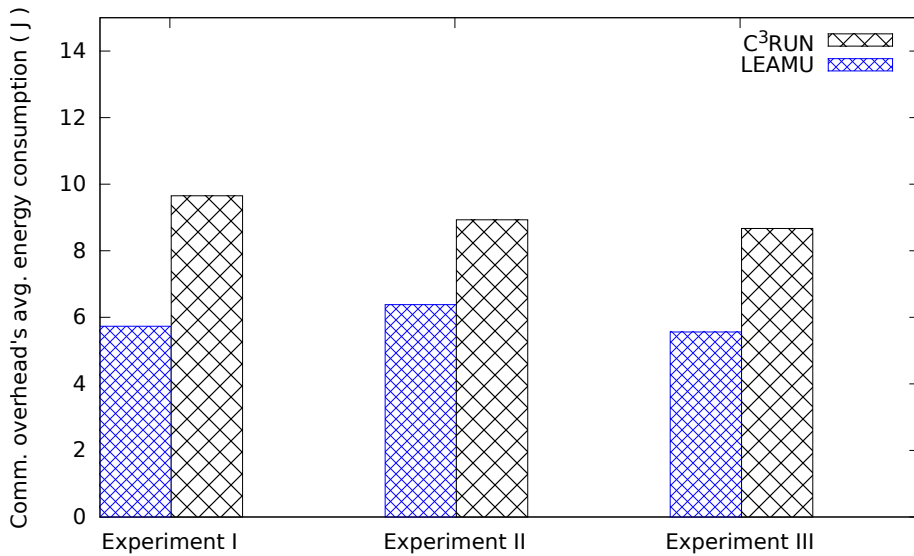


Figure 3.8: Average energy consumption of communication overhead

The performance of LEAMU remains better as compared to C³RUN except for *Experiment I* in which it shows comparable results. Thanks to technological advancements in the domain of UAVs, sudden failures (*Experiment I*) occur very rarely that leads us to conclude that LEAMU's overall performance is better in all failure scenarios compared to C³RUN.

3.5 Conclusion

Network recovery in a distributive manner is the cornerstone of the proposed LEAMU scheme that is a desirable feature in AdHoc networks. LEAMU not only provides routing services but also acts as a fail-safe method during UAV failure scenarios. The anticipation of a forthcoming UAV failure and replacement of a suitable candidate in advance ensures a better end-to-end link continuity. This feature also facilitates LEAMU to run the election campaign in the entire network and come up with a single robust candidate to avoid frequent recovery requests arising due to the selection of unhealthy replacing UAVs.

In future work, we are aiming to propose a recovery scheme that will deal with more than one failure at the same time. Differentiating between many recovery requests arising at the same time by their respective sequence numbers may help in the realization of this extension.

Part II

UAV and WSN collaboration to enhance connectivity and surveillance efficiency

A self-governing collaborative architecture of UAV and WSN for dynamic surveillance

Abstract

This chapter proposes a unique closed-loop control architecture for traffic surveillance on highways comprised of Wireless Sensor Network (WSN) and UAV, named Collaborative Highway Surveillance (CHS). The proposed architecture works without human intervention and enables the capture of over-speeding drivers on a highway. All nodes in the WSN act as a relay except some nodes, placed at hotspots where over-speeding is likely, have additional functionality of speed sensing. A WSN not only helps in providing routing services to UAVs but can also dynamically guide the UAV to the best hotspot to position itself. The UAV is the entity that also detects vehicles' over-speeding. If any vehicle exceeds the speed limit allowed, the UAV immediately informs this event to the Mobile Base Station (MBS). However, to conserve the energy resources of the WSN, low-level speed violations are forwarded to the MBS only when the UAV gets closer to the MBS. Performance results obtained with our proposed architecture show an increase in surveillance efficiency with an improved response time compared with cooperative and stand-alone unguided UAV networks.¹

¹ N. Bashir, S. Boudjit, S. Zeadally. 2022. "A closed-loop control architecture of UAV and WSN for traffic surveillance on highways," *Computer Communications*. [Accepted 8 Apr 2022]

Chapter content

4.1	Introduction	63
4.2	Related work	65
4.3	UAV-WSN collaborative system's description	67
4.3.1	Wireless sensor network hierarchy	69
4.3.2	Working of the proposed collaborative surveillance system	69
4.3.3	Probabilistic based trajectory control model for UAV	69
4.3.4	Hop number assignment and immediate reporting mechanism	72
4.4	Performance evaluation	73
4.4.1	Simulation scenario I - hotspot location 2.0 km away from BS	74
4.4.2	Simulation scenario II - hotspot location 3.9 km away from BS	76
4.4.3	Simulation scenario III - hotspot location 5.3 km away from BS	77
4.4.4	Simulation scenario IV - with varying hotspot locations	77
4.5	Conclusion	80

4.1 Introduction

Current surveillance applications employing UAVs are not utilizing them at their full potential because of the limited on-board carrying capacity of batteries, communication range, and inefficient architecture [81]. To have a real-time response while considering the limited transmission range of a UAV, single-UAV based surveillance systems need to have a Mobile Base Station (MBS) nearby. Additionally, it becomes challenging to pursue an over-speeding vehicle by a UAV due to its speed limitations. On the other hand, swarm-based UAV systems have also proposed in the literature [17], but having a higher number of UAVs, just for the sake of connectivity, makes the system expensive. For the highway surveillance scenario, a connected fleet of UAVs will produce a lot of redundant data but a sparse topology, on the other hand, will lead to a delay-tolerant network.

A Wireless Sensor Network (WSN) is an arrangement of sensor nodes to measure different physical parameters such as temperature, vibrations, humidity, and many more. Sensing, transceiver, processing, and power units constitute a low cost embedded sensor node. A WSN is known for its limited communication and scarce energy resources [82]. Collaboration schemes exist in the literature between WSN and UAV networks to better deal with these shortcomings [83, 84, 85, 86, 87]. Integration of these two aforementioned technologies can be very useful for highways surveillance applications, wherein a WSN could provide different services to a UAV network.

In this chapter, we propose a closed-loop control architecture for traffic surveillance on highways. The architecture integrates UAV and large scale WSN networks. The proposed idea, called Collaborative Highway Surveillance (CHS), inspired by closed-loop control systems wherein a feedback signal (actual captured violations) and the reference input (expected violations) act as system excitation to regulate the desired output of detecting² a maximum number of violations that exceed the speed limit. All nodes are placed along the highways including some of them, equipped with speed-sensing capability too, at hotspot locations. These hotspots are areas where over-speeding is most likely to occur or where we need to enforce speed restrictions. A WSN, as a whole, not only provides routing services but also facilitates a UAV in targeting the best hotspot. The UAV is the entity that identifies over-speeding vehicles with the help of a high-resolution speed camera on board. The objective of the CHS is not to pursue an over-speeding vehicle but to catch future over-speeding vehicles around the hotspot locations. To prevent rapid energy resource depletion of the underlying WSN, only high-speed violators are reported immediately.

²It is worth mentioning here that a UAV in CHS does not follow an over-speeding vehicle due to the speed limitations of a UAV and instead informs the MBS to act quickly.

Less severe speed violations, on the other hand, are reported only once the UAV gets closer to the MBS.

Research contributions of this chapter:

The main contributions of the proposed UAV-WSN collaborative scheme as follows.

- **Maximize the detection of speed violations:** The proposed scheme considers over-speeding hotspots and aims to detect the maximum number of speed violations on highways. The CHS takes into account the limited energy resources of a wireless sensor network and reports only severe speed violations.
- **Real-time reporting:** To avoid rash driving consequences, severe speed violations are reported immediately to the mobile base station thanks to the path pro-actively maintained by the wireless sensor network.
- **Autonomous operation:** The wireless sensor network guides the UAV by providing input for the probabilistic-based trajectory model which is used in the autonomous real-time path planning and tracking for the UAV.
- **Cost-effectiveness:** The CHS is a cost-effective scheme for highway surveillance because it involves a single UAV and inexpensive sensor nodes compared to existing solutions that involve expensive roadside units, a helicopter, and a ground pilot to operate the UAV.
- **Efficacy of the CHS:** We evaluate the proposed approach using several performance metrics to demonstrate its efficacy. However, we would like to point out that there is no previous work that has combined UAV and wireless sensor network in such a unique way for traffic surveillance. To demonstrate the effectiveness of the proposed scheme, we compare it with closely related cooperative and non-cooperative UAV based road traffic surveillance schemes. The CHS succeeds in detecting the maximum number of violations and avoids unnecessary movements between the hotspots.

The rest of the chapter is organized as follows. Section 4.2 presents related works on WSN, UAV, and their integration. Section 4.3 describes the main framework of the proposed strategy. Section 4.4 presents an evaluation of the proposed architecture. Section 4.5 makes some concluding remarks.

4.2 Related work

This section highlights recent results of research efforts that have focused on the monitoring of areas while utilizing UAVs and their integration with other networks. Additionally, we also identify both the strengths and weaknesses of past approaches aimed at highways' surveillance.

In [88], the authors developed an Airborne Traffic Surveillance System (ATSS) mainly based on UAVs and microwave IP networks for the dissemination of highways surveillance data to BS. After capturing the video, the UAV transmits it to the two towers. These towers are deployed along the highway and equipped with a computer to encode the video received. The encoded video is then transmitted to the BS using a microwave network. Besides capturing video, the UAV also captures any other required information such as related to the weather. This technique is not suitable for traffic surveillance in terms of its flexibility of deployment as well as the cost associated with the whole system.

To perform UAV traffic surveillance, the authors of [89] proposed a video relay model which uses existing public networks such as Mobile Broadband Networks (MBNs). Video is transmitted directly to the ground BS available in the vicinity of the UAV. The ground BS uses a wireless card to connect with a MBN. The end-user has to be aware of the BS's IP address as it keeps on changing each time it connects with the network. In another scheme proposed by the same authors [89], a server with a fixed IP acts as a host of the data communication link. The IP address of the server is known beforehand to the end-user as well as to the ground BS. This scheme is dependent on the availability of nearby ground stations as well as access to a broadband network.

The authors of [4] proposed a three-layered architecture for road surveillance. Layer 1 is comprised of a practical implementation of UAV deployment on the highway. The deployed UAV is equipped with speed sensing cameras to capture and report to the mobile police. A first-time speed violation results in a warning to the driver while a second-time violation results in the issuance of a ticket. The video, along with GPS location, is recorded and sent to the mobile police's BS. The second layer, based on 5G technology, is the communication bridge between the first layer and the mobile police's BS. The third layer is the actual layer that handles the highway traffic. Cellular network planning becomes more complex due to higher mobility, line-of-sight interference, and energy constraint associated with UAV all of which make this scheme non-viable currently.

In [90], the authors proposed a multi-UAV cooperative traffic monitoring scheme with two new methods. The first method tries to cover as many vehicles as possible, and the second one tries to detect a high number of events such as the position and speed of vehicles. Trajectories of the

UAVs are adaptive in the first method and mainly based on the movement of the targeted group of vehicles as a whole. The second method, on the other hand, has trajectories based on mobility models of vehicles. Overall, this multi-UAV scheme lacks real-time support to detect and report speed violators to the mobile police's BS.

In [91], the authors integrate UAV and WSN technologies in their proposed architecture wherein, most of the time, the UAVs act as a data collecting unit while the WSN acts as a sensing unit. WSNs use small size and low-cost sensor nodes that are deployed in many collaborative distributed applications [92]. The authors of [83] proposed a UAV-WSN collaborative scheme to increase the surveillance area and enhance the overall performance of the UAV-WSN system. Sensor nodes deployed over a large area gather environmental data, and later, the UAV collects this information from the sensor nodes. To design an optimal trajectory with predefined way-points, guaranteed communication time along with the smallest total path length, is a key objective of this scheme. This WSN-UAV integration also helps in sensor localization and clustering for optimal ground coverage. This scheme lacks a real-time aspect rendering it unsuitable for traffic surveillance systems.

In [93], the authors developed an efficient road safety framework based on Hybrid Sensor and Vehicular Network (HSVN). They proposed three solutions in this scheme to extend the WSN lifetime while ensuring the minimum delivery delay. First, the WSN deployed on the road is divided into clusters to reduce communication overhead. Second, cluster heads are chosen based on the location of the sensors and their remaining energies. Third, dynamic sleeping periods are used to conserve the limited energy resources of sensor nodes. This technique is dependent on a roadside unit to transmit any detected events to BS.

In [94], the authors proposed a Driving Guidance System (DGS) based on WSN that helps drivers to drive carefully. DGS consists of two sub-systems i) a weather information sub-system and, ii) a speed measurement sub-system. Drivers are informed about the current road conditions (e.g., icy, wet, and so on) in addition to weather information. Sensors deployed along the road detect the speed of vehicles, and the speed camera installed at the roadside detects any over-speeding vehicle. In S3 [95], the authors described a school zone safety system that is also based on WSN. S3 monitors the speed limits and prevents illegal parking around the school zone. These schemes rely on roadside units and back-haul connectivity to transmit any observed events to BS.

In chapter 2, we proposed a Collaborative scheme between UAVs and VANETs (CUV) for road surveillance. This approach works with a predefined fleet trajectory and relies on vehicles on the ground during failure scenarios.

Fargeas et al. [96] proposed a cooperative surveillance system that can pursue vehicles on roads using mutually disconnected ground sensors and multiple UAVs. UAVs, when used as a stand-alone vehicle, cannot detect intruders and must seek guidance from the deployed ground sensors. This approach uses revisit deadlines to create paths for surveillance and requires a UAV to get into the proximity of the sensor to get local information. The UAV informs the BS after getting informed about the intruder, and then the appropriate UAV pursues the intruder. The precise definition for revisit time and the delay incurred in getting information from the separate sensor network make this scheme less effective.

On highways, a WSN can detect an over-speeding vehicle but it requires collaboration with other systems for complete identification. A WSN is not adequate as a standalone entity in this application domain, and its potential becomes limited and costly because of its dependence on roadside infrastructures such as fixed camera towers or cellular networks. It is worth pointing out that since a vehicular ad-hoc network is still an emerging technology, a UAV network in conjunction with a vehicular network, is not seen as a practical, feasible solution right now.

Table 4.1 presents a brief comparison of various UAV-based highway traffic surveillance schemes in terms of their weaknesses, strengths, and the kind of trajectory used.

4.3 UAV-WSN collaborative system's description

The proposed CHS architecture works as a closed-loop feedback control system. Fig. 4.1 illustrates the operation of CHS wherein the *Process* represents captured violations by the UAV. The system uses these violations (actual) and the violations reported by the WSN (expected) as an input. To achieve the desired state of capturing the maximum number of violations, the probabilistic based controller then places the UAV at the best hotspot.

In this section, we describe our proposed highway surveillance scheme along with complete details of the collaborative interaction between the WSN and the UAV.

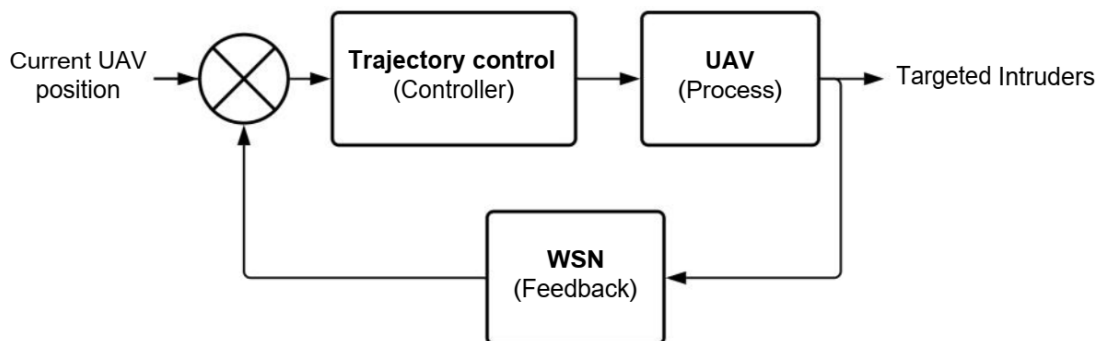


Figure 4.1: Operation of CHS as a closed-loop control system

Table 4.1: Comparison of various UAV based traffic surveillance schemes for highways

Technique	Strengths	Weaknesses	Surveillance trajectory
Srinivasan et al. [88]	Deployment flexibility Lower transmission delay Best video stream selection	Deployment cost Dependency on other network operators Limited coverage range	Fixed
Chen et al. [89]	Ease of deployment Live or off-line video transmission	Deployment cost Dependency on other network operators Limited coverage range	Fixed
Khan et al. [4]	Deployment flexibility Real-time traffic monitoring and reporting	Dependency on 5G network Line-of-sight interference	Fixed
Elloumi et al. [90]	Better coverage and event detection rate	Lack of real-time support Need to share information among different UAVs	Adaptive based on vehicles mobility models
Bashir et al. [97]	Robust to UAV failures Real-time transmission of surveillance data	Deployment of roadside BS Dependency on vehicular networks in case of failure	Fixed
Fargeas et al. [96]	Cooperative surveillance and pursuance of over-speeding vehicles	Disconnected sensor network UAV needs to visit every sensor to get local information Definition of revisit deadline is not clear	Fixed but adaptive for pursuance of over-speeding vehicles
CHS (Proposed scheme)	Autonomous real-time path planning Real-time reporting of severe violations	Limited WSN resources	Probabilistic based dynamic trajectory

4.3.1 Wireless sensor network hierarchy

In CHS, the WSN network plays a dual role in routing as well as guiding the UAV to the best place to position itself. Based on the part played by a wireless node, we introduce a two-level hierarchy wherein Reporting Nodes (RNs) are at the top of the WSN hierarchy. RNs, having speed sensors, are placed alongside the road where speed restrictions have to be maintained or around possible over-speeding areas. Sensors in RNs are assumed to be of the same type as in [94, 98]. These sensors work by measuring the disruption in the Earth's magnetic field caused by vehicles. Anisotropic magneto-resistive magnetic field sensor measures these distortions. Helping Nodes (HNs), without any sensor, are second in the hierarchy and provide connectivity for both RNs and the UAV to the MBS.

4.3.2 Working of the proposed collaborative surveillance system

As Fig. 4.2 shows, UAV, RNs, HNs, and MBS are the main components of the proposed CHS scheme. The UAV monitors the speed of vehicles using a speed-sensing camera. Unlike other surveillance techniques, a UAV does not follow a predefined trajectory. RNs guide the UAV in CHS. Each time an over-speeding vehicle passes over an RN, an Over-speeding Counter (OC) increments its value. At each RN, the OC value is reset and reported periodically to the MBS and the UAV. Depending on the OC value, the UAV decides to stay either at the current position or move to a new over-speeding hotspot. RNs are capable of detecting and reporting over-speeding vehicles, but the absence of a speed camera on these nodes makes it impossible to identify those vehicles. In CHS, the UAV does the actual detection and identification of speed limit violators. Based on the vehicles' over-speeding intensity, the UAV decides on how to communicate with the MBS. Severe speed violations³ are reported immediately by the UAV to let the MBS act quickly with the over-speeding vehicles. To save the energy of the WSN, less severe violations, on the other hand, are only reported when the UAV is near to the MBS.

4.3.3 Probabilistic based trajectory control model for UAV

In this subsection, we introduce a probabilistic model to control the flight trajectory of the UAV in CHS. Periodic reporting of OC values can be treated as a Poisson experiment in which the OC value is taken as a mean success value over a certain period. Equation 4.1 represents

³In this chapter, a vehicle with 50% more speed than the specified speed limit is considered a severe speed violation. However, the value of this limit is beyond the scope of this chapter. It is usually fixed by the concerned authorities (i.e., by the police).

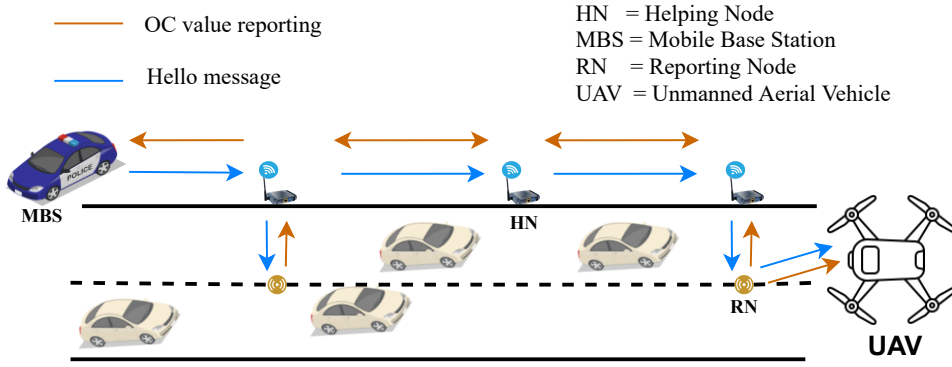


Figure 4.2: *The architecture of CHS*

the Poisson distribution function for a random variable Y . In this equation, λ is a mean success value, e is the Euler's number, and y is the actual number of successes in a specified period.

$$P(Y = y) = \frac{\lambda^y e^{-\lambda}}{y!}, \quad (4.1)$$

T is a set of all reporting time intervals $\{t_0, t_1, t_2, \dots, t_n, \dots, t_e\}$ of OC values by RNs, wherein $t_n - t_{n-1}$ is the period for reporting, t_e is the last reporting time and eventually the end of the simulation. The OC values received at t_{n-1} become the mean value for calculating the probability at t_n with the recently received value as a potential candidate for y . The highest OC value received at t_n is used as a success value, i.e., y for the probability calculation at t_n .

OC_{rt_n} is the OC value received by the UAV at t_n , wherein $t_n \in T$, and r is the identity of RN. The UAV calculates the Poisson probability for each RN after receiving the OC values from all RNs. We assume that OC_{max} is the highest OC value received at the UAV at t_n . Equation 4.2 gives the probability of receiving less than OC_{max} over-speeding vehicles at RN r .

$$P_r(Y < OC_{max}) = \sum_{y=0}^{OC_{max}-1} \frac{(OC_{rt_{n-1}})^y \times e^{-OC_{rt_{n-1}}}}{y!}, \quad (4.2)$$

Our aim is to find the probability P_r such that an RN r will receive at least OC_{max} over-speeding vehicles which equation 4.3 gives as.

$$P_r(Y \geq OC_{max}) = 1 - P_r(Y < OC_{max}), \quad (4.3)$$

The probability calculations are the same (i.e., using equation 4.3) for all RNs except the one which has the UAV in its vicinity. In Fig. 4.3, we assume that RN-2 has the highest probability of getting over-speeding vehicles within the next reporting period, and the UAV decides to move

from RN-1 to RN-2 location. The UAV will miss all over-speeding vehicles around RN-2 while traveling a distance of $d_{1,2}$ toward it until it reaches RN-2. So, the calculation for the probability of RN, currently having the UAV in its proximity, should include the time to move to other RNs.

Consider two RNs separated by a distance $d_{r,x}$, wherein r the identity of an RN having the

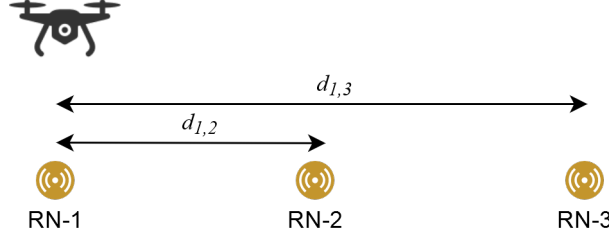


Figure 4.3: Distance between RN-1 and other reporting nodes

UAV nearby whereas x is any other RN with identity x . Equation 4.4 gives the time $t_{r,x}$ required to travel through distance $d_{r,x}$ with a drone speed v .

$$t_{r,x} = \frac{d_{r,x}}{v}, \quad (4.4)$$

Due to the addition of time $t_{r,x}$, equation 4.5 gives the new mean value $OC_{r,x}$ at RN r when the UAV is nearby.

$$OC_{r,x} = \frac{(t_n - t_{n-1}) + t_{r,x}}{t_n - t_{n-1}} \times OC_{rt_{n-1}}, \quad (4.5)$$

So, having a new mean value ($OC_{r,x}$), equation 4.2 and 4.3 are modified into equation 4.6 and 4.7, respectively, to give probability $P_{r,x}$ to compare the probability of RN r with that of RN x .

$$P_{r,x}(Y < OC_{max}) = \sum_{y=0}^{OC_{max}-1} \frac{(OC_{r,x})^y \times e^{-OC_{r,x}}}{y!}, \quad (4.6)$$

Similarly, equation 4.3 is transformed into equation 4.7.

$$P_{r,x}(Y \geq OC_{max}) = 1 - P_{r,x}(Y < OC_{max}), \quad (4.7)$$

So, using equation 4.3, the UAV calculates the probabilities for all RNs except the one located near to it. The probability calculations of RN, having UAV in its vicinity, are done using equation 4.7. Fig. 4.3 shows a scenario with three RN nodes named as RN-1, RN-2, and RN-3. Using equation 4.3, the UAV calculates the probability of RN-2 and RN-3 as P_2 and P_3 , while the other two probabilities (i.e., $P_{1,2}$ and $P_{1,3}$) are calculated using equation 4.7. $P_{1,2}$ is the probability of RN-1 with respect to RN-2 and in the same way, $P_{1,3}$ is the probability of RN-1 with respect to the RN-3.

Algorithm 6 *UAV movement control algorithm*

Input:

- 1: $\mathcal{P}, \mathcal{R}, \mathcal{L}$
 - 2: **if** $P_{max} \in \mathcal{P}$ **then**
 - 3: Stay at the current position
 - 4: **else**
 - 5: **if** $P_{max} \in \mathcal{R}$ **then**
 - 6: Move right to the next immediate RN location
 - 7: **end if**
 - 8: **if** $P_{max} \in \mathcal{L}$ **then**
 - 9: Move left to the next immediate RN location
 - 10: **end if**
 - 11: **end if**
-

Thus, for Fig. 4.3, after every OC reporting interval, the UAV will have $P_2, P_3, P_{1,2}$, and $P_{1,3}$ as probability values. So, for N number of RNs, there are $2(N-1)$ probability values. Algorithm 6 describes

the set \mathcal{P} contains probabilities of an RN, having the UAV nearby, corresponding to other RNs. The set \mathcal{R} contains all the probability values for RNs which are located right to the UAV. In Fig. 4.4, RN-2 and RN-3 are the right-sided RNs that are located to the right of the UAV as well RN-1. The set \mathcal{L} contains all probabilities for RNs which are located left to the current position of the UAV. In Fig. 4.4, RN-0 is the only left-handed RN located to the left of the UAV and RN-1. P_{max} is the highest probability value among the \mathcal{P}, \mathcal{R} , and \mathcal{L} sets.

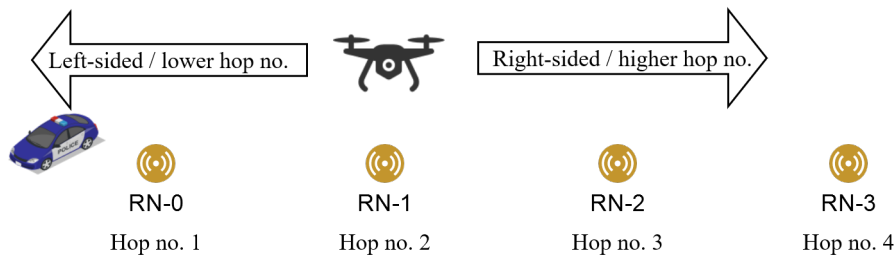


Figure 4.4: *Depiction of movement control algorithm*

4.3.4 Hop number assignment and immediate reporting mechanism

The MBS is responsible for sending hello messages. These messages help in identifying the neighbors and the assignment of hop numbers to the UAV and wireless nodes. The MBS sends a

hello message with a zero hop number. Any wireless node receiving this message updates its hop number to "1" and retransmits the same message with an incremented hop number. Upon reception of this hello message, the UAV does not forward this message but rather updates its hop number accordingly. Fig. 4.4 represents hop numbers assigned as a result of the hop number assignment procedure. During the detection of any extreme speed violation, the UAV immediately informs the MBS about the registration number of the vehicle. There is no specific path formation technique, and data packets at each node are forwarded to a lower hop number to reach the MBS. Other speed violations are transmitted when both the UAV and the MBS are close to each other.

4.4 Performance evaluation

To evaluate our proposed collaborative architecture, we simulated it in Network Simulator-2 under four different simulation scenarios with varying traffic conditions. The proposed work aims to have fewer UAV frequent movements between hotspots (RNs) while not missing too many over-speeding vehicles over the intended region of interest. We compare our proposed scheme with stand-alone unguided [4] and cooperative [96] traffic surveillance schemes. The approach in [4] is labeled "Static" in the simulation results and is validated by placing the UAV at all hotspot locations. The cooperative approach [96] is labeled "Stepwise" in the simulation results as the UAV visits all the hotspot locations step by step. Our evaluation uses the following metrics, i.e., (i) the UAV location tracking based on reported OC values, (ii) the number of actual violations captured by the UAV over time, and (iii) the reporting time to MBS for extreme-speed violations (i.e., to validate real-time aspect).

In the simulations, we have a police vehicle acting as the MBS, three RNs, and 78 deployed sensor nodes covering almost 7 km of the highway. At the start of the simulation, all the vehicles were placed at the end of the highway and started randomly with a randomly chosen speed. Three RNs (RN-2.0, RN-3.9, and RN-5.3) report OC values placed at 2.0, 3.9, and 5.3 km away from the MBS. In these simulations, a vehicle with a speed over 28 m/s, is considered as an over-speeding vehicle, and a speed above or equal to 50 m/s is considered as a severe speed violation. In all the simulations, the UAV detects over-speeding vehicles within 100 m range. However, the actual detection range depends on the resolution of the embedded camera and the altitude of a UAV [99]. Focusing on routing issues, we consider a UAV to be a point mass moving at constant speed, and it is worth pointing out that a detailed study on the UAV's dynamics is beyond the scope of this chapter but we will address this in our future work. UAV receives

Table 4.2: *Simulation parameters*

Parameter	Value
Network size	10 m x 10000 m
Speed of UAV	15 m/s [100]
Transmission range (UAV, RN, and HN)	100 m [101, 99]
Number of wireless nodes	78
Number of vehicles	200 – 300
Vehicle normal speed range	13 – 27 m/s [102]
Vehicle speed change duration	40 s
OC reporting interval	100 s
MAC protocol	MAC/802_11p
Simulation time	0 – 2500 sec

trajectory coordinates and hotspots location information before the start of a mission. Each node (i.e., UAV, RN, and HN) is equipped with an omni-directional antenna with the Two Ray Ground propagation model for communication. Table 4.2 presents the remaining simulation parameters.

4.4.1 Simulation scenario I - hotspot location 2.0 km away from BS

In simulation-I, the region around RN-2.0 is defined as a hotspot region. This region has the highest probability of finding over-speeding vehicles. In Fig. 4.5, the y-axis shows the reported OC values and UAV location information after every OC reporting interval. The x-axis shows the simulation time. Fig. 4.5 shows that RN-2.0, located 2.0 km away from the MBS, reported the highest number of speed violations. Initially, the UAV was placed in the region around RN-2.0. It can be inferred from Fig. 4.5 that the UAV tries to target the over-speeding region by following the highest persistently reported OC values. It can be observed that the UAV prefers to stay at the current position when the difference between the reported OC values is very small.

Fig. 4.6 shows the actual number of violations detected by the UAV during the entire simulation time using different trajectory schemes. In CHS, the probabilistic model controls the UAV trajectory autonomously while targeting the best hotspot.

In the Static-2.0 scenario (i.e., UAV placed at RN-2.0 in [4]), the UAV always remained at 2 km away from the MBS during the whole simulation time. The same is true for Static-3.9 and

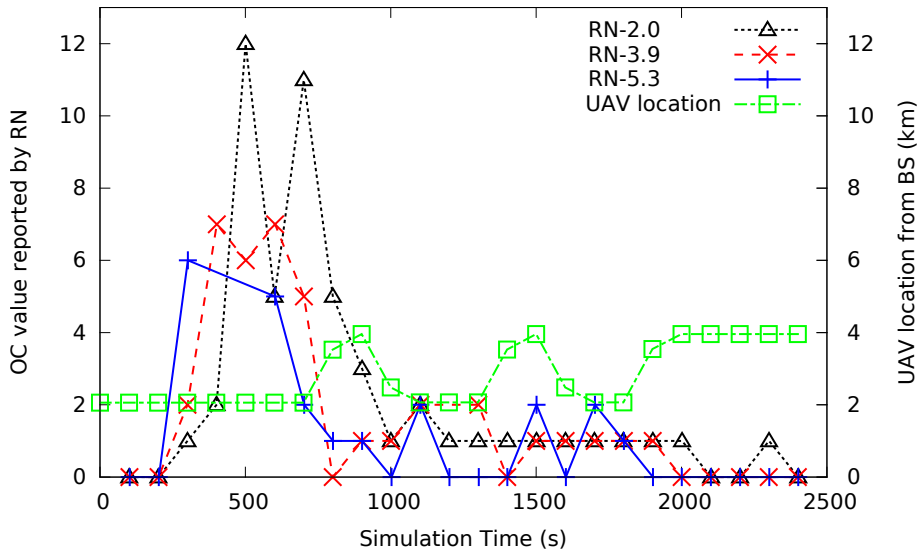


Figure 4.5: Reported OC values and UAV location [RN-2.0 as a hotspot region]

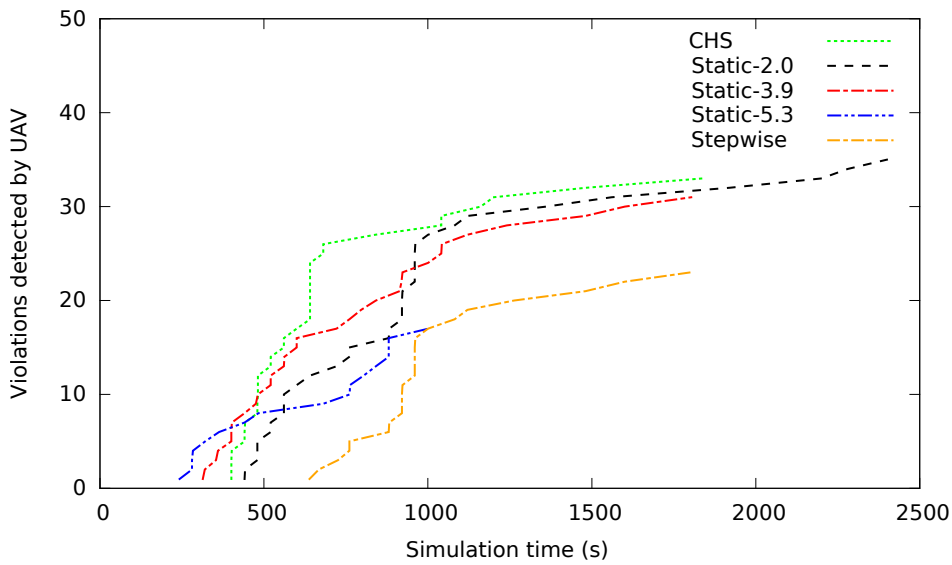


Figure 4.6: Violations detected by the UAV [RN-2.0 as a hotspot region]

Static-5.3 scenarios, wherein the UAV remained at 3.9 and 5.3 km away from the MBS during complete simulation time. In the Stepwise scheme, the UAV follows a predefined trajectory that starts from the MBS. The UAV takes a new position after each specifically defined time interval. Fig. 4.6 shows that the Static-2.0 plan captures the highest number of speed violations because the UAV was placed exactly around the hotspot region. This scheme would be less effective if the hotspot was around any other region. The Stepwise scheme proves to be the least effective scheme because the UAV follows a predefined trajectory, and it does not care if more violations

occur at other points of the trajectory. The proposed CHS scheme shows almost the same result as that of the Static-2.0 scenario because it reacts dynamically to the OC changes and aims for the best hotspot.

4.4.2 Simulation scenario II - hotspot location 3.9 km away from BS

In simulation-II, the region around RN-3.9 is defined as a hotspot region. This region now has the highest probability of finding over-speeding vehicles. Fig. 4.7 shows that RN-3.9 reported the highest number of speed violations, more specifically in the first half of the simulation. Accordingly, the UAV tries to approach a highly reported RN-3.9 region by moving from its initial position of RN-2.0. In the later part of the simulation, RN-5.3 reports the highest number of over-speeding violations that result in the further movement of the UAV toward this RN.

Fig. 4.8 shows actual violations detected by the UAV while using different trajectory schemes. This time Static-3.9 showed the highest number of speed violations detected because the UAV remained around the RN-3.9 region during the entire simulation time. Unlike in simulation-I, Static-2.0 showed the least number of detections in this simulation. In CHS, the UAV again starts its operation around RN-2.0, and then based on the reported OC values, decides to move to the RN-3.9 region. In the later part of the simulation, RN-5.3 reports more violations owing to which the UAV decides to move further to approach the location around RN-5.3.

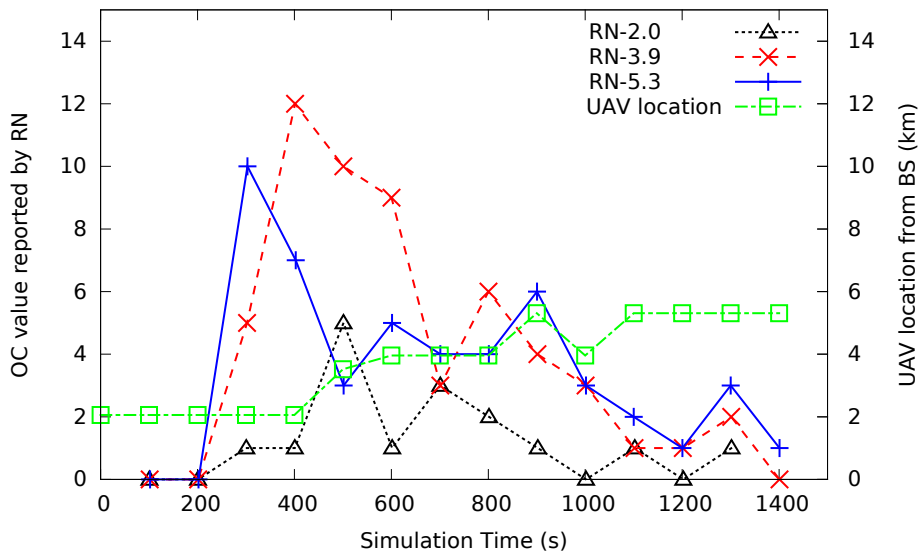


Figure 4.7: Reported OC values and UAV location [RN-3.9 as a hotspot region]

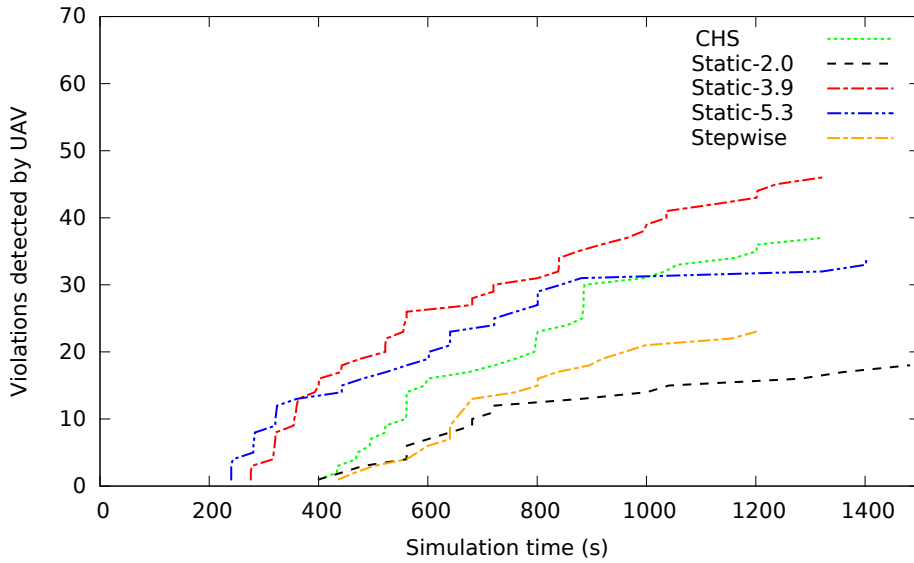


Figure 4.8: *Speed violations detected by the UAV [RN-3.9 as a hotspot region]*

4.4.3 Simulation scenario III - hotspot location 5.3 km away from BS

In simulation-III, the new hotspot is defined around the region RN-5.3. This region now has the highest probability of finding over-speeding vehicles. Fig. 4.9 confirms that RN-5.3 reported the highest number of speed violations. In CHS, to avoid too much long-distance back and forth movements, the UAV does not move beyond one RN in one trip. In Fig. 4.9, at the start of the simulation, RN-5.3 reports the highest OC values, but the UAV avoids direct movement from RN-2.0 to RN-5.3. If the UAV had moved directly to RN-5.3, it would have to return back to RN-3.9 due to the frequent variation of the OC values. In the later part of the simulation, RN-5.3 reports the highest values owing to which UAV follows this region.

Fig. 4.10 shows actual violations detected by the UAV during the entire simulation time by different trajectory control schemes with the RN-5.3 region as a hotspot. In this figure, Static-5.3 reported the highest number of detected violations due to the availability of the UAV around the RN-5.3 region during the entire simulation time. The Static-3.9 scheme detected the second-highest number of speed violations because a higher number of vehicles violated the speed limit around this region after RN-5.3. Our proposed scheme yields a similar performance to the Static-3.5 and Static-3.9 scenarios in terms of the number of over-speeding vehicles detected.

4.4.4 Simulation scenario IV - with varying hotspot locations

To further assess the performance of our proposed scheme, we simulated a scenario with peak speed violations occurring at different instants of simulation time. In Fig. 4.11, at first, RN-2.0

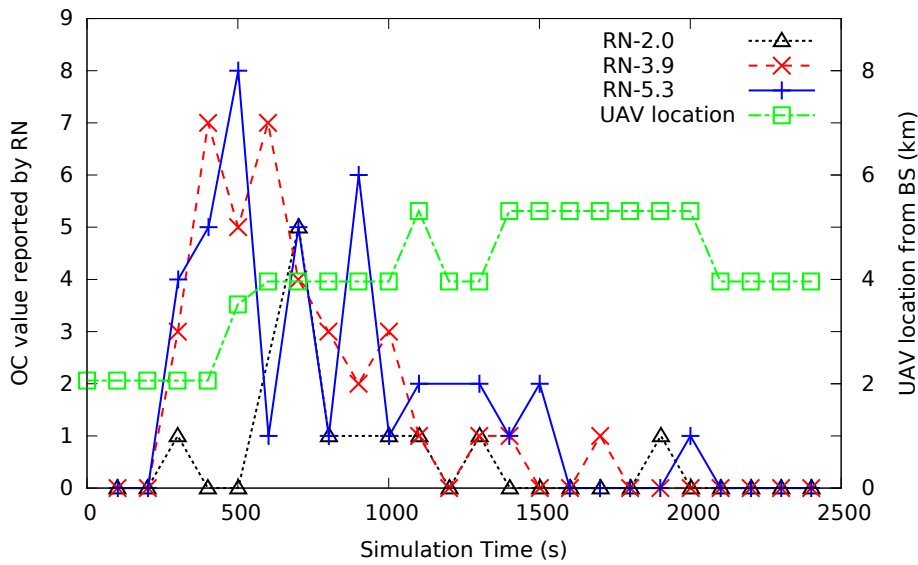


Figure 4.9: Reported OC values and UAV location [RN-5.3 as a hotspot region]

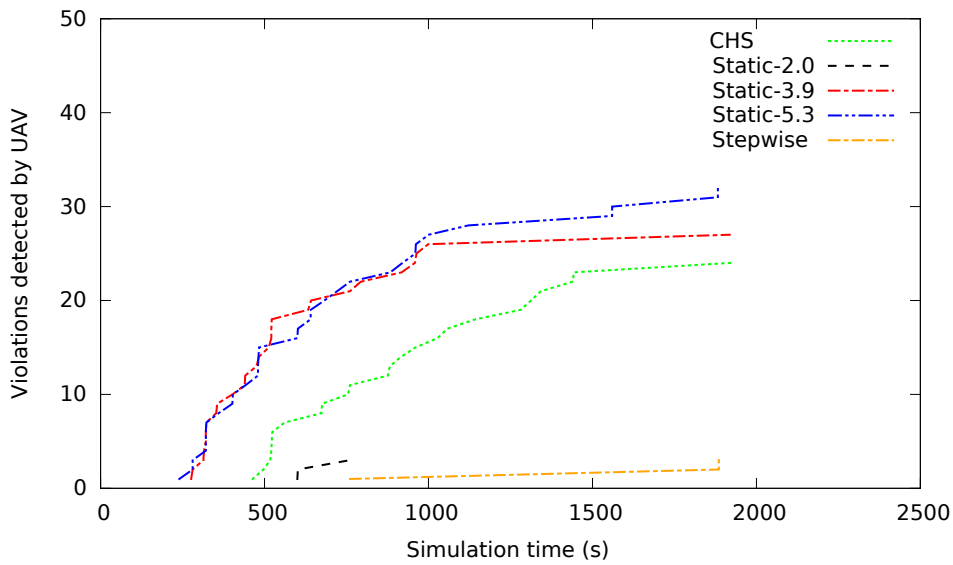


Figure 4.10: Speed violations detected by the UAV [RN-5.3 as a hotspot region]

reports higher violations starting from 500-sec with a peak value at 700-sec of the simulation time. The UAV decides to stay at the same place until RN-3.9 starts to report higher violations from 800-sec with two peak values at 900-sec and 1100-sec.

The UAV advances towards the RN-3.9 region and stays there for the longest time due to persistently highest reported violations by RN-3.9. In the later phase of the simulation, the UAV decides to move to the RN-5.3 location because this region reported the highest violations.

Fig. 4.12 presents actual violations captured by different schemes for this simulated scenario.

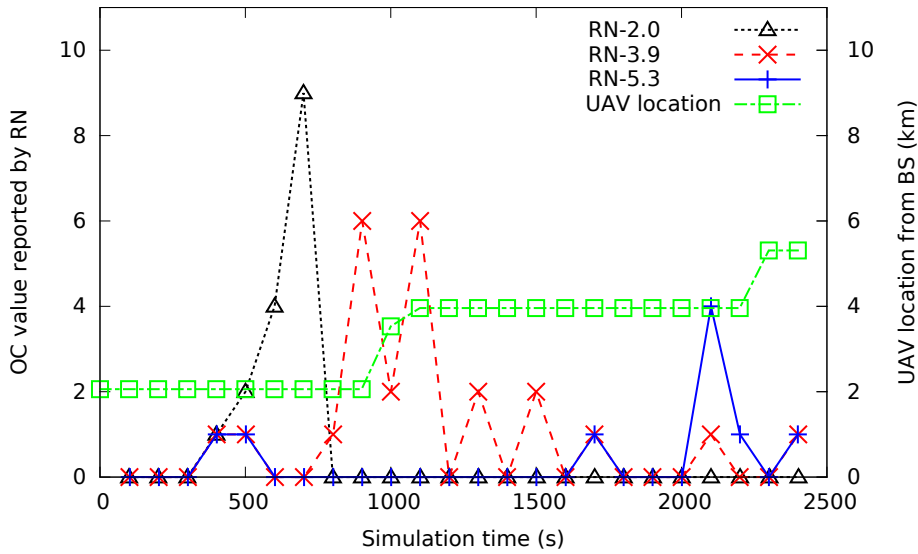


Figure 4.11: Reported OC values and UAV location

As can be inferred from Fig. 4.12, our proposed CHS scheme outclassed all other methods by capturing the maximum number of speed violations. The UAV in the proposed CHS scheme does not stick to one hotspot but visits the best hotspot locations by exploiting the information provided by the deployed WSN.

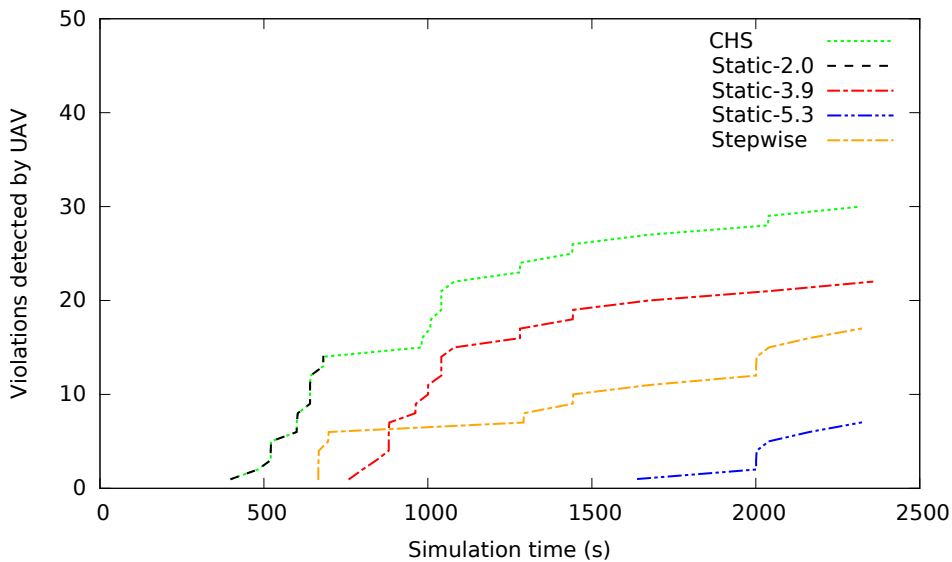


Figure 4.12: Speed violations detected by the UAV

The static deployment schemes are not suitable because either people become aware of the deployed systems or the traffic does not show the same trend all the time. The stepwise scheme is the least effective scheme because of its predefined trajectory and disconnected sensor network.

Our proposed CHS scheme remains aware of its surroundings and adapts well to traffic variations thanks to the real-time feedback provided by the connected wireless sensor network.

4.5 Conclusion

Collaborative highway traffic surveillance is a dynamic collaboration based architecture that integrates WSN and UAV technologies. WSN acts as a backbone network providing routing services as well as dynamically guiding the UAV to target the best hotspot. CHS inherits excellent features from a closed-loop system and independently regulates itself to the desired output. Having no dependency on humans or expensive roadside units makes the proposed architecture a cost-efficient solution compared to current highway surveillance systems.

This chapter does not consider the energy resource limitations of a UAV. However, timely replacement with a new UAV or introducing a wireless charging option at the hotspot locations can address this issue. A swarm of UAVs performs better as compared to single-UAV-based systems with additional benefits such as scalability and robustness. In future work, we will explore these benefits with a multi-UAV-based surveillance system. Such a system will be more robust along with the possibility of covering larger surveillance areas.

A MAC-aware routing protocol for wireless sensor networks

Abstract

WSN is the backbone of the collaborative architecture in the previous chapter. Transmission is the most energy-consuming process and decisive factor for determining the lifetime of a WSN and eventually the collaborative architecture. Wireless MAC protocols based on CSMA/CA are also prone to collisions that result in a repetitive call to a back-off algorithm. In CSMA, the inclusion of a collision avoidance scheme increases the overhead, results in an underutilization of the channel, and causes higher delays incurred by the data packets. In this chapter, a Novel Packet Scheduling Strategy (PSSN) is presented to diminish the chances of collisions along a single path in a shared wireless medium. The PSSN is a query-driven approach and works without RTS and CTS control signals. Each packet is scheduled after a fixed interval decided by the PSSN keeping in view the objective of collision reduction along a single path. Being a cross-layer solution, the proposed scheme is compared with AODV and AOMDV routing protocols with 802.11p as an underlying MAC protocol. The proposed protocol outperforms the above two with respect to energy efficiency and delays while maintaining a better throughput level.¹

¹N. Bashir, S. Boudjit. 2020, October. "A Collision Avoiding Packet Scheduling and Energy-Efficient Routing Technique for Video Wireless Sensor Networks," In *Proceedings of International Symposium on Networks, Computers and Communications (ISNCC)*, Montreal, QC, Canada, pp. 1–6.

Chapter content

5.1	Introduction	83
5.2	Related work	84
5.3	Proposed work	85
5.3.1	Problem formulation with possible collision scenarios	86
5.3.2	Network setup phase	86
5.3.3	Packet scheduling interval calculation	87
5.3.4	Routing path selection criteria	89
5.4	Performance evaluation	89
5.4.1	Data packet delivery ratio	90
5.4.2	Energy consumed per packet delivery	91
5.4.3	Average delay	92
5.4.4	Data packet delivery ratio vs. number of nodes	93
5.5	Conclusion	94

5.1 Introduction

Wireless sensor networks have been used by numerous real and non-real-time applications of internet of things [103]. After sensing desired parameters from a region of interest, the sensing unit of the sensor node hands over the collected data to the communication system. Depending upon the transmission range of a sensor node, the acquired data is relayed to the BS in a single-hop or multi-hop fashion[104, 105, 106].

Protocol stack for WSN comprises of the following layers, i.e., application, transport, network, data-link, and physical [107]. Each layer has its own set of defined rules, e.g., the network layer takes care of routing data provided by the above transport layer, and the transport layer regulates end-to-end data flow[108]. The data-link layer controls access to the medium and shares communication resources with all sensor nodes. Contention-based and scheduling-based protocols are commonly used MAC protocols with Carrier-Sense Multiple Access (CSMA) and Time Division Multiple Access (TDMA) as examples, respectively.

An intelligent way to access the shared wireless medium is necessary for having collision-free transmission, specifically for a network like WSN having scarce energy resources. MAC protocols with CSMA/Collision Avoidance (CSMA/CA) aim to reduce the chances of collisions using Request To Send (RTS) and Clear To Send (CTS) control signals. The use of the collision avoidance technique still comes up with a probability of collision and repeated calls to back-off algorithms. Due to unsuccessful packet transmissions, each packet has to wait in the queue that not only results in increased end-to-end delays but extra energy consumption for its retransmission. The layered model idea is based on a principle of separation and does not guarantee optimal performance for networks such as WSN. Therefore, cross-layer protocols are preferred as they can overcome architectural limitations to bring up an overall efficient solution[109].

The work presented in this chapter proposes a cross-layer energy-efficient and collision-avoiding routing scheme for a sensor network, called PSSN. The basic idea of the PSSN is to provide a new collision-free channel access scheme with the objective to reduce overall overhead. The PSSN scheme works without RTS and CTS controlling signals of MAC_802.11p protocol. The packets at a source node are directly moved to the buffer and each packet is scheduled after an interval called Scheduling Interval (SI). The SI value taken for the operation is decided so that the chances of collision within an existing path are minimized. A major part of the end-to-end delay incurred by data packets comes from the wait time at a source node while other nodes receive and forward packets with a minimum delay.

The remaining chapter is organized as follows. Section 5.2 presents the related work. Sec-

tion 5.3 presents the core functionality of the proposed protocol. In Section 5.4, the PPSN scheme is evaluated. The chapter is concluded in Section 5.5.

5.2 Related work

Cross-layer protocols that consider MAC sub-layer and network layer fall into three categories, i.e., i) Joint functionality of MAC and routing protocols (MAC and Network), ii) Routing that consider MAC information (MAC aware routing), and iii) MAC protocol using routing information (routing aware MAC)[110]. This section presents MAC-aware routing protocols for WSNs as the proposed idea falls within this category.

Ad hoc On-Demand Multiple Distance Vector (AOMDV) [111] is an extension of AODV protocol[50] with multiple paths as an eminent feature. In AOMDV, path formation is based on the distance vector concept and multiple node disjoint loop-free paths are discovered from a source node to a destination node. Routing in AOMDV is done in a hop-by-hop manner owing to which intermediate nodes between source and destination maintain multipath information in their routing table. The availability of alternate paths in the intermediate nodes results in the reduction of the frequent route discovery processes. Maintenance of multiple paths as compared to AODV results in an increased routing overhead and eventually leads to a lower network performance [112].

Hop-by-hop Markov decision-based routing (DCRs) [113] is an energy-efficient and controllable expected delay scheme in duty-cycled WSNs. DCR requires very less routing information and pursues localized energy use optimization to achieve a delay constraint success ratio. Whenever next-hop candidate wakes up, a node having data packets to transmit first re-computes its optimal next-hop and chooses whether to wait for a next-hop or transmit using the best available next-hop using Markov findings. Even though delay guarantee is provided in DCR but it does not consider any reliability metric.

In [114], the authors present transmission power control-based Opportunistic Routing (TCOR) scheme which conserves energy by reducing transmission power and still maintaining communication reliability. TCOR proposes an energy cost function considering multiple available transmission power levels for the selection of a next-hop forwarder. Even though this routing scheme can provide reliability but nothing is considered for the end-to-end delay.

A maximally Radio-disjoint Multipath routing (MR2) [115] aims to increase network lifetime of WSN and to provide the necessary bandwidth to multimedia traffic through utilizing non-interfering paths. At the start, only a single channel is established while an additional path

comes into play in case of path congestion or lack of bandwidth. While looking for non-interfering paths, MR2 approach may select longer paths.

Interference-Minimized Multipath (I2MR) [116] is a routing scheme for WSNs having high rate streaming capability. I2MR with the use of multipath transmission increases throughput and reduces overhead along with reducing localization support. This protocol first evaluates the quality of multiple paths taking into account the effect of wireless interference. Zone disjoint multiple paths are established and these paths are loaded with the highest supportable load to further increase throughput.

A Low-overhead Interference-Minimized Multipath Routing Protocol (LIEMRO) [117] tries to discover multiple paths from a source to a destination node considering interference level, residual energy, and link quality parameters. A new path is added to the multi-path list only if it results in increased throughput, otherwise, it is removed from the list. LIEMRO also uses a load balancing algorithm to distribute traffic among different available paths based on the interference level of paths. This protocol does not perform well due to high channel competition.

Geographic Energy-Aware non-interference Multipath (GEAM) [118] routing protocol divides the network area into different districts to have non-interfering zones. GEAM transmits data simultaneously on these paths and load is adjusted according to the remaining energy of the nodes along these paths. There are no fixed paths in this protocol owing to which it tackles well with dynamic topology and hole problem. Inadequacy to guarantee QoS constraints on all the selected paths and necessity of position information at each node are among the major drawbacks of this protocol.

5.3 Proposed work

Due to collisions during wireless communication, energy wasted in retransmission and delay incurred because of repeated calls to the back-off algorithm degrades the performance of any wireless sensor network badly. Many researchers have tried to resolve interference among different multiple paths but no one has ever tried to find and resolve the root cause of collisions within a single path. To the best of our knowledge, PSSN is a unique attempt to reduce the effect of collisions along a single path through scheduling the packets to reduce or completely eliminate the chances of collisions in a query-driven WSNs.

5.3.1 Problem formulation with possible collision scenarios

In Fig. 5.1, consider part of a network with node **A** being as a source node. Node **A** wants to communicate with node **B** and issues an RTS to node **B**. Sensing wireless environment to be idle, node **B** issues a CTS for node **A**. In this scenario, complication starts when node **C** also wants to communicate with node **D** and issues RTS before its senses a CTS signal from node **B**. Node **C** starts transmitting data packet after getting CTS from node **D**. In spite of using RTS/CTS control signals, we still have a collision of data packets at node **B**.

In PSSN, we will develop a routing technique with a scheduling control mechanism to avoid collisions along a specific active path.

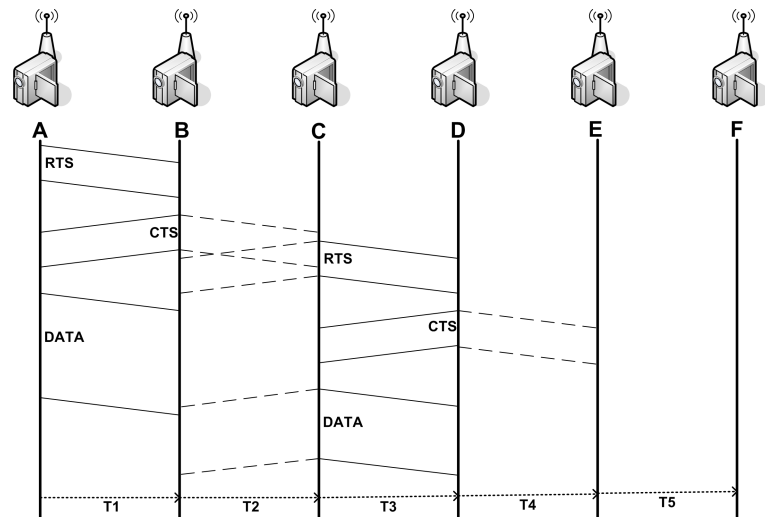


Figure 5.1: A collision scenario at MAC layer

5.3.2 Network setup phase

In this proposed work, we use the same initialization phase as presented in [82] which consists of Neighbors Discovery (ND) and Hop-Update (HU) phase. As soon as the network is deployed, the ND phase is started wherein each node broadcasts a hello message to its one-hop neighbors. After the accomplishment of ND phase, BS initiates the HU phase and broadcasts a hop update packet with hop number field value equal to "1". Any node within the direct transmission range of BS will set its hop number to "1" and rebroadcasts the message with hop count value incremented by one. This hop number assignment phase is continued until each node of the network is assigned with a hop number. Once the ND and HU phases are over, each node becomes aware of its neighbors and the number of hops to reach the BS. After completing the HU phase, the hop numbers assigned to each node are shown for the network in Fig. 5.2.

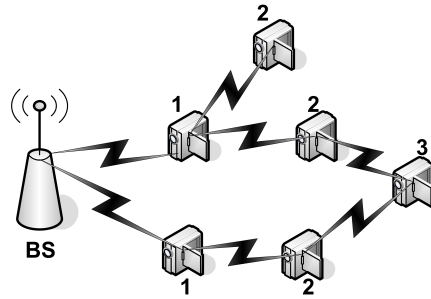


Figure 5.2: A network part after hop number assignment (HU)

5.3.3 Packet scheduling interval calculation

Packets are moved to the buffer as soon as generated and scheduled for transmission according to the scheduling interval. This specific scheduling mechanism only applies to a source node while other nodes within the path receive and transmit packets immediately. Data packet from the buffer is scheduled after every SI , and the value of SI is taken such that the chances of collision along the path are minimized.

In this subsection, we will estimate SI value for the source node **A** in Fig. 5.1. Assuming nodes **A** and **B** are in the transmission range of each other with maximum distance apart. T_1 is the transmission time of a single packet between node **A** and **B**, and accordingly, the same is true for all other nodes in Fig. 5.1.

Case I: In Fig. 5.3, let's assume a case with SI value to be equal to T_1 . Source node **A** transmits a new packet after every time interval of T_1 . In Fig. 5.3, at the end of the first interval (i.e., T_1), there will be a probability of collision as packets at both the nodes i.e., **A** and **B**, are due to be transmitted.

Case II: Now, consider a second scenario in Fig. 5.4 with SI having value T_1+T_2 . After this

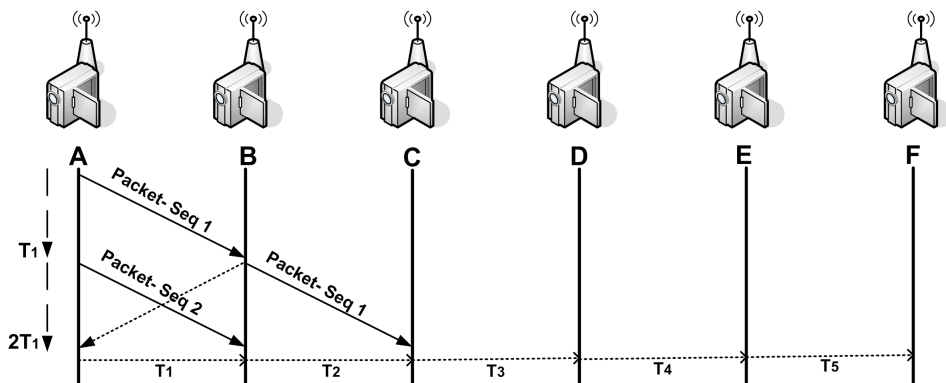


Figure 5.3: Case I: SI having value equal to T_1

time interval, nodes **A** and **C** will have packets to transmit and these packets can still collide in

the region around **B**.

Case III: Finally, in Fig 5.5, SI takes the value of $T_1+T_2+T_3$. With the lapse of this time

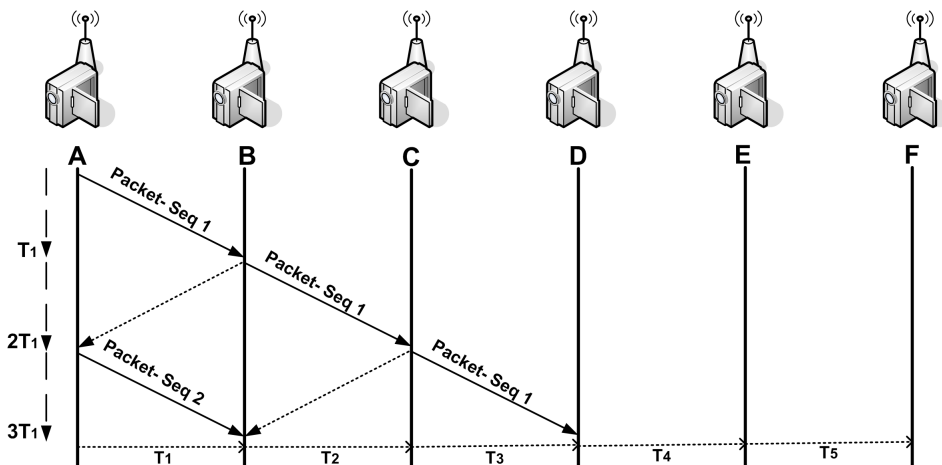


Figure 5.4: Case II: SI having value equal to T_1+T_2

interval, the first packet from node **A** would have reached until node **D**. At this time, **A** will be ready to schedule another packet while **D** will be in the preparation to transmit this newly received packet. It can be inferred from the Fig. 5.5 that there will be no collision even both of these nodes try to transmit at the same time.

So, to have a minimum chance of collisions, the minimum time T_{SI} required for SI is given by

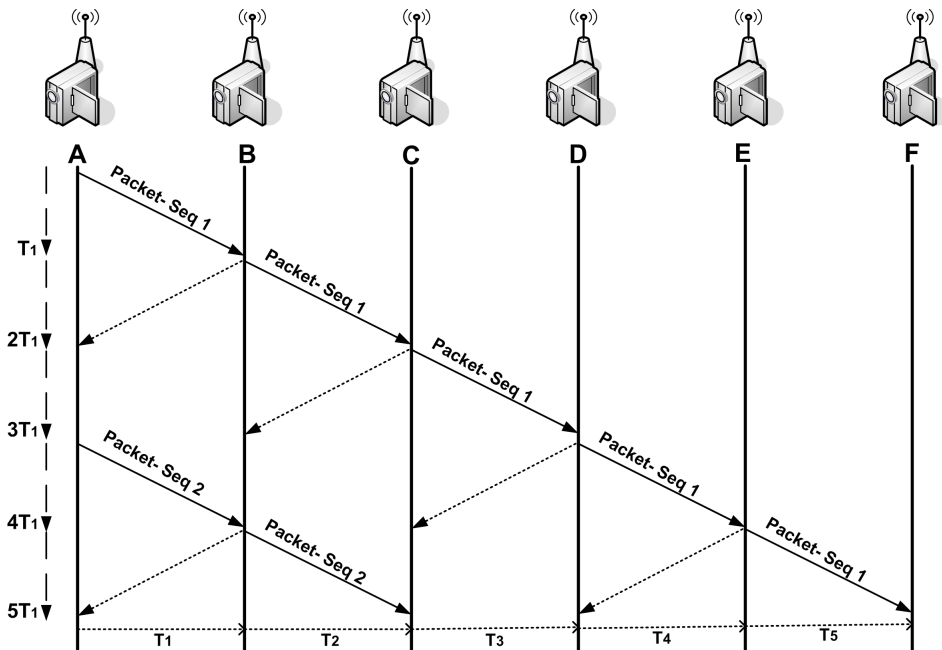


Figure 5.5: Case III: SI having value equal to $T_1+T_2+T_3$

equation 5.1.

$$T_{SI} \geq T_1 + T_2 + T_3, \quad (5.1)$$

Assuming that all nodes are equally distanced apart and having the same transmission time, we have.

$$T_{SI} \geq 3 \times T_1, \quad (5.2)$$

So, equation 5.2 reveals that to have minimum chances of a collision, the scheduling interval should be at least three times the maximum transmission time between two maximally distant apart nodes.

5.3.4 Routing path selection criteria

After the initialization phase, each node becomes aware of its hop number from the BS as well as its total number of neighbors along with their respective hop numbers. Whenever any node receives a query from the BS, the source node selects a lower hop number node than itself for transmission. The same procedure is repeated by this selected node which further selects a lower hop number node than itself for transmission of the received data packets. The same procedure is repeated by all the intermediate nodes until the destination is reached.

Algorithm 7 represents the packet scheduling procedure for a source node and nodes involved along an active path. After the lapse of each SI value, the scheduling timer at a source node \mathcal{S} raises a binary flag \mathcal{T} , and a packet \mathcal{P} at the top of the queue is scheduled for transmission.

5.4 Performance evaluation

To validate the assumptions made by the proposed PSSN scheme and to analyze its performance with the existing ones, we simulated PSSN, AODV [50], and AOMDV [111] in Network Simulator-2 (ver. 2.35). The evaluation of the protocols is based on metrics, i.e., (i) Packet delivery ratio or throughput, (ii) Energy consumed per packet delivery, and (iii) the average delay incurred by data packets during the entire operation.

In this simulation, a random topology of wireless sensor nodes is considered with single base station. Query-driven approach is used in this simulation in which BS requests for a real time transmission from any specific sensor node. Unless specified, simulations are done for 100, 200, and 400 nodes, and average results are plotted to have a better depiction of the protocols. PSSN protocol uses Mac/802_11p with no CTS and RTS controlling signals and no retry is made

Algorithm 7 *Packet scheduling procedure*

Input:1: $\mathcal{T}, \mathcal{S}, \mathcal{P}$ **Output:** $Forward(\mathcal{P})$: Unicast this data packet \mathcal{P} to a lower hop number node2: **if** $\mathcal{S} = true$ **then**3: **while** true **do**4: **if** $\mathcal{T} = true$ **then**5: $\mathcal{T} \leftarrow false$ 6: $Forward(\mathcal{P})$ 7: **end if**8: **end while**9: **else**10: $Forward(\mathcal{P})$ 11: **end if**

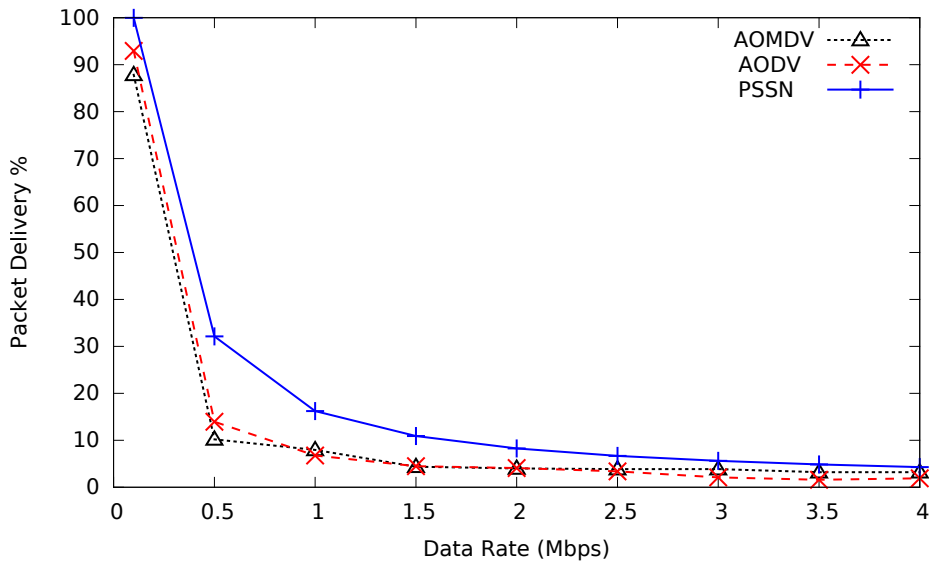
for the lost data packets. Each node is equipped with an Omni-directional antenna. For this simulation, Droptail [58] queue type is used with the Two Ray Ground propagation model for communication. The parameters used for the simulations are listed in Table 5.1.

5.4.1 Data packet delivery ratio

Data Packet Delivery Ratio (PDR) or throughput is defined as the ratio of the number of packets received at the BS to the number of packets sent by the source node. Fig. 5.6 presents the performance of PSSN, AODV, and AOMDV protocols in terms of PDR with variation in data transmission rate. PSSN protocol shows better results for the each data rate as compared to AODV and AOMDV. The use of the novel scheduling technique in PSSN results in almost no collisions at the MAC level and hence, becomes the reason for its better behavior as compared to the other two protocols. It can be observed from Fig. 5.6 that PDR at higher data rates becomes almost unchangeable for all the protocols because limited buffer capacity fails to handle higher data rates. So, owing to the reason mentioned, protocols at higher data rates work in saturation mode and larger amount of the packets are dropped from the buffer. It can be inferred from Fig. 5.6 that PSSN outperforms in terms of PDR when a buffer is not utilized at its full capacity.

Table 5.1: *Simulation Parameters*

Parameter	Value
Network size	5000 m x 5000 m
Number of nodes	100 – 400
Maximum packets transmitted	20000
Transmission range of each node	100 m
Data rate	0.01 – 4 Mbps
Interface queue length	1500
Initial energy	50 J
Transmission energy	2330 nJ/bit
Receiving energy	1131 nJ/bit
Simulation time	300 sec
MAC protocol	Mac/802_11p (Modified)

**Figure 5.6:** *Data packet delivery ratio (throughput)*

5.4.2 Energy consumed per packet delivery

We define Energy Consumed per packet Delivery (ECD) as the ratio of total energy consumed in all the sensor nodes to the total number of packets received at the BS. ECD gives an overall idea of a protocol in terms of energy consumption for delivering a single data packet. Fig. 5.7

demonstrates the ECD performance comparison for the simulated protocols with the variation in data rate. PSSN performs better among AODV and AOMDV owing to the absence of packet collisions at the MAC level, CTS/RTS signals, and repetitive retransmissions. The maintenance of multiple paths in AOMDV results in the worst performance among all of these compared protocols. In PSSN, energy consumed is the combination of fixed energy i.e., energy used for the initialization phase and data transmission energy. With the increase in data rate, the PDR decreases and the fixed energy dominates in the calculation for ECD which shows an increase in the ECD.

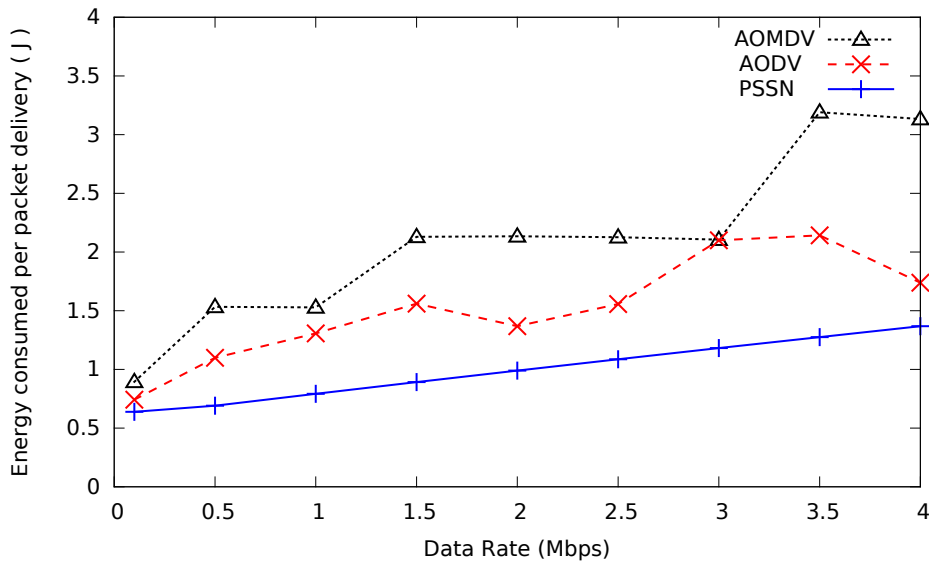


Figure 5.7: Energy consumed per packet delivery

5.4.3 Average delay

The average delay in this chapter is defined as an average of the delay faced by all the data packets to reach BS. The average delay versus data rate for the simulated protocols is presented in Fig. 5.8. PSSN outclasses AODV and AOMDV protocols in terms of average delay faced by the data packets to reach BS. There is no CTS/RTS signaling in PSSN, and utilizing a novel collision avoidance scheduling technique turns out to be the reason for the best performance among the compared protocols. AODV and AOMDV, on the other hand, use periodic hello message, path on demand, and use of RTS/CTS with MAC protocol result in the higher average delay. AOMDV performs worst in terms of delay factor due to the probable use of longer node disjoint multiple paths.

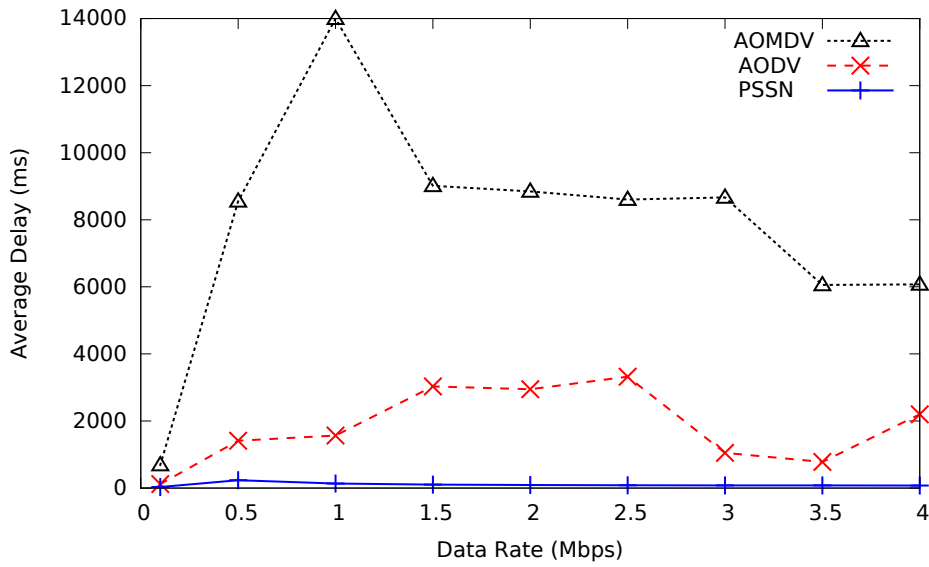


Figure 5.8: Average delay

5.4.4 Data packet delivery ratio vs. number of nodes

Fig. 5.9 illustrates the evaluation for the proposed scheme under the different number of nodes in the network. The performance of PSSN is almost unaffected by the variation in the number of nodes in the network. However, both AODV and AOMDV show variation in PDR with the fluctuation in the number of nodes. In AOMDV, multiple paths are formed in a flooding manner that ultimately results in a more overhead when the size of the network is increased, and the same becomes the reason for such a behavior of AOMDV.

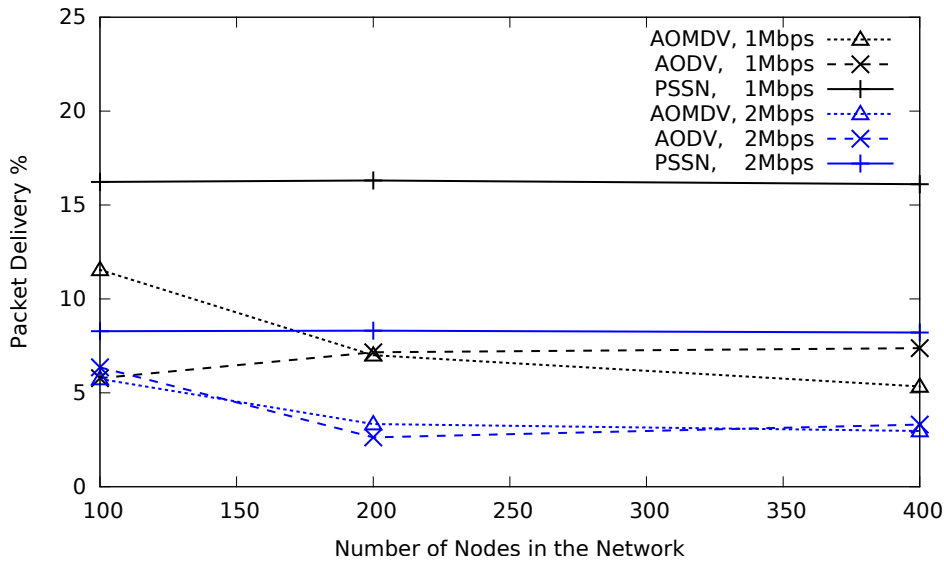


Figure 5.9: Data Packet delivery ratio vs. number of nodes

5.5 Conclusion

The PSSN scheme is a unique cross-layer routing approach to reduce the chances of collisions along a path without using CTS/RTS signals. To prevent collisions of data packets, PSSN uses a scheduling interval, after which each new data packet is scheduled for transmission. Energy preserved through avoiding repetitive retransmissions and extra delay avoided by not going into a back-off algorithm becomes the decisive factor for the outclass performance of the proposed PSSN scheme.

Part III

Path planning techniques for UAVs in an urban environment

A UAV path planning approach considering obstacles and environmental uncertainties

Abstract

Irrespective of the application, it is imperative to have an autonomous path planning to utilize UAVs to their full potential. Collision-free trajectories are expected from the path planning process to ensure the safety of UAVs and humans on the ground. This chapter proposes a path planning technique where collision avoidance is mathematically proven under an uncertainty prerequisite, that the UAV follows its requested moving position within some threshold distance. This scheme ensures UAV safety even if underlying control's system limitations are compromised. Obstacles play a guiding role in selecting collision-free trajectories. These obstacles are modeled as rectangular shapes with interest points defined around their corners. These points further define collision-free permissible edges, and later we apply the Dijkstra algorithm to these edges before having the desired trajectory. Regardless of the size of deployment area, our proposed scheme incurs low computational load due to the dependency on pre-defined interest points only thereby making it suitable for real-time path planning. Simulation results obtained using MATLAB's UAV Toolbox show that the proposed method succeeds in getting short collision-free trajectories¹.

¹ N. Bashir, S. Boudjit, G. Dauphin, S. Zeadally. 2022. "An obstacle avoidance approach for UAV path planning," *IEEE Transactions on Vehicular Technology*. [undergoing revision after first decision]

Chapter content

6.1	Introduction	99
6.2	Related work	100
6.3	Problem statement and research contributions of this chapter	102
6.3.1	Research contributions of this chapter	104
6.4	Proposed obstacle avoidance scheme	105
6.4.1	Optimization problem statement	105
6.4.2	Suboptimal solution with the Dijkstra algorithm	108
6.4.3	Discussing the location of the predefined points	109
6.5	Simulation testbed and experimental procedures	110
6.5.1	Performance metrics	111
6.6	Results and discussion	112
6.6.1	Total distance traveled by the UAV	112
6.6.2	Trajectory tracking error vs. UAV's turning angles	114
6.6.3	Trajectory tracking error vs. UAV's moving speed	115
6.6.4	Computational load comparison	117
6.7	Conclusion	117

6.1 Introduction

Many of the currently deployed applications for UAVs have autopilot functionalities along with the capability to fly them according to the preplanned path or even make real-time decisions in case of any unforeseen scenario [119, 120]. However, some of the commercially used UAVs still use off-board pilots and fixed trajectories. Path planning approaches are grouped into two classes that formulate path planning as an optimization problem [121]. Heuristic methods are those wherein the optimality of the solutions compromises for better computational time efficiency. Non-heuristic approaches, however, provide optimal solutions but demand high computational resources.

Path planning is a way to find a feasible, optimal/near-optimal, shortest, smooth, and a low-cost path between a starting point and the desired destination point by considering specific operational constraints [119]. These constraints usually involve velocity, acceleration, environmental disturbances such as wind, sensor uncertainties, and flying over restricted areas. Generally, path planning process consists of two stages, namely, graph building and pathfinding [122]. A graph construction considers the start and end points, and all available vertices obtained by matrix decomposition or area tessellation.

Different tessellation resolutions result in a different number of vertices which in turn determine the computational work needed [123]. A higher number of vertices require a high computational load, while a lower value requires a smoothing process to get stable trajectories [119]. This smoothing process is necessary to avoid having any acute angle turns within a path. The pathfinding process assigns the respective cost of each vertex and selects a flight path with a minimum overall cost. In this context, Dijkstra, A* or genetic algorithms are applied to these vertices to get optimal trajectories [122]. As a result of consideration of the entire environment into the tessellation process, these algorithms render a higher computational load for the existing works.

UAVs must have an obstacle avoidance mechanism to prevent collisions by maintaining a safe distance from nearby objects. The efficiency of this mechanism highly relies on the accurate operation of positioning sensors. These sensors help UAVs in the navigation process by providing positioning information during the entire flight. These conventional positioning systems cannot be relied upon in urban areas having large buildings with a higher probability of signal blockage. Existing path planning schemes rely entirely on the underlying control system to track the designed trajectory and do not consider if this system fails to meet its expected limits.

Environmental disturbances or sensor uncertainties and, specifically, their role in UAV path

planning is an outstanding research field that needs further investigation. In this chapter, we assume that uncertainties related to the environment, the sensors, and the navigation system are modeled into an uncertainty prerequisite, that is the existence of a threshold distance between the UAV and its requested moving position. We propose a collision-free path planning technique that reduces the needed computational load by replacing area tessellation with interest points connected into a graph. These interest points are located around each corner of rectangles, each modeling an obstacle and all together model the environment.

The rest of the chapter is organized as follows. Section 6.2 presents related research on UAV path planning and in particular in the presence of obstacles. Section 6.3 describes the problem statement and main contributions of this chapter, followed by Section 6.4 which describes the mathematical model of our proposed scheme. Section 6.5 provides details about the testbed used and experimental procedures. Section 6.6 discusses the results obtained. Finally, Section 6.7 concludes the chapter.

6.2 Related work

A robust UAV path planning strategy must possess important attributes which should provide a computationally efficient solution while complying with the given constraints. The strategy development depends on different planning requirements such as real-time planning, performance optimization, risk minimization, and obstacle avoidance [124, 125]. Many path planning techniques are available in the literature that leverage results from other research fields such as potential field algorithms from physics, probabilistic approaches from mathematics, and graph-based solutions from the computer science field. Many traditional path planning techniques have been proposed such as artificial potential field, probability road-map, and rapidly exploring trees methods [126].

The Artificial Potential Field (APF) [127] path planning is a popular method to avoid obstacles having a concise mathematical model and simple algorithm structure [128]. It creates an attractive and repulsive field for destination and restricted areas, respectively, and the route by displacement is planned based on the resultant force. [129] is one of the first publications on APF path planning. Besides its application for single UAV, the APF approach applies to multi-UAV systems as well. The APF approach faces local minima and destination unreachable scenarios due to closely spaced obstacles and the presence of an obstruction between the destination and a UAV respectively.

Many improvements [130, 131, 132, 133] have since been made to the original APF. In [134], the

authors developed an APF based reactive controller for UAVs to avoid collision with terrain as well as from each other. Their scheme models obstacles as points with latitude, longitude, and altitude information provided by a Digital Elevation Map. To address the issue of local minimum in APF, the authors of [135] present an improved artificial potential field that finds an optimal path and successfully avoids collisions with obstacles. In [136], the authors proposed a hybrid model involving APF and optimal control theory. The additional force introduced is considered to be an optimization variable that transforms path planning into an optimization problem. The optimal control law is applied to solve the optimization problem after converting the constrained problem into an unconstrained optimization problem.

Probabilistic RoadMap (PRM) and Rapidly Exploring Trees (RRT) come under the domain of sample-based path planning algorithms [137]. In PRM, at first, sampling of the configuration space is done using a probabilistic model. A connected graph is created by applying a local planner which connects the configuration sampled to the nearest configuration space. At last, any graph search algorithm can be applied to the connected graph to get a possible path from a start point to the desired endpoint. The authors of [138] proposed a PRM based 3D path planning approach for a complex environment. The octree algorithm divides the configuration space into voxels. The PRM random method selects samples from all the available voxels. The connected graph produced by the local planar is utilized further by the A* algorithm to have a feasible path. RRT is another path planning method that uses random spatial sampling for high-dimensional spaces. RRT grows a tree with its roots at the start configuration. With each sample taken, the tree grows to include more feasible trajectories. Many RRT-based path planning approaches have been reported in the literature [139, 140, 141, 142].

Graph-based search techniques are extensively explored in many fields and are popular in the UAV path planning domain [137]. In these approaches, a grid map represents the entire environment. Depending on the presence of obstacles, each cell in the grid represents either an occupied or a free cell. Any graph exploration algorithm can be applied to the graph to find a feasible path between the start and destination cell. Fast searching capabilities make these algorithms very useful for real-time path planning, but the generation of non-smooth trajectories renders them inefficient for large environments. Dijkstra algorithm finds the shortest path between the start cell and the destination cell [143], and many graph-based path planning techniques, in conjunction with the Dijkstra algorithm, have been implemented [144, 145, 146, 147]. A* is another graph traversal or path search algorithm which has attributes of optimality and completeness [148]. The performance of the A* algorithm highly depends on the heuristic

function used. Recent works such as [122, 128] are the few path planning implementations involving the A* algorithm. Several other graph-based algorithms are available in the literature that finds smooth flight trajectories by incorporating some smoothing process [119, 124].

The computational needs of path planning algorithms grow exponentially with an increase in the dimensional size of the configuration space along with the reduction in the operational time [128, 149]. Moreover, in algorithms such as APF, an improper definition may lead to a local minimum or unreachable destination scenarios [150]. Most path planning techniques rarely consider environmental disturbances and sensor uncertainties. In [128], the authors considered positioning errors that require an additional map of predicted satellite positions generated by a 3D building model. All path planning approaches, even those considering positioning error maps in their designing process, still rely entirely on the UAV's control system to track the desired trajectory. None of the existing approaches have addressed overshooting of a UAV beyond its desired control system's limitations.

To address these shortcomings, in this chapter, we propose a simple yet robust path planning method. We model obstacles as rectangles, and for the configuration space, instead of involving the entire environment, we only includes interest points defined around corners of the rectangles². Additionally, we consider a disk-based uncertainty scheme to make our model resilient to uncertainties arising as a result of environmental disturbances.

Table 6.1 provides a summary of various path planning approaches for UAVs. For each approach, this table enlists the research objective, the method adopted, strengths, and weaknesses.

6.3 Problem statement and research contributions of this chapter

The main objective of this work is to develop a robust and computationally low UAV path planning technique. We model obstacles as rectangles, and unlike other methods, we seek guidance from the interest points defined around their corners. These interest points are defined by considering the threshold distance in the uncertainty prerequisite. Contrary to existing path planning techniques, wherein every cell of the grid environment represents the vertex of a graph, our model only considers defined interest points as vertices. Fig. 6.1 shows how a graph-based path planning technique configures the environment compared to our proposed approach. Fig. 6.1 depicts a 15×16 square units environment wherein the left side of the figure

²Most obstacles (buildings, no fly-zones) can be represented by polygons (rectangles) [122, 135, 144], and tools like visibility graphs [151] allow us to view polygonal obstacles as graphs. We can successfully bypass a rectangular shape obstacle by the use of four points defined around the obstacle.

Table 6.1: *Summary of various path planning methods for UAVs*

Reference	Research objective/Approach followed	Strengths	Weaknesses
Oland et al. [134]	→ Collision and terrain avoidance with multiple UAVs Kinematic model with dynamic feedback linearization using APF	Maintaining rigid formation and low altitude flight in a rolling terrain Uniformly stable controller	Need of a digital elevation map Need of precise definition for APF
Lifen et al. [135]	→ Address local minima issue in APF Collision free trajectories using APF with change in repulsive potential function	Optimal path especially in complex environment	Dependent on obstacle size Need of precise definition for APF
Chen et al. [136]	→ Solution to additional control force in APF to deal with time varying variable Remodeling the functional optimization model by taking the additional control force as an independent variable	Solution to dead-end problem in APF Shorter and smoother trajectories with irregular obstacles	Dependent on precise definition of repulsive and attractive potential
Yan et al. [138]	→ Path planning in 3D environment with less time complexity Octree algorithm to divide the work space into voxels and random selection of free voxels	Allows UAVs to fly through narrow-passage areas	Random selection of voxel may lead to longer paths
Maini et al. [144]	→ Obstacle free paths satisfying UAV kinematic constraints Visibility graph to represent environment and validation for maximum steering angle	Suitable for on-line implementation Easy validation of the steering angle constraint	No tolerance for sensor uncertainties No definition as how to add points of obstacles as vertices Steering angle constraint may result in longer paths
Our proposed method	→ Collision-free trajectories taking environmental disturbances into account Defining interest points around rectangular obstacles while taking environmental disturbances into account	Shortest possible linear paths Less configuration space complexity Tolerance to sensor uncertainties Suitable for real-time path planning Acts as guard if underlying control system fails to track the desired trajectory within limitations	Absence of UAV kinematics into path planning model (left as future work)

shows the configuration space adopted by the graph-based technique. It divides the entire area into 240 cells that constitute the vertices of a graph. The right side of Fig. 6.1 shows how the proposed scheme delineates interest points around the obstacles. It selects only 20 interest points that become vertices of a graph. Permissible obstacle-avoiding edges are defined, which in collaboration with the Dijkstra algorithm formulate a path. Fig. 6.2 shows the uncertainty

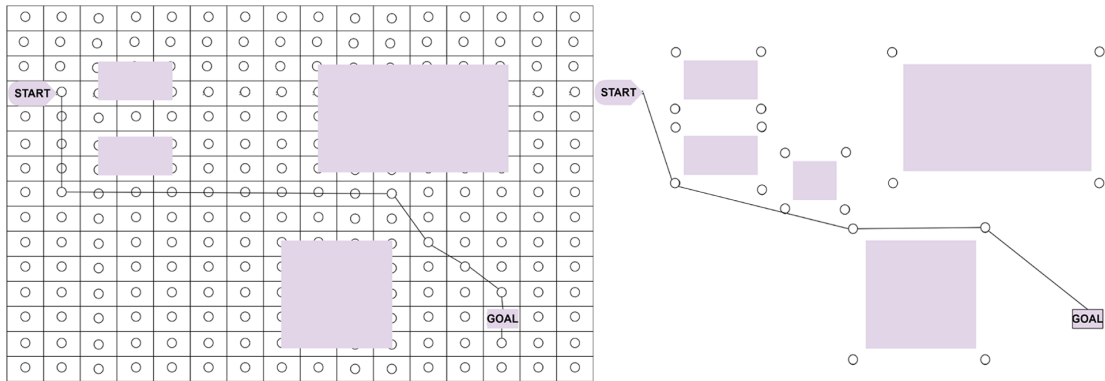


Figure 6.1: Comparison of configuration space using the graph-based approach and our proposed scheme

prerequisite modeling a disk, centered on the requested moving position and, whose radius is the threshold distance. The dotted line in Fig. 6.2 shows the desired trajectory between the start and the goal point. This trajectory is planned by including two interest points, i.e., P_1 and P_2 . The UAV is assumed to be present within a disk area of radius ρ centered on each point of the trajectory. In Fig. 6.2, the two solid lines surrounding a dotted line shows the location of the UAV. As these lines are quite far from any obstacle, it exemplifies the underlying mechanism ensuring collision avoidance.

6.3.1 Research contributions of this chapter

We summarize the main research contributions of this chapter as follows:

- We model uncertainties arising due to environmental disturbances as an uncertainty prerequisite and consider them in the UAV path planning process to design collision-free trajectories. This uncertainty prerequisite as a threshold acts as a guard distance and prevents the UAV from collision even if the underlying control system fails to meet its defined limitations.
- The proposed path planning method considers obstacles in an environment as rectangles since a tool like a visibility graph allows modeling of the environment as a graph. We seek guidance from obstacles by including their corners as graph nodes into the area tessellation

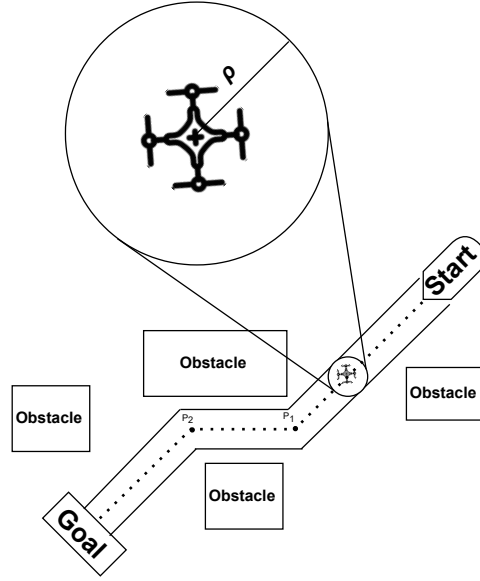


Figure 6.2: *Illustration of the trajectory designed by the proposed scheme with the uncertainty model*

process instead of considering the entire environment, thus reducing the computational load on the path planning.

- We demonstrate the efficacy of the proposed scheme, in terms of reduced computational load, shorter and collision-free trajectories, through simulations carried out in MATLAB's UAV Toolbox.

6.4 Proposed obstacle avoidance scheme

This section describes our proposed scheme. Subsection 6.4.1 defines the way we model obstacles as rectangles and the selection criteria for interest points. Subsection 6.4.2 introduces a suboptimal solution that finds admissible collision-avoiding trajectories while employing the Dijkstra algorithm. This section concludes with a discussion on the location of interest points defined.

6.4.1 Optimization problem statement

The design of the trajectory is regarded as an optimization problem, that of leaving from a starting point A at $t = 0$ and reaching the fastest way to an ending point B , while avoiding all obstacles modeled as rectangles of different widths, lengths and orientations. These rectangles are denoted $\mathcal{R}_1 \dots \mathcal{R}_R$ and defined as the set of points inside and on their borders.

The first assumption we are making here is that A , B , \mathcal{R}_r are all known to the UAV before its departure. We make use of a time-dependent virtual point denoted as $V(t)$ defined on $[0, T]$

where T is the time of flight. This virtual point moves at a speed no greater than v .

$$\forall t_1, t_2 \in [0, T], d(V(t_2), V(t_1)) \leq v |t_2 - t_1| \quad (6.1)$$

where d is the usual Euclidean distance. The second assumption, illustrated in figure 6.3, states that the UAV navigation system is able to follow $V(t)$ staying at a distance strictly below ρ during the time of flight.

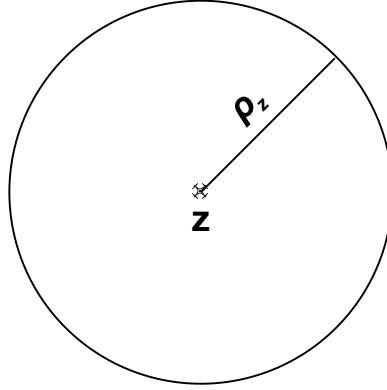


Figure 6.3: Disk centered on $V(t)$ where the UAV is assumed to be according to assumption 2.

$$\forall t \in [0, T], d(V(t), D(t)) < \rho \quad (6.2)$$

The collision of the UAV with an obstacle is modeled as:

$$\exists r \leq R, \quad \exists t \in [0, T], \quad D(t) \in \mathcal{R}_r \quad (6.3)$$

Using assumption 2, equation (6.3) is derived into a constraint on $V(t)$ ensuring collision avoidance.

$$\forall r \leq R, \forall t \in [0, T], \quad d(V(t), \mathcal{R}_r) \geq \rho \quad (6.4)$$

We denote the border of each rectangle \mathcal{R}_r as $\partial\mathcal{R}_r$, it is the set of points on one of the four line segments bordering \mathcal{R}_r .

The following theorem shows why the problem statement concerns only the borders of the rectangles.

Theorem 1. *Let $M(t)$ be a continuous mapping from $[0, T]$ to \mathbb{R}^2 and \mathcal{R} a rectangle with $\partial\mathcal{R}$ as border.*

$$\left. \begin{array}{l} d(M(0), \mathcal{R}) > 0 \\ \forall t \in [0, T] \quad d(M(t), \partial\mathcal{R}) \geq \rho \end{array} \right\} \Rightarrow \forall t \in [0, T] \quad d(M(t), \mathcal{R}) \geq \rho$$

We present a sketch of the proof in appendix A.1.

Each rectangle border $\partial\mathcal{R}_r$ is also defined by its four line segments denoted as $C_{4r-3}D_{4r-3}$, $C_{4r-2}D_{4r-2}$, $C_{4r-1}D_{4r-1}$, and $C_{4r}D_{4r}$. The collection of all line segments is denoted C_iD_i with $i \in \mathcal{I}$.

In order to apply theorem 1, we add a third assumption, that the beginning position is far enough from all obstacles.

$$\forall r \leq R, d(A, \mathcal{R}_r) \geq \rho \quad (6.5)$$

with $\rho > 0$. We are considering a set of admissible trajectories denoted as \mathcal{V} . Figure 6.4 illustrates an example of admissible trajectory.

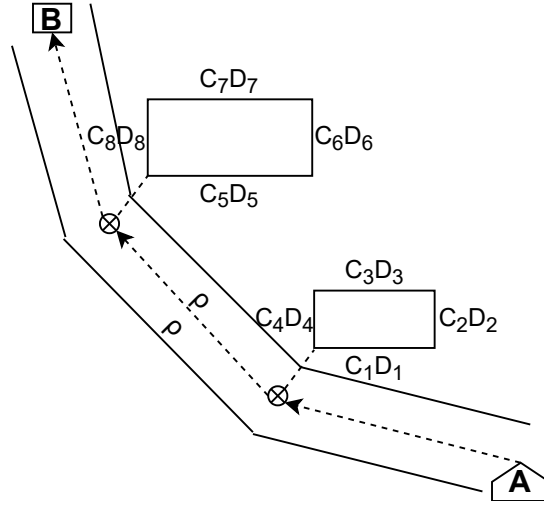


Figure 6.4: $V(t)$ is on the line joining A and B. The contour surrounding this line delineates the set of all points at a distance below ρ , where the UAV is assumed according to assumption 2.

$$\mathcal{V} = \left\{ V \in (\mathbb{R}^2)^{[0,T]} \left| \begin{array}{l} V(0) = A \\ V(T) = B \\ \forall t_1, t_2 \in [0, T], \quad d(V(t_2), V(t_1)) \leq v|t_2 - t_1| \\ \forall i \in \mathcal{I}, \forall t \in [0, T], \quad d(V(t), \overline{C_iD_i}) \geq \rho \end{array} \right. \right\} \quad (6.6)$$

$T_{\mathcal{V}}$ operates on \mathcal{V} and yields the time of flight.

$$T_{\mathcal{V}}(V) = \min \{t > 0 \mid V(t) = B\} \quad (6.7)$$

The optimization problem is finding:

$$V^* = \operatorname{argmin}_{V \in \mathcal{V}} T_{\mathcal{V}}(V) \quad (6.8)$$

6.4.2 Suboptimal solution with the Dijkstra algorithm

To make the optimization problem more tractable, we make a fourth assumption. The trajectory is a set of connected line segments joining A and B , where the segment ends are chosen among a predefined set of points denoted as $(P_j)_{j \in \mathcal{J}}$ and $V(t)$ moves at a speed v on each line segment. Denoting $\mathcal{V}' \subset \mathcal{V}$ as the mappings fulfilling this third assumption, we get a weighted graph representation of \mathcal{V}' as figure 6.5 shows.

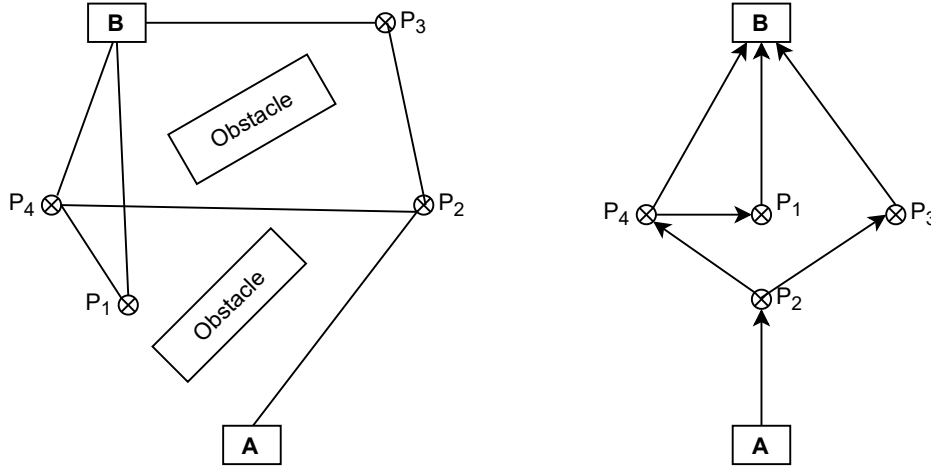


Figure 6.5: Four paths joining A and B consistent with assumption 3, shown on the left as positions of $V(t)$ and on the right using a graph structure.

This graph is denoted as $((P_j)_{j \in \mathcal{J}}, \mathcal{E}, T_{\mathcal{E}})$.

- $A \in \{P_j | j \in \mathcal{J}\}$ is the root.
- $B \in \{P_j | j \in \mathcal{J}\}$ is the sink.
- $(P_{j_1}, P_{j_2}) \in \mathcal{E}$ is an admissible edge connecting P_{j_1} and P_{j_2} if the line segment $\overline{P_{j_1}P_{j_2}}$ is at a distance equal to or greater than ρ of any obstacle.

$$(P_{j_1}, P_{j_2}) \in \mathcal{E} \Leftrightarrow \forall i \in \mathcal{I}, d(\overline{C_i D_i}, \overline{P_{j_1}, P_{j_2}}) \geq \rho \quad (6.9)$$

- Each edge is given a value which is the time of travel.

$$T_{\mathcal{E}}((P_{j_1}, P_{j_2})) = \frac{1}{v} d(P_{j_1}, P_{j_2}) \quad (6.10)$$

The following theorem shows the equivalence between finding the suboptimal solution $\hat{V} = \operatorname{argmin}_{V \in \mathcal{V}'} T_{\mathcal{V}}(V)$ and finding the least weighted path joining A and B . The latter is precisely what the Dijkstra algorithm solves efficiently.

Theorem 2. *A mapping V in \mathcal{V}' is a path $P_{j_1}P_{j_2}\dots P_{j_N}$ of $((P_j)_j, \mathcal{E}, T_{\mathcal{E}})$ joining A and B and its time of flight is the sum of all weights of the edges traversed.*

$$T_{\mathcal{V}}(V) = \sum_{n=2}^N T_{\mathcal{E}}((P_{j_{n-1}}, P_{j_n})) \quad (6.11)$$

We present a sketch of the proof in appendix A.2.

6.4.3 Discussing the location of the predefined points

The appropriate choice of (P_j) is crucial to the performance of the algorithm and their number is a trade-off between numerical complexity and performance. Interest points (P_j) should be located near each obstacle's line-segment's end and organized so as to allow paths enclosing each obstacle.

Considering a specific predefined point P and a unique obstacle \overline{CD} , not colliding line segments joining P have a range of valid angles, and the size of this range is denoted as $\Delta\Theta$.

$$\Delta\Theta = \left| \left\{ \Theta \mid \widehat{(\overrightarrow{MP}, \overrightarrow{CD})} = \theta \Rightarrow d(\overline{MP}, \overline{CD}) \geq \rho \right\} \right| \quad (6.12)$$

Figure 6.6 shows a horizontal obstacle of length $l = 1$ and five different locus of predefined points, each associated with a specific value of $\Delta\Theta$: $\frac{3\pi}{2} - \frac{\pi}{5}$, $\frac{3\pi}{2} - \frac{\pi}{10}$, $\frac{3}{2}\pi$, $\frac{3\pi}{2} + \frac{\pi}{10}$, $\frac{3\pi}{2} + \frac{\pi}{5}$. It is worth noting that smaller values of $\Delta\Theta$ are associated with predefined points closer to the obstacle.

Our proposition considers four predefined points for each obstacle which are sufficient to bypass the obstacle, located at a distance of $\rho\sqrt{2}$ of each segment end and having an angle of $\pm\frac{\pi}{4}$. These points, denoted as C^- , C^+ , D^- , D^+ are located on the intermediate locus associated to $\Delta\theta = \frac{3\pi}{2}$ as figure 6.6 shows.

$$\begin{cases} \angle(\overrightarrow{DC}, \overrightarrow{CC^-}) = \frac{\pi}{4} & CC^- = \rho\sqrt{2} \\ \angle(\overrightarrow{DC}, \overrightarrow{CC^+}) = -\frac{\pi}{4} & CC^+ = \rho\sqrt{2} \\ \angle(\overrightarrow{CD}, \overrightarrow{CD^-}) = -\frac{\pi}{4} & CD^- = \rho\sqrt{2} \\ \angle(\overrightarrow{CD}, \overrightarrow{CD^+}) = \frac{\pi}{4} & CD^+ = \rho\sqrt{2} \end{cases} \quad (6.13)$$

The following theorem states that any path joining a predefined point with some appropriate angle ensures a safe distance from the obstacle.

Theorem 3. *Let (C, D) be an obstacle, and C^-, C^+, D^-, D^+ be its associated predefined points*

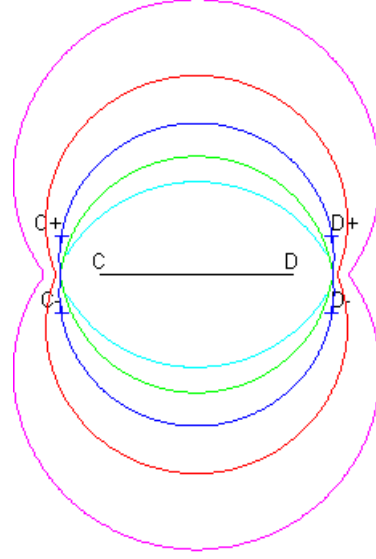


Figure 6.6: Five loci of predefined points corresponding to, $\frac{3\pi}{2} - \frac{\pi}{5}$, $\frac{3\pi}{2} - \frac{\pi}{10}$, $\frac{3}{2}\pi$, $\frac{3\pi}{2} + \frac{\pi}{10}$, $\frac{3\pi}{2} + \frac{\pi}{5}$, when counting outwards. At the center is the horizontal obstacle of length $l = 1$ and $\rho = 0.1$. The proposed predefined points are C^- , C^+ , D^- , D^+ .

and M be a given point.

$$\begin{aligned}
 \angle(\vec{CD}, \vec{C^-M}) \in [\frac{\pi}{2}, 2\pi] &\Rightarrow d(\overline{CD}, \overline{C^-M}) \geq \rho \\
 \angle(\vec{CD}, \vec{C^+M}) \in [0, \frac{3\pi}{2}] &\Rightarrow d(\overline{CD}, \overline{C^+M}) \geq \rho \\
 \angle(\vec{CD}, \vec{D^-M}) \in [-\pi, \frac{\pi}{2}] &\Rightarrow d(\overline{CD}, \overline{D^-M}) \geq \rho \\
 \angle(\vec{CD}, \vec{D^+M}) \in [-\frac{\pi}{2}, \pi] &\Rightarrow d(\overline{CD}, \overline{D^+M}) \geq \rho
 \end{aligned} \tag{6.14}$$

It is worth noting that all four statements are consistent with $\Delta\Theta = \frac{3\pi}{2}$ because

$$\left| \left[\frac{\pi}{2}, 2\pi \right] \right| = \left| \left[0, \frac{3\pi}{2} \right] \right| = \left| \left[-\pi, \frac{\pi}{2} \right] \right| = \left| \left[-\frac{\pi}{2}, \pi \right] \right| = \frac{3\pi}{2}$$

We present a sketch of the proof in appendix A.3.

6.5 Simulation testbed and experimental procedures

We conducted simulations under different environmental conditions by varying the number of obstacles and their dimensions, simulation area, UAV turning angles, UAV speed, start, and destination locations. We implemented the compared schemes in MATLAB-R2021a's UAV Toolbox³. We used different grid sizes ranging from a smaller configuration space of 400×600 to a larger one with 20000×20000 cells. Similarly, to evaluate schemes under a different number of

³<https://www.mathworks.com/products/uav.html>

Table 6.2: *Simulation parameters and environment*

Parameter	Value
Grid size	$400 \times 600 - 20000 \times 20000$
Number of obstacles	4 – 19
ρ	10 m [134]
UAV model	multirotor
UAV speed limit	5 – 9 m/s [152]
Gain for heading controller	2
Roll angle limit	45°
Number of simulations	250 – 300
CPU	Intel(R) Core(TM) i5-6200U 2.30 GHz
RAM	12.00 GB
Operating system	Microsoft Windows 10 Pro

obstacles, we varied the number of obstacles from a smaller value of 4 to a higher value of 19. We considered the multirotor UAV model to determine the trajectory tracking effectiveness of the proposed scheme. We used ρ value as in [134] and it relies on accuracy of Global Navigation Satellite System (GNSS) and uncertainty caused due to environmental disturbances. In all simulation environments, the start and destination locations are shown by solid red and green circular points, respectively. The rectangular obstacles with solid blue circular points represent interest points proposed by our path planning technique. Table 6.2 presents the simulation parameters along with the operating system parameters.

6.5.1 Performance metrics

We used the following performance metrics in our performance evaluation tests:

- Total distance traveled by a UAV from a start point to the endpoint.
- Trajectory tracking error with the variation in UAV's turning angles.
- Trajectory tracking error with the variation in UAV's moving speed.
- Algorithm computational load by looking at the program execution time.

6.6 Results and discussion

This section presents experimental results for the proposed pathfinding technique along with the artificial potential field method. We compare our proposed scheme with the APF method, which is known to have a precise mathematical model with a low computational load and is suitable for real-time applications [128]. Like most other previously proposed approaches [135, 136, 144], this APF method does not consider a specific uncertainty model. Instead, it uses a free parameter to balance the trade-off between obstacles and reducing the path length. To achieve a fair comparison with our proposal, for each specific set of obstacles of the following experiment, an iterative algorithm seeks a value for this free parameter minimizing the path length while remaining at the threshold distance ρ of all obstacles.

6.6.1 Total distance traveled by the UAV

A path planning technique should keep a balance between the length of a planned trajectory and a collision avoidance algorithm. A path planning algorithm looking for the shortest path may collide with an obstacle. On the other hand, if more weight is given to collision avoidance algorithm, this will result in longer trajectories. In this context, to evaluate the performance of the proposed scheme, we performed 250 experiments with a different number of obstacles and their dimensions, simulation area, start, and destination locations.

Fig. 6.7 displays the distance traveled for the proposed scheme and the APF method for 200 experiments. To better understand the shortest distance between the start and destination locations, we also plot the Shortest Direct Distance (SDD) that the UAV would have traveled in the absence of obstacles. The distance traveled by APF, the proposed scheme, and SDD is represented by the black plus sign, red solid circular point, and blue steric symbol, respectively. In Fig. 6.7, the y-axis shows the distance traveled while the x-axis shows the experiment number. As can be inferred from Fig. 6.7, the proposed scheme designs a trajectory with a shorter length compared to APF in every experiment. The difference in the total distance traveled is smaller on the left part of the figure wherein the number of obstacles and simulation grid dimensions are lower. This difference becomes more prominent on the right side of Fig. 6.7, where there is a higher number of obstacles and larger grid dimensions. The proposed scheme resulted in the same distance as indicated by SDD because it tries to follow the shortest path while bypassing any obstruction using pre-defined interest points.

To observe the difference of traveled distance more closely, Fig. 6.8 shows the simulation results for the large dimensional size of simulation environment with 5000×5000 grid size, varying

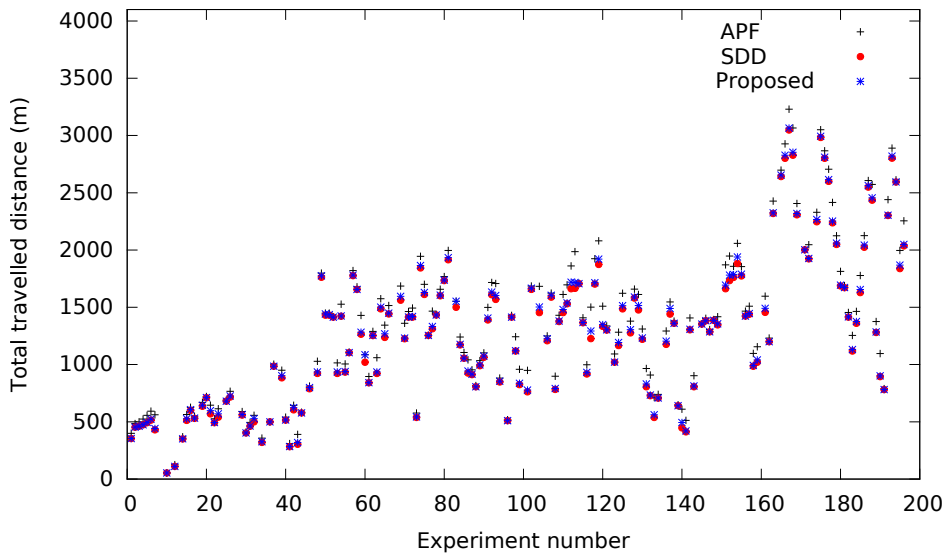


Figure 6.7: Total distance traveled by the UAV [maximum grid size 2500×2500]

start and destination points, and number of obstacles approaching 19. The highest observed traveled difference occurs for experiment number 28, wherein this value reaches 1433 m.

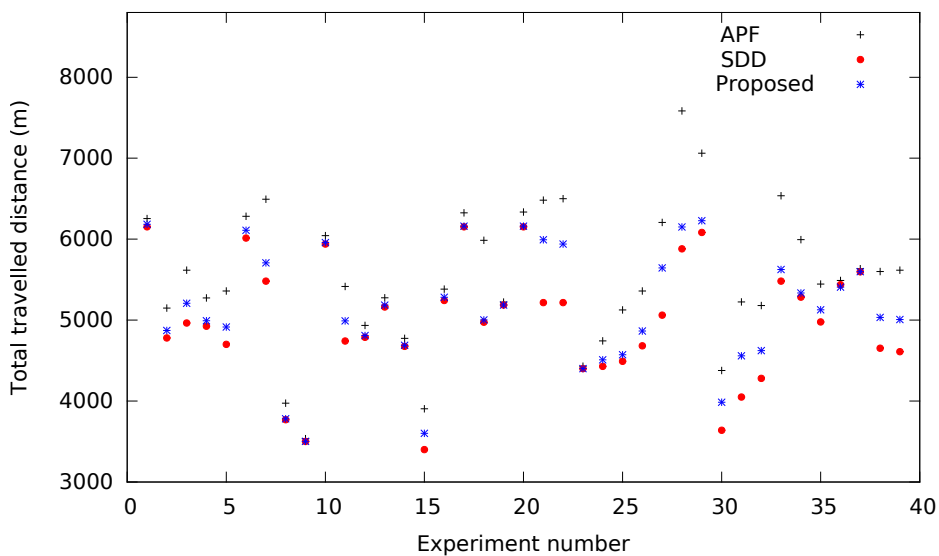


Figure 6.8: Total distance traveled by the UAV [grid size 5000×5000]

Fig. 6.9 illustrates trajectories designed by the compared schemes under different simulation scenarios. The red contour surrounding the desired proposed black line trajectory is the region within which the UAV is expected to take during its flight. It can be inferred from the figure that APF takes a longer curved paths compared to the proposed method that uses pre-defined interest points to get a linear shorter paths to reach the destination.

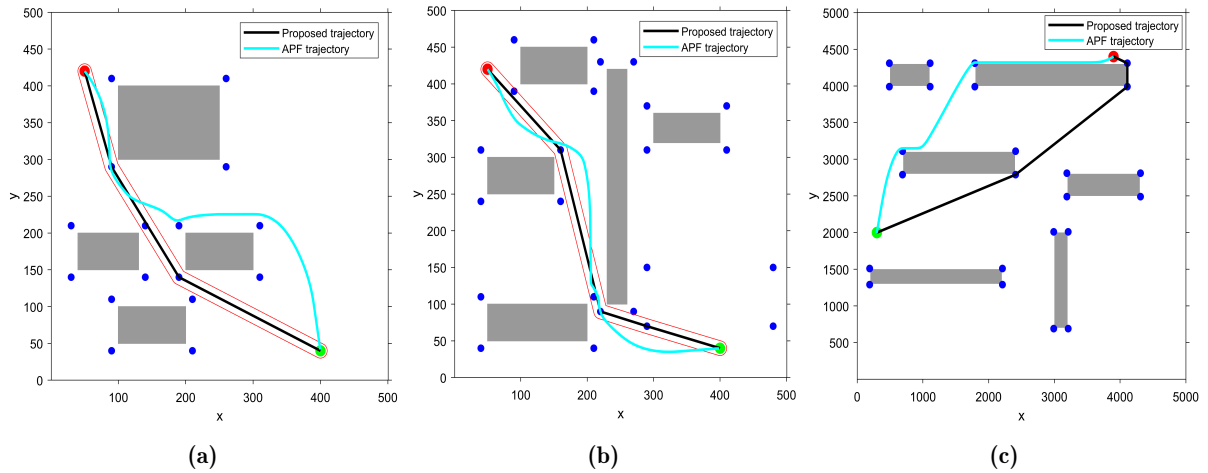


Figure 6.9: Trajectories for the schemes compared (a) 4 obstacles, 500×500 grid size (b) 5 obstacles, 500×500 grid size (c) 6 obstacles, 5000×5000 grid size

6.6.2 Trajectory tracking error vs. UAV's turning angles

This subsection evaluates our claim for designing collision-free trajectories under varying UAV turning angles. We consider a scenario as shown in Fig. 6.10 with five obstacles and the UAV speed of 7 m/s. Fig. 6.11 shows zoomed-in parts for various UAV turning angles around vertex

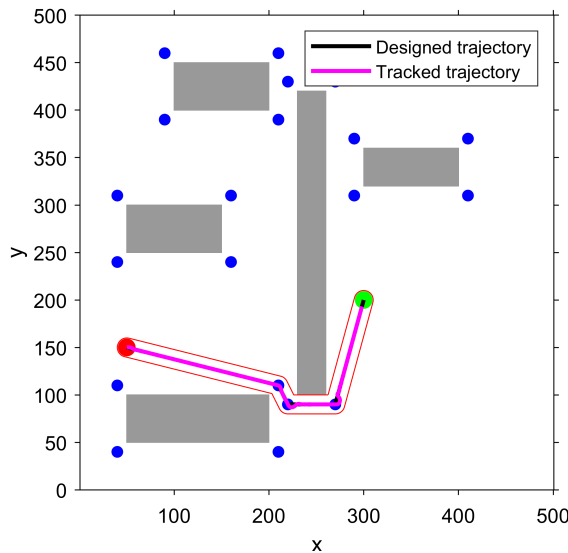


Figure 6.10: Simulation scenario with 75° turning angle

represented by the dashed arrow. It can be observed from the figure that the tracked trajectory points remain within the defined limits, i.e., within the red contour. As can be inferred from Fig. 6.11, the UAV trajectory deviates widely with an increase in UAV turning angle to maintain its dynamics constraints. Fig. 6.12 presents the trajectory tracking errors for the various UAV's

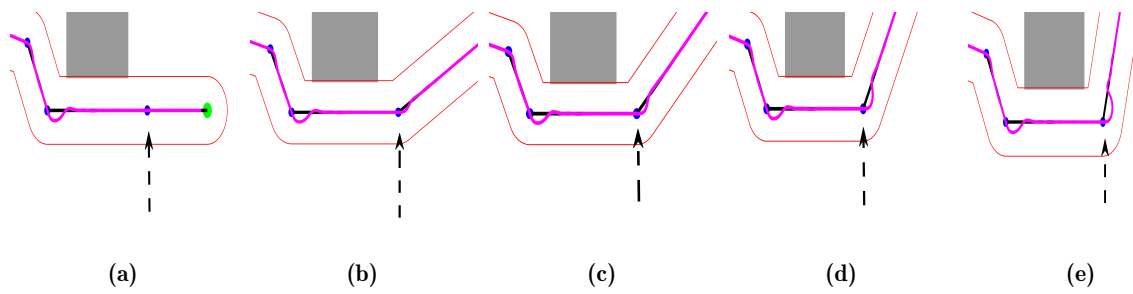


Figure 6.11: Zoomed-in plots with arrow pointing towards vertices with angles (a) 0° (b) 30° (c) 45° (d) 60° (e) 75°

turning angles. This figure also substantiates that the tracking error gets increased as we move from 0 degrees to 75 degrees. However, it is worth mentioning that trajectory deviation remains within limits and validates our collision-free trajectory designing claim.

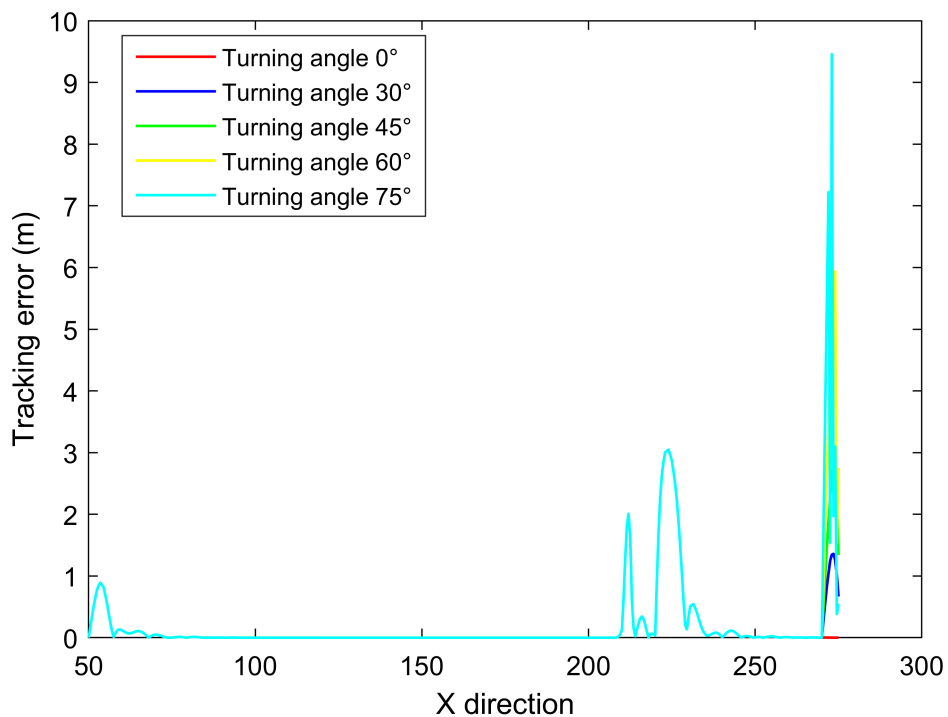


Figure 6.12: Tracking error for different turning angles

6.6.3 Trajectory tracking error vs. UAV's moving speed

This section analyzes the effect of UAV speed on trajectory tracking. As shown in Fig. 6.13, we consider a simulation scenario with five obstacles and varying UAV speeds. Fig. 6.14 presents trajectory tracking errors for different UAV speeds. Tracking error for 5 m/s is the lowest among

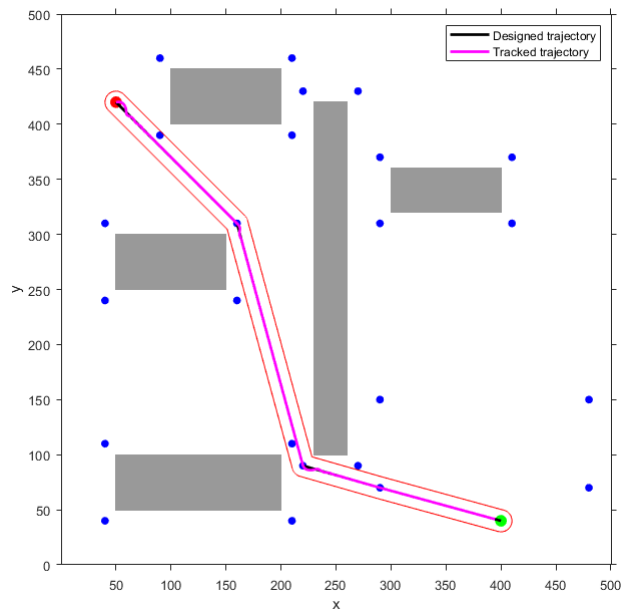


Figure 6.13: Scenario considered to determine tracking error under different UAV's speed

all and becomes higher as UAV speed approaches 9 m/s because UAV's control system output stabilizes more quickly at lower speed values. Tracking errors for the various UAV speed tests lie within the defined contour threshold and validate collision-free trajectory tracking for the proposed scheme.

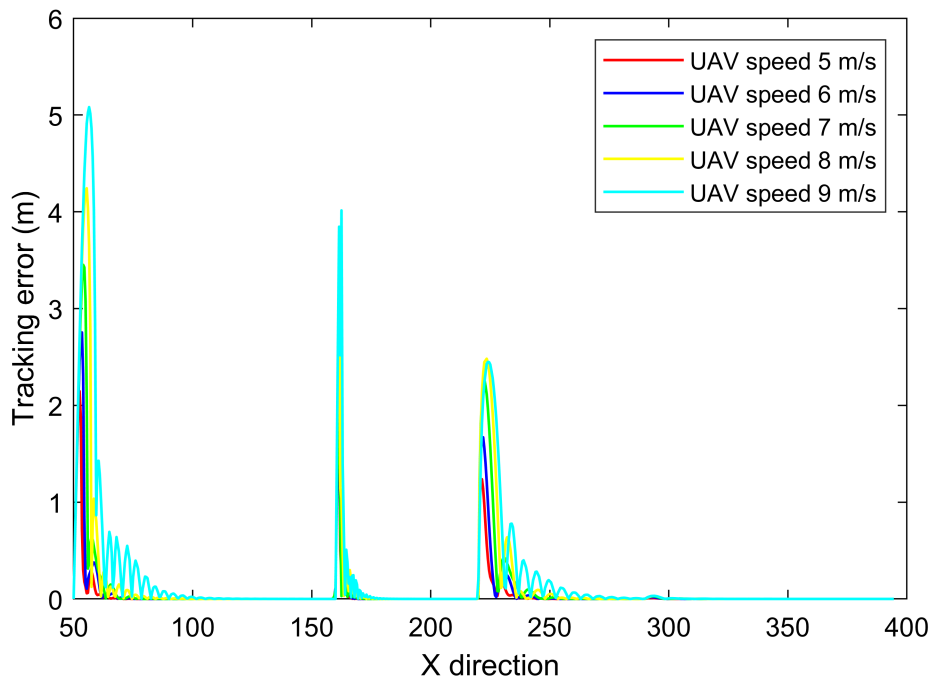


Figure 6.14: Tracking error for different UAV's speed

6.6.4 Computational load comparison

To estimate the computational load offered by the proposed scheme, we simulated different scenarios by varying the distance between the start and the destination location and the simulation grid size. Fig. 6.15 presents the computational load offered by the path planning techniques in terms of execution times under varying simulated grid sizes. Both schemes incur comparable load until 6000×6000 grid size, after which APF incurs higher execution time. As Fig. 6.15 shows, the proposed method, in terms of execution time is not affected by the variation in the grid size. The reason behind this result is that APF considers the entire environment during the path planning process whereas the proposed scheme is only concerned with the interest points.

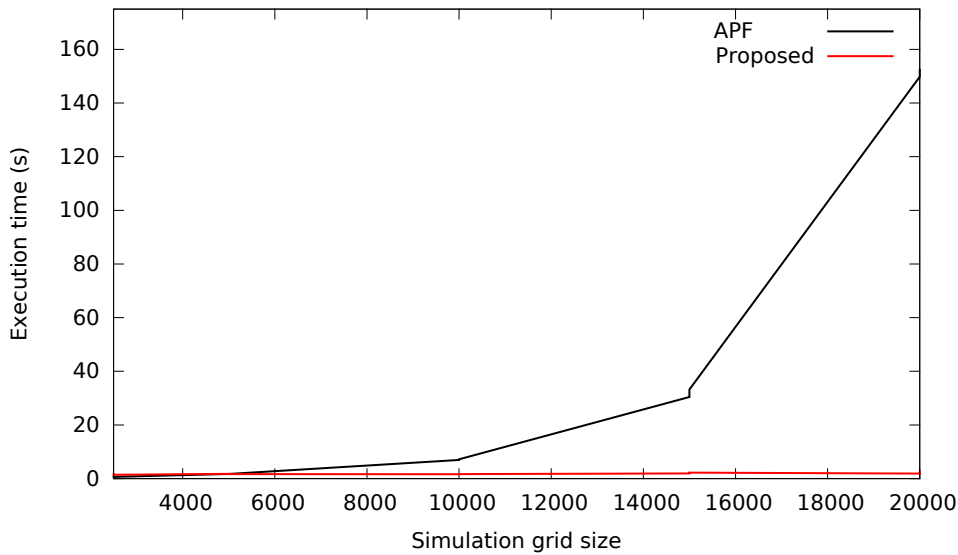


Figure 6.15: Impact of simulation grid size on execution time [19 obstacles]

Fig. 6.16 plots the execution time with respect to the distance between the start and destination locations. From the figure, we observe that although both schemes are unaffected by the variation in the distance, but it is worth mentioning here that our proposed method outperforms APF in terms of execution time.

6.7 Conclusion

A collision-free path planning is of paramount importance to ensure the safety of UAVs and humans on the ground. Our proposed UAV path planning technique works by taking advantage of interest points defined around the rectangular obstacles. We used the disk-based uncertainty model to eliminate the chances of a collision by taking care of the environmental disturbances. This uncertainty model as a threshold distance provides an extra layer of safety to the underlying

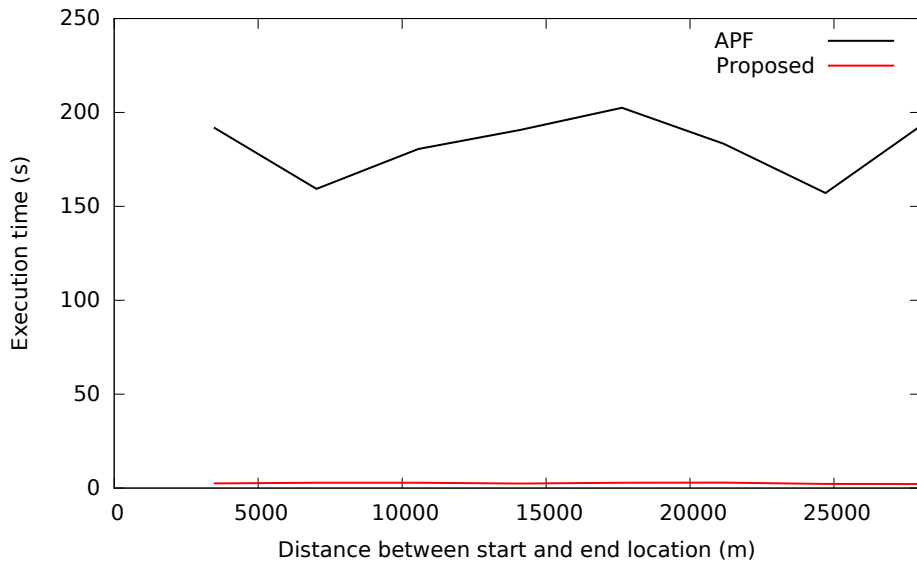


Figure 6.16: Impact of the distance between the start and destination location on execution time [grid size 20000×20000 , 19 obstacles]

control system, i.e., to act as a guard if the control system overshoots its defined limits. Moreover, due to the low computational demand regardless of the deployed environment dimensions, our approach becomes promising for real-time path planning.

As turning points show higher tracking errors, UAV safety can be enhanced further by assigning a higher threshold value to these points. This work is also applicable for multiple destination locations, with each location as a new interest point. In future work, we plan to develop a UAV-based network comprising roadside units to provide new interest points at runtime. We also plan to modify our proposed model by integrating UAV kinematics into the path planning method.

A connectivity aware path planning for a fleet of unmanned aerial vehicles

Abstract

Unmanned Aerial Vehicles (UAVs) are known for their highly dynamic nature, as a result of which their applications demand high design consideration in urban areas. It is imperative to have trajectories that avoid UAV-to-UAV and UAV-to-obstacle collision to ensure the safety of a fleet and people on the ground. Moreover, many applications, like temporary network provision, require continuous backhaul fleet connectivity. This chapter simultaneously addresses UAVs' path planning and routing issues to propose connectivity-aware path planning for a fleet of UAVs in an urban environment. The proposed scheme is a graph-based offline path planning for a fleet of UAV with line formation that ensures continuous backhaul fleet connectivity. This feature allows any UAV to play the role of leader and guide the entire fleet according to the desired speed. Thanks to the continuous backhaul connectivity, the Base Station (BS) can disseminate commands to the connected fleet as required. Fleet line formation acts as a backbone network and allows additional UAVs or ground users to become a part of this network. The proposed approach is implemented in MATLAB and evaluated in a network simulator. The simulation results demonstrate that the proposed scheme provides collision-free trajectories while ensuring continuous BS connectivity¹.

¹ N. Bashir, S. Boudjit, G. Dauphin. 2022. "A connectivity aware path planning for a fleet of UAVs in an urban environment," *IEEE Transactions on Intelligent Transportation Systems*. [Under review]

Chapter content

7.1	Introduction	121
7.2	Related work	122
7.3	Scope of the proposed work	124
7.3.1	Initialization and application scenarios	124
7.3.2	Problem statement	126
7.4	Proposed connectivity-aware path planning algorithm	128
7.4.1	Kinematics of UAVs along a given path	128
7.4.2	Path planning	134
7.4.2.1	Generating the graph nodes	135
7.4.2.2	Generating edges	135
7.4.2.3	Generating weights	136
7.4.2.4	Generating a valid path	137
7.5	Simulation testbed and experimental procedures	139
7.5.1	Performance metrics	140
7.6	Results and discussion	141
7.6.1	UAVs as a source of data	141
7.6.1.1	UAV fleet moving with varying speed	141
7.6.1.2	UAV fleet moving with fixed speed	143
7.6.2	Ground users as a source of data	145
7.6.2.1	UAV fleet moving with varying speed	146
7.6.2.2	UAV fleet moving with fixed speed	147
7.7	Conclusion	149

7.1 Introduction

The formation of a UAV fleet is a challenging task due to the unique dynamics associated with UAVs[27]. Path planning and data routing for a fleet of UAVs are influenced by these characteristics and demand careful consideration. Having a fleet of UAVs in a mission increases the chances of collisions[28, 29], moreover high relative mobility results in frequent link failures[30]. To control the formation of UAVs, a continuous exchange of information among UAVs, such as the current locations and velocities, is necessary[35, 36]. Considering the dynamic nature of UAVs, the absence of state information due to communication problems may lead to severe consequences for the UAVs and people on the ground[37]. Moreover, high communication demand to control a UAV formation with low bandwidth resources of UAVs network renders no space for application requiring high quality-of-Service (QoS) such as search and rescue operation during a natural disaster scenario. Collision-free path planning is of paramount importance to introduce autonomous operating capability into a fleet of UAVs[153].

Designing routing protocols for a dynamic and resource-constrained UAV network poses serious challenges. Path planning and formation control should go hand in hand as conventional ad hoc routing protocols are not capable enough to cope with such a rapidly changing topology of UAVs networks. A UAV network may have different connectivity requirements based on a specific application. These requirements include always connected, periodic, and delay-tolerant connectivity. In an always-connected network, all UAVs have Base Station (BS) connectivity all the time. In a periodic approach, UAVs get an opportunity to exchange information with each other at relay points. Delay-tolerant UAV networks are devoid of continuous connectivity, and the exchange of information becomes possible only in the proximity of BS. Having just fleet connectivity is sufficient to establish cooperative relationships among UAVs, but having no connection with BS results in the lack of real-time aspect.

This chapter simultaneously addresses path planning and routing issues for a fleet of UAVs to provide collision-free trajectories, continuous monitoring, and end-to-end link continuity with BS. We present a graph-based path planning method with obstacles modeled as line segments. Four points defined around each obstruction represent nodes of the graph that help in going around those obstructions. Traceable non-colliding edges which do not yield sharp turns are defined afterward. The proposed technique iteratively applies the Dijkstra algorithm and removes any non-compliant edges until it finds a valid path.

The main contributions of this chapter are as follows:

- We consider a connectivity-aware path planning model for a fleet of UAVs with line formation. The proposed path planning ensures a collision-free trajectory starting from the departure position to the landing position. The model also takes into account a safe distance margin during the path planning to encompass uncertainties that may arise due to environmental disturbances.
- The line formation forms a connected dominating set acting as the backbone of the network. This formation ensures continuous coverage over the designed trajectory and provides end-to-end link connectivity with the BS. Moreover, at any moment, BS can change the course of the mission by sending new control messages to the connected network. Any UAV can make a request to lead the entire fleet in order to make the fleet move at some specific speed or request for hovering.
- The proposed scheme is scalable in the sense that any new UAV can come and join the backbone of the network to utilize the high QoS routing resources.
- The proposed scheme is implemented and evaluated in MATLAB and the network simulator, respectively. The efficacy of this scheme remains satisfactory in terms of backhaul and fleet connectivity, collision avoidance, and fleet scalability with the variations in fleet speed, leader's location, and number and nature of data originating nodes.

We organized the rest of the chapter as follows. Section 7.2 presents the related work in the field of UAVs covering path planning and routing issues. Section 7.3 describes the system considered. Section 7.4 provides in detail the proposed path plan method for a fleet of UAVs. Section 7.5 presents simulation testbed and experimental procedures. Results along with the discussion are presented in section 7.6. Section 7.7 concludes the chapter.

7.2 Related work

This section reviews related work on path planning, routing protocol designing, and connectivity-aware path planning for UAV networks.

Zhang et al.[154] propose a new UAV path planning approach using a navigational system-based localization error map in an urban environment. It avoids hazardous areas by reducing the effect of multipath and non-line-of-sight signal reception through predicting position errors in different areas. This approach uses a 3D building model, broadcast almanac, and ray-tracing simulation to generate the position error map and utilizes a modified A* algorithm to generate feasible

trajectories. Additional work of error map generation and processing of the map before flight demands a high computational load.

Yin et al.[155] propose a multi-objective UAV path planning approach for a dynamic urban environment. This approach explores feasible paths while ensuring the safety of a UAV and guarantees travel time. It uses two types of safety index maps to tackle static and dynamic obstacles. The offline search uses a static safety map to avoid static obstacles and reduce travel time. The online search uses a dynamic safety map to go around unexpected obstacles quickly. The computational complexity of generating a static safety index map is high in this path planning.

Chen et al.[156] come up with a path planning technique for multi-UAV formation in a known and realistic environment using a modified Artificial Potential Field (APF) method. This modification includes additional control force with its solution provided by the optimal control method. This approach introduces path planning and particle dynamics models for a single UAV. A virtual velocity rigid body and virtual target point formulate a path planning for a UAV formation. This approach requires a precise definition of repulsive potential to avoid a virtual point from entering into obstacle areas.

Filippis et al.[157] propose a UAV path planning for a 3D urban environment. This method uses a graph-based Theta* search algorithm to reduce the path length by including a lesser number of node points. Moreover, the Theta* algorithm used, results in smooth trajectories having fewer unnecessary altitude changes. This method reduces the searching time by using an effective nodal expansion technique where obstacles result in the blockage of a path. This method has higher computational complexity as compared to the A* algorithm.

Yoon et al.[158] propose an adaptive UAVs path planning to deliver delay-sensitive information during a natural disaster situation. The main objective of this technique is to find optimal UAV paths and serve the maximum number of nodes within a specific packet deadline. A distributed path planning mechanism determines the next visiting point constrained by delivery and packet deadline time. A task division method reduces the overall travel time by collaboratively distributing tasks among different UAVs. This approach lacks a real-time aspect and does not incorporate obstacles into its model.

In[37], the authors propose a coordination protocol for maintaining a swarm of UAVs during a mission. In this centralized approach, a master UAV synchronizes all other UAVs at intermediate points during a mission. This method attains a high level of swarm cohesion and a lower level of synchronization delays even with lossy communication channels. This technique does not

consider obstacles, and BS cannot feed waypoints in real-time. It is imperative to have a reliable routing specifically for applications like disaster management, rescue operations, and battlefields. Toorchi et al.[160] propose a skeleton-based intelligent routing protocol for dynamic networks of UAVs. It tries to reduce routing complexity by exploiting the structure of a swarm formation. A swarm formation-based geometric addressing model approximates the location of each UAV that is necessary to identify the role for each UAV. During a change in UAV formation, UAVs move according to formation morphing technique to have minimum impact on the geometric addresses. An adaptive leaf-like pipe acts as a central framework for routing purposes. The absence of path planning and no connectivity with BS are the shortcomings of this scheme.

Hayat et al.[159] propose multiple objective path planning for Search And Rescue (SAR) operations with QoS requirements. This path planning approach considers two adaptive strategies. In the first one, search, inform, and monitoring tasks are optimized simultaneously. In the second case, search and inform are optimized initially, followed by monitoring optimization to get optimum positions. The SAR mission starts by detecting a static target in the shortest possible time. A UAV, after the detection of a target, carries the location information to the BS. This scheme optimizes coverage and connectivity but lacks in responding quickly, specifically when UAVs are deployed far enough from BS and require more time to form a connected network. Table 7.1 summarizes all path planning and routing techniques discussed in this section.

7.3 Scope of the proposed work

7.3.1 Initialization and application scenarios

We consider an Inform and Monitoring (IM) mission with a fleet of UAVs along with the deployment of multiple BSs in an urban environment. Before the mission starts, the proposed path planner is informed about the location of obstacles, departing and landing points and angles, communication distance, the uncertainty of the navigation system in terms of safe margin distance, speed of UAVs, location of intermediate BSs to visit during the mission. The connectivity-aware path planner estimates the number of UAVs required using the information given and informs each UAV accordingly about its mission-specific collision-free trajectory. The tracking of the designed trajectory by a fleet of UAVs mainly depends upon the specific application. The proposed scheme considers the following application scenarios:

- Provision of situational information, continuous monitoring, and temporary network connectivity at a natural disaster location quickly. Moreover, the fleet on their way will inform

Table 7.1: Summary of various path planning and routing techniques for UAVs

Reference	Research objective/Approach followed	Strengths	Weaknesses
Zhang et al.[154]	<p>→Reduce multipath effect in urban path planning</p> <p>Use of predicted positions error map to avoid hazardous areas</p>	Safely operable in urban environment with low altitude	<p>Generation of error prediction maps</p> <p>High computational complexity of error maps</p> <p>Path may have sharp turning angles</p>
Filippis et al.[157]	<p>→Shorter and smooth paths in 3D urban environment</p> <p>Having less number of nodes and avoiding unnecessary altitude changes</p>	Low searching time in case of re-planning	Higher computational complexity
Yoon et al.[158]	<p>→Optimal path and serve maximum nodes within packet specific deadline</p> <p>Distributed path planning constrained by delivery and packet deadline time</p>	Optimal paths with overall travel time and lower packet delivery delays	Lacking real-time aspect and does not consider obstacles
Fabra et al.[37]	<p>→Increase coverage area and accelerate mission completion time</p> <p>Use of swarm synchronization of all UAVs at intermediate points</p>	<p>Swarm cohesion</p> <p>Lower synchronization delays even with lossy communication channels</p>	<p>No real-time aspect</p> <p>No obstacle consideration</p>
Toorchi et al.[160]	<p>→Reduce routing complexity for dynamic UAVs networks</p> <p>Skeleton-based (swarm structure) intelligent routing protocol</p>	<p>Higher throughput</p> <p>Adaptable to changes in formation</p>	<p>Absence of path planning</p> <p>No connectivity with BS (No real-time aspect)</p>
Hayat et al.[159]	<p>→Optimize coverage and connectivity in search and rescue missions</p> <p>Joint optimization of search, inform and monitoring tasks</p> <p>Incorporation of communication in the path design process</p>	Tunable to application requirements to prioritize coverage over connectivity or the other way around	Lacking in response time, specifically when UAVs are deployed far enough from BS
Our proposed method	<p>→Connectivity-aware collision-free path planning to provide real-time aspect for delay sensitive applications</p> <p>Path planning constrained to fleet connectivity and ensuring all time connection with at least one BS</p>	<p>Collision-free paths and tolerable to sensor uncertainties</p> <p>Real-time communication</p> <p>Scalable in a sense that any UAV can join connected dominating set to use routing resources</p> <p>Continuous coverage over the designed trajectory</p>	Lacks consideration of UAV dynamics, but path planning considers a parameter to compensate for uncertainties that may arise due to environmental disturbances

all vehicles on the roads about the disaster and keep them away from this specific path to help the rescue team interfere quickly.[Fig. 7.1a]

- Continuous tracking and monitoring of a public rally. The first UAV can lead the fleet according to the variation in the speed of the rally.[Fig. 7.1b]
- To monitor the public in real-time to enforce some special rules at different hotspot locations, like social distancing rule specifically during the current pandemic of COVID-19. Furthermore, any UAV can request the entire fleet to stop (hover) to report any violation in real-time.[Fig. 7.1c]

Each BS periodically broadcasts beacons that ensure each UAV connectivity with a specific BS and with each other. Each beacon message contains BS identity and timestamp. Data packets take the reverse of the path taken by the beacon messages to reach BS.

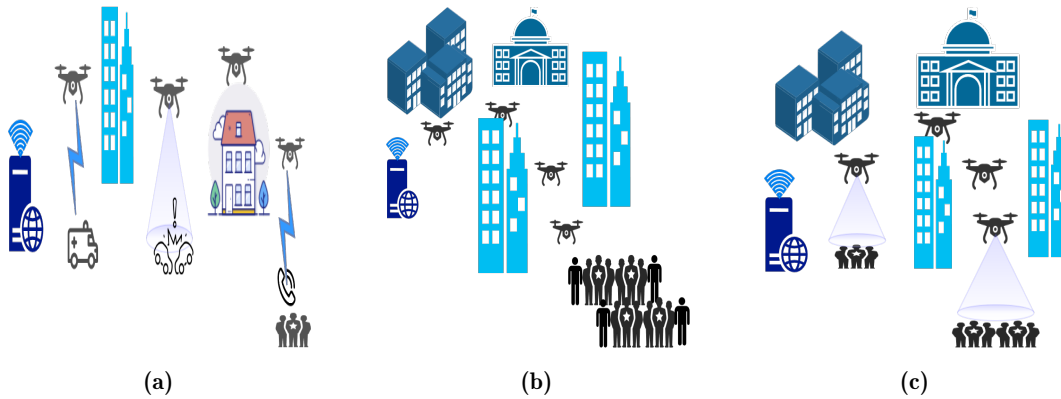


Figure 7.1: Application scenarios (a) Natural disaster scenario. (b) Continuous tracking and monitoring of a rally. (c) Monitoring of public areas in real-time

7.3.2 Problem statement

The three missions described in section 7.3.1 require from the UAVs the following four objectives and the evaluation of, N , the number of UAVs needed.

- Collisions are avoided between any two UAVs.

$$\forall t, n_1 \neq n_2, R_{n_1}(t) \neq R_{n_2}(t) \quad (7.1)$$

where $R_n(t)$ stands for the location of the n^{th} -UAV at time t .

- ii. Collisions are avoided between any UAV and any obstacle.

$$\forall t, n, i, R_n(t) \notin \overline{C_i D_i} \quad (7.2)$$

where $\overline{C_i D_i}$ are the line segments modeling obstacles.

- iii. All UAVs may communicate at all time with A or B which are base stations.

$$R_n(t) \boxtimes A \text{ or } R_n(t) \boxtimes B \quad (7.3)$$

where \boxtimes denotes the ability to do UAV-network communication.

- iv. The speed at which all UAVs are on average travelling can be set freely.

$$\forall n, \left| \frac{d}{dt} V_n(t) \right| \stackrel{a.e.}{=} v(t) \quad (7.4)$$

where $V_n(t)$ is the n^{th} virtual UAV that the n^{th} UAV follows closely. This equation holds only once the n^{th} -UAV has left its starting point and as long as it has not reached its final point.

In our proposal, these objectives derive from precise and constraining rules to which all UAVs are to abide. These rules are displayed in the form of four assumptions.

- i. The navigation system of the n^{th} -UAV $R_n(t)$ is able to follow $V_n(t)$ within a safe margin ρ_c .

$$\forall n, R_n(t) V_n(t) < \rho_c \quad (7.5)$$

where $R_n(t) V_n(t)$ is the Euclidean distance between $R_n(t)$ and $V_n(t)$.

- ii. The prescribed speed $v(t)$, at time t , is not known to any UAV before time t . It should not be greater than a maximum speed denoted v_{\max} (i.e. v_{\max} has to be smaller than the technical speed threshold as it should be followed by the UAV's navigation system possibly on a long period of time).

$$v(t) \leq v_{\max} \quad (7.6)$$

To simplify the mathematical technicalities, $v(t)$ is assumed to be a finite linear combination of indicator functions having a positive limit as $t \rightarrow +\infty$. The supports of these indicator functions are left-closed and right-open intervals.

- iii. Each UAV is able to communicate with another UAV when their mutual distance is below ρ_r . Their ability to directly communicate with one another is denoted \bowtie .

$$R_{n_1}(t)R_{n_2}(t) \leq \rho_R \Leftrightarrow R_{n_1}(t)\bowtie R_{n_2}(t) \quad (7.7)$$

The departure and terminal point, denoted **A** and **B**, are also base stations.

$$\begin{aligned} R_n(t)\mathbf{A} \leq \rho_R &\Leftrightarrow R_n(t)\bowtie\mathbf{A} \\ R_n(t)\mathbf{B} \leq \rho_R &\Leftrightarrow R_n(t)\bowtie\mathbf{B} \end{aligned} \quad (7.8)$$

ρ_R is generally much greater than ρ_c , we assume here that at least

$$\rho_R \geq 2(1 + \sqrt{2})\rho_c \quad (7.9)$$

- iv. The line segments $\overline{C_iD_i}$, modeling the obstacles, are motionless and known prior to departure.

7.4 Proposed connectivity-aware path planning algorithm

7.4.1 Kinematics of UAVs along a given path

We consider in this section being provided with a path $\mathbf{P} \in \mathbb{P}$ defined in the following definition. We propose here a precise time and space description of the location of each UAV and give theorems showing that this description meets the objectives listed in section 7.3.2 when \mathbf{P} complies with some prerequisites.

Definition 1. \mathbb{P} is a set of paths, each defined as an array of distinct points.

$$\mathbf{P} = [P_1 \dots P_K] \in \mathbb{P} \Leftrightarrow \forall k \in \{1 \dots K\}, P_k \in \mathbb{M} \text{ and } [k \neq k' \Rightarrow P_k \neq P_{k'}] \quad (7.10)$$

where \mathbb{M} is a set of points in a 2D-space.

We consider that the provided path is *complete* when it joins **A** and **B**. The set of all complete paths is denoted \mathbb{P}_c .

$$\mathbb{P}_c = \{[P_1 \dots P_K] \in \mathbb{P} \mid P_1 = \mathbf{A} \text{ and } P_K = \mathbf{B}\} \quad (7.11)$$

The length of a path is

$$\mathcal{L}(\mathbf{P}) = \sum_{k=1}^{K-1} P_k P_{k+1} \quad (7.12)$$

To describe how UAVs are displayed in the departure area and in the arrival area, we define two rays starting at A and B and having an angle of $\pi + \Theta_A$ and Θ_B with \overrightarrow{AB} .

$$\begin{aligned} \Delta_A &= \left\{ M \mid \angle(\overrightarrow{AB}, \overrightarrow{AM}) = \pi + \Theta_A \right\} \text{ and} \\ \Delta_B &= \left\{ M \mid \angle(\overrightarrow{AB}, \overrightarrow{BM}) = \Theta_B \right\} \end{aligned} \quad (7.13)$$

These rays should be far enough from any obstacles.

$$\forall i, d(\Delta_A, \overline{C_i D_i}) \geq \rho_c \text{ and } d(\Delta_B, \overline{C_i D_i}) \geq \rho_c \quad (7.14)$$

These rays should also be far enough from each others.

$$d(\Delta_A, \Delta_B) \geq 2\rho_c \quad (7.15)$$

To avoid collisions during departure and arrival, UAVs are displayed along Δ_A at departure and Δ_B at arrival. The distance between each consecutive slots is $2\rho_c\sqrt{2}$.

The orientation's motion in Δ_A and Δ_B are described with two unitary vectors $\overrightarrow{e_0}$ and $\overrightarrow{e_K}$.

$$\angle(\overrightarrow{AB}, \overrightarrow{e_0}) = \Theta_A \text{ and } \angle(\overrightarrow{AB}, \overrightarrow{e_K}) = \Theta_B \quad (7.16)$$

The exact starting and finishing locations of each virtual UAV are N points denoted $P_{0,n}$ and $P_{K+1,n}$.

$$\overrightarrow{AP_{0,n}} = -(n-1)\overrightarrow{e_0} \text{ and } \overrightarrow{BP_{K+1,n}} = (N-n)\overrightarrow{e_K} \quad (7.17)$$

To ease notations, an unneeded index n is added to P_k .

$$\forall k \in \{1 \dots K\}, \quad P_{k,n} = P_k \quad (7.18)$$

Once \mathbf{P} is defined, the exact location of $V_n(t)$ can be derived from the distance traveled. The following definition computes $\mathcal{S}(t)$ a distance, s , that a non-stopping UAV would have traveled at time t .

Definition 2. $\mathcal{S}(t)$ maps time into traveled distance.

$$\mathcal{S}(t) = \int_0^t v(\tau) d\tau \quad (7.19)$$

Describing the kinematics of each UAV requires the computation of an inverse mapping, yielding the time t by which this non-stopping UAV would have traveled a given distance s . Theorem 4 provides such an inverse mapping and gives an explicit definition. Its proof is in appendix B.1 (p. 161).

Theorem 4. *Let $v(t)$ follow assumption ii, there exists $t = \Phi[s]$ with the following property:*

$$\begin{cases} \mathcal{S}(\Phi[s]) = s \\ \frac{d}{ds}\Phi[s] = \frac{1}{v(\Phi[s])} & \text{if } v(\Phi[s]) \neq 0 \\ \lim_{s' \rightarrow s^-} \Phi[s'] < \Phi[s] & \text{if } v(\Phi[s]) = 0 \end{cases} \quad (7.20)$$

Moreover at time t , $\Phi[s]$ is known to all UAVs when $\Phi[s] \leq t$

With the following definition, we get a second inverse mapping of $\mathcal{S}(t)$ that uses only the information available to all UAVs at time t .

Definition 3.

$$\Phi_t[s] = \begin{cases} \Phi[s] & \text{if } \Phi[s] \leq t \\ +\infty & \text{if not} \end{cases} \quad (7.21)$$

To ease notations, we will be using $\Phi[s]$ instead of $\Phi_t[s]$.

Note that in the case of the no speed-change scenario, $\mathcal{S}(t)$ and $\Phi[s]$ are linear:

$$\mathcal{S}(t) = vt \text{ and } \Phi[s] = \frac{s}{v} \quad (7.22)$$

with $v = v(t)$ being the constant speed.

We define two time-varying lags, they are expressed in terms of distance traveled: $\Delta\mathcal{S}_c$, meant to avoid collision and $\Delta\mathcal{S}_r$, meant to maintain radio-connection.

$$\Delta\mathcal{S}_c = 2\rho_c\sqrt{2} \text{ and } \Delta\mathcal{S}_r = \rho_R - 2\rho_c \quad (7.23)$$

The travel is composed of five phases as illustrated in figure 7.2 for the no speed-change scenario. This travel is described for each UAV by durations Δt_k and distances $L_{k,n}$ that have or could have been traveled. $L_{k,n}$ are displayed later on.

$$\forall k \in \{-1 \dots K+2\}, \forall n \in \{1 \dots N\}, \Delta t_{k,n} = \Phi[L_{k+1,n}] - \Phi[L_{k,n}] \text{ and } L_{-1,n} = 0 \quad (7.24)$$

This travel contains five phases, note that the first UAV ($n = 1$) skips the two first phases and the last UAV ($n = N$) skips the two last phases.

- i. At $k = -1$, for $n > 1$, the n^{th} virtual UAV remains at $P_{0,n}$, waiting the maximum amount of time for which the connectivity with the previous UAV is preserved. The traveled distance is:

$$L_{0,n} = (n-1)(\Delta\mathcal{S}_r - \Delta\mathcal{S}_c)$$

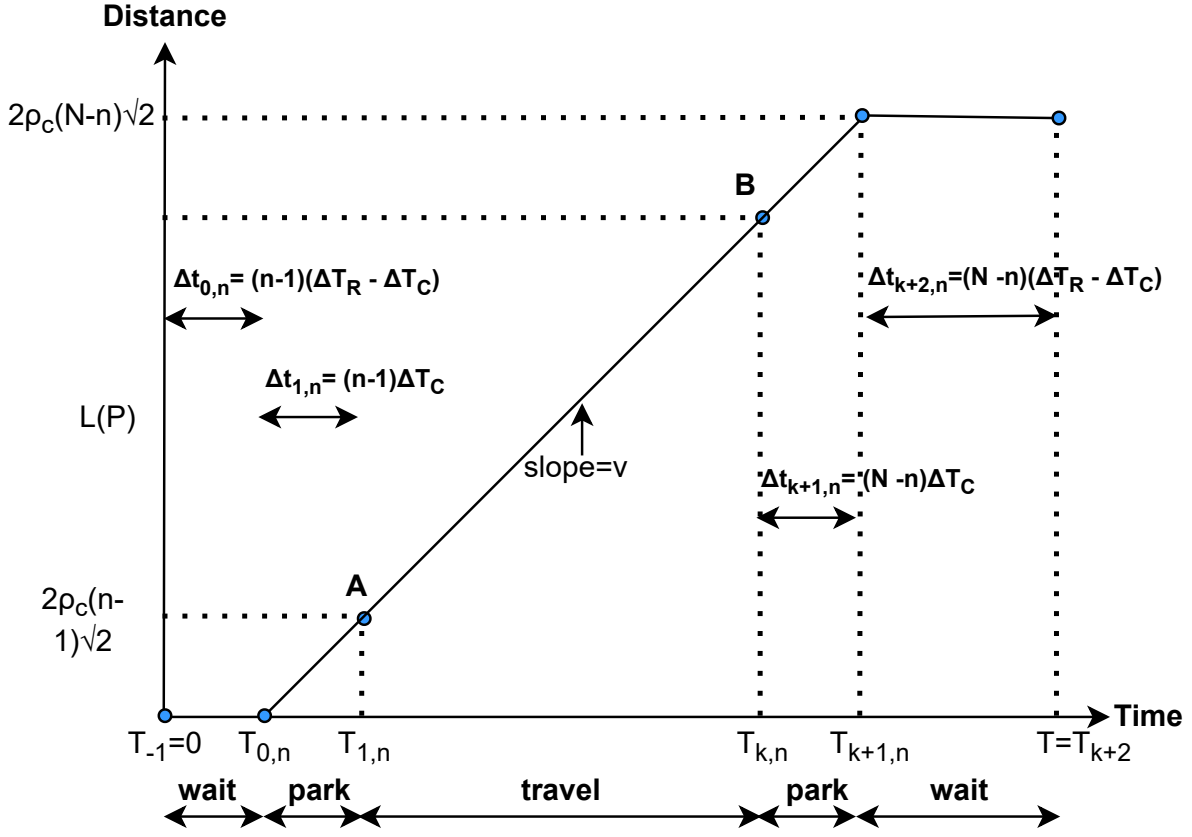


Figure 7.2: Line chart of the distance traversed by the n^{th} UAV for the no speed-change scenario.

- ii. At $k = 0$, for $n > 1$, the n^{th} virtual UAV travels from $P_{0,n}$ to $P_1 = A$, a distance that prevents mutual UAV collisions in the departure area, Δ_A . The traveled distance is:

$$L_{1,n} = L_{0,n} + (n-1)\Delta S_c = (n-1)\Delta S_r$$

- iii. At $k \in \{1 \dots K-1\}$, the n^{th} virtual UAV travels from $P_1 = A$ to $P_K = B$ along the path \mathbf{P} covering a distance of $\mathcal{L}(\mathbf{P})$

$$L_{K,n} = L_{1,n} + \mathcal{L}(\mathbf{P})$$

This phase is decomposed into $K-1$ sub-phases, each being a travel from P_k to P_{k+1}

$$L_{k+1,n} = L_{k,n} + P_k P_{k+1}$$

- iv. At $k = K$, for $n < N$, the n^{th} virtual UAV enters in the arrival area Δ_B and travels up to $P_{K+1,n}$ covering a distance preventing mutual UAV collisions.

$$L_{K+1,n} = L_K + (N-n)\Delta S_c$$

v. At $k = K + 2$, for $n < N$, the n^{th} virtual UAV waits at $\mathbf{P}_{K+1,n}$ for the other UAVs to reach their final destination points.

$$\begin{aligned} L_{K+2,n} &= L_{K+1,n} + (N - n)(\Delta\mathcal{S}_r - \Delta\mathcal{S}_c) \\ &= \mathcal{L}(\mathbf{P}) + (N - 1)\Delta\mathcal{S}_r \end{aligned}$$

The overall travel duration, T is

$$T = \sum_{k=1}^{K+2} \Delta t_{k,n} = \Phi[\mathcal{L}(\mathbf{P}) + (N - 1)\Delta\mathcal{S}_r] \quad (7.25)$$

In the case of the no speed-change scenario, note that the duration of phases 1,2,4 and 5, namely $\Delta t_{0,n}$, $\Delta t_{1,n}$, $\Delta t_{K+1,n}$, $\Delta t_{K+2,n}$ are specific to each UAV, not the third phase nor T .

We define unit vectors $\overrightarrow{e_k}$, one for each segment line contained in the path \mathbf{P} and two for $\Delta_{\mathbf{A}}$ and $\Delta_{\mathbf{B}}$.

$$\overrightarrow{e_k} = \frac{\overrightarrow{\mathbf{P}_k \mathbf{P}_{k+1}}}{|\mathbf{P}_k \mathbf{P}_{k+1}|} \text{ for } k \in \{1 \dots K - 1\}$$

We are now making use of a saturated ramp function defined as:

$$\mathcal{R}_a^b(x) = \begin{cases} 0 & \text{when } x \leq a \\ x - a & \text{when } x \in [a, b] \\ b - a & \text{when } x \geq b \end{cases} \quad (7.26)$$

The location of each virtual UAV can then be described in a shortened equation:

$$\forall t \in [0, T], \forall n \in \{1 \dots N\}, \mathbf{V}_n(t) = \mathbf{A} - (n - 1)\Delta\mathcal{S}_c \overrightarrow{e_0} + \sum_{k=0}^{K+1} \mathcal{R}_{L_{k,n}}^{L_{k+1,n}}(\mathcal{S}(t)) \overrightarrow{e_k} \quad (7.27)$$

The required number of UAVs is equal to:

$$N = \left\lceil \frac{\mathcal{L}(\mathbf{P})}{\rho_R - 2\rho_c} \right\rceil \quad (7.28)$$

Remark 1. Equations (7.17), (7.23), (7.24), (7.27), (7.28), tell us that the information to be broadcast to all UAVs is composed of N , $\mathbf{P}_1 \dots \mathbf{P}_K$, ρ_c , ρ_R , $\theta_{\mathbf{A}}$, $\theta_{\mathbf{B}}$ and $(v(\tau))_{\tau \leq t}$.

Theorem 5 validates equation (7.27). Its proof is in appendix B.2 (p. 163).

Theorem 5. Let equation (7.27) be the description of the moving virtual UAV $\mathbf{V}_n(t)$ using $v(t)$.

Then this description is consistent with the path \mathbf{P} .

$$\begin{cases} \mathbf{V}_n(t) \in \Delta_{\mathbf{A}} & \text{if } t \in [0, \Phi[L_{1,n}]] \\ \mathbf{V}_n(t) \in \overline{\mathbf{P}_k \mathbf{P}_{k+1}} & \text{if } t \in [\Phi[L_{k,n}], \Phi[L_{k+1,n}]] \text{ for } k \in \{1 \dots K - 1\} \\ \mathbf{V}_n(t) \in \Delta_{\mathbf{B}} & \text{if } t \in [\Phi[L_{K,n}], T] \end{cases} \quad (7.29)$$

This description is also consistent with objective iv.

$$\left\{ \begin{array}{ll} \left| \frac{d}{dt} \mathbf{V}_n(t) \right| = 0 & \text{if } t \in [0, \Phi[L_{0,n}]] \\ \left| \frac{d}{dt} \mathbf{V}_n(t) \right| \stackrel{a.e.}{=} v(t) & \text{if } t \in [\Phi[L_{0,n}], \Phi[L_{K+1,n}]] \\ \left| \frac{d}{dt} \mathbf{V}_n(t) \right| = 0 & \text{if } t \in [\Phi[L_{K+1,n}], T] \end{array} \right. \quad (7.30)$$

Theorem 6 ensures the avoidance of any obstacle collision using assumption 1, it thereby yields a first prerequisite on the path. It makes use of, d , the Euclidean distance between two sets of points. Proof of theorem 6 is in appendix B.3 (p. 163).

Theorem 6. *When considering a complete path whose line segments are sufficiently far from any obstacles, the movement of the virtual points as described in equation (7.27) cannot lead to obstacle collision.*

$$\left. \begin{array}{l} [\mathbf{P}_1 \dots \mathbf{P}_K] \in \mathbb{P}_c \text{ and } \forall i, \forall k \in \{1 \dots K-1\}, \\ d(\overline{\mathbf{P}_k \mathbf{P}_{k+1}}, \overline{\mathbf{C}_i \mathbf{D}_i}) \geq \rho_c, d(\Delta_A, \overline{\mathbf{C}_i \mathbf{D}_i}) \geq \rho_c, \\ d(\Delta_B, \overline{\mathbf{C}_i \mathbf{D}_i}) \geq \rho_c \end{array} \right\} \Rightarrow \forall t, n, i \quad \mathbf{R}_n(t) \notin \overline{\mathbf{C}_i \mathbf{D}_i} \quad (7.31)$$

Theorem 7 adds some other prerequisites and ensures with assumption 4, the avoidance of any mutual UAV collisions. Proof of theorem 7 is in appendix B.4 (p. 164).

Theorem 7. *Let $\mathbf{P} = [\mathbf{P}_1 \mathbf{P}_2 \dots \mathbf{P}_K]$ be a complete path, whose consecutive line segments have an absolute angle no greater than $\frac{\pi}{2}$ and for which any two non-consecutive line segments are always at a distance of at least $2\rho_c$ from each other. Then the movement of the virtual points as described in equation (7.27) cannot yield any mutual UAV collision.*

For $k \in \{2 \dots K-1\}$

$$|\theta_A - \angle(\overrightarrow{\mathbf{AB}}, \overrightarrow{\mathbf{P}_1 \mathbf{P}_2})| \leq \frac{\pi}{2}, |\angle(\overrightarrow{\mathbf{P}_{k-1} \mathbf{P}_k}, \overrightarrow{\mathbf{P}_k \mathbf{P}_{k+1}})| \leq \frac{\pi}{2}, |\theta_B - \angle(\overrightarrow{\mathbf{AB}}, \overrightarrow{\mathbf{P}_{K-1} \mathbf{P}_K})| \leq \frac{\pi}{2}$$

$$d(\Delta_A, \Delta_B) \geq 2\rho_c, d(\Delta_A, \overline{\mathbf{P}_k \mathbf{P}_{k+1}}) \geq 2\rho_c, d(\Delta_B, \overline{\mathbf{P}_{k-1} \mathbf{P}_k}) \geq 2\rho_c$$

(7.32)

For $k, k' \in \{1 \dots K-1\}$

$$|k - k'| > 1 \Rightarrow d(\overline{\mathbf{P}_k \mathbf{P}_{k+1}}, \overline{\mathbf{P}_{k'} \mathbf{P}_{k'+1}}) \geq 2\rho_c,$$

$$\Rightarrow \forall t, n_1 \neq n_2, \quad \mathbf{R}_{n_1}(t) \neq \mathbf{R}_{n_2}(t)$$

Theorem 8 and assumption 1 ensure that all UAVs remain connected with A or B. Proof of theorem 8 is in appendix B.5 (p. 168).

Theorem 8. *Let \mathbf{P} be a complete path and considering a number of UAVs no smaller than the value proposed in equation (7.28). Then the movement of the virtual points as described in equation (7.27) do not lose connectivity.*

$$\mathbf{P} \in \mathbb{P}_c \text{ and } N > \frac{\mathcal{L}(\mathbf{P})}{\rho_R - 2\rho_c} - 1 \quad \Rightarrow \quad \forall t, n, \left(R_n(t) \boxtimes A \text{ or } R_n(t) \boxtimes B \right) \quad (7.33)$$

The five prerequisites needed by theorem 6 and 7 and defined in equations (7.31) and (7.32) give rise to a new definition, that of a *valid* path denoted \mathbb{P}_v .

Definition 4. $[\mathbf{P}_1 \dots \mathbf{P}_K] \in \mathbb{P}_c$ is a valid path if and only if

i. it is far enough from any obstacle:

$$\forall k \in \{1 \dots K - 1\}, \forall i, d(\overline{\mathbf{P}_k \mathbf{P}_{k+1}}, \overline{\mathbf{C}_i \mathbf{D}_i}) \geq \rho_c \quad (7.34)$$

ii. it does not cross or come close to the departure and arrival areas:

$$\forall k \in \{2 \dots K - 1\}, d(\overline{\mathbf{P}_k \mathbf{P}_{k+1}}, \Delta_A) \geq 2\rho_c, \quad \text{and} \quad d(\overline{\mathbf{P}_{k-1} \mathbf{P}_k}, \Delta_B) \geq 2\rho_c \quad (7.35)$$

iii. its first and last bends are not too sharp:

$$\left| \theta_A - \angle(\overrightarrow{\mathbf{AB}}, \overrightarrow{\mathbf{P}_1 \mathbf{P}_2}) \right| \leq \frac{\pi}{2}, \quad \text{and} \quad \left| \theta_B - \angle(\overrightarrow{\mathbf{AB}}, \overrightarrow{\mathbf{P}_{K-1} \mathbf{P}_K}) \right| \leq \frac{\pi}{2} \quad (7.36)$$

iv. its others bends are also not too sharp:

$$\forall k \in \{2 \dots K - 1\}, \left| \angle(\overrightarrow{\mathbf{P}_{k-1} \mathbf{P}_k}, \overrightarrow{\mathbf{P}_k \mathbf{P}_{k+1}}) \right| \leq \frac{\pi}{2}, \quad (7.37)$$

v. it does not cross or come close to its own path:

$$\forall k, k' \in \{1 \dots K - 1\}, |k - k'| > 1 \quad \Rightarrow \quad d(\overline{\mathbf{P}_k \mathbf{P}_{k+1}}, \overline{\mathbf{P}_{k'} \mathbf{P}_{k'+1}}) \geq 2\rho_c \quad (7.38)$$

7.4.2 Path planning

We propose here an algorithm for providing iteratively a valid path. It builds a weighted directed graph and then repeatedly increases some edge weights to help the Dijkstra algorithm to find a valid path. This need to help the Dijkstra algorithm stems from the non-local nature of conditions 4 and 5 of definition 4. When an edge is traversed by a valid path, other edges may, as a consequence, be excluded from being traversed by this path. The Dijkstra algorithm

is repeatedly applied, each time increasing the weights of the problematic edges, until a valid path is found. Nodes are generated in section 7.4.2.1, edges in section 7.4.2.2, and weights in section 7.4.2.3. Finally section 7.4.2.4 discloses the proposed Dijkstra-based heuristic extracting a valid path from this directed graph. Note that ρ_R is not being used for path planning.

7.4.2.1 Generating the graph nodes

The nodes of the graph being built are points denoted $M \in \mathbb{M}$, they are generated using the provided knowledge of the environment. Below is the list of these \mathbb{M} -points.

- A, B are two required points.
- Four points $C_i^+, C_i^-, D_i^-, D_i^+$ are defined for each line segments $\overline{C_i D_i}$, they are displayed so as to allow going around the obstacle. These four points are located at a distance of $\sqrt{2}\rho_c$ of each segment end, they have an angle of $\pm\frac{\pi}{4}$ with respect to $\overline{C_i D_i}$.

$$\left\{ \begin{array}{l} \angle(\overrightarrow{D_i C_i}, \overrightarrow{C_i C_i^-}) = \frac{\pi}{4} \quad d(C_i C_i^-) = \rho_c \sqrt{2} \\ \angle(\overrightarrow{D_i C_i}, \overrightarrow{C_i C_i^+}) = -\frac{\pi}{4} \quad d(C_i C_i^+) = \rho_c \sqrt{2} \\ \angle(\overrightarrow{C_i D_i}, \overrightarrow{C_i D_i^-}) = -\frac{\pi}{4} \quad d(C_i D_i^-) = \rho_c \sqrt{2} \\ \angle(\overrightarrow{C_i D_i}, \overrightarrow{C_i D_i^+}) = \frac{\pi}{4} \quad d(C_i D_i^+) = \rho_c \sqrt{2} \end{array} \right. \quad (7.39)$$

- Four extra points, denoted A^-, A^+, B^-, B^+ , are also considered to withhold the use of sharp turns at departure and arrival. They are located at a distance of $\rho_c \sqrt{2}$ of A and B and have an angle of $\theta_A \pm \frac{\pi}{2}$ and $\theta_B \pm \frac{\pi}{2}$ with \overrightarrow{AB} .

$$\left\{ \begin{array}{l} \angle(\overrightarrow{AB}, \overrightarrow{AA^+}) = \theta_A - \frac{\pi}{2} \quad d(AA^+) = \rho_c \sqrt{2} \\ \angle(\overrightarrow{AB}, \overrightarrow{AA^-}) = \theta_A + \frac{\pi}{2} \quad d(AA^-) = \rho_c \sqrt{2} \\ \angle(\overrightarrow{AB}, \overrightarrow{BB^+}) = \theta_B + \frac{\pi}{2} \quad d(BB^+) = \rho_c \sqrt{2} \\ \angle(\overrightarrow{AB}, \overrightarrow{BB^-}) = \theta_B - \frac{\pi}{2} \quad d(BB^-) = \rho_c \sqrt{2} \end{array} \right. \quad (7.40)$$

7.4.2.2 Generating edges

\mathbb{E} , the set of edges, is derived from equations (7.34), (7.35) and (7.36) of definition 4.

Definition 5. *The set of edges, $\mathbb{E} \subset \mathbb{M} \times \mathbb{M}$, contains all pairs of points sufficiently distant from any obstacles, and which do not yield too sharp bend at departure and arrival.*

$$(M, M') \in \mathbb{E} \Leftrightarrow \begin{cases} \forall i \, d(\overline{MM'}, \overline{C_i D_i}) \geq \rho_c, \\ d(\overline{MM'}, \Delta_A) \geq 2\rho_c, \\ d(\overline{MM'}, \Delta_B) \geq 2\rho_c \\ \text{if } (M = A) \text{ and } \left| \angle(\overrightarrow{AB}, \overrightarrow{AM'}) - \theta_A \right| \leq \frac{\pi}{2} \\ \text{if } (M' = B) \text{ and } \left| \angle(\overrightarrow{AB}, \overrightarrow{MB}) - \theta_B \right| \leq \frac{\pi}{2} \end{cases} \quad (7.41)$$

7.4.2.3 Generating weights

Let us denote l the iteration number and \mathcal{W}_l the edge-weight map at the l^{th} iteration. At the first iteration, each edge weight is assigned to its length.

$$\mathcal{W}_1((M, M')) = d(MM') \quad (7.42)$$

The average weight is also computed.

$$W = \frac{1}{|\mathbb{E}|} \sum_{e \in \mathbb{E}} \mathcal{W}_1(e) \quad (7.43)$$

where $|\mathbb{E}|$ is the number of edges in \mathbb{E} .

By applying the Dijkstra algorithm on the graph $(\mathbb{M}, \mathbb{E}, \mathcal{W}_l)$, we get $\mathbf{P} \in \mathbb{P}_c$ complying with equations (7.34), (7.35) and (7.36) of definition 4. Denoting an edge by e , we define $\mathcal{I}(\mathbf{P}, e)$ an indicator of how e is problematic, in that it accounts for how many times e is not compliant with (7.37) and (7.38).

To ease the technical description of how weights are evolving, we define some path-related attributes. For a given path \mathbf{P} and an edge e , $\mathcal{K}(\mathbf{P}, e)$ is an index, $\overline{\mathcal{K}(\mathbf{P}, e)}$ a line segment and $\overrightarrow{\mathcal{K}(\mathbf{P}, e)}$ a vector. The \mathcal{K} -indexes are defined as:

$$\begin{aligned} &\text{Given } \mathbf{P} = [P_1 \dots P_K] \text{ and } e = (M, M') \\ &\text{if } \exists k \leq K - 1, P_k = M \text{ and } P_{k+1} = M' \\ &\text{then } \mathcal{K}(\mathbf{P}, e) = k \text{ and } \overline{\mathcal{K}(\mathbf{P}, e)} = \overline{P_k P_{k+1}} \text{ and } \overrightarrow{\mathcal{K}(\mathbf{P}, e)} = \overrightarrow{P_k P_{k+1}} \\ &\text{else } \mathcal{K}(\mathbf{P}, e) = 0 \end{aligned} \quad (7.44)$$

Note that definition 1 ensures the unicity of k when it exists. The following definition makes use

of $\mathbf{1}$ mapping propositions into $\{0, 1\}$:

$$\mathbf{1}(\mathcal{P}) = \begin{cases} 1 & \text{if } \mathcal{P} \text{ is true} \\ 0 & \text{if } \mathcal{P} \text{ is false} \end{cases}$$

Definition 6. $\mathcal{I}(\mathbf{P}, \mathbf{e})$ denotes an integer valued function.

$$\begin{aligned} \mathcal{I}: \quad \mathbb{P} \times \mathbb{E} &\quad \rightarrow \quad \mathbb{N} \\ ([\mathbf{P}_1 \dots \mathbf{P}_K], \mathbf{e}) &\quad \mapsto \quad \mathcal{I}(\mathbf{P}, \mathbf{e}) \end{aligned} \tag{7.45}$$

with $\mathcal{I}(\mathbf{P}, \mathbf{e}) = 0$ when $\mathcal{K}(\mathbf{P}, \mathbf{e}) = 0$

and otherwise

$$\mathcal{I}(\mathbf{P}, \mathbf{e}) = \sum_{k \neq \mathcal{K}(\mathbf{P}, \mathbf{e})} \mathbf{1} \left(d(\overline{\mathcal{K}(\mathbf{P}, \mathbf{e})}, \overline{\mathbf{P}_k \mathbf{P}_{k+1}}) < 2\rho_c \right) + \sum_{|k - \mathcal{K}(\mathbf{P}, \mathbf{e})| \leq 1} \mathbf{1} \left(|\angle(\overrightarrow{\mathcal{K}(\mathbf{P}, \mathbf{e})}, \overrightarrow{\mathbf{P}_k \mathbf{P}_{k+1}})| > \frac{\pi}{2} \right) \tag{7.46}$$

Problematic edges have their weights increased as this induces the Dijkstra algorithm to yield paths avoiding such edges. At the l^{th} iteration, the modified edge-weight map is assigned to:

$$\mathcal{W}_l(\mathbf{e}) := \mathcal{W}_{l-1}(\mathbf{e}) + \mathcal{I}(\mathbf{P}_{l-1}, \mathbf{e})W \tag{7.47}$$

where \mathbf{P}_{l-1} is the path found at the previous iteration.

7.4.2.4 Generating a valid path

The proposed algorithm generating a valid path (Algorithm 8) is in three parts. The first part builds a graph with its nodes, edges and weights. The second part yields a complete path. The third part tests if this path is valid. And if not, weights of problematic edges are increased, making it less likely that these problematic edges are again traversed. The second and third parts are repeated until a valid path is found.

Algorithm 8 Generating a valid path**INPUT:** A : initial point, B terminal point, $(C_i D_i)$ set of obstacles, ρ_c collision safe-distance**OUTPUT:** \mathbf{P} trajectory**INITIALIZATION:** \mathbb{M} is set as in section 7.4.2.1 \mathbb{E} is set by definition 5 $l := 1$ \mathcal{W}_1 is set by equation (7.42) W is set by equation (7.43)**loop** $\forall M \in \mathbb{M}, \mathcal{V}(M) := +\infty$ **for** $(M, M') \in \mathbb{E}$ $\mathcal{V}(M') := \min[\mathcal{V}(M'), \mathcal{V}(M) + \mathcal{W}_l((M, M'))]$ **end for** $\mathbf{P} := [A]$ **while** $\mathcal{F}(\mathbf{P}) \neq B$ **do** $\hat{M}' := \arg \min_{(\mathcal{F}(\mathbf{P}), M') \in \mathbb{E}} \mathcal{V}(M')$ $\mathbf{P} := [\mathbf{P} \hat{M}']$ **end while****if** $\mathbf{P} \in \mathbb{P}_v$ **then****exit loop** and return \mathbf{P} **end if** $l := l + 1$ **for** $e \in \mathbb{E}$ **do** $\mathcal{W}_l(e) := \mathcal{W}_{l-1}(e) + \mathcal{I}(\mathbf{P}, e)W$ **end for****end loop**

To simplify the description of the Dijkstra algorithm, we denote by $\mathcal{F}(\mathbf{P})$ the final destination of a path.

$$\mathcal{F}([P_1 \dots P_K]) = P_K \quad (7.48)$$

Concerning the **loop** on l , it is of great importance for the Dijkstra algorithm to examine edges in a specific order, that of the breadth-first search algorithm.

Remark 2. Based on this algorithm and on remark 1, the transmitted information includes $A, B, C_i, D_i, \rho_c, \rho_R$.

7.5 Simulation testbed and experimental procedures

The proposed path planning scheme is implemented and evaluated in MATLAB and Network Simulator-2 (NS-2), respectively. Applying the proposed technique, MATLAB provides a collision-free flyable trajectory according to the given environmental conditions. MATLAB feeds trajectory coordinates to NS-2 to evaluate the tracking of the designed path and end-to-end connectivity performance. We consider off-line path planning, but any UAV can lead the entire fleet dynamically at any desired speed or request hovering. Any UAV within the fleet or node on the ground can be a source of data. We consider rectangular-shaped obstacles of random sizes. The path designing process takes a safe distance margin of 5^2 meters. Each UAV is equipped with an Omni-directional antenna with communication architecture based on 802.11p. Each UAV can transmit within a range of 220m. MATLAB provides coordinate information to be sampled by NS-2 with a sampling frequency of 10 and a virtual UAV speed of 10 m/s. Fig. 7.3a and Fig. 7.3b present all the possible routes between starting and destination points and the final selected one by the proposed scheme, respectively. Table 7.2 enlists the complete simulation parameters.

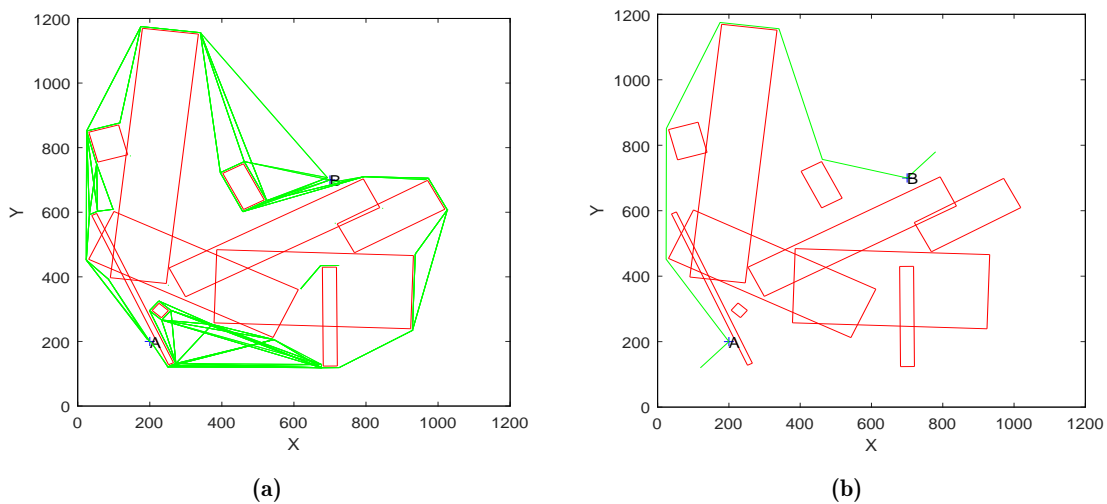


Figure 7.3: (a) All possible routes between start and destination point (b) The selected path by the proposed scheme

²This value is taken according to the GPS accuracy as provided by <https://www.gps.gov/systems/gps/performance/accuracy/>

Table 7.2: *Simulation parameters*

Parameter	Value
ρ (safe margin distance)	5 m
Number of obstacles	10
Transmission distance	220 m
MAC protocol	802.11p
Hello interval	1 s
Data rate	0.8 kbps
Number of UAVs	9
Starting coordinate	(200,200)
Destination coordinate	(700,700)
Max. UAV speed	10 m/s
Ground users	6
Coordinates sampling interval	0.1 s
θ_A and θ_B	0

7.5.1 Performance metrics

We use the following performance parameters to evaluate backhaul connectivity, fleet connectivity, collision avoidance, and scalability features for the proposed approach.

- To validate the real-time aspect and backhaul connectivity, we plot the number of packets received at BS with respect to time.
- To demonstrate the fleet connectivity and integrity in response to change in speed requests by the leaders, we plot followers' response time for all such speed change requests.
- To highlight the collision avoidance and evaluate trajectory tracking, we plot, among all UAVs, the minimum UAV-UAV and UAV-obstacles distances for the entire flight time.
- To show the scalability feature in terms of routing, we plot the number of packets received at the BS and packets dropped for the ground users.

7.6 Results and discussion

This section presents experimental results to validate the proposition of the proposed scheme. The first subsection enlists results in which data is generated only by the UAVs within the fleet and for fixed and varying fleet speed. The second subsection provides simulation results for scenarios wherein only ground users transmit data under the constant and changing speed of the UAV fleet.

7.6.1 UAVs as a source of data

7.6.1.1 UAV fleet moving with varying speed

In this scenario, the UAV fleet tracks a pre-given trajectory, and each UAV follows the leader for speed variations. Fig. 7.4a presents, among all UAVs within the fleet, the minimum distance between any two UAVs during the whole simulation time. Flat line in this figure shows that at least two UAVs are at ground either at starting or at landing positions. Similarly, Fig. 7.4b presents, for all the UAVs and obstacles, the minimum distance between any UAV and obstacle during the entire simulation time. It can be inferred from these two figures that none of the UAV collides with other UAV or obstacles and this shows the effectiveness of collision-free trajectory and successful following of leader instructions for the speed variations.

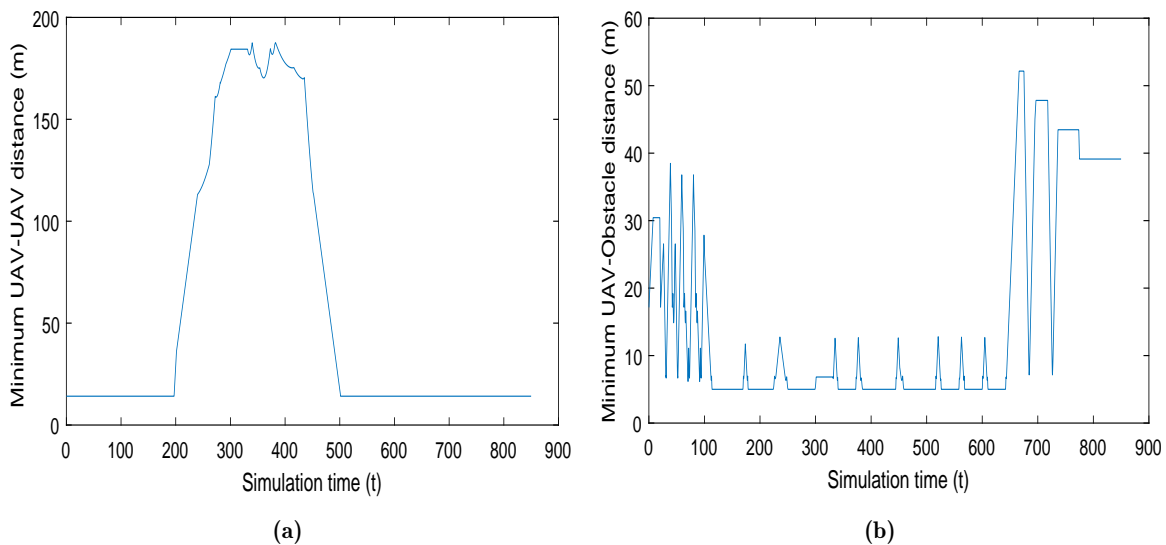


Figure 7.4: Minimum distance with simulation time (a) UAV-UAV (b) UAV-Obstacles

Fig. 7.5a and Fig. 7.5b illustrate the reaction time for the follower UAVs to the speed change requests by the leader with 4 and 8 transmitting UAVs per fleet, respectively. UAV-1 plays the

role of leader and instructs follower UAVs to move according to the given speed instructions. In Fig. 7.5a, UAV-2 is the neighbor of the leader and shows a reaction time below 0.1 s that can lead to a maximum of 1 m error with the speed of 10 m/s. UAV-9 (i.e., farthest from the leader) gets the speed change message from the leader with a maximum delay of 0.58 s. As we consider a safe distance of 5 m in our path planning process, so with a speed of 10 m/s, we can afford a maximum of 0.5 s reaction time between any two consecutive UAVs to avoid a possible collision. It is observed in Fig. 7.5a and Fig. 7.5b that the reaction time between any two neighboring UAVs stays far below the threshold limit of 0.5 s leading us to conclude that all follower UAVs track the desired trajectory within their limits. Fig. 7.5a and Fig. 7.5b also show that an increase in the number of transmitting UAVs does not affect the reaction time significantly.

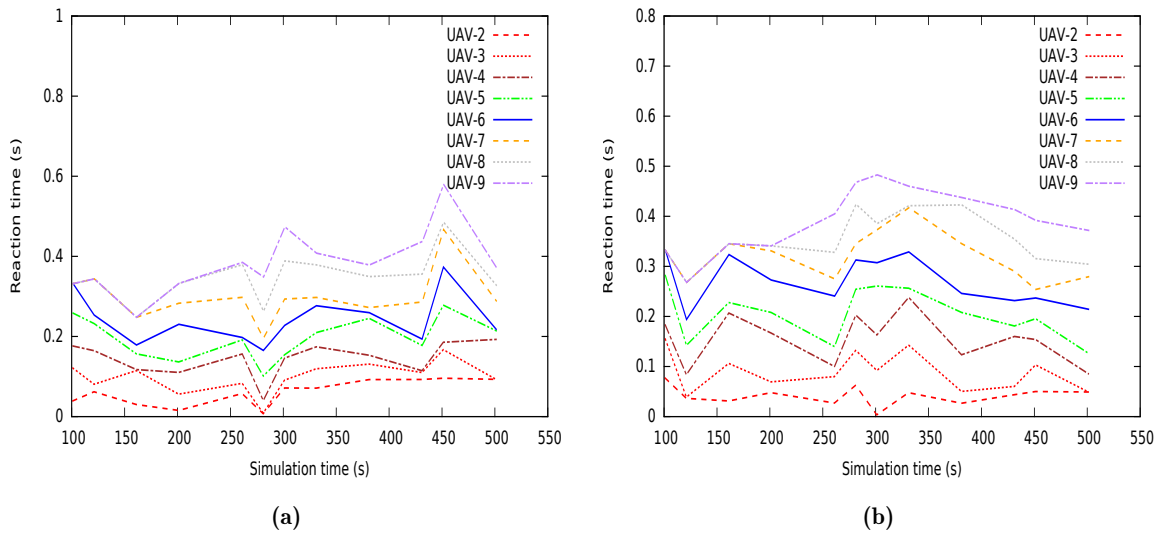


Figure 7.5: UAV's reaction time to speed change requests (a) 4 transmitting UAVs (b) 8 transmitting UAVs

To evaluate backhaul connectivity with BS, Fig. 7.6 plots the number of packets received at BS with time. In this figure, the non-zero slope line indicates data reception at the BS, while any zero slope line implies no data reception at BS. Consequently, Fig. 7.6 demonstrates continuous entire fleet connectivity with BS except around 365 s simulation time due to connection handover from one BS to another. Fig. 7.7 presents the delay incurred by the data packets traveling from the UAVs to BS. As it should be obvious, Fig. 7.7 shows that increasing the number of transmitting UAVs increases the delay incurred by the data packets. During the start and near the end of the mission, data packets encounter a lower hop count to reach BS due to fewer aerial UAVs resulting in a lower delay. During the middle part of mission time, the number of hops to

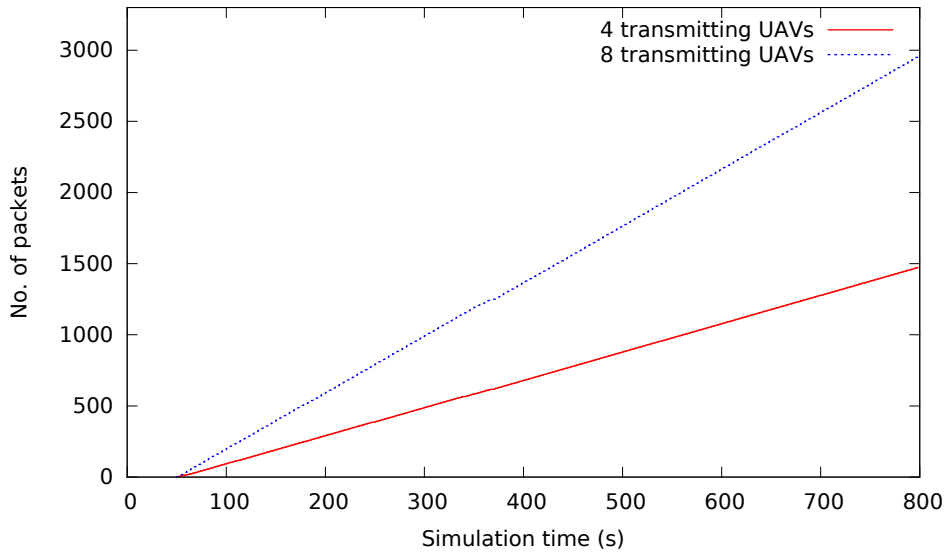


Figure 7.6: *Number of packets received at BS (end-to-end connectivity check)*

BS increases due to the increased aerial UAVs leading to higher delay values.

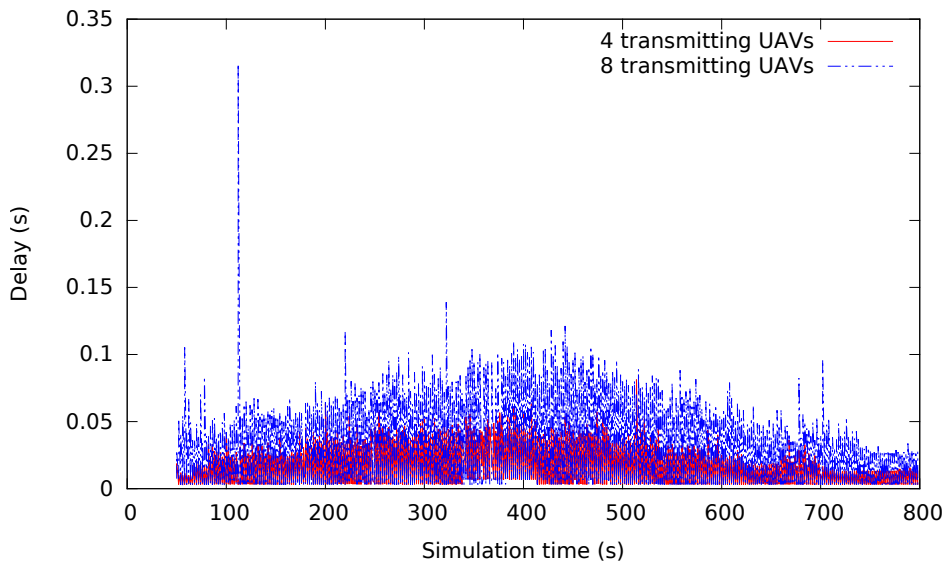


Figure 7.7: *Data packets delay to reach BS*

7.6.1.2 UAV fleet moving with fixed speed

This subsection demonstrates the efficacy of the proposed scheme under a fixed UAV tracking speed. Fig. 7.8 shows UAVs' minimum distance variation from any other UAV for a fixed fleet speed of 2, 5, and 10 m/s. Similarly for the same experiments, Fig. 7.9 represents, among all UAVs and obstacle, the minimum UAV-obstacle distance during entire simulation time with fixed fleet speed of 2, 5, and 10 m/s. All UAVs in the fleet maintain a safe distance from each other

and that with obstacles which is observable in Fig. 7.8 and Fig. 7.9.

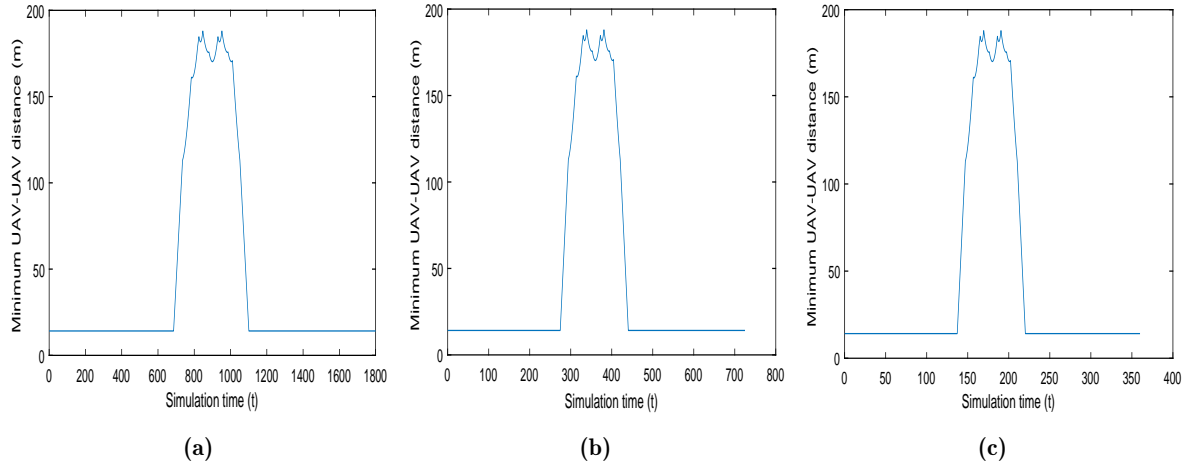


Figure 7.8: Minimum UAV-UAV distance with simulation time (a) Fleet speed 2 m/s (b) Fleet speed 5 m/s (c) Fleet speed 10 m/s

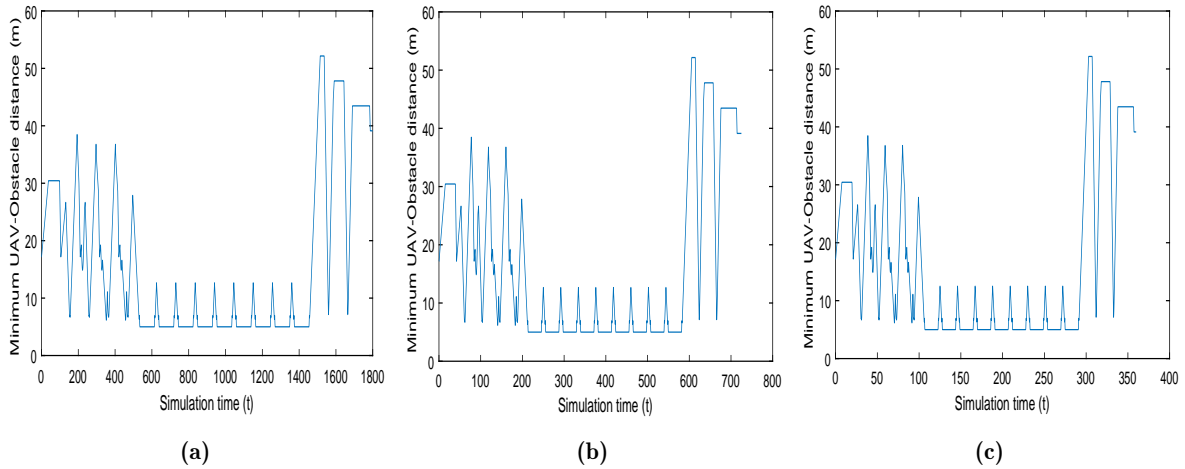


Figure 7.9: Minimum UAV-Obstacle distance with simulation time (a) Fleet speed 2 m/s (b) Fleet speed 5 m/s (c) Fleet speed 10 m/s

Fig. 7.10 plots the number of packets received at BS with time to validate end-to-end connectivity for a fleet with a speed of 2, 5, and 10 m/s. The UAV fleet with 2 m/s speed takes 1800 seconds to complete its mission and remains connected all the time with BS. Even though the fleet retains end-to-end connectivity with higher speed values but lags in comparison to lower speeds due to frequent fleet link changes with BSs.

Fig. 7.11 demonstrates the delay faced by data packets for UAV fleet speed of 2, 5, and 10 m/s.

This figure concludes that the average delay remains the same for all these speed variations.

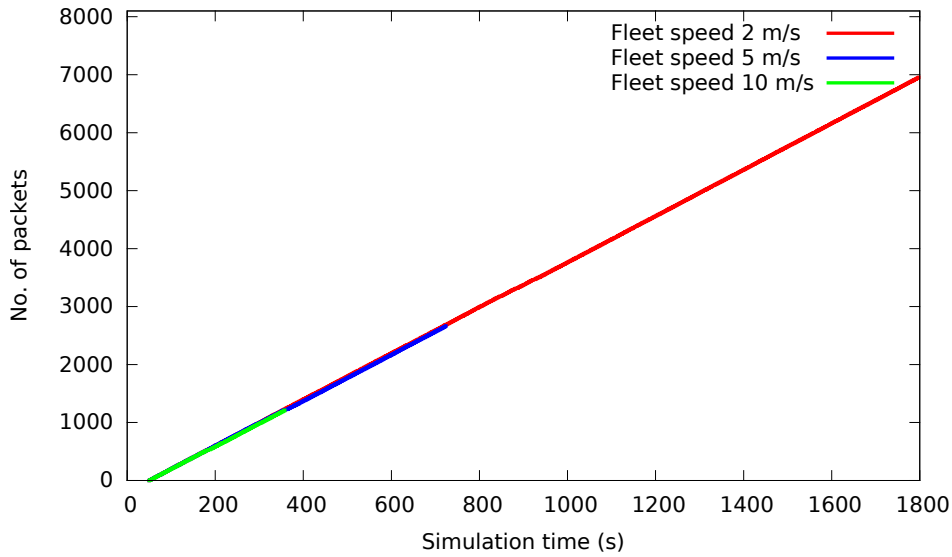


Figure 7.10: *Number of packets received at BS (end-to-end connectivity check)*

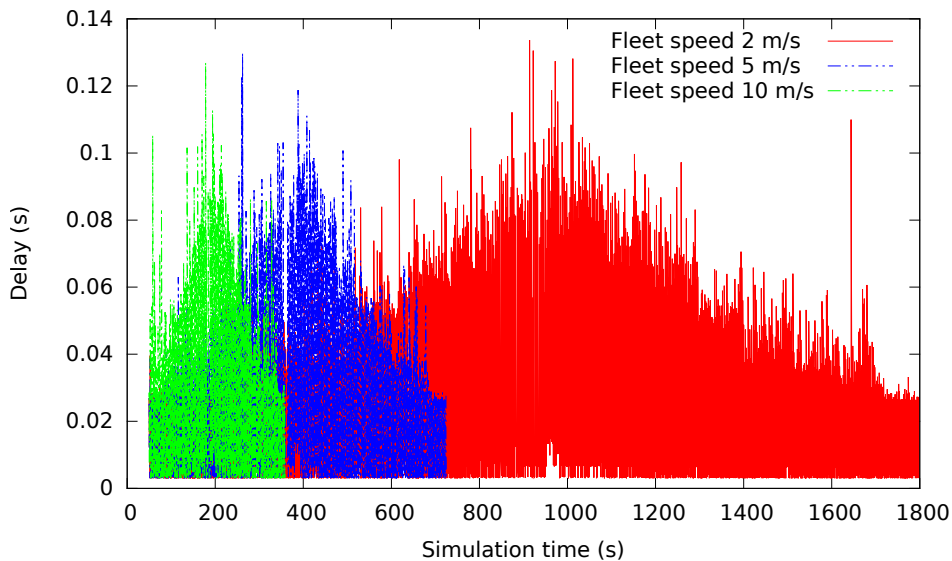


Figure 7.11: *Data packets delay to reach BS*

7.6.2 Ground users as a source of data

In this section, we evaluate the fleet response to connectivity requests arising from the uniformly deployed ground users. We also assess the effect of a leader's position within a fleet on the system performance. The first section presents results for varying fleet speed, whereas the second part assesses system performance for fixed speed values.

7.6.2.1 UAV fleet moving with varying speed

Fig. 7.12 plots reaction time for follower UAVs in response to speed change requests made by the leaders with ground users as a source of data. Fig. 7.12 shows that the reaction time difference between neighboring UAVs remains modest and results in the elimination of any possible collisions among them. This figure also concludes that the average reaction time for UAVs drops down when a leader lies in the middle of a fleet.

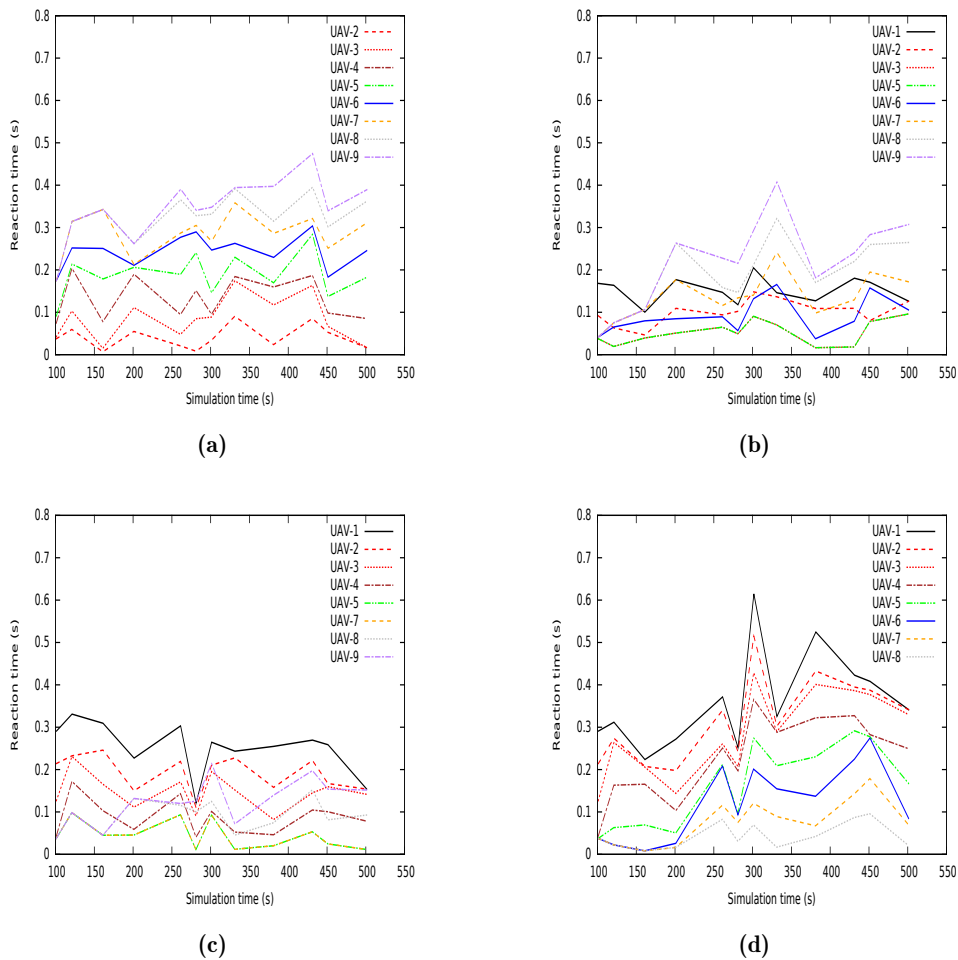


Figure 7.12: UAV's reaction time to speed change requests (a) UAV-1 as leader (b) UAV-4 as leader (c) UAV-6 as leader (d) UAV-9 as leader

Fig. 7.13 presents the number of packets received at the BS and dropped packets at ground users. During the start and near the end of the mission, most of the UAVs reside at the departure and arrival points, respectively, and less coverage offered to ground users results in higher drop rates. Fig. 7.13 shows that the number of packets drops to zero as the entire fleet becomes airborne while ensuring connectivity to the ground users. It is also observable in Fig. 7.13 that

the leader position has a very marginal effect on the packets received and dropped. Fig. 7.14 plots data packet delays for the ground users to reach BS and for different leaders' locations within the fleet. Fig. 7.14 shows that leaders' position plays only a slight role in determining the delay incurred by data packets.

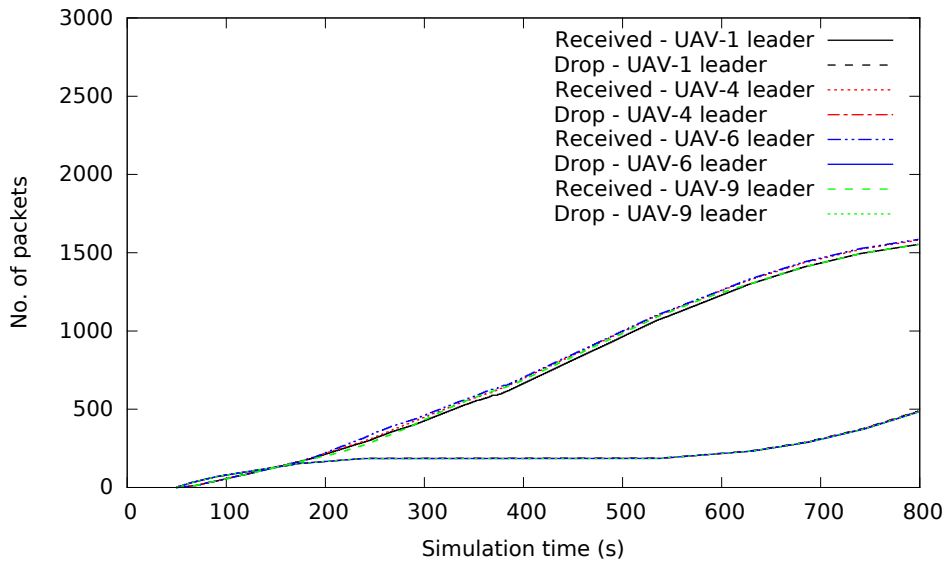


Figure 7.13: Number of packets received at BS (end-to-end connectivity check for ground users)

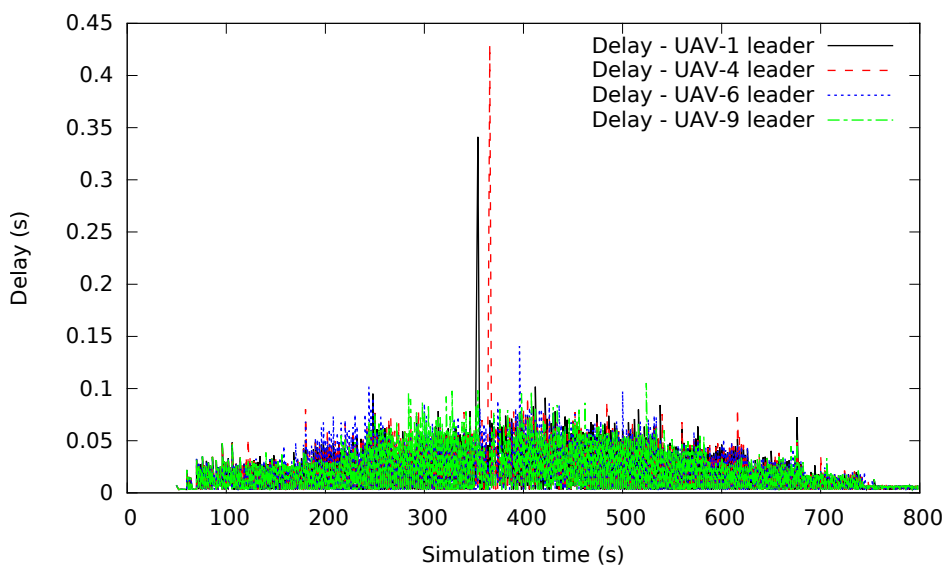


Figure 7.14: Data packets delay to reach BS

7.6.2.2 UAV fleet moving with fixed speed

Fig. 7.15 plots the number of packets received at BS and dropped at ground users with a UAV fleet having speed values of 2, 5, and 10 m/s. UAV fleet completes its mission in the 1800 s

with a speed of 2 m/s. BS receives 3211 packets compared to 5250 transmitted packets with a success ratio of 61.16. Furthermore, 1666 packets get dropped due to no link connectivity with the fleet. In Fig. 7.15, for a fleet speed of 5 m/s, BS, with a success ratio of 62.1, receives 1276 out of 2028 send packets having a drop count of 602. Finally, BS successfully receives packets with a 62.7 percentage and ground users dropping 243 out of 930 packets. Fig. 7.16 plots the delay incurred by the data packets for ground users to reach BS under 2, 5, and 10 m/s fleet speed. It can be observed from the figure that the delay faced by the data packets under these conditions is not affected by the variation in the fleet speed. Fig. 7.15 and Fig. 7.16 show that a higher fleet speed allows coverage provision to deployed ground users quickly compared to lower speed values wherein the fleet takes more time to reach all deployed ground nodes.

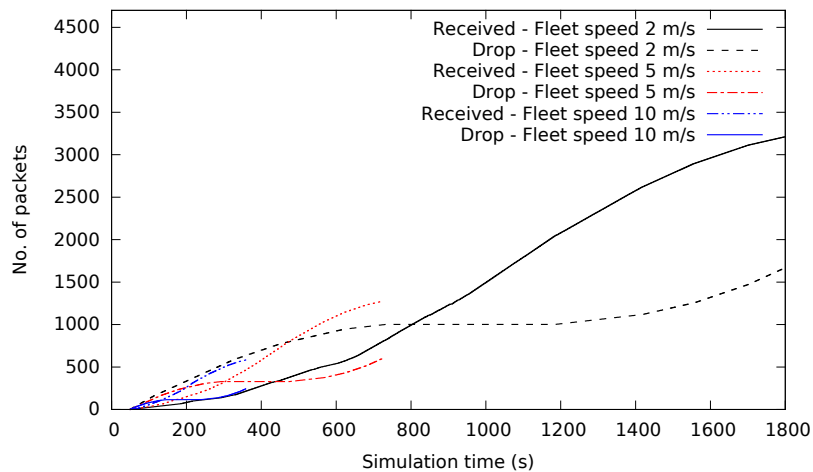


Figure 7.15: Number of packets received at BS (end-to-end connectivity check for ground users)

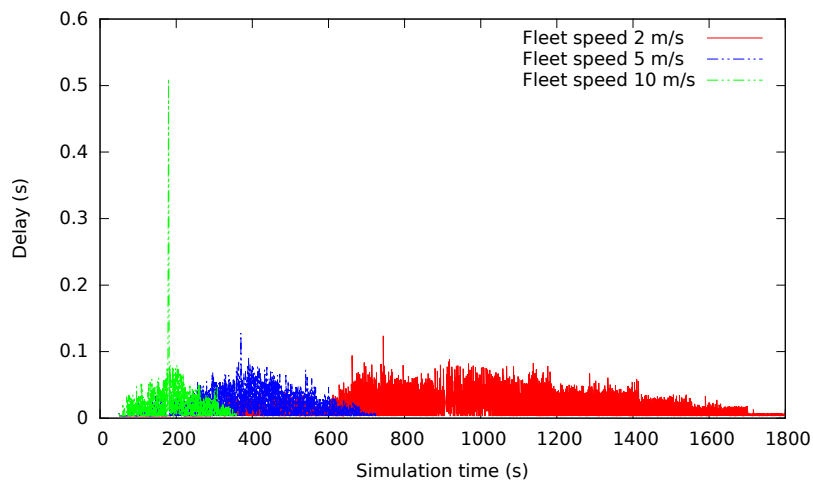


Figure 7.16: Data packets delay to reach BS from ground users

7.7 Conclusion

In this chapter, we have proposed collision-free UAV fleet path planning constrained to backhaul connectivity. The proposed approach considers offline graph-based path planning with obstacles modeled as line segments. The path planning model includes a safe margin distance parameter to tolerate any uncertainty arising due to environmental disturbances. All the UAVs receive trajectory coordinates before departure, and any UAV can lead the entire fleet dynamically owing to the continuous fleet and backhaul connectivity. Line formation for the fleet acts as a backbone of the network and adds a scalability feature to our scheme in the sense that any additional UAV or ground user can become a part of the fleet. The proposed approach is mathematically proved, implemented in Matlab, and evaluated in the network simulator. The simulation results demonstrate that all UAVs track their designated paths within permissible limits even with the variation in the leader's position, speed variations, and the fleet provides all-time link connectivity to all UAVs within the fleet or outside users.

Conclusion and future work

Conclusion and future work

Chapter content

8.1	Achieved work	154
8.2	Thesis contributions	155
8.3	Future work	156

8.1 Achieved work

Even though recent advancements in UAVs' technology have empowered the realization of economical UAV-based highway surveillance solutions, yet inherent limitations of UAVs prevent exploiting them at their full potential. Multiple UAVs in collaboration with each other work beyond the capabilities of a single UAV to compensate for the limitations, i.e., limited energy resources and lower transmission range, associated with a single UAV. The dynamic nature of a UAV network and the requirement for real-time aspects in many surveillance applications make existing ad-hoc routing solutions nonviable. Due to typical UAV dynamics, a fleet of UAVs frequently finds UAV-UAV link failures and loss of fleet-BS associativity. These link failures are followed by a frequent exchange of topology control messages that become the reason for an increased routing overhead. Sometimes link failure results in a network breakdown due to the permanent removal of a bridging UAV from the fleet. In such scenarios, a link recovery mechanism fails to recover the link and requires physical replacement by another UAV to recover the disconnected network.

Above all, UAV deployment for any application should ensure UAV safety and people on the ground, specifically in an urban environment with high navigational uncertainties. In this regard, the path planning process should provide collision-free path planning while offering less computing load considering the limited computational resources of UAVs. Moreover, all-time backhaul connectivity demands routing requirements to be considered simultaneously along with the path planning process.

Below is the summary of the complete thesis work, followed by the section listing all main contributions of the thesis.

Firstly, a routing protocol, having less overhead, is presented for a fleet of UAVs. This work intends to increase fleet-BS associativity by deploying multiple base stations while reducing routing overhead. Each BS periodically broadcasts DNA messages in the entire fleet, and at each UAV, data packets take the reverse of the path taken by the DNA. The absence of Hello messages and no dedicated route establishment procedure make this routing efficient in terms of throughput, delay, and energy consumed. The second work proposes UAVs and VANETs collaboration that deals with the sparse and highly dynamic topology of UAV networks. UAVs take help from nearby vehicles during the non-availability of neighboring UAVs required for link recovery. Additionally, this scheme proposes a procedure to compensate for the loss of BS association messages and resolves the issue at the network layer. The third contribution provides an anticipatory link failure recovery approach to ensure consistent fleet-BS connectivity. This

work anticipates the departure of a UAV with near-to-end energy resources or its sudden exit and replaces it with a suitable candidate beforehand. The core point of this scheme is to select a robust replacer candidate to reduce the occurrence of further recovery requests.

The fourth contribution is the collaboration of UAV and WSN to increase the effective coverage area and adapt UAV position according to dynamic traffic patterns to catch maximum speed intruders. WSN is at the heart of this architecture and provides routing services besides guiding the UAV to target the best hotspot location. The fifth contribution considers the limited energy resources of WSN and proposes a cross-layer routing solution to reduce collisions along a complete path. Energy conserved by avoiding re-transmissions and time saved by not going into the backoff algorithm becomes the reason for the energy-efficient routing solution.

The sixth contribution is the obstacle avoidance path planning for a UAV considering navigational uncertainties in an urban environment. This technique models obstacles as rectangles and includes only their corners, instead of the entire environment, in the path planning process, thus reducing computational load. The uncertainty model as a threshold distance provides an extra layer of safety to UAV if the underlying control system overshoots its desired values. Finally, the last contribution is a connectivity-aware graph-based offline path planning approach for a fleet of UAVs. This technique simultaneously considers collision-free path planning and routing constraints to avoid UAV-UAV, UAV-obstacle collisions and ensure consistent fleet connectivity with the BS. The line formation considered for the fleet allows scalability feature by allowing any UAV to become a part of the connected network. Any UAV within the fleet can play a leadership role and lead the entire fleet at any desired value.

8.2 Thesis contributions

The main contributions of this thesis are listed below:

- To increase the probability of BS-fleet associativity, designing of less overhead bearing multi-hop routing for a fleet of UAVs with multiple BSs deployment.
- To deal with the sparsity and the dynamic nature of UAV network, UAVs-VANETs collaboration to have robust routing and link failure recovery.
- To deal with network breakdown scenarios, anticipatory link failure recovery approach to ensure consistent BS-fleet link.
- To effectively increase the coverage area and catch maximum over-speed violations, UAV-WSN collaboration to dynamically guide UAV according to varying traffic patterns on a

highway.

- Cross-layer solution of packet scheduling to increase the lifetime of a WSN.
- Collision-free path planning for a UAV with a low computational load.
- Connectivity-aware path planning for a fleet of UAVs to ensure continuous backhaul connectivity.

8.3 Future work

To explore further opportunities, the proposed research work in this thesis can be extended in many directions. The probable future directions are listed below.

- In all contributions, we consider only static ground BSs with the same underlying MAC layer technology. With the recent deployment of 5G, we can extend these works by considering flying BS equipped with 5G technology. This development will offer higher fleet-BS connectivity and data rates.
- In Chapter 4, the UAV-WSN collaboration can be extended to enhance the effective surveillance area by including multiple UAVs. UAVs, even out of the communication range of each other, can coordinate via WSN to optimize surveillance tasks.
- In Chapter 5, we consider a single source node. This work can be extended to make it applicable for multiple source nodes by introducing a central controlling node. The source node may ask the central node for path use, or the central node can issue a path access schedule for all source nodes.
- Implementation of the proposed off-line path planning techniques in chapters 6 and 7 on real platforms to demonstrate their effectiveness in a real environment. Moreover, modification of these techniques to react in real-time for any unforeseen scenario. To have more autonomy in UAVs, we aim to introduce artificial intelligence-based real-time UAV path planning with path optimization and collision avoidance as main objectives.



Appendix for Chapter 6

A.1 Sketch of proof of theorem 1 - Consideration of rectangles' borders

Let us first prove theorem 1 for a specific rectangle centered and of size 2×2 denoted \mathcal{R}^* with $M(t)$ a continuous mapping from $[0, T]$ to \mathbb{R}^2 such that $d(M(0), \mathcal{R}) > 0$ and for $t \in [0, T]$, $d(M(t), \mathcal{R}) \geq \rho$. We define $t \mapsto f(t)$ on $[0, T]$ as

$$\begin{cases} f(t) = d(M(t), \mathcal{R}^*) & \text{if } d(M(t), \mathcal{R}^*) > 0 \\ f(t) = -d(M(t), \partial\mathcal{R}^*) & \text{if } d(M(t), \mathcal{R}^*) < d(M(t), \partial\mathcal{R}^*) \\ f(t) = 0 & \text{if } d(M(t), \partial\mathcal{R}^*) = 0 \end{cases}$$

It is the following analytic expression of both distances which shows that f is well defined, continuous and that the first condition happens when $M(t)$ is outside the rectangle, the second condition when $M(t)$ is inside the rectangle and the third condition when $M(t)$ is on its border.

$$\begin{cases} \xi(x) = (||x| - 1|) \mathbf{1}_{\mathbb{R} \setminus [0,1]}(x) & \eta(x) = (||x| - 1|) \mathbf{1}_{[0,1]}(x) \\ d(M(t), \mathcal{R}^*) = \sqrt{\xi(x)^2 + \xi(y)^2} \\ d(M(t), \partial\mathcal{R}^*) = \sqrt{\xi(x)^2 + \xi(y)^2} + \min(\eta(x), \eta(y)) \end{cases}$$

where x and y are the coordinates of M and $\mathbf{1}_{\mathcal{S}}(x) = 0$ if $x \notin \mathcal{S}$ and $\mathbf{1}_{\mathcal{S}}(x) = 1$ if $x \in \mathcal{S}$, \mathcal{S} being any set. Based on the first assumption, we have $f(0) > 0$ and if the implication we are proving was false, then there would exist $t \in [0, T]$ such that $f(t) \leq 0$ which would imply the existence of $t' \in [0, T]$ with $f(t') = 0$ meaning that $d(M(t'), \partial\mathcal{R}^*) = 0$ and contradicting the second assumption.

We extend this proof to any rectangle \mathcal{R} by considering a transformation \mathcal{T} that is a combination of translation, rotation and rescaling, whose inverse is denoted \mathcal{T}^{-1} . It fulfills the following conditions with $\tau > 0$.

$$\left\{ \begin{array}{l} M \in \mathcal{R} \Leftrightarrow \mathcal{T}(M) \in \mathcal{R}^* \\ M \in \partial\mathcal{R} \Leftrightarrow \mathcal{T}(M) \in \partial\mathcal{R}^* \\ d(\mathcal{T}(M), \mathcal{T}(M')) = \tau d(M, M') \\ M(t)\text{continuous} \Leftrightarrow \mathcal{T}(M(t))\text{continuous} \end{array} \right.$$

These conditions prove that

$$d(M, \mathcal{R}) = \min_{M' \in \mathcal{R}} d(M, M') = \min_{\mathcal{T}(M') \in \mathcal{R}^*} d(M, M') = \frac{1}{\tau} \min_{\mathcal{T}(M') \in \mathcal{R}^*} d(\mathcal{T}(M), \mathcal{T}(M')) = \frac{1}{\tau} d(\mathcal{T}(M), \mathcal{R}^*)$$

and in the same way that $d(M, \partial\mathcal{R}) = \frac{1}{\tau} d(\mathcal{T}(M), \partial\mathcal{R}^*)$.

Hence the theorem proven for \mathcal{R}^* is also true for \mathcal{R} .

A.2 Sketch of proof of theorem 2 - Time of flight of a UAV

The proof is based on these four statements.

- The weighted partial sums of a given path are time stamps of the corresponding mapping $V \in \mathcal{V}'$.

$$t_n = \frac{1}{v} \sum_{k=2}^n d(P_{j_{k-1}}, P_{j_k}) \quad (\text{A.1})$$

- The weight of a path is the time of flight of a mapping in $V \in \mathcal{V}'$.

$$T_{\mathcal{V}}(V) = \frac{1}{v} \sum_{k=2}^N d(P_{j_{k-1}}, P_{j_k}) \quad (\text{A.2})$$

- Each traversed edge is a line segment of the trajectory traveled at speed v .

$$n \in \{2, \dots, N\}, t \in [t_{n-1}, t_n] \Rightarrow \overrightarrow{P_{j_{n-1}} V(t)} = \frac{t - t_{n-1}}{t_n - t_{n-1}} \overrightarrow{P_{j_{n-1}} P_{j_n}} \quad (\text{A.3})$$

- Based on equation (9), edges traversed by the path define line segments in the trajectory that remain at a safe distance from any obstacle.

$$\forall n \in \{2, \dots, N\}, (P_{j_{n-1}}, P_{j_n}) \in \mathcal{E} \Leftrightarrow \forall i \in \mathcal{I}, d(V(t), \overline{C_i D_i}) \geq \rho \quad (\text{A.4})$$

A.3 Sketch of proof of theorem 3 - Safe distance from the obstacles

As the four statements are similar, we exhibit only the proof of the first statement and assume that M is located such that $\theta = \angle(\vec{CD}, \vec{C^-M}) \in [\frac{\pi}{2}, 2\pi]$. Figure A.1 illustrates the two following arguments.

- If $\theta \in [\frac{\pi}{2}, \frac{3\pi}{2}]$, then M is located in the left half-plane delineated by line (C^-C^+) which is located at a distance of ρ from the obstacle \overline{CD} . As this property is shared too by C^- and using convexity, we get that $d(\overline{CD}, \overline{C^-M}) \geq \rho$.
- If $\theta \in [\frac{3\pi}{2}, 2\pi]$, then M is located in the left half-plane delineated by line (C^-D^-) which is located at a distance of ρ from the obstacle \overline{CD} . As this property is shared too by C^- and using convexity, we get also that $d(\overline{CD}, \overline{C^-M}) \geq \rho$.

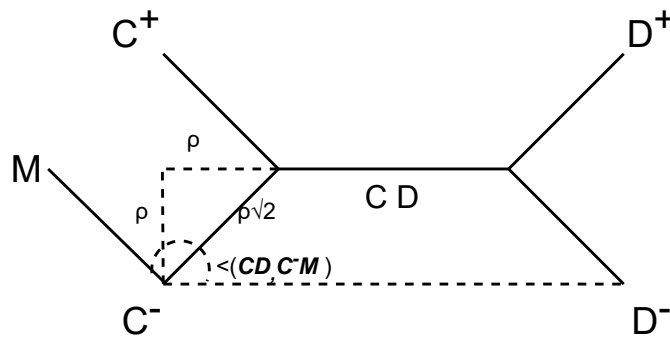


Figure A.1: Graphical construction illustrating theorem 3. As $\angle(CD, C^-M) = \frac{3\pi}{4}$, M is on the left of a half-plane delineated by line (C^-C^+) located at a distance of ρ of the obstacle CD .

Appendix for Chapter 7

B.1 Proof of theorem 4 - Inverse mapping fro each UAV

The proof is organized in three parts. First we use analytical formulas of $v(t)$, $\mathcal{S}(t)$ and $\Phi[s]$, the latter uses a given value of $s \geq 0$. Then we derive the desired properties. And thirdly we prove that $v(\tau)$, $\mathcal{S}(\tau)$ and $\Phi[s'']$ are known to all UAVs at time t if $\tau \leq t$ and $\Phi[s''] \leq t$.

i. We first compute analytic formulas.

- Based on assumption ii, we denote $v(t)$ as a finite linear combination of indicator functions also denoted with $\mathbf{1}$.

$$v(t) = \sum_{l=0}^{L-1} v_l \mathbf{1}_{[t_l, t_{l+1})}(t) \quad (\text{B.1})$$

where $[t_l, t_{l+1})$ are the left-closed and right-open intervals of the different indicator functions. Because a speed is non-negative and based on assumption ii, we also have

$$\begin{aligned} \forall l, v_l &\in [0, v_{\max}] \\ t_L &= +\infty \text{ and } v_{L-1} > 0 \end{aligned} \quad (\text{B.2})$$

- Using equation (7.26) defining the ramp function and integrating equation (B.1), we get

$$\mathcal{S}(t) = \sum_{l=0}^{L-1} v_l \mathcal{R}_{t_l}^{t_{l+1}}(t) \quad (\text{B.3})$$

- Because $\mathcal{S}(t)$ can be constant on some intervals, there are different ways of defining its inverse. We have chosen to define $\Phi[s]$ as

$$\Phi[s] = \max\{t | \mathcal{S}(t) \leq s\} \quad (\text{B.4})$$

An equivalent definition is

$$\Phi[s] = t_{l_s} + \frac{s - \mathcal{S}(t_{l_s})}{v_{l_s}} \quad (\text{B.5})$$

where

$$l_s = \max\{0 \leq l < L | \mathcal{S}(t_{l_s}) \leq s\} \quad (\text{B.6})$$

This definition of l_s ensures that $v_{l_s} \neq 0$, note that it says nothing as to whether $v_{l_s-1} = 0$.

We also have

$$\Phi[s] \in [t_{l_s}, t_{l_s+1}) \quad (\text{B.7})$$

ii. We now derive the expected properties.

- Equation (B.5) and the linearity of $\mathcal{S}(t)$ within each intervals $[t_l, t_{l+1})$ show that

$$\mathcal{S}(\Phi[s]) = \mathcal{S}(t_{l_s}) + \mathcal{R}_{\mathcal{S}(t_{l_s})}^{\mathcal{S}(t_{l_s+1})}(s) = s \quad (\text{B.8})$$

- We first assume that we do not have both, $s = \mathcal{S}(t_{l_s})$ and $v_{l_s-1} = 0$.

Equation (B.5) and (B.7) prove that

$$\frac{d}{ds}\Phi(s) = \frac{1}{v_{l_s}} = \frac{1}{v(\Phi(s))}$$

- We then assume that we do have both $s = \mathcal{S}(t_{l_s})$ and $v_{l_s-1} = 0$.

Because $s = \mathcal{S}(t_{l_s-1}) > \mathcal{S}(t_{l_s-2})$, we consider

$$s' \in [\mathcal{S}(t_{l_s-2}), \mathcal{S}(t_{l_s-1}))$$

and see that

$$\Phi(s') = t_{l_s-2} + \frac{s' - \mathcal{S}(t_{l_s-2})}{v_{l_s}} \rightarrow t_{l_s-1} < t_{l_s}$$

when $s' \rightarrow s^-$.

- iii. Finally we study the availability of these quantities at time t . Let us assume that $\tau \leq t$ and s'' such that $\Phi(s'') \leq t$.

- By defining $l_\tau = \max\{l | t_l \leq \tau\}$, we get $v(\tau) = v_{l_\tau}$ which is known to all UAVs at time t .
- Equation (B.3) shows that

$$\mathcal{S}(\tau) = \sum_{l=0}^{l_\tau-1} v_l(t_{l+1} - t_l) + v_{l_\tau}(\tau - t_{l_\tau})$$

which is also known to all UAVs at time t .

- Applying equation (B.5) to s'' , we have

$$\Phi[s''] = t_{l_{s''}} + \frac{s'' - \mathcal{S}(t_{l_{s''}})}{v_{l_{s''}}}$$

$t_{l_{s''}}$, $v_{l_{s''}}$ and $\mathcal{S}(t_{l_{s''}})$ are known to all UAVs at time t since:

- $t_{l_{s''}} \leq \Phi[s''] \leq t$
- $v_{l_{s''}} = v(t_{l_{s''}})$
- $\mathcal{S}(t_{l_{s''}}) = \int_0^{t_{l_{s''}}} v(\tau) d\tau$

Therefore $\Phi[s'']$ is also known to all UAVs at time t .

B.2 Proof of theorem 5 - Location of each virtual UAV

Equation (7.27) can be recast into five equations describing each of the five phases listed below equation (7.24).

$$\begin{aligned} V_n(t) = & \overbrace{\begin{aligned} & A - (n-1)\Delta\mathcal{S}_c \overrightarrow{e_0} & \text{if } \mathcal{S}(t) \in [0, L_{0,n}] \\ & A + (\mathcal{S}(t) - L_{1,n}) \overrightarrow{e_0} & \text{if } \mathcal{S}(t) \in [L_{0,n}, L_{1,n}] \\ & P_k + (\mathcal{S}(t) - L_{k,n}) \overrightarrow{e_k} & \text{if } \mathcal{S}(t) \in [L_{k,n}, L_{k+1,n}] \\ & & k \in \{1 \dots K-1\} \\ & B + (\mathcal{S}(t) - L_{K,n}) \overrightarrow{e_K} & \text{if } \mathcal{S}(t) \in [L_{K,n}, L_{K+1,n}] \\ & B - (N-n)\Delta\mathcal{S}_c \overrightarrow{e_K} & \text{if } \mathcal{S}(t) \in [L_{K+1,n}, L_{K+2,n}] \end{aligned}} \end{aligned} \quad (\text{B.9})$$

Equations (7.29) and (7.30) are then easily derived.

B.3 Proof of theorem 6 - UAV's obstacle avoidance condition

The proposed proof of theorem 6 uses the following lemma.

Lemma 1. Let P_k and P_{k+1} be two points defining a line segment $\overline{P_k P_{k+1}}$ sufficiently far from any obstacle in that

$$\forall i, d(\overline{P_k P_{k+1}}, \overline{C_i D_i}) \geq \rho_c \quad (\text{B.10})$$

Let M be a point of this line segment.

$$M \in \overline{P_k P_{k+1}}$$

Then for any point M' , we have

$$MM' < \rho_c \quad \Rightarrow \quad M' \notin \overline{C_i D_i}$$

Proof. We are actually proving here the contrapositive of this lemma.

Let M be a point in $\overline{P_k P_{k+1}}$ and M' another point in $\overline{C_i D_i}$, for some value of i and k .

Equation (B.10) implies that $MM' \geq \rho_c$. \square

This lemma is used in the following proof of theorem 6.

Proof. Let $t \in [0, T]$. Let $\mathbf{P} = [P_1 \dots P_K]$ be a complete path with Δ_A and Δ_B its departure and arrival areas all of them being sufficiently far from any obstacles (i.e. left of equation (7.31) holds). Equation (7.27) proves that there exists $k \in \{0 \dots K\}$ such that

$$V_n(t) \in \overline{P_{k,n} P_{k+1,n}} \quad (\text{B.11})$$

Equations (7.13), (7.16), and (7.17) prove that

$$\overline{P_{0,n} P_1} \subset \Delta_A \text{ and } \overline{P_K P_{K+1,n}} \subset \Delta_B$$

Therefore $d(\overline{P_{k,n} P_{k+1,n}}, \overline{C_i D_i}) \geq \rho_c$ holds also for $k \in \{0, K\}$.

Left of equation (7.5) in assumption 1 tells us that $R_n V_n(t) < \rho_c$. Lemma 1 ends the proof. \square

B.4 Proof of theorem 7- Avoidance of any mutual UAV collisions

Based on equations (7.9) and (7.23), we have

$$\Delta \mathcal{S}_r \geq \Delta \mathcal{S}_c \quad (\text{B.12})$$

Lemma 2. Let α and β be two non-negative real numbers.

$$\sqrt{\alpha^2 + \beta^2} \geq \frac{\alpha + \beta}{\sqrt{2}} \quad (\text{B.13})$$

Proof.

$$\begin{aligned}\sqrt{\alpha^2 + \beta^2} &\geq \frac{\alpha + \beta}{\sqrt{2}} \Leftrightarrow \alpha^2 + \beta^2 \geq \frac{(\alpha + \beta)^2}{2} \\ &\Leftrightarrow \frac{\alpha^2}{2} + \frac{\beta^2}{2} \geq \alpha\beta \Leftrightarrow \frac{1}{2}(\alpha - \beta)^2 \geq 0\end{aligned}$$

□

Lemma 3. *Let E, F, G be three points such that*

$$|\angle(\overrightarrow{EF}, \overrightarrow{FG})| \leq \frac{\pi}{2}$$

Let $M_e \in \overline{EF}$ and $M_g \in \overline{FG}$

$$M_e M_g \geq \frac{FM_e + FM_g}{\sqrt{2}} \quad (\text{B.14})$$

This proof uses the scalar product denoted \bullet and its corresponding norm $\| \cdot \|$ derived from the Euclidean distance.

Proof. We first prove that $\overrightarrow{FM_e} \bullet \overrightarrow{FM_g} \leq 0$ then use this result and that of lemma 2 to end the proof.

i. Because $M_e \in \overline{EF}$ and $M_g \in \overline{FG}$, there exist $\lambda, \mu \in [0, 1]$ such that

$$\begin{aligned}\overrightarrow{FM_e} &= \lambda \overrightarrow{FE} \\ \overrightarrow{FM_g} &= \mu \overrightarrow{FG}\end{aligned}$$

We can now conclude on the sign of $\overrightarrow{FM_e} \bullet \overrightarrow{FM_g}$

$$\overrightarrow{FM_e} \bullet \overrightarrow{FM_g} = \lambda \mu \overrightarrow{FE} \bullet \overrightarrow{FG} \cos \angle(\overrightarrow{FE}, \overrightarrow{FG}) \leq 0$$

ii. Thanks to some scalar product operations and to lemma 2, we get

$$\begin{aligned}M_e M_g &= \left\| \overrightarrow{FM_g} - \overrightarrow{FM_e} \right\| = \\ &= \sqrt{FM_g^2 - 2\overrightarrow{FM_e} \bullet \overrightarrow{FM_g} + FM_e^2} \geq \\ &= \sqrt{FM_g^2 + FM_e^2} \geq \frac{FM_e + FM_g}{\sqrt{2}}\end{aligned}$$

□

Before handling the proof itself, we recast equation (B.9) describing the position of each virtual UAV into two equations, first a description of where each virtual UAV is inside a global path and then a description of this global path.

$$S_n(s) = \mathcal{R}_{L_{0,n}}^{L_{K+1,n}}(s) - (n-1)\Delta\mathcal{S}_c \quad (\text{B.15})$$

$$\begin{aligned} \mathbf{V}(s) &= \mathbf{A} - (N-1)\Delta\mathcal{S}_c \overrightarrow{e_0} + \mathcal{R}_{-(N-1)\Delta\mathcal{S}_c}^0(s) \\ &+ \sum_{k=0}^K \mathcal{R}_{L_{k,1}}^{L_{k+1,1}}(s) \overrightarrow{e_k} \end{aligned} \quad (\text{B.16})$$

Simple computations show that

$$\mathbf{V}_n(t) = \mathbf{V}(S_n(S(t))) \quad (\text{B.17})$$

Lemma 4. *Let n_1 and n_2 denote two different UAV and let $S_{n_1}(s)$ and $S_{n_2}(s)$ be their relative positions on the global path. We have*

$$|S_{n_1}(s) - S_{n_2}(s)| \geq \Delta\mathcal{S}_c \quad (\text{B.18})$$

Proof. There is no loss of generality in assuming that $n_2 > n_1$. The proof depends on s and we consider three possibilities.

i. When $s \leq L_{0,n_2}$, we have

$$S_{n_2}(s) = -(n_2-1)\Delta\mathcal{S}_c \text{ and } S_{n_1}(s) \geq -(n_1-1)\Delta\mathcal{S}_c$$

By subtracting the two equations, we prove equation (B.18).

$$S_{n_2}(s) - S_{n_1}(s) \leq (n_1 - n_2)\Delta\mathcal{S}_c \leq -\mathcal{S}_c$$

ii. When $s \geq L_{K+1,n_1}$, we have

$$\begin{aligned} S_{n_2}(s) &\leq \mathcal{L}(\mathbf{P}) - (N - n_2)\Delta\mathcal{S}_c \\ S_{n_1}(s) &= \mathcal{L}(\mathbf{P}) - (N - n_1)\Delta\mathcal{S}_c \end{aligned}$$

By subtracting the two equations, we prove again equation (B.18).

$$S_{n_2}(s) - S_{n_1}(s) \leq (n_1 - n_2)\Delta\mathcal{S}_c \leq -\mathcal{S}_c$$

iii. When $s \in [L_{0,n_2}, L_{K+1,n_1}]$, we note that this implies $s \in [L_{0,n_1}, L_{K+1,n_2}]$ and therefore that

$$S_{n_2}(s) - S_{n_1}(s) = (n_1 - n_2)\Delta\mathcal{S}_r \leq -\Delta\mathcal{S}_r \leq -\Delta\mathcal{S}_c$$

□

Below is the proof of theorem 7.

Proof. Let $t \in [0, T]$ and n_1, n_2 denote two different UAVs. We again assume that $n_2 > n_1$, meaning that the n_2^{th} -UAV is following the n_1^{th} -UAV. Let $\mathbf{P} = [\mathbf{P}_1 \dots \mathbf{P}_K]$ be a complete path with Δ_A and Δ_B its departure and arrival areas, all of which fulfill left of equation (7.32). First we note that we only need to prove that $\mathbf{V}_{n_1}(t)\mathbf{V}_{n_2}(t) \geq 2\rho_c$, as left of equation (7.5) in assumption 1 transforms this statement into $\mathbf{R}_{n_1}(t) \neq \mathbf{R}_{n_2}(t)$.

Lemma 4 and equation (7.23) show that

$$|S_n(S(t)) - S_{n_1}(S(t))| \geq 2\rho_c\sqrt{2} \quad (\text{B.19})$$

And to prove that $\mathbf{V}_{n_1}(t)\mathbf{V}_{n_2}(t) \geq 2\rho_c$ we consider three possibilities, namely both UAVs are at time t : in the same line segment, in two consecutive line segments or in two distant line segments.

i. When both UAVs are in the same line segment, equations (B.17) and (B.16) show that

$$\begin{aligned} \mathbf{V}_{n_1}(t)\mathbf{V}_{n_2}(t) &= S_{n_2}(S(t)) - S_{n_1}(S(t)) \\ &\geq 2\rho_c\sqrt{2} \geq 2\rho_c \end{aligned}$$

ii. When both UAVs are in consecutive line segments, let \mathbf{J} denote their intersecting point.

Equations (B.17) and (B.16) prove that

$$\begin{aligned} \mathbf{V}_{n_1}(t)\mathbf{J} + \mathbf{J}\mathbf{V}_{n_2}(t) &= S_{n_2}(S(t)) - S_{n_1}(S(t)) \\ &\geq 2\rho_c\sqrt{2} \end{aligned}$$

Thanks to left of equation (7.32), consecutive line segments have acute bending

$$\left| \angle \left(\overrightarrow{\mathbf{V}_{n_1}(t)\mathbf{J}}, \overrightarrow{\mathbf{J}\mathbf{V}_{n_2}(t)} \right) \right| \leq \frac{\pi}{2}$$

Lemma 3 shows that $\mathbf{V}_{n_1}(t)\mathbf{V}_{n_2}(t) \geq 2\rho_c$

iii. When both UAVs are in non-consecutive line segments, then their mutual distance is greater than the distance between these line segments. This distance is lower bounded by $2\rho_c$, thanks to left of equation (7.32).

□

B.5 Proof of theorem 8 - All UAVs connectivity with starting or end locations

The definition below gives a more formal definitions of \bowtie and \boxtimes .

Definition 7. Let D_1 and D_2 be two points with wireless communication capabilities. The symbol \bowtie denotes here the ability to communicate directly.

$$D_1 \bowtie D_2 \Leftrightarrow D_1 D_2 \leq \rho_R \quad (\text{B.20})$$

Let D'_j a set of points. The symbol \boxtimes denotes here the ability to communicate using a routing communication path.

$$D_1 \boxtimes D_2, \Leftrightarrow \exists D'_{j_1} \dots D'_{j_J}, \text{ such that} \quad (\text{B.21})$$

$$D_1 \bowtie D'_{j_1} \bowtie D'_{j_2}, \dots, D'_{j_{J-1}} \bowtie D'_{j_J}, \text{ and } D'_{j_J} \bowtie D_2$$

\bowtie is regarded as a reflexive and symmetric relation. \boxtimes is transitive and also reflexive and symmetric, it is therefore regarded as an *equivalence relation*.

Similarly to lemma 4, we prove that consecutive UAVs are not too far apart.

Lemma 5. Let n and $n+1$ denote two consecutive drones.

$$0 \leq S_n(s) - S_{n+1}(s) \leq \Delta \mathcal{S}_r \quad (\text{B.22})$$

Proof. Left of equation (B.22) is only claiming that the $(n+1)^{\text{th}}$ is following the n^{th} , which is true. Right of equation (B.22) depends on the value of s , and three possibilities are considered.

- i. When $s \leq n(\Delta \mathcal{S}_r - \Delta \mathcal{S}_c)$ that is smaller than $L_{0,n+1}$, then either both UAVs are on their parking slots or the first one is leaving the parking slot.

In the first outcome, $S_n(s) - S_{n+1}(s)$ is smaller than $\Delta \mathcal{S}_c$ which is smaller than $\Delta \mathcal{S}_r$.

In the second outcome,

$$\begin{aligned} S_n(s) - S_{n+1}(s) &= \{-(n-1)\Delta \mathcal{S}_c\} - \{s - n(\Delta \mathcal{S}_r - \Delta \mathcal{S}_c)\} \\ &\leq \Delta \mathcal{S}_r \end{aligned}$$

- ii. When $s \geq \mathcal{L}(\mathbf{P}) + (n-1)\Delta \mathcal{S}_r + (N-n)\Delta \mathcal{S}_c$ that is greater than $L_{K+1,n+1}$ then either both UAVs are on their parking slots or the last one is reaching the parking slot.

In the first outcome, $S_n(s) - S_{n+1}(s)$ is smaller than $\Delta \mathcal{S}_c$ which is smaller than $\Delta \mathcal{S}_r$.

In the second outcome,

$$\begin{aligned} & S_n(s) - S_{n+1}(s) \\ &= \{\mathcal{L}(\mathbf{P}) + (N - n)\Delta\mathcal{S}_c\} - \{s - n\Delta\mathcal{S}_r\} \\ &\leq \Delta\mathcal{S}_r \end{aligned}$$

iii. When $s \in [L_{0,n+1}, L_{K+1,n}]$, then both UAVs are moving and

$$S_n(s) - S_{n+1}(s) = \Delta\mathcal{S}_r$$

□

Lemma 6. Let a_i be an increasing sequence of I numbers and $\overrightarrow{e_i}$ be unitary vectors. Let $\overrightarrow{f}(x)$ be a function defined as

$$\overrightarrow{f}(x) = \sum_{i=1}^{I-1} \mathcal{R}_{a_i}^{a_{i+1}}(x) \overrightarrow{e_i}$$

Then $\overrightarrow{f}(x)$ is a contraction mapping.

$$\left\| \overrightarrow{f}(x_1) - \overrightarrow{f}(x_2) \right\| \leq |x_1 - x_2| \quad (\text{B.23})$$

where $\| \cdot \|$ is the norm.

Proof. With no less of generality, we assume that $x_1 < x_2$.

- First we consider that both x_1 and x_2 are in $[a_1, a_I]$. Let i, j such that $x_1 \in [a_i, a_{i+1})$ and $x_2 \in [a_j, a_{j+1})$, we get

$$\begin{aligned} \overrightarrow{f}(x_2) - \overrightarrow{f}(x_1) &= (x_2 - a_j) \overrightarrow{e_j} \\ &+ \sum_{k=i+1}^{j-1} (a_{k+1} - a_k) \overrightarrow{e_k} + (a_i - x_1) \overrightarrow{e_i} \end{aligned}$$

Because the vectors $\overrightarrow{e_k}$ are unitary and because of the triangle inequality, we have

$$\begin{aligned} & \left\| \overrightarrow{f}(x_2) - \overrightarrow{f}(x_1) \right\| \\ & \leq (x_2 - a_j) + \sum_{k=i+1}^{j-1} (a_{k+1} - a_k) + (a_i - x_1) \\ & = x_2 - x_1 \end{aligned}$$

- If $x_1 \leq a_1$, $f(x_1) = f(a_1)$ and by applying equation (B.23) to (a_1, x_2) , we have

$$\left\| \overrightarrow{f}(x_2) - \overrightarrow{f}(x_1) \right\| \leq x_2 - a_1 \leq x_2 - x_1$$

- If $x_2 \geq a_I$, $f(x_2) = f(a_I)$ and by applying equation (B.23) to (x_1, a_I) , we have

$$\left\| \overrightarrow{f}(x_2) - \overrightarrow{f}(x_1) \right\| \leq a_I - x_1 \leq x_2 - x_1$$

□

Below is the proof of theorem 8.

Proof. Let \mathbf{P} be a complete path and $N > \frac{\mathcal{L}(\mathbf{P})}{\Delta\mathcal{S}_r} - 1$. The proof is in four parts, dealing first with connectivity among UAVs, then with connectivity first to A, then to B, and finally with A or B.

- To prove that all UAVs are connected to each other, it is sufficient to prove that each UAV is connected to its follower when it has one, that is

$$\forall n \in \{1 \dots N - 1\}, \mathbf{R}_n(t) \bowtie \mathbf{R}_{n+1}(t)$$

Let $n \in \{1 \dots N - 1\}$ refer to a UAV. Equation (B.17), lemmas 5 and 6 yield an upper bound on $\mathbf{V}_n(t)\mathbf{V}_{n+1}(t)$.

$$\begin{aligned} & \mathbf{V}_n(t)\mathbf{V}_{n+1}(t) \\ &= \left\| \overrightarrow{\mathbf{AV}}(S_{n+1}(S(t))) - \overrightarrow{\mathbf{AV}}(S_n(S(t))) \right\| \\ &\leq |S_{n+1}(S(t)) - S_n(S(t))| \\ &\leq \Delta\mathcal{S}_r \end{aligned}$$

The triangle inequality, equation (7.23) and assumptions i and iii prove that this UAV is directly connected with its follower.

$$\begin{aligned} & \mathbf{R}_n(t)\mathbf{R}_{n+1}(t) \\ &\leq \mathbf{R}_n(t)\mathbf{V}_n(t) + \mathbf{V}_n(t)\mathbf{V}_{n+1}(t) + \mathbf{R}_{n+1}(t)\mathbf{V}_{n+1}(t) \\ &\leq 2\rho_c + \Delta\mathcal{S}_r = \rho_r \end{aligned}$$

- We prove here that connectivity to A is ensured when

$$S(t) \in [0, N\Delta\mathcal{S}_r] \tag{B.24}$$

This interval comes from the following calculation.

$$[0, N\Delta\mathcal{S}_r] \subset \bigcup_{n=1}^N [L_{1,n} - \Delta\mathcal{S}_r, L_{1,n} + \Delta\mathcal{S}_r]$$

Hence to prove the statement, it suffices to show for all n that

$$S(t) \in [L_{1,n} - \Delta\mathcal{S}_r, L_{1,n} + \Delta\mathcal{S}_r] \Rightarrow R_n(t) \bowtie \mathbf{A}$$

Let $n \in \{1 \dots N\}$ and t such that $S(t)$ belongs to $[L_{1,n} - \Delta\mathcal{S}_r, L_{1,n} + \Delta\mathcal{S}_r]$.

Because $\mathbf{A} = \mathbf{V}(0) = \mathbf{V}(S_n(L_{1,n}))$, equation (B.17), lemmas 5 and 6 show that

$$\begin{aligned} \mathbf{V}_n(t)\mathbf{A} &= \|\mathbf{V}(S_n(S(t)) - \mathbf{V}(S_n(L_{1,n}))\| \\ &\leq |S_n(S(t)) - S_n(L_{1,n})| \\ &\leq |S(t) - L_{1,n}| \leq \Delta\mathcal{S}_r \end{aligned}$$

The triangle inequality, equation (7.23) and assumptions i and iii prove that $R_n(t) \bowtie \mathbf{A}$.

iii. We prove here that connectivity to \mathbf{B} is ensured when

$$S(t) \in [\mathcal{L}(\mathbf{P}) - \Delta\mathcal{S}_r, \mathcal{L}(\mathbf{P}) + N\Delta\mathcal{S}_r] \tag{B.25}$$

This interval comes from the following calculation.

$$\begin{aligned} &[\mathcal{L}(\mathbf{P}) - \Delta\mathcal{S}_r, \mathcal{L}(\mathbf{P}) + N\Delta\mathcal{S}_r] \subset \\ &\bigcup_{n=1}^N [L_{K,n} - \Delta\mathcal{S}_r, L_{K,n} + \Delta\mathcal{S}_r] \end{aligned}$$

Hence to prove the statement, it suffices to show for all n that

$$S(t) \in [L_{K,n} - \Delta\mathcal{S}_r, L_{K,n} + \Delta\mathcal{S}_r] \Rightarrow R_n(t) \bowtie \mathbf{B}$$

Let $n \in \{1 \dots N\}$ and t such that $S(t)$ belongs to $[\mathcal{L}(\mathbf{P}) - \Delta\mathcal{S}_r, \mathcal{L}(\mathbf{P}) + N\Delta\mathcal{S}_r]$.

Because $\mathbf{B} = \mathbf{V}(\mathcal{L}(\mathbf{P})) = \mathbf{V}(S_n(L_{K,n}))$, equation (B.17), lemmas 5 and 6 again show:

$$\mathbf{V}_n(t)\mathbf{B} = \|\mathbf{V}(S_n(S(t)) - \mathbf{V}(S_n(L_{K,n}))\| \leq \Delta\mathcal{S}_r$$

The triangle inequality, equation (7.23) and assumptions i and iii prove that $R_n(t) \bowtie \mathbf{B}$.

iv. All-time connectivity (i.e. $\forall n, t, R_n(t) \bowtie \mathbf{A}$ or $R_n(t) \bowtie \mathbf{B}$) is directly derived from equations (B.24) and (B.25) when

$$\begin{aligned} &[0, N\Delta\mathcal{S}_r] \cup [\mathcal{L}(\mathbf{P}) - \Delta\mathcal{S}_r, \mathcal{L}(\mathbf{P}) + N\Delta\mathcal{S}_r] \\ &\supset [0, \mathcal{L}(\mathbf{P}) + (N-1)\Delta\mathcal{S}_r] \end{aligned}$$

that is when

$$N\Delta\mathcal{S}_r \geq \mathcal{L}(\mathbf{P}) - \Delta\mathcal{S}_r$$

which is equivalent to $N \geq \frac{\mathcal{L}(\mathbf{P})}{\Delta\mathcal{S}_r} - 1$.

□

Bibliography

- [1] “Road traffic injuries,” Accessed: Dec. 15, 2021, Available: <https://www.who.int/news-room/fact-sheets/detail/road-traffic-injuries>.
- [2] B. Coifman, M. McCord, R. G. Mishalani, K. Redmill. 2004. “Surface transportation surveillance from unmanned aerial vehicles,” In *Proceedings of the 83rd Annual Meeting of the Transportation Research Board*, Washington, D.C, p. 28.
- [3] T. Anjuman, S. Hasanat-E-Rabbi, C. K. A. Siddiqui, M. M. Hoque. 2020. “Road traffic accident: A leading cause of the global burden of public health injuries and fatalities,” In *Proceedings of the International Conference on Mechanical Engineering*, Bangladesh, pp. 29–31.
- [4] N. A. Khan, N. Z. Jhanjhi, S. N. Brohi, R. S. A. Usmani, A. Nayyar. 2020. “Smart traffic monitoring system using Unmanned Aerial Vehicles (UAVs),” *Computer Communications*, vol. 157, pp. 434–443.
- [5] A. Beg, A. R. Qureshi, T. Sheltami, A. Yasar. 2021. “UAV-enabled intelligent traffic policing and emergency response handling system for the smart city,” *Personal and Ubiquitous Computing*, vol. 25, pp. 33–50.
- [6] F. Pan, Y. Yang, L. Zhang, C. Ma, J. Yang, X. Zhang. 2020. “Analysis of the impact of traffic violation monitoring on the vehicle speeds of urban main road: Taking China as an example,” *Journal of advanced transportation*, vol. 2020, p. 6304651.
- [7] “What’s behind the steep rise in road deaths in France?,” Accessed: Dec. 15, 2021, Available: <https://www.thelocal.fr/20190328/whats-behind-the-steep-rise-in-road-deaths-in-france>.
- [8] Z. Zhe, N. Yifeng, S. Lincheng. 2020. “Adaptive level of autonomy for human-UAVs collaborative surveillance using situated fuzzy cognitive maps,” *Chinese Journal of Aeronautics*, vol. 33, pp. 2835–2850.

- [9] H. Kim, J. Ben-Othman. 2018. “A collision-free surveillance system using smart UAVs in multi domain IoT,” *IEEE communications letters*, vol. 22, pp. 2587–2590.
- [10] W. Ejaz, A. Ahmed, A. Mushtaq, M. Ibnkahla. 2020. “Energy-efficient task scheduling and physiological assessment in disaster management using UAV-assisted networks,” *Computer Communications*, vol. 155, pp. 150–157.
- [11] H. S. Munawar, F. Ullah, S. Qayyum, S. I. Khan, M. Mojtahedi. 2021. “UAVs in Disaster Management: Application of Integrated Aerial Imagery and Convolutional Neural Network for Flood Detection,” *Sustainability*, vol. 13, p. 7547.
- [12] M. Erdelj, M. Król, E. Natalizio. 2017. “Wireless sensor networks and multi-UAV systems for natural disaster management,” *Computer Networks*, vol. 124, pp. 72–86.
- [13] W. Fawaz. 2018. “Effect of non-cooperative vehicles on path connectivity in vehicular networks: A theoretical analysis and UAV-based remedy,” *Vehicular Communications*, vol. 11, pp. 12–19.
- [14] H. Zhao, H. Wang, W. Wu, J. Wei. 2018. “Deployment algorithms for UAV airborne networks toward on-demand coverage,” *IEEE Journal on Selected Areas in Communications*, vol. 36, pp. 2015–2031.
- [15] G. Peng, Y. Xia, X. Zhang, L. Bai. 2018. “UAV-aided networks for emergency communications in areas with unevenly distributed users,” *Journal of Communications and Information Networks*, vol. 3, pp. 23–32.
- [16] K. Ro, J. S. Oh, L. Dong. 2007. “Lessons learned: Application of small uav for urban highway traffic monitoring,” In *Proceedings of 45th AIAA aerospace sciences meeting and exhibit*, Nevada, p. 596.
- [17] X. Meng, W. Wang, B. Leong. 2015. “Skystitch: A cooperative multi-UAV-based real-time video surveillance system with stitching,” In *Proceedings of the 23rd ACM international conference on Multimedia*, Australia, pp. 261–270.
- [18] J. Lee, Z. Zhong, K. Kim, B. Dimitrijevic, B. Du, S. Gutesa. 2015. “Examining the applicability of small quadcopter drone for traffic surveillance and roadway incident monitoring,” *Transportation Research Board 94th Annual Meeting*, Washington, D.C, p. 15.

- [19] “French Police Use Drones to Catch Dangerous Drivers,” Accessed: Dec. 15, 2021, Available: <https://dronelife.com/2017/11/14/french-police-drones/>.
- [20] “How drones can be used to combat COVID-19 ,” Accessed: Jul. 22, 2021, Available: <https://www.unicef.org/supply/documents/how-drones-can-be-used-combat-covid-19>.
- [21] “Innovating to Fight COVID-19: Four Ways Drones are Contributing,” Accessed: Apr. 15, 2020, Available: <https://enterprise.dji.com/fr/news/detail/fight-covid-19-with-drones>.
- [22] I. K. Ha, Y. Z. Cho. 2018. “A probabilistic target search algorithm based on hierarchical collaboration for improving rapidity of drones,” *Multidisciplinary Digital Publishing Institute*, vol. 18, p. 2535.
- [23] S. R. Bassolillo, E. D’Amato, I. Notaro, L. Blasi, M. Mattei. 2020. “Decentralized mesh-based model predictive control for swarms of UAVs,” *Sensors*, vol. 20, p. 4324.
- [24] S. Hayat, E. Yanmaz, C. Bettstetter, T. X. Brown. 2020. “Multi-objective drone path planning for search and rescue with quality-of-service requirements,” *Autonomous Robots*, vol. 44, pp. 1183–1198.
- [25] R. Carli, G. Cavone, N. Epicoco, M. Di Ferdinando, P. Scarabaggio, M. Dotoli. 2020, October. “Consensus-Based Algorithms for Controlling Swarms of Unmanned Aerial Vehicles,” In *Proceedings of International Conference on Ad-Hoc Networks and Wireless*, Bari, Italy, pp. 84–99.
- [26] L. De Filippis, G. Guglieri, F. Quagliotti. 2012. “Path planning strategies for UAVS in 3D environments,” *Journal of Intelligent & Robotic Systems*, vol. 65, pp. 247–264.
- [27] M. Khachumov, V. Khachumov. 2018. “The model of UAV formation based on the uniform allocation of points on the sphere,” In *Proceedings of MATEC Web of Conferences*, Russia, Article ID 03022.
- [28] G. Flores-Caballero, A. Rodríguez-Molina, M. Aldape-Pérez, M. G. Villarreal-Cervantes. 2020. “Optimized path-planning in continuous spaces for unmanned aerial vehicles using meta-heuristics,” *IEEE Access*, vol. 8, pp. 176774–176788.
- [29] G. Vásárhelyi, C. Virágh, G. Somorjai, T. Nepusz, A. E. Eiben, T. Vicsek. 2018. “Optimized flocking of autonomous drones in confined environments,” *Science Robotics*, vol. 3, Article ID eaat3536.

- [30] H. Nawaz, H. M. Ali, A. A. Laghari. 2021. "UAV communication networks issues: a review," *Archives of Computational Methods in Engineering*, vol. 28, pp. 1349–1369.
- [31] M. A. Dehghani, M. B. Menhaj. 2016. "Communication free leader–follower formation control of unmanned aircraft systems," *Robotics and Autonomous Systems*, vol. 80, pp. 69–75.
- [32] J. N. Yasin, S. A. S. Mohamed, M. H. Haghbayan, J. Heikkonen, H. Tenhunen, M. M. Yasin, J. Plosila. 2020. "Energy-efficient formation morphing for collision avoidance in a swarm of drones," *IEEE Access*, vol. 8, pp. 170681–170695.
- [33] Y. Chen, J. Yu, X. Su, G. Luo. 2015. "Path planning for multi-UAV formation," *Journal of Intelligent & Robotic Systems*, vol. 77, pp. 229–246.
- [34] R. D. Arnold, H. Yamaguchi, T. Tanaka. 2018. "Search and rescue with autonomous flying robots through behavior-based cooperative intelligence," *Journal of International Humanitarian Action*, vol. 3, pp. 1–18.
- [35] G. Lee, D. Chwa. 2018. "Decentralized behavior-based formation control of multiple robots considering obstacle avoidance," *Intelligent Service Robotics*, vol. 11, pp. 127–138.
- [36] T. M. Cabreira, L. B. Brisolara, P. R. Ferreira Jr. 2019. "Survey on coverage path planning with unmanned aerial vehicles," *Drones*, vol. 3, p. 4.
- [37] F. Fabra, W. Zamora, P. Reyes, J. A. Sanguesa, C. T. Calafate, J. C. Cano, P. Manzoni. 2020. "MUSCOP: Mission-based UAV swarm coordination protocol," *IEEE Access*, vol. 8, pp. 72498–72511.
- [38] H. Xiang, L. Tian, "Development of a low-cost agricultural remote sensing system based on an autonomous unmanned aerial vehicle (UAV)," *Biosystems engineering*, vol. 108, pp. 174–190, February 2011.
- [39] H. Shakhathreh, A.H. Sawalmeh, A. Al-Fuqaha, Z. Dou, E. Almaita, I. Khalil, N. S. Othman, A. Khreishah, M. Guizani, "Unmanned aerial vehicles (UAVs): A survey on civil applications and key research challenges," *IEEE Access*, vol. 7, pp. 48572–48634, April 2019.
- [40] I. Bekmezci, O. K. Sahingoz, Ş. Temel, "Flying ad-hoc networks (FANETs): A survey," *Ad Hoc Networks*, vol. 11, pp. 1254–1270, May 2013.

-
- [41] Z. Sun, P. Wang, M. C. Vuran, M. Al-Rodhaan, A. Al-Dhelaan, I. F. Akyildiz, "BorderSense: border patrol through advanced wireless sensor networks," *Ad Hoc Networks*, vol. 9, pp. 468–477, May 2011.
- [42] E. P. de Freitas, T. Heimfarth, I. F. Netto, C. E. Lino, C. E. Pereira, A. M. Ferreira, F. R. Wagner, T. Larsson, "UAV relay network to support WSN connectivity," *ICUMT, IEEE*, pp. 309–314, October 2010.
- [43] I. Maza, F. Caballero, J. Capitán, J. R. Martínez-De-Dios, A. Ollero, "Experimental results in multi-UAV coordination for disaster management and civil security applications," *Journal of Intelligent and Robotics Systems*, pp. 563–585, January 2011.
- [44] H. Chao, Y. Cao, Y. Chen, "Autopilots for small fixed-wing unmanned air vehicles: a survey," In *International Conference on Mechatronics and Automation*, pp. 3144–3149, August 2007.
- [45] B. S. Morse, C. H. Engh, M. A. Goodrich, "UAV Video Coverage Quality Maps and Prioritized Indexing for Wilderness Search and Rescue," In *Proceedings of the 5th ACM/IEEE International Conference on Human–Robot Interaction*, pp. 227–234, March 2010.
- [46] J. Jiang, and G. Han, "Routing protocols for unmanned aerial vehicles," *IEEE Communications Magazine*, vol. 56, pp. 58–63, 2018.
- [47] A. Purohit, F. Mokaya, P. Zhang, "Collaborative indoor sensing with the SensorFly aerial sensor network," In *ACM/IEEE 11th International Conference on Information Processing in Sensor Networks*, pp. 145–146, April 2012.
- [48] O. K. Sahingoz, "Networking models in flying ad-hoc networks (FANETs): Concepts and challenges," *Journal of Intelligent & Robotic Systems*, vol. 74, pp. 513–527, April 2014.
- [49] D. B. Johnson, Y. C. Hu, and D. A. Maltz, "The dynamic source routing protocol (DSR) for mobile ad hoc networks for IPv4 ," RFC 4728, Internet Engineering Task Force, February 2007.
- [50] S. R. Das, E. M. Belding-Royer, and C. E. Perkins, "Ad hoc on-demand distance vector (AODV) routing," *IETF RFC 3561*, July 2003.
- [51] W. Xia, Y. Wen, C. H. Foh, D. Niyato, and H. Xie, "A survey on software-defined networking," *IEEE Communications Surveys & Tutorials*, vol. 17, pp. 27–51, June 2014.
- [52] T. Clausen, P. Jacquet, "Optimized Link State Routing," *IETF RFC 3626*, October 2003.

-
- [53] J. H. Forsmann, R. E. Hiromoto, J. Svoboda, "A time-slotted on-demand routing protocol for mobile ad hoc unmanned vehicle systems," In *SPIE Unmanned Systems Technology*, pp. 65611P, May 2007.
- [54] A. I. Alshbatat, L. Dong, "Cross layer design for mobile ad-hoc unmanned aerial vehicle communication networks," In *International Conference on Networking, Sensing and Control*, pp. 331–336, April 2010.
- [55] L. Lin, Q. Sun, J. Li, F. Yang, "A novel geographic position mobility oriented routing strategy for UAVs," *Journal of Computational Information Systems*, vol. 8, pp. 709–716, 2012.
- [56] E. Kuiper, S. Nadjm-Tehrani, "Mobility models for UAV group reconnaissance applications," In *International Conference on Wireless and Mobile Communications*, pp. 33–33, July 2006.
- [57] D. L. Gu, G. Pei, H. Ly, M. Gerla, X. Hong, "Hierarchical routing for multi-layer ad-hoc wireless networks with UAVs," In *Proceedings of 21st Century Military Communications. Architectures and Technologies for Information Superiority*, pp. 310–314, October 2000.
- [58] R. Stanojevic, R. N. Shorten, C. M. Kellett, "Adaptive tuning of drop-tail buffers for reducing queueing delays," *IEEE Communications Letters*, vol. 10, no. 7, pp. 570–572, 2006.
- [59] L. Gupta, R. Jain, and G. Vaszkun, "Survey of important issues in UAV communication networks," *IEEE Communications Surveys & Tutorials*, vol. 18, pp. 1123–1152, November 2015.
- [60] A. Otto, N. Agatz, J. Campbell, B. Golden, and E. Pesch, "Optimization approaches for civil applications of unmanned aerial vehicles (UAVs) or aerial drones: A survey," *Networks*, vol. 72, pp. 411–458, December 2018.
- [61] J. Jiang, and G. Han, "Routing protocols for unmanned aerial vehicles," *IEEE Communications Magazine*, vol. 56, pp. 58–63, 2018.
- [62] I. Bekmezci, O. K. Sahingoz, and S. Temel, "Flying ad-hoc networks (FANETs): A survey," *Ad Hoc Networks*, vol. 11, pp. 1154–1170, May 2013.
- [63] C. M. Cheng, P. H. Hsiao, H. T. Kung, and D. Vlah, "Maximizing throughput of UAV-relaying networks with the load-carry-and-deliver paradigm," *IEEE Wireless Communications and Networking Conference (WCNC)*, pp. 4417–4424, March 2007.

-
- [64] C. Tsu-Wei, G. Mario, “Global state routing: A new routing scheme for ad-hoc wireless networks,” In IEEE International Conference on Communications (ICC), vol. 1, pp. 171–175, June 1998.
- [65] 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, pp. 234–244, October 1994.
- [66] R. Shirani, M. St-Hilaire, T. Kunz, Y. Zhou, J. Li, and L. Lamont, “Combined reactive-geographic routing for unmanned aeronautical ad-hoc networks,” 8th International Conference on Wireless Communications and Mobile Computing (IWCMC), pp. 820–826, August 2012.
- [67] F. H. Tseng, H. P. Chiang, and H. C. Chao, “Black Hole along with Other Attacks in MANETs: A Survey,” Journal of Information Processing Systems, vol. 14, pp. 56–78, February 2018.
- [68] Z. J. Haas, M. R. Pearlman, and P. Samar, “The zone routing protocol (ZRP) for ad hoc networks,” IETF Internet Draft draft-ietf-manet-zone-zrp-04, July 2002.
- [69] V. Park, and S. Corson, “Temporally-ordered routing algorithm (TORA),” IETF Internet Draft, July 2001.
- [70] C. P. Singh, O. P. Vyas, and M. K. Tiwari, “A survey of simulation in sensor networks,” In: International Conference on Computational Intelligence for Modelling Control & Automation, Vienna, Austria, pp. 867–872, December 2008.
- [71] M. Korkalainen, M. Sallinen, N. Karkkainen, and P. Tukeva, “Survey of wireless sensor networks simulation tools for demanding applications,” In: Proceedings of the IEEE Fifth international conference on networking and services, Valencia, Spain, pp. 102–106, April 2009.
- [72] A. Abuarqoub, F. Alfayez, M. Hammoudeh, T. Alsboui, A. Nisbet, “Simulation issues in wireless sensor networks: a survey,” In: Proceedings of the 2012 Sixth International Conference on Sensor Technologies and Applications, Rome, Italy, pp. 222–228, August 2012.

- [73] A. S. Tonneau, N. Mitton, J. Vandaele, “How to choose an experimentation platform for wireless sensor networks? A survey on static and mobile wireless sensor network experimentation facilities,” *Ad Hoc Networks*, vol. 30, pp. 115–127, 2015.
- [74] H. V. Abeywickrama, B. A. Jayawickrama, Y. He, E. Dutkiewicz, “Comprehensive energy consumption model for unmanned aerial vehicles, based on empirical studies of battery performance,” *IEEE Access*, vol. 6, pp. 58383–58394, 2018.
- [75] H. Shakhathreh, A. H. Sawalmeh, A. Al-Fuqaha, Z. Dou, E. Almaita, I. Khalil, N. S. Othman, A. Khreishah, M. Guizani, “Unmanned aerial vehicles (UAVs): A survey on civil applications and key research challenges,” *IEEE Access*, vol. 7, pp. 48572–48634, 2019.
- [76] S. Rosati, K. Kruźelecki, G. Heitz, D. Floreano, B. Rimoldi “Dynamic routing for flying ad hoc networks,” *IEEE Transactions on Vehicular Technology*, vol. 65, pp. 1690–1700, 2015.
- [77] N. E. H. Bahloul, S. Boudjit, M. Abdennebi, D. E. Boubiche, “A flocking-based on demand routing protocol for unmanned aerial vehicles,” *Journal of Computer Science and Technology*, vol. 33, pp. 263–276, 2018.
- [78] P. Basu, J. Redi, “Movement control algorithms for realization of fault-tolerant ad hoc robot networks,” *IEEE network*, vol. 18, pp. 36–44, 2004.
- [79] W. Tian, Z. Jiao, M. Liu, M. Zhang, D. Li, “Cooperative communication based connectivity recovery for UAV networks,” *Proceedings of the ACM Turing Celebration Conference, China*, pp. 1–5, 2019.
- [80] Y. Chen, D. Baek, A. Bocca, A. Macii, M. Poncino, “A Case for a Battery-Aware Model of Drone Energy Consumption,” *IEEE International Telecommunications Energy Conference (INTELEC)*, Turin, Italy, pp. 1–8, 2018.
- [81] H. V. Abeywickrama, B. A. Jayawickrama, Y. He, E. Dutkiewicz. 2018. “Comprehensive energy consumption model for unmanned aerial vehicles, based on empirical studies of battery performance,” *IEEE Access*, vol. 6, pp. 58383–58394.
- [82] N. Bashir, Z. H. Abbas, G. Abbas. 2019. “On Demand Cluster Head Formation with Inherent Hierarchical Clustering and Reliable Multipath Routing in Wireless Sensor Networks,” *Adhoc & Sensor Wireless Networks*, vol. 45, no. 1-2, pp. 59–91.

- [83] D. Popescu, C. Dragana, F. Stoican, L. Ichim, G. Stamatescu. 2018. “A collaborative UAV-WSN network for monitoring large areas,” *Sensors*, vol. 18, p. 4202.
- [84] C. A. Trasviña-Moreno, R. Blasco, Á. Marco, R. Casas, A. Trasviña-Castro. 2017. “Unmanned aerial vehicle based wireless sensor network for marine-coastal environment monitoring,” *Sensors*, vol. 17, p. 460.
- [85] C. Zhan, Y. Zeng, R. Zhang. 2017. “Energy-efficient data collection in UAV enabled wireless sensor network,” *IEEE Wireless Communications Letters*, vol. 7, pp. 328–331.
- [86] C. Wang, F. Ma, J. Yan, D. De, S. K. Das. 2015. “Efficient aerial data collection with UAV in large-scale wireless sensor networks,” *International Journal of Distributed Sensor Networks*, vol. 11, Art. no. 286080.
- [87] S. Berrahal, J. H. Kim, S. Rekhis, N. Boudriga, D. Wilkins, J. Acevedo. 2016. “Border surveillance monitoring using quadcopter UAV-aided wireless sensor networks,” *Journal of Communications Software and Systems*, vol. 12, pp. 67–82.
- [88] S. Srinivasan, H. Latchman, J. Shea, T. Wong, J. McNair. 2004. “Airborne traffic surveillance systems: video surveillance of highway traffic,” In *Proceedings of the ACM 2nd international workshop on Video surveillance & sensor networks*, New York, USA, pp. 131–135.
- [89] Y. M. Chen, L. Dong, J. S. Oh. 2007. “Real-time video relay for uav traffic surveillance systems through available communication networks,” In *IEEE wireless communications and networking conference*, Hong Kong, pp. 2608–2612.
- [90] M. Elloumi, R. Dhaou, B. Escrig, H. Idoudi, L. A. Saidane. 2018. “Monitoring road traffic with a UAV-based system,” In *IEEE Wireless Communications and Networking Conference (WCNC)*, Barcelona, Spain, pp. 1–6.
- [91] D. Popescu, F. Stoican, G. Stamatescu, O. Chenaru, L. Ichim. 2019. “A survey of collaborative UAV-WSN systems for efficient monitoring,” *Sensors*, vol. 19, p. 4690.
- [92] F. Losilla, A. J. Garcia-Sanchez, F. Garcia-Sanchez, J. Garcia-Haro, Z. J. Haas. 2011. “A comprehensive approach to WSN-based ITS applications: A survey,” *Sensors*, vol. 11, pp. 10220–10265.
- [93] T. Bensiradj, S. Moussaoui. 2019. “Strategy efficient to extend the lifetime of wireless sensor networks in a framework of hybrid sensors and vehicular networks for road safety,” *IET Wireless Sensor Systems*, vol. 9, pp. 416–423.

- [94] S. E. Yoo, P. K. Chong, T. Park, Y. Kim, D. Kim, C. Shin, K. Sung, H. Kim. 2008. “DGS: Driving guidance system based on wireless sensor network,” In *22nd International Conference on Advanced Information Networking and Applications-Workshops (aina workshops)*, Okinawa, pp. 628–633.
- [95] S. E. Yoo, P. K. Chong, D. Kim. 2009. “S3: School zone safety system based on wireless sensor network,” *Sensors*, vol. 9, pp. 5968–5988.
- [96] J. Las Fargeas, P. Kabamba, A. Girard. 2015. “Cooperative surveillance and pursuit using unmanned aerial vehicles and unattended ground sensors,” *Sensors*, vol. 15, pp. 1365–1388.
- [97] N. Bashir, S. Boudjit. 2020. “An Energy-Efficient Collaborative Scheme for UAVs and VANETs for Dissemination of Real-Time Surveillance Data on Highways,” In *IEEE 17th Annual Consumer Communications & Networking Conference (CCNC)*, Las Vegas, NV, USA, pp. 1–6.
- [98] A. N. Knaian. 2000. “A wireless sensor network for smart roadbeds and intelligent transportation systems,” Doctoral dissertation, Massachusetts Institute of Technology.
- [99] I. K. Ha, Y. Z. Cho. 2018. “A probabilistic target search algorithm based on hierarchical collaboration for improving rapidity of drones,” *Sensors*, vol. 18, p. 2535.
- [100] M. Elloumi, R. Dhaou, B. Escrig, H. Idoudi, L. A. Saidane. 2018. “Monitoring road traffic with a UAV-based system,” In *IEEE Wireless Communications and Networking Conference (WCNC)*, Barcelona, Spain, pp. 1–6.
- [101] B. D. Deebak, F. Al-Turjman. 2020. “A hybrid secure routing and monitoring mechanism in IoT-based wireless sensor networks,” *Ad Hoc Networks*, vol. 97, p. 102022.
- [102] N. Dutta, M. D. Fontaine. 2019. “Improving freeway segment crash prediction models by including disaggregate speed data from different sources,” *Accident Analysis & Prevention*, vol. 132, p. 105253.
- [103] I. F. Akyildiz, T. Melodia, K. R. Chowdhury, “A survey on wireless multimedia sensor networks,” *Computer networks*, vol. 51, no. 4, pp. 921–960, March 2007.
- [104] C. Y. Chong, S. P. Kumar, “Sensor networks: evolution, opportunities, and challenges,” *Proceedings of the IEEE*, vol. 91, no. 8, pp. 1247–1256, August 2003.

- [105] J. Yick, B. Mukherjee, D. Ghosal, "Wireless sensor network survey," *Computer networks*, vol. 52, no. 12, pp. 2292–2330, August 2008.
- [106] P. Natarajan, P. K. Atrey, M. Kankanhalli, "Multi-camera coordination and control in surveillance systems: A survey," *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, vol. 11, no. 4, pp. 57, June 2015.
- [107] W. Osamy, A. A. El-Sawy, A. M. Khedr. 2020. "Effective TDMA scheduling for tree-based data collection using genetic algorithm in wireless sensor networks," *Peer-to-Peer Networking and Applications*, vol. 13, pp. 796–815.
- [108] H. M. A. Fahmy. 2016. "Wireless sensor networks: concepts, applications, experimentation and analysis," *Springer*.
- [109] T. Melodia, M. C. Vuran, D. Pompili. 2005, July. "The state of the art in cross-layer design for wireless sensor networks," In *Proceedings of International workshop of the EuroNGI network of excellence*, Villa Vigoni, Italy, pp. 78–92.
- [110] L. Louail, V. Felea. 2016, July. "Routing and TDMA joint Cross-Layer design for wireless sensor networks," In *Proceedings of International Conference on Ad-Hoc Networks and Wireless*, Lille, France, pp. 111–123.
- [111] M. K. Marina, S. R. Das, "Ad hoc on-demand multipath distance vector routing," *Wireless communications and mobile computing*, vol. 6, no. 7, pp. 969–988, 2006.
- [112] Z. J. Haas, M. R. Pearlman, "The performance of query control schemes for the zone routing protocol," *IEEE/ACM Transactions on networking*, vol. 9, no. 4, pp. 427–438, 2001.
- [113] J. Hao, Z. Yao, K. Huang, B. Zhang, C. Li, "An energy-efficient routing protocol with controllable expected delay in duty-cycled wireless sensor networks," *IEEE International Conference on Communications (ICC)*, pp. 6215–6219, 2013.
- [114] R. W. Coutinho, A. Boukerche, L. F. Vieira, A. A. Loureiro, "Transmission power control-based opportunistic routing for wireless sensor networks," *Proceedings of the 17th ACM international conference on Modeling, analysis and simulation of wireless and mobile systems*, pp. 219–226, 2014.
- [115] M. Maimour, "Maximally radio-disjoint multipath routing for wireless multimedia sensor networks," *Proceedings of the 4th ACM workshop on Wireless multimedia networking and performance modeling*, pp. 26–31, 2008.

- [116] J. Y. Teo, Y. Ha, C. K. Tham, “Interference-minimized multipath routing with congestion control in wireless sensor network for high-rate streaming,” *IEEE Transactions on Mobile Computing*, vol. 7, no. 9, pp. 1124–1137, 2008.
- [117] M. Radi, B. Dezfouli, K. A. Bakar, S. A. Razak, M. A. Nematbakhsh, “Interference-aware multipath routing protocol for QoS improvement in event-driven wireless sensor networks,” *Tsinghua Science and Technology*, vol. 16, no. 5, pp. 475–490, 2011.
- [118] B. Y. Li, P. J. Chuang, “Geographic energy-aware non-interfering multipath routing for multimedia transmission in wireless sensor networks,” *Information Sciences*, vol. 249, pp. 24–37, 2013.
- [119] H. A. Satai, M. M. A. Zahra, Z. I. Rasool, R. S. Abd-Ali, C. I. Pruncu. 2021. “Bézier curves-based optimal trajectory design for multirotor UAVs with any-angle pathfinding algorithms,” *Sensors*, vol. 21, p. 2460.
- [120] V. San Juan, M. Santos, J. M. Andújar, 2018. “Intelligent UAV map generation and discrete path planning for search and rescue operations,” *Complexity*, vol. 2018, p. 17.
- [121] C. Goerzen, Z. Kong, B. Mettler. 2010. “A survey of motion planning algorithms from the perspective of autonomous UAV guidance,” *Journal of Intelligent and Robotic Systems*, vol. 57, pp. 65–100.
- [122] M. Naazare, D. Ramos, J. Wildt, D. Schulz. 2019, September. “Application of graph-based path planning for uavs to avoid restricted areas,” In *Proceedings of IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR)*, Würzburg, Germany, pp. 139–144.
- [123] M. Radmanesh, M. Kumar, P. H. Guentert, M. Sarim. 2018. “Overview of path-planning and obstacle avoidance algorithms for UAVs: a comparative study,” *Unmanned systems*, vol. 6, pp. 95–118.
- [124] L. De Filippis, G. Guglieri, F. Quagliotti. 2012. “Path planning strategies for UAVS in 3D environments,” *Journal of Intelligent & Robotic Systems*, vol. 65, pp. 247–264.
- [125] S. Goudarzi, N. Kama, M. H. Anisi, S. Zeadally, S. Mumtaz. 2019. “Data collection using unmanned aerial vehicles for internet of things platforms,” *Computers & Electrical Engineering*, vol. 75, pp. 1–15.

- [126] Y. Zhao, Z. Zheng, Y. Liu. 2018. "Survey on computational-intelligence-based UAV path planning," *Knowledge-Based Systems*, vol. 158, pp. 54–64.
- [127] P. Vadakkepat, K. C. Tan, W. Ming-Liang. 2000, July. "Evolutionary artificial potential fields and their application in real time robot path planning," In *Proceedings of the 2000 congress on evolutionary computation*, La Jolla, CA, USA, pp. 256–263.
- [128] G. Zhang, L.T. Hsu. 2019. "A new path planning algorithm using a GNSS localization error map for UAVs in an urban area," *Journal of Intelligent & Robotic Systems*, vol. 94, pp. 219–235.
- [129] O. Khatib. 1986. "Real-time obstacle avoidance for manipulators and mobile robots," *Autonomous robot vehicles*, Springer, pp. 396–404.
- [130] L. Lifen, S. Ruoxin, L. Shuandao, W. Jiang. 2016, August. "Path planning for UAVS based on improved artificial potential field method through changing the repulsive potential function," In *Proceedings of Chinese Guidance, Navigation and Control Conference*, Nanjing, China, pp. 2011–2015.
- [131] X. Chen, J. Zhang. 2013, August. "The three-dimension path planning of UAV based on improved artificial potential field in dynamic environment," In *Proceedings of 5th International Conference on Intelligent Human-Machine Systems and Cybernetics*, Hangzhou, China , pp. 144–147.
- [132] Y. Yao, X. S. Zhou, K. L. Zhang, D. Dong. 2010. "Dynamic trajectory planning for unmanned aerial vehicle based on sparse A* search and improved artificial potential field," *Kongzhi Lilun Yu Yingyong/Control Theory and Applications*, vol. 27, pp. 953–959.
- [133] Z. Yingkun. 2018, June. "Flight path planning of agriculture UAV based on improved artificial potential field method," In *Proceedings of Chinese Control And Decision Conference*, Shenyang, China , pp. 1526–1530.
- [134] E. Oland, R. Kristiansen. 2013, March. "Collision and terrain avoidance for UAVs using the potential field method," In *Proceedings of IEEE Aerospace Conference*, Big Sky, MT, USA, pp. 1–7.
- [135] L. Lifen, S. Ruoxin, L. Shuandao, W. Jiang. 2016, August. "Path planning for UAVS based on improved artificial potential field method through changing the repulsive potential

- function,” In *Proceedings of IEEE Chinese Guidance, Navigation and Control Conference (CGNCC)*, Nanjing, China, pp. 2011–2015.
- [136] Y. B. Chen, G. C. Luo, Y. S. Mei, J. Q. Yu, X. L. Su. 2016. “UAV path planning using artificial potential field method updated by optimal control theory,” *International Journal of Systems Science*, vol. 47, pp. 1407–1420.
- [137] C. T. Recchiuto, A. Sgorbissa. 2018. “Post-disaster assessment with unmanned aerial vehicles: A survey on practical implementations and research approaches,” *Journal of Field Robotics*, vol. 35, pp. 459–490.
- [138] F. Yan, Y. S. Liu, J. Z. Xiao. 2013. “Path planning in complex 3D environments using a probabilistic roadmap method,” *International Journal of Automation and computing*, vol. 10, pp. 525–533.
- [139] M. Kothari, I. Postlethwaite, D. W. Gu. 2010. “A Suboptimal Path Planning Algorithm Using Rapidly-exploring Random Trees,” *International Journal of Aerospace Innovations*, vol. 2.
- [140] M. W. Achtelik, S. Weiss, M. Chli, R. Siegwart. 2013, May. “Path planning for motion dependent state estimation on micro aerial vehicles,” In *Proceedings of IEEE international conference on robotics and automation*, Karlsruhe, Germany, pp. 3926–3932.
- [141] M. Kothari, I. Postlethwaite, D. W. Gu. 2009, December. “Multi-UAV path planning in obstacle rich environments using rapidly-exploring random trees,” In *Proceedings of IEEE on Decision and Control (CDC) held jointly with 2009 28th Chinese Control Conference*, Shanghai, China, pp. 3069–3074.
- [142] Sun, Q., Li, M., Wang, T. and Zhao, C., 2018, June. ‘UAV path planning based on improved rapidly-exploring random tree,’ In *Proceedings of IEEE on Chinese Control and Decision Conference*, Shenyang, China, pp. 6420–6424.
- [143] E. W. Dijkstra. 1959. “A note on two problems in connexion with graphs,” *Numerische mathematik*, vol. 1, pp. 269–271.
- [144] P. Maini, P. B. Sujit. 2016, June. “Path planning for a uav with kinematic constraints in the presence of polygonal obstacles,” In *Proceedings of IEEE international conference on unmanned aircraft systems (ICUAS)*, Arlington, VA, USA, pp. 62–67.

- [145] F. L. L. Medeiros, J. D. S. Da Silva. 2010, October. “A Dijkstra algorithm for fixed-wing UAV motion planning based on terrain elevation,” In *Brazilian Symposium on Artificial Intelligence*, Berlin, Heidelberg, pp. 213–222.
- [146] Y. Qu, Y. Zhang, Y. Zhang. 2018. “A global path planning algorithm for fixed-wing UAVs,” *Journal of Intelligent & Robotic Systems*, vol. 91, pp. 691–707.
- [147] X. Chen, X. Chen. 2014, May. “The UAV dynamic path planning algorithm research based on Voronoi diagram,” In *26th IEEE chinese control and decision conference*, Changsha, China, pp. 1069–1071.
- [148] P. E. Hart, N. J. Nilsson, B. Raphael. 1968. “A formal basis for the heuristic determination of minimum cost paths,” *IEEE transactions on Systems Science and Cybernetics*, vol. 4, pp. 100–107.
- [149] K. Yang, S. Sukkarieh. 2008, September. “3D smooth path planning for a UAV in cluttered natural environments,” In *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, Nice, France, pp. 794–800.
- [150] W. Chen, X. Wu, Y. Lu. 2015. “An improved path planning method based on artificial potential field for a mobile robot,” *Cybernetics and information technologies*, vol. 15, pp. 181–191.
- [151] M. De Berg, M. Van Kreveld, M. Overmars, O. C. Schwarzkopf. 2000. “Visibility graphs,” *Computational geometry*, pp. 307–317.
- [152] M. Doole, J. Ellerbroek, V. L. Knoop, J. M. Hoekstra. 2021. “Constrained Urban Airspace Design for Large-Scale Drone-Based Delivery Traffic,” *Aerospace*, vol. 15, p. 38.
- [153] C. Goerzen, Z. Kong, B. Mettler. 2010. “A survey of motion planning algorithms from the perspective of autonomous UAV guidance,” *Journal of Intelligent and Robotic Systems*, vol. 57, pp. 65–100.
- [154] G. Zhang, L. T. Hsu. 2019. “A new path planning algorithm using a GNSS localization error map for UAVs in an urban area,” *Journal of Intelligent & Robotic Systems*, vol. 94, pp. 219–235.
- [155] C. Yin, Z. Xiao, X. Cao, X. Xi, P. Yang, D. Wu. 2017. “Offline and online search: UAV multiobjective path planning under dynamic urban environment,” *IEEE Internet of Things Journal*, vol. 5, pp. 546–558.

- [156] Y. Chen, J. Yu, X. Su, G.Luo. 2015. "Path planning for multi-UAV formation," *Journal of Intelligent & Robotic Systems*, vol. 77, pp. 229–246.
- [157] L. De Filippis, G. Guglieri, F. Quagliotti. 2012. "Path planning strategies for UAVS in 3D environments," *Journal of Intelligent & Robotic Systems*, vol. 65, pp. 247–264.
- [158] J. Yoon, Y. Jin, N. Batsoyol, H. Lee. 2017, March. "Adaptive path planning of UAVs for delivering delay-sensitive information to ad-hoc nodes," In *Proceedings of IEEE Wireless Communications and Networking Conference (WCNC)*, San Francisco, CA, USA, pp. 247–264.
- [159] S. Hayat, E. Yanmaz, C. Bettstetter, T. X. Brown. 2020. "Multi-objective drone path planning for search and rescue with quality-of-service requirements," *Autonomous Robots*, vol. 44, pp. 1183–1198.
- [160] N. Toorchi, F. Hu, E. S. Bentley, S. Kumar. 2020. "Skeleton-Based Swarm Routing (SSR): Intelligent Smooth Routing for Dynamic UAV Networks," *IEEE Access*, vol. 9, pp. 1286–1303.

Title : Multi-hop routing in a drone network for road surveillance

Keywords : Collaborative networks, Fault-tolerant routing, highway surveillance, link failure recovery, multi-hop routing, UAVs, UAV's path planning, VANETs, WSN.

Abstract : Existing road surveillance systems are either too expensive such as helicopters, due to their high operating and maintenance costs, or inadequate to tackle the dynamic nature of traffic on highways such as roadside traffic radars due to their limited coverage and mobility constraints. Unmanned Aerial Vehicles (UAVs) with excellent maneuvering skills have a promising future for next-generation surveillance systems. UAVs work together in an ad-hoc manner to counter limited energy resources and lower transmission range constraints associated with a single UAV. However, multiple UAVs with dynamic topology suffer frequent link failures and require specific fault-tolerant routing solutions. In addition to this, a robust path planning process is inevitable as the deployment of a UAV fleet in an urban environment with navigation system uncertainties raises safety concerns.

To this end, instead of restricting solely to UAVs, this thesis explores UAV's collaboration with other architectures to come up with robust end-to-end surveillance solutions.

At first, a less overhead bearing routing solution is presented for a fleet of UAVs to enhance backhaul connectivity in the presence of multiple Base Stations (BS). This work is enhanced to seek services of a Vehicular Ad hoc Network (VANET) during the non-availability of a nearby UAV to compensate for link failures. The proposed anticipatory network recovery approach ensures a better end-to-end link continuity by taking care of link failures arising due to the permanent displacement of a UAV.

The second part presents a collaborative work of UAV and Wireless Sensor Network (WSN) to enhance connectivity and surveillance efficiency on a highway. WSN is the backbone of the network and provides routing services to the UAV besides dynamically guiding it to target the best hotspot for catching maximum speed violations. This work follows a cross-layer routing solution to increase WSN lifetime by reducing the chances of collisions along a path.

The last part of the thesis introduces connectivity-aware and collision-free UAV path planning in an urban environment to ensure backhaul connectivity and UAV fleet safety, respectively.