

ISTANBUL TECHNICAL UNIVERSITY ★ GRADUATE SCHOOL

**ARTIFICIAL INTELLIGENCE BASED AND DIGITAL TWIN ENABLED
AERONAUTICAL AD-HOC NETWORK MANAGEMENT**

Ph.D. THESIS

Tuğçe BİLEN

Department of Computer Engineering

Computer Engineering Programme

DECEMBER 2022

ISTANBUL TECHNICAL UNIVERSITY ★ GRADUATE SCHOOL

**ARTIFICIAL INTELLIGENCE BASED AND DIGITAL TWIN ENABLED
AERONAUTICAL AD-HOC NETWORK MANAGEMENT**



Ph.D. THESIS

**Tuğçe BİLEN
(504172508)**

Department of Computer Engineering

Computer Engineering Programme

Thesis Advisor: Prof. Dr. Berk CANBERK

DECEMBER 2022

İSTANBUL TEKNİK ÜNİVERSİTESİ ★ LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ

**YAPAY ZEKA TABANLI VE DİJİTAL İKİZ DESTEKLİ
GEÇİCİ HAVASAL AĞ YÖNETİMİ**

DOKTORA TEZİ

**Tuğçe BİLEN
(504172508)**

Bilgisayar Mühendisliği Anabilim Dalı

Bilgisayar Mühendisliği Programı

Tez Danışmanı: Prof. Dr. Berk CANBERK

ARALIK 2022

Tuğçe BİLEN, a Ph.D. student of ITU Graduate School student ID 504172508 successfully defended the thesis entitled “ARTIFICIAL INTELLIGENCE BASED AND DIGITAL TWIN ENABLED AERONAUTICAL AD-HOC NETWORK MANAGEMENT”, which he/she prepared after fulfilling the requirements specified in the associated legislations, before the jury whose signatures are below.

Thesis Advisor : **Prof. Dr. Berk CANBERK**
Istanbul Technical University

Jury Members : **Prof. Dr. Özgür GÜRBÜZ**
Sabancı University

Assoc. Prof. Dr. Nazım Kemal ÜRE
Istanbul Technical University

Prof. Dr. Sema F. OKTUĞ
Istanbul Technical University

Assoc. Prof. Dr. Muhammed Ali AYDIN
Istanbul University

Date of Submission : **25 November 2022**

Date of Defense : **20 December 2022**





*To my Father and Mother,
Babama ve Anneme,*



FOREWORD

First of all, I would like to thank my advisor, Prof. Dr. Berk CANBERK. I am very lucky to have had the opportunity to work with him during my BSc, MSc, and PhD degrees in the Computer Engineering Department of Istanbul Technical University. I would like to thank him for supporting me with all his patience and sharing all his knowledge and experience from the very beginning of my academic life until I got my PhD degree. I am very grateful to him for teaching me a lot of valuable information about academic life and research.

I would like to express my sincere thanks to my thesis progress committee Prof. Dr. Özgür GÜRBÜZ and Assoc. Prof. Dr. Nazım Kemal ÜRE. They guided me by making very positive and constructive comments in all my thesis progress. Their comments and guidance were very valuable to me. Thanks to them, I was able to evaluate many points in my thesis from different perspectives. I am grateful to them for all their help, support, and kindness.

I would like to thank our Dean Prof. Dr. Sema F. OKTUĞ. She is a very valuable role model for me. I'll never forget her understanding, support, and kindness.

Additionally, I would like to thank our research group, Broadband Communication and Network Automation Research Group, BCRG. I am very happy and proud to be a member of this nice and helpful family.

Lastly and most importantly, I would like to express my special thanks to my lovely family my mother, father, and brothers. None of this would have been possible without the support, trust, and patience of my family. It's all thanks to them, and I'm forever grateful. They always support me in every moment of my life. Thanks to them for being next to me for each difficult time. Their guidance, continuous support, and smiling face as always motivated me well. I dedicate this thesis to my family for their endless love, support, and encouragement.

My father Ruhi was with me when I wrote the above paragraph. He won't be with me when I present the thesis. I know he sees me and he's proud. You are always with me Dad, and you always will be. I love you so much and I will love you forever.

December 2022

Tuğçe BİLEN
(Computer Engineer, M.Sc.)

TABLE OF CONTENTS

	<u>Page</u>
FOREWORD	ix
TABLE OF CONTENTS	xiii
ABBREVIATIONS	xv
SYMBOLS	xvii
LIST OF TABLES	xix
LIST OF FIGURES	xxii
SUMMARY	xxiii
ÖZET	xxvii
1. INTRODUCTION-AANETs FOR IN-FLIGHT CONNECTIVITY	1
1.1 Purpose	1
1.2 Introduction	1
1.2.1 Contributions of the chapter	3
1.2.2 Organization of the chapter	4
1.3 Aeronautical Ad-Hoc Networks (AANETs)	4
1.4 Open Research Challenges for AANETs	8
1.4.1 Data link layer issues	8
1.4.1.1 Link stability requirement in AANET	8
1.4.1.2 Link connectivity requirement in AANET	10
1.4.1.3 Medium access control (MAC) requirement in AANET	11
1.4.2 Network layer issues	13
1.4.2.1 IGW selection	13
1.4.2.2 Aircraft clustering	14
1.4.2.3 Routing management	15
1.4.2.4 Handover management	21
1.4.3 Transport layer issues	22
1.4.4 Additional issues	24
1.4.4.1 Effects of the aircraft antennas on connectivity	24
1.4.4.2 Effects of regulations	25
1.5 Conclusion	26
2. ARTIFICIAL INTELLIGENCE-BASED AANETs	27
2.1 Purpose	27
2.2 Introduction	27
2.2.1 Contributions of the chapter	29
2.2.2 Organization of the chapter	30
2.3 Necessity of AI for AANETs	30
2.4 AI-Driven AANET Management	32
2.4.1 Topology formation management with unsupervised learning	33
2.4.1.1 Clustering-enabled AANET topology formation	33
2.4.1.2 Self-organizing maps-enabled topology formation	35
2.4.2 Topology sustainability management with supervised learning	36
2.4.3 Reinforcement learning-based routing management	38
2.5 Conclusion	40
3. TOPOLOGY FORMATION MANAGEMENT FOR AANETS	43
3.1 Purpose	43
3.2 Introduction	43
3.2.1 Topology formation challenges in AANETs	44
3.2.2 Related work	45
3.2.3 Contributions of the chapter	47
3.2.4 Organization of the chapter	47
3.3 Three-Phased Clustered Topology Formation Managment for AANETs	48
3.3.1 AANET topology	48
3.3.2 Topology formation framework	49

3.3.2.1	Aircraft clustering formation	49
3.3.2.2	A2A link determination	53
3.3.2.3	Cluster head selection	56
3.4	Performance Evaluation	59
3.4.1	Simulation environment	59
3.4.2	Simulation results	60
3.5	Conclusion	66
4.	TOPOLOGY SUSTAINABILITY MANAGEMENT FOR AANETS	67
4.1	Purpose	67
4.2	Introduction	68
4.2.1	Topology sustainability challenges in AANET	68
4.2.2	Related work	70
4.2.3	Contributions of the chapter	71
4.2.4	Organization of the chapter	72
4.3	LVQ-Based Topology Sustainability Management for AANETs	72
4.3.1	AANET topology	72
4.3.2	Topology sustainability framework	74
4.3.2.1	Winning cluster selection	74
4.3.2.2	Intra-cluster link determination	76
4.3.2.3	Attribute update	77
4.4	Performance Evaluation	77
4.4.1	Simulation environment	77
4.4.2	Simulation results	78
4.5	Conclusion	82
5.	TOPOLOGY MAPPING MANAGEMENT FOR AANETS	85
5.1	Purpose	85
5.2	Introduction	85
5.2.1	Topology mapping challenges in AANET	86
5.2.2	Related work	87
5.2.3	Contributions of the chapter	87
5.2.4	Organization of the chapter	88
5.3	GRU-Aided Intra-Cluster Topology Mapping Management for AANETs	88
5.3.1	AANET topology	88
5.3.2	Topology mapping framework	89
5.3.2.1	Forget gate	90
5.3.2.2	Update gate	91
5.4	Performance Evaluation	92
5.4.1	Simulation environment	93
5.4.2	Simulation results	94
5.5	Conclusion	96
6.	ROUTING MANAGEMENT FOR AANETS	97
6.1	Purpose	97
6.2	Introduction	97
6.2.1	Routing management challenges in AANETs	98
6.2.2	Related work	100
6.2.3	Contributions of the chapter	101
6.2.4	Organization of the chapter	102
6.3	Q-Learning (QLR)-Driven Routing Management for AANETs	102
6.3.1	AANET topology	102
6.3.2	Routing management framework	106
6.3.2.1	$(Q_{t-1}(s, a))$ and $\max_{a'} Q(s', a')$ determination	107
6.3.2.2	Dynamic reward $(R(s, a))$ determination	112
6.4	Performance Evaluation	116
6.4.1	Simulation environment	116
6.4.2	Simulation results	119
6.5	Conclusion	125
7.	DIGITAL TWIN FOR AANET MANAGEMENT	127
7.1	Purpose	127
7.2	Introduction	128
7.2.1	Management challenges for AANETs	128
7.2.2	The digital twin evolution for hard-to-follow AANETs	129

7.2.3	Related work	131
7.2.4	Contributions of the chapter	132
7.2.5	Organization of the chapter	133
7.3	Digital Twin-Enabled Management for AANETs (DT-AANET)	133
7.3.1	DT-AANET framework.....	133
7.3.1.1	Physical AANET twin	133
7.3.1.2	Controller	135
7.3.2	Case studies for DT-AANET framework	137
7.3.2.1	Unsupervised learning-based topology formation management ..	137
7.3.2.2	Supervised learning-based topology sustainability management .	140
7.4	Performance Evaluation	142
7.4.1	Simulation environment	142
7.4.2	Simulation results	143
7.5	Conclusion	145
8.	CONCLUSIONS	147
	REFERENCES	149
	CURRICULUM VITAE.....	168





ABBREVIATIONS

3D	: Three Dimensional
A2A	: Air-to-Air
A2G	: Air-to-Ground
AANETs	: Aeronautical Ad Hoc Networks
AeroMRP	: Aeronautical Multipath Reliable Protocol
AI	: Artificial Intelligence
ARPAM	: Ad-hoc Routing Protocol for Aeronautical Mobile Ad-Hoc Networks
ADS-B	: Automatic Dependent Surveillance-Broadcast
ADT	: Azure Digital Twin
DDVC	: Doppler Velocity Clustering
DLDC	: Dynamic Link Duration Clustering
DMDR	: Delay aware Multipath Doppler Routing
DT	: Digital Twin
DT-AANET	: Digital Twin-Enabled AANET Topology Management
DTDL	: Digital Twins Definition Language
FANETs	: Flying ad-hoc networks
AeroMTP	: Fountain Code-Based Multipath Transport Protocol
GPS	: Global Positioning System
GLSR	: Geographic Load Share Routing
GPSR	: Greedy Perimeter Stateless Routing
GRAA	: Geographic Routing Protocol for Aircraft Ad Hoc Network
GRU	: Gated Recurrent Unit
GS	: Ground Station
HSRP	: Hierarchical Space Routing Protocol
IFC	: In-Flight Connectivity
IGW	: Internet Gateway
L-DACS	: L-band Digital Aeronautical Communication System
LCC	: Lease Cluster Change
LHMM	: Layered Hidden Markov Model
LOS	: Line-of-Sight
LVQ	: Learning Vector Quantization
MAC	: Medium Access Control
MST	: Minimum Spanning Tree
NMI	: Nautical Mile
QLR	: Q-learning
PLAR	: Path Link Availability Routing Protocol
RGR	: Reactive Greedy Reactive
SOMs	: Self-Organizing Maps
TCP	: Transmission Control Protocol
UAVs	: Unmanned Aerial Vehicles



SYMBOLS

α	: Learning rate
Δ	: Cluster Delay Bound
γ	: Discount factor
γ_i	: Arrivals of aircraft i from the directly connected neighbors
λ_j, λ_i	: Mean Arrival Rate of Aircraft j, i
μ_j, μ_i	: Mean Service Rate of Aircraft j, i
ρ_j, ρ_i	: Utilization of Aircraft j, i
σ	: Sigmoid Function in GRU
$\sigma_{a_j}^2, \sigma_{A_i}^2$	: Variance of Inter-Arrival Times of Aircraft j, i
$\sigma_{s_j}^2, \sigma_{S_i}^2$	: Variance of Service Times of Aircraft j, i
θ_l	: Theta Index
$\theta_{l_{Modified}}$	: Theta Index-Modified
a	: Action performed by an aircraft at state s
a'	: Action performed by an aircraft at state s'
A_i	: Arrival time of packets to aircraft i
B	: Packet Bandwidth
C_{0i}^2	: Squared coefficient of variation of the inter-arrival times for the accepted packets from the directly connected neighbors
$C_{a_j}^2, C_{A_i}^2$	: Squared coefficient of variation of arrival times of aircraft j, i
C_{Di}^2	: Squared coefficient of variation between the departures of the aircraft queue
C_l	: Centrality Index
C_{ji}^2	: Squared coefficient of variation of the inter-arrival times for the accepted packets from the other airplanes
C_k	: Aircraft Cluster
$C_{s_j}^2, C_{S_i}^2$	: Squared coefficient of variation of service times of aircraft j, i
d	: Distance between Ground Station and Internet Gateway Aircraft
d_{ij}	: Distance between Aircraft i and j
$\vec{D_{ij}}$	: Relative Position Vector between the Aircraft j and i
d_j	: Degree of Aircraft j
$E(V_S)$	: Expected Value of Virtual Stability Index
F_t	: Forget Gate Definition in GRU
H_0	: Null Hypothesis
H_1	: Alternate Hypothesis
H_c	: Candidate Hidden State in GRU
H_t	: Current Hidden State Vector
$H_{(t-1)}$	: Previous Hidden State Vector
I_t	: Input Vector in GRU
L	: Packet Length
L_{q_j}, L_{q_i}	: Queuing Load of Aircraft j, i

m_i	: Weight Vector for LVQ
$Q_{t-1}(s, a)$	: Current state-action function
R	: Transmission Range of an Airplane
r_{GS}	: Coverage range of Ground Station
$R_{ji}(s, a)$	: Reward function between aircraft j and i
r_{ji}	: Transfer probability of packets from aircraft j to i
$R(s, a)$	: Reward of an action a at state s
S	: Transmission Speed
s	: State of an aircraft
s'	: State coming after from s
S_0, S_1, S_2, k	: Intermediate Stages
$S(i)$	: Euclidean Distance
$s_i(\lambda_i, C_{Ai}^2)$	: Air-to-air link state
T_{ij}	: Total Delay between Aircraft i and j
T_{Lij}	: Longevity of an A2A link between aircraft i and j
T_{prj}	: Processing Delay between Aircraft i and j
T_{pij}	: Propagation Delay between Aircraft i and j
T_{qj}, T_{qi}	: Queuing Delay of Aircraft j, i
T_{rij}	: Transmission Delay between Aircraft i and j
U_t	: Update Gate Definition in GRU
$Var(V_S)$	: Variance of Virtual Stability Index
$\vec{V}_{ij}$	: Relative Velocity Vector between the Aircraft j and i
V_S	: Virtual Stability Index
W_{ij}	: Connectivity Matrix of an Aircraft
W_{IC}	: Weight Value of Input Vector for Candidate State in GRU
W_{IF}	: Weight Value of Input Vector for Forget Gate in GRU
W_{IU}	: Weight Value of Input Vector for Update Gate in GRU
W_{HC}	: Weight Value of Hidden State Vector for Candidate State in GRU
W_{HF}	: Weight Value of Hidden State Vector for Forget Gate in GRU
W_{HU}	: Weight Value of Hidden State Vector for Update Gate in GRU
x_n	: Input Vector in LVQ
Z_{V_S}	: Z-Score of Virtual Stability Index

LIST OF TABLES

	<u>Page</u>
Table 1.1 : Frequency dependent oxygen loss.....	6
Table 1.2 : Attenuation by cloud and rain.	6
Table 1.3 : Routing protocols for AANETs.	18
Table 1.4 : Summary of open research problems.	24
Table 2.1 : AANET related opportunities and challenges of AI-driven methods..	35
Table 6.1 : Existing techniques for AANET routing management.....	100
Table 6.2 : Mapping of QLR components to AANET.	105
Table 6.3 : Elements of QLR-driven AANET routing management.	106
Table 7.1 : Role of digital twin on network management.....	131



LIST OF FIGURES

	<u>Page</u>
Figure 1.1 : Aeronautical network types for IFC.	2
Figure 1.2 : The AANET topology.	5
Figure 1.3 : The pattern of flight.....	7
Figure 1.4 : Data link layer issues in AANET.	9
Figure 1.5 : Network layer issues in AANET.	13
Figure 1.6 : A possible AANET clustering.	15
Figure 1.7 : Basis of AANET routing.	17
Figure 1.8 : Sample handover procedure on AANET.....	21
Figure 1.9 : Transport layer issues in AANET.	23
Figure 2.1 : An exemplary illustration of AANET topology.	28
Figure 2.2 : AI supports and methodologies for AANET-specific characteristics.	31
Figure 2.3 : The root causes chains of AANET management challenges.	31
Figure 2.4 : Q-learning and deep Q-learning for AANETs	40
Figure 3.1 : The topology formation challenges and phases of chapter.	44
Figure 3.2 : The system model of AANET with proposed framework.	48
Figure 3.3 : Topology formation framework and its illustration.	49
Figure 3.4 : G/G/1 queuing model of an aircraft in AANET.	53
Figure 3.5 : Simulation details.	59
Figure 3.6 : Evaluations for the climb phase.	61
Figure 3.7 : Evaluations for the cruise phase.....	63
Figure 3.8 : Evaluations for the descent phase.	65
Figure 4.1 : Topology sustainability challenges in AANET.....	69
Figure 4.2 : AANET topology and system model.	73
Figure 4.3 : The modules and flowchart of topology sustainability framework. ...	75
Figure 4.4 : Simulation results.	80
Figure 4.5 : Late and wrong transfer decision results.....	81
Figure 5.1 : AANET specific characteristics and challenges.....	86
Figure 5.2 : The AANET topology.	89
Figure 5.3 : GRU-aided intra-cluster topology mapping model.....	90
Figure 5.4 : Flow of the GRU-aided intra-cluster topology mapping model.	91
Figure 5.5 : Performance evaluation results.	95
Figure 6.1 : Passenger and income changes by year.	98
Figure 6.2 : AANET routing mechanism.....	99
Figure 6.3 : Routing management challenges in AANET.	100
Figure 6.4 : The AANET topology.	103
Figure 6.5 : Q-learning for AANET routing.	104
Figure 6.6 : Updated-LHMM for AANET routing management.	107

Figure 6.7 : Decomposition-based two parameter characterization-aided queuing model of a sample AANET topology.	109
Figure 6.8 : Illustrations of a sample packet transfer using the updated-LHMM strategy.	112
Figure 6.9 : The effects of α and γ on (a) packet delivery ratio and (b) routing delay.	116
Figure 6.10 : Reward performance of exploration-exploitation phases.	117
Figure 6.11 : Routing Delay performance of AeroRP, GLSR, HSRP, RGR, and QLR-based routing methodologies according to the (a) Velocity of airplanes (b) Number of airplanes (c) Queuing load of airplanes (d) Network traffic intensity	120
Figure 6.12 : Packet loss rate performance of AeroRP, GLSR, HSRP, RGR, and QLR-based routing methodologies according to the (a) Velocity of airplanes (b) Number of airplanes (c) Queuing load of airplanes (d) Network traffic intensity	121
Figure 6.13 : Accuracy performance of AeroRP, GLSR, HSRP, RGR, and QLR-based routing methodologies according to the (a) Simulation time (b) Number of airplanes	122
Figure 6.14 : Network load performance of AeroRP, GLSR, HSRP, RGR, and QLR-based routing methodologies according to the (a) Simulation time (b) Number of airplanes	123
Figure 7.1 : Challenging characteristics of hard-to-follow AANETs and DT Evolution.	129
Figure 7.2 : DT-AANET topology management framework.	134
Figure 7.3 : Comparison of standard K-means and customized K-means.	138
Figure 7.4 : Communication range-related calculations.	139
Figure 7.5 : Summary of LVQ-based topology sustainability.	140
Figure 7.6 : Simulation architecture based Microsoft-ADT.	142
Figure 7.7 : Simulation environment and results.....	144

ARTIFICIAL INTELLIGENCE BASED AND DIGITAL TWIN ENABLED AERONAUTICAL AD-HOC NETWORK MANAGEMENT

SUMMARY

The number of passengers using aircraft has been increasing gradually over the following years. With the increase in the number of passengers, significant changes in their needs have been made. In-flight connectivity (IFC) has become a crucial necessity for passengers with the evolving aeronautical technology. The passengers want to connect to the Internet without interruption regardless of their location and time. The aeronautical networks attract the attention of both industry and academia due to these reasons. Currently, satellite connectivity and air-to-ground (A2G) networks dominate existing IFC solutions. However, the high installation/equipment cost and latency of the satellites reduce their efficiency. Also, the terrestrial deployment of A2G stations reduces the coverage area, especially for remote flights over the ocean.

One of the novel solutions is the Aeronautical Ad-hoc Networks (AANETs) to satisfy the IFC's huge demand by also solving the defects of satellite and A2G connectivities. The AANETs are based on creating air-to-air (A2A) links between airplanes and transmitting packets over these connections to enable IFC. The AANETs dramatically increase the Internet access rates of airplanes by widening the coverage area thanks to these established A2A links.

However, the mobility and atmospheric effects on AANETs increase the A2A link breakages by leading to frequent aircraft replacement and reducing link quality. Accordingly, the mobility and atmospheric effects create the specific characteristics for AANETs. More specifically, the ultra-dynamic link characteristics of high-density airplanes create an unstructured and unstable topology in three-dimensional space for AANETs. To handle these specific characteristics, we first form a more stable, organized, and structured AANET topology. Then, we should continuously enable the sustainability and mapping of this created AANET topology by considering broken A2A links. Finally, we can route the packets over this formed, sustained, and mapped AANET topology. However, the above-explained AANET-specific characteristics restrict the applicability of conventional topology and routing management algorithms to AANET by increasing its complexity. More clearly, the AANET specific characteristics make its management challenging by reducing the packet delivery success of AANET with higher transfer delay.

At that point, artificial intelligence (AI)-based solutions have been adapted to AANET to cope with the high management complexity by providing intelligent frameworks and architectures. Although AI-based management approaches are widely used in terrestrial networks, there is a lack of a comprehensive study that supports AI-based solutions for AANETs. Here, the AI-based AANET can take topology

formation, sustainability, and routing management decisions in an automated fashion by considering its specific characteristics thanks to learning operations. Therefore, AI-based methodologies have an essential role in handling the management complexity of this hard-to-follow AANET environment as they support intelligent management architectures by also overcoming the drawbacks of conventional methodologies. On the other hand, these methodologies can increase the computational complexity of AANETs. At that point, we propose the utilization of the Digital Twin (DT) technology to handle computational complexity issues of AI-based methodologies.

Based on these, in this thesis, we aim to propose an AI-based and DT-enabled management for AANETs. This system mainly consists of four main models as AANET Topology Formation Management, AANET Topology Sustainability Management, AANET Topology Mapping Management, and AANET Routing Management. Here, our first aim is to form a stable, organized, and structured AANET topology. Then, we will enable the sustainability of this formed topology. We also continuously map the formed and sustained AANET topology to airplanes. Finally, the packets of airplanes are routed on this formed, sustained, and mapped AANET topology. We will create these four models with different AI-based methodologies and combine all of them under the DT technology in the final step.

In the Topology Formation Management, we will propose a three-phased topology formation model for AANETs based on unsupervised learning. The main reason for proposing an unsupervised learning-based algorithm is that we have independently located airplanes with unstructured characteristics in AANETs before forming the topology. They could be considered as the unlabeled training data for unsupervised learning. This management model utilizes the spatio-temporal locations of aircraft to create a more stable, organized, and structured AANET topology in the form of clusters. More clearly, the first phase corresponds to the aircraft clustering formation, and here, we aim to increase the AANET stability by creating spatially correlated clusters. The second phase consists of the A2A link determination for reducing the packet transfer delay. Finally, the cluster head selection increases the packet delivery ratio in AANET.

In the Topology Sustainability Management, we will propose a learning vector quantization (LVQ) based topology sustainability model for AANETs based on supervised learning. The main reason for proposing a supervised learning-based algorithm is that we already have an AANET topology before the A2A link breakage, and we can use it in supervised learning for training. Accordingly, we can consider the clusters in AANET topology as a pattern; then, we can find the best matching cluster of an aircraft observing A2A link breakages through pattern classification instead of creating topology continuously. This management model works in three phases: winning cluster selection, intra-cluster link determination, and attribute update to increase the packet delivery ratio with reduced end-to-end latency.

In the Topology Mapping Management, we will propose a gated recurrent unit (GRU) based topology mapping model for AANETs. In topology formation, we create AANET topology in the form of clusters by collecting airplanes having similar features under the same set. In topology sustainability, we sustain the formed clustered-AANET topology with supervised learning. However, these formed and

sustained AANET topologies must be continuously mapped to the clustered airplanes to notify them about the current situation. This procedure could be considered a part of sustainability management. Here, we continuously notify the airplanes with GRU at each timestamp about topological changes. This management model works in two main parts as forget and update gates.

In Routing Management, we propose a q-learning (QLR) based routing management model for AANETs. For this aim, we map the AANET environment to reinforcement learning. Here, the QLR-based management model aims to let the airplanes find their routing path through exploration and exploitation. Accordingly, the routing algorithm can adapt to the dynamic conditions of AANETs. In this management model, we adapt the Bellman Equation to the AANET environment by proposing different methodologies for its related QLR components. Accordingly, this model mainly consists of two main parts current state& maximum state-action determination and dynamic reward determination.

Therefore, we execute the topology formation, sustainability, and routing management modules through unsupervised, supervised, and reinforcement learning-based algorithms. Additionally, we take advantage of neural networks in topology mapping management.

After managing the topology and routing of AANETs with AI-based models, in the DT-enabled AANET management, we will support them with the DT technology. The DT can virtually replicate the physical AANET components through closed-loop feedback in real-time to solve the computational challenges of AI-based methodologies. Therefore, we will introduce the utilization of DT technology for the AANET orchestration and propose a DT-enabled AANET (DT-AANET) management model. This model consists of the Physical AANET Twin and Controller, including the Digital AANET Twin with Operational Module. Here, the Digital AANET Twin virtually represents the physical environment. Also, the operational module executes the implemented AI-based models.

Therefore, in this thesis, we aim to propose an AI-based and DT-enabled management for AANETs. In this management system, we will first aim to propose AI-based methodologies for AANET topology formation, topology sustainability, topology mapping, and routing issues. Then, we will support these AI-based methodologies with DT technology. This proposed complete management model increased the packet delivery success of AANETs with reduced end-to-end latency.



YAPAY ZEKA TABANLI VE DİJİTAL İKİZ DESTEKLİ GEÇİCİ HAVASAL AĞ YÖNETİMİ

ÖZET

Günümüzde hava yolu kullanan yolcu sayısı giderek artmaktadır ve yolcu sayısının artmasıyla birlikte ihtiyaçlarında da önemli değişiklikler olmaktadır. Uçak içi bağlantı (IFC), gelişen havacılık teknolojisi ile birlikte yolcular için çok önemli bir gereklilik haline gelmiştir. Yolcular bulundukları konum ve zamandan bağımsız olarak internete kesintisiz olarak bağlanmak istemektedir. Bu nedenlerden ötürü havacılık ağları hem endüstrinin hem de akademinin ilgisini oldukça çekmektedir. Şu anda, uydu bağlantısı ve havadan karaya (A2G) ağlar, mevcut uçak içi bağlantı çözümleri arasında en ön sıralarda gelmektedir. Ancak uyduların kurulum/ekipman maliyetleri ile gecikme sürelerinin yüksek olması verimliliklerini düşürmektedir. Ayrıca, havadan karaya ağ istasyonlarının karasal alanlarda konuşlandırılması, özellikle okyanus üzerindeki uzak uçuşlar için kapsama alanını azaltmaktadır.

Uydu ve havadan karaya ağların sorunlarını da çözerek uçak içi bağlantının büyük talebini karşılamak için sunulan oldukça yeni çözümlerden biri geçici havasal ağlardır (AANETs). Geçici havasal ağlar, uçak içi bağlantıyı etkinleştirmek için uçaklar arasında havadan havaya (A2A) bağlantılar oluşturmaya ve bu bağlantılar üzerinden paketleri iletmeye dayanmaktadır. Geçici havasal ağlar, kurulan bu havadan havaya bağlantılar sayesinde kapsama alanını genişleterek uçakların internet erişim oranlarını önemli ölçüde artırmaktadır.

Ancak, havasal ağlar üzerindeki hareketlilik ve atmosferik etkiler uçakların sık sık yer değiştirmesine ve bağlantı kalitesinin düşmesine yol açarak uçaklar arasındaki bağlantı kopmalarını artırır. Yani, hareketlilik ve atmosferik etkiler, geçici havasal ağların kendine özgü özelliklerinin ortaya çıkmasına öncülük eder. Bu noktada daha spesifik olarak düşünersek, yüksek yoğunluktaki uçakların oldukça dinamik bağlantı özellikleri, geçici havasal ağlar için üç boyutlu uzayda yapılandırılmamış ve kararsız bir topoloji yaratır. Dolayısıyla geçici havasal ağların kendilerine özgü özelliklerinin üstesinden gelebilmek için önce daha kararlı, organize ve yapılandırılmış bir ağ topolojisi oluşturmamız gerekmektedir. Ardından, uçaklar arasında sıklıkla kopan havadan havaya bağlantıları düşünerek, oluşturulan bu geçici havasal ağ topolojisinin sürekli olarak sürdürülebilirliğini ve uçaklara atanmasını sağlamamız gerekmektedir. Son olarak, paketleri bu oluşturulmuş, sürdürülmüş ve uçaklara atanmış geçici havasal ağ topolojisi üzerinden yönlendirebiliriz. Bununla birlikte, yukarıda açıklanan geçici havasal ağlara özgü özellikler, bu ağların karmaşıklığını artırarak geleneksel topoloji ve yönlendirme yönetimi algoritmalarının uygulanabilirliğini kısıtlar. Daha açık bir ifadeyle, geçici havasal ağlara özgü özellikler, geçici havasal ağların daha yüksek aktarım gecikmesi ile paket teslim başarısını azaltarak yönetimini zorlaştırmaktadır.

Bu noktada, akıllı yapılar ve mimariler sağlayarak yüksek yönetim karmaşıklığı ile başa çıkmak için yapay zeka (AI) tabanlı çözümler geçici havasal ağlara uyarlanabilmektedir. Yapay zeka tabanlı yönetim yaklaşımları karasal ağlarda yaygın olarak kullanılmasına rağmen, geçici havasal ağlar için yapay zeka tabanlı çözümleri destekleyen kapsamlı bir çalışma eksikliği vardır. Bu noktada, yapay zeka tabanlı geçici havasal ağlar, öğrenme işlemleri sayesinde kendine özgü özellikleri göz önünde bulundurarak topoloji oluşturma, sürdürülebilirlik ve yönlendirme yönetimi kararlarını otomatik bir şekilde alabilir. Yani, yapay zeka tabanlı metodolojiler, geleneksel yöntemlerin dezavantajlarının üstesinden gelerek akıllı yönetim mimarilerini destekledikleri için, bu takip edilmesi zor geçici havasal ağ ortamının yönetim karmaşıklığının ele alınmasında önemli bir role sahiptir. Öte yandan, bu metodolojiler geçici havasal ağların hesaplama karmaşıklığını artırabilir. Bu noktada, yapay zeka tabanlı metodolojilerin hesaplama karmaşıklığı sorunlarını ele almak için dijital ikiz (DT) teknolojisinin kullanılmasını öneriyoruz.

Bunlara dayanarak, bu tezde, geçici havasal ağlar için yapay zeka tabanlı ve dijital ikiz destekli bir yönetim önermeyi amaçlıyoruz. Bu sistem temel olarak geçici havasal ağ Topoloji Oluşturma Yönetimi, Topoloji Sürdürülebilirlik Yönetimi, Topoloji Atama Yönetimi ve Yönlendirme Yönetimi olmak üzere dört ana modelden oluşmaktadır. Burada ilk amacımız kararlı, organize ve yapılandırılmış bir geçici havasal ağ topolojisi oluşturmaktır. Ardından oluşan bu topolojinin sürdürülebilirliğini sağlayacağız. Ayrıca oluşturulan ve sürdürülen geçici havasal ağ topolojisini sürekli olarak uçaklara atayacağız. Son olarak, uçak paketlerini bu oluşturulan, sürdürülen ve sürekli olarak uçaklara atanan geçici havasal ağ topolojisi üzerinde yönlendireceğiz. Bu dört modeli farklı yapay zeka tabanlı metodolojilerle oluşturacağız ve son adımda hepsini dijital ikiz teknolojisi altında birleştireceğiz.

Topoloji Oluşturma Yönetiminde, geçici havasal ağlar için denetimsiz öğrenmeye dayalı üç aşamalı bir topoloji oluşturma modeli önereceğiz. Denetimsiz öğrenme tabanlı bir algoritma önermemizin temel nedeni, topolojiyi oluşturmadan önce geçici havasal ağların bağımsız olarak konumlanmış ve yapılandırılmamış özelliklere sahip uçaklardan oluşmasıdır. Bu uçaklar, denetimsiz öğrenme için etiketlenmemiş eğitim verileri olarak kabul edilebilirler. Bu yönetim modeli, kümeler şeklinde daha kararlı, organize ve yapılandırılmış bir geçici havasal ağ topolojisi oluşturmak için uçağın uzaysal-zamansal konumlarını kullanır. Daha açık olarak, ilk aşama uçak kümeleme oluşumuna karşılık gelir ve burada mekansal olarak ilişkili kümeler oluşturularak geçici havasal ağ kararlılığının artırılması hedeflenmektedir. İkinci aşama, paket aktarım gecikmesini azaltmak için havadan havaya bağlantı belirlemesinden oluşur. Son olarak, küme başı seçimi, geçici havasal ağların paket teslim oranını artırır.

Topoloji Sürdürülebilirlik Yönetiminde, geçici havasal ağlar için denetimli öğrenmeye dayalı vektör niceleme öğrenme (LVQ) tabanlı bir topoloji sürdürülebilirlik modeli önereceğiz. Denetimli öğrenme tabanlı bir algoritma önermemizin temel nedeni, havadan havaya bağlantıların kopmasından önce zaten bir geçici havasal ağ topolojimizin olması ve bunu eğitim için denetimli öğrenmede kullanabilmemizdir. Buna göre geçici havasal ağ topolojisindeki kümeleri bir şablon olarak ele alabiliriz; daha sonra, sürekli olarak topoloji oluşturmak yerine, şablon sınıflandırması yoluyla havadan havaya bağlantı kopmalarına sahip bir uçağın en uygun kümesini bulabiliriz.

Bu yönetim modeli uçtan uca azaltılmış gecikme ile paket teslim oranını artırmak için üç aşamada çalışmaktadır. Bu aşamalar, kazanan küme seçimi, küme içi bağlantı belirleme ve öznitelik güncelleme şeklindedir.

Topoloji Atama Yönetiminde, geçici havasal ağlar için kapılı tekrarlayan hücreler (GRU) tabanlı bir topoloji atama modeli önereceğiz. Topoloji oluşumunda benzer özelliklere sahip uçakları aynı küme altında toplayarak kümeler şeklinde bir havasal ağ topolojisi oluşturuyoruz. Topoloji sürdürülebilirliğinde, oluşturulan kümelenmiş-geçici havasal ağ topolojisini denetimli öğrenme ile sürdürüyoruz. Ancak, bu oluşturulan ve sürdürülen havasal ağ topolojileri, mevcut bağlantılar hakkında bilgilendirme sağlanması amacıyla kümelenmiş uçaklara sürekli olarak atanmalıdır. Bu prosedür, sürdürülebilirlik yönetiminin bir parçası olarak düşünülebilir. Bu yönetim ile topolojik değişiklikler hakkında uçakları kapılı tekrarlayan hücreler ile her zaman biriminde sürekli olarak bilgilendiriyoruz. Bu yönetim modeli, unutma ve güncelleme kapıları olmak üzere iki ana bölümde çalışmaktadır.

Yönlendirme Yönetiminde, geçici havasal ağlar için q-öğrenme (QLR) tabanlı bir yönlendirme yönetimi modeli öneriyoruz. Bu amaçla, geçici havasal ağ ortamını pekiştirici öğrenme ile eşleştiriyoruz. Burada, q-öğrenme tabanlı yönetim modeli, uçakların keşif ve kullanım yoluyla rotalarını bulmalarına izin vermeyi amaçlar. Yani, bu yönlendirme algoritması, geçici havasal ağların dinamik koşullarına uyum sağlayabilir. Bu yönetim modelinde, ilgili q-öğrenme bileşenleri için farklı metodolojiler önererek Bellman Denklemine geçici havasal ağ ortamına uyarlıyoruz. Buna göre, bu model temel olarak mevcut durum & maksimum durum-eylem belirleme ve dinamik ödül belirleme olmak üzere iki ana bölümden oluşmaktadır.

Yani topoloji oluşturma, sürdürülebilirlik ve yönlendirme yönetimi modüllerini denetimsiz, denetimli ve pekiştirmeli öğrenme tabanlı algoritmalar aracılığıyla yürütüyoruz. Ek olarak, topoloji atama yönetiminde sinir ağlarından yararlanıyoruz.

Yapay zeka tabanlı modellerle geçici havasal ağların topolojisini ve yönlendirmesini yönettikten sonra, dijital ikiz destekli havasal ağ yönetiminde bu modelleri dijital ikiz teknolojisi ile destekleyeceğiz. Dijital ikiz, yapay zeka tabanlı metodolojilerin hesaplama zorluklarını çözmek için gerçek zamanlı olarak kapalı döngü geri bildirim yoluyla fiziksel ağ bileşenlerini sanal olarak çoğaltabilir. Bu nedenle, geçici havasal ağ yönetimi için dijital ikiz teknolojisinin kullanımına giriş yapacağız ve dijital ikiz etkin bir AANET (DT-AANET) yönetim modeli önereceğiz. Bu model, fiziksel geçici havasal ağ ikizi ve operasyonel modülden oluşmaktadır. Ek olarak, operasyonel modül dijital geçici havasal ağ ikizi ve denetleyiciden oluşmaktadır. Burada dijital geçici havasal ağ ikizi, fiziksel ortamı sanal olarak temsil eder. Ayrıca operasyonel modül, önerilen yapay zeka tabanlı modelleri yürütür.

Özetle, bu tezde, geçici havasal ağlar için yapay zeka tabanlı ve dijital ikiz destekli bir yönetim önermeyi amaçlıyoruz. Bu yönetim sisteminde öncelikle geçici havasal ağ topoloji oluşumu, topoloji sürdürülebilirliği, topoloji ataması ve yönlendirme sorunları için yapay zeka tabanlı metodolojiler önereceğiz. Ardından, bu yapay zeka tabanlı metodolojileri dijital ikiz teknolojisi ile destekleyeceğiz. Önerilen tüm bu yönetim modeli ile, uçtan uca azaltılmış gecikme süresiyle geçici havasal ağların paket teslim başarısını artıracaktır.



1. INTRODUCTION-AANETs FOR IN-FLIGHT CONNECTIVITY

1.1 Purpose

The aeronautical networks attract the attention of both industry and academia since Internet access during flights turns to the crucial demand from luxury with the evolving technology. This in-flight connectivity (IFC) necessity is currently dominated by satellite connectivity and air to ground (A2G) network solutions. However, the high installation/equipment cost and latency of the satellite connectivity reduce its efficiency. The A2G networks are utilized through the ground stations deployed in terrestrial areas to solve these satellites' problems. This terrestrial deployment reduces the coverage area of A2G networks, especially for remote flights over the ocean. The aeronautical ad hoc networks (AANETs) are designed to provide IFC while solving the primary defects of dominating solutions. The AANET is an entirely novel solution under the vehicular networks since it consists of airplanes with ultra-dynamic and unstructured characteristics. These specific characteristics cause special challenges for AANETs when combined with the environmental and mobility effects. This chapter presents a holistic review of these open AANET challenges by investigating them in data link, network, and transport layers. Before giving the details of these challenges, this chapter explores the main features of AANETs for IFC. Therefore, we provide our specific interest in the AANET by investigating its particular features and open research challenges in this chapter.

1.2 Introduction

The number of passengers using aircraft increases gradually over the following years. International Air Transport Association (IATA) estimates that there will be 8.2 billion aircraft passengers in 2037 [1]. With the increase in the number of passengers, significant changes in their needs have been made. The passengers want to connect to the Internet without interruption regardless of their location and time. Accordingly,

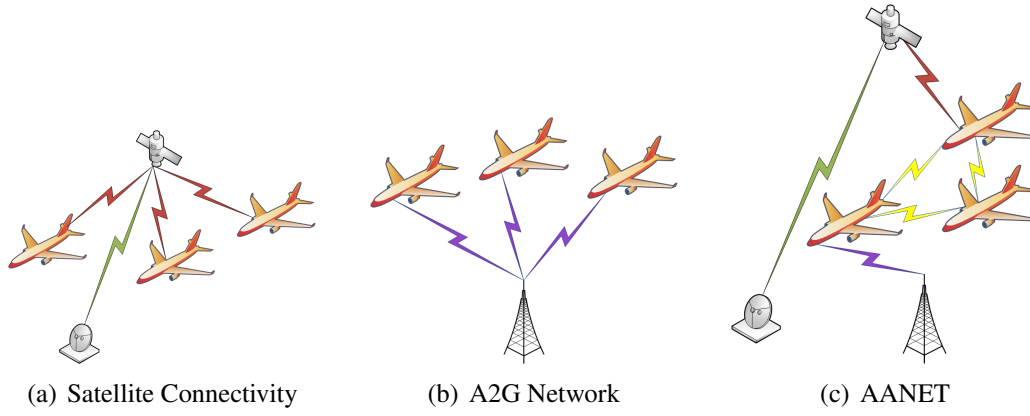


Figure 1.1 : Aeronautical network types for IFC.

passengers want to reach real-time Internet browsing, text messaging, live television, online gaming, and e-mailing during a flight [2]. This situation shows that IFC has become an essential requirement for passengers during a flight [3], [4]. More specifically, IFC is a critical selection criteria for roughly 54% of passengers, and they agree to pay extra fees for this service [5]. Also, approximately 75% of passengers are ready to change the airline to get faster and uninterrupted Internet access, while 20% have changed the airline they use [6]. As a result, recently, IFC became a critical income source for the airlines [7]. According to a market report released in 2016, the total revenue obtained from IFC is expected to increase from \$700 million in 2015 to nearly \$5.4 billion by 2025 with a 23% CAGR over the ten years [8]. The number of aircraft that provide this service needs to increase to enable this income [9]. Also, the number of commercial flights is expected to grow from 5300 in 2015 to 23100 in 2025 [10]. More generally, it is expected that IFC will create 130\$ billion global markets up to 2035 [11].

Technological advances have made IFC an essential part of the aviation domain [12]. The key figures in the previous paragraph show the importance and popularity of IFC in aviation. This evolving interest leads to more funding and research in IFC by taking the industry and academia's attention. As a result, many publications, products, and projects are in the literature to provide and develop different IFC solutions. These works in literature divide the aeronautical networks into three categories of satellite connectivity, A2G network, and AANETs as shown in Figure 1.1.

Satellite communication is the oldest method for IFC, and it is also used for aviation control services, which ensure the safety of aircraft [13]. Although the coverage area of the satellite communication is large, the delay and high cost become the main problems for Internet access during the flight. Many ground stations specialized in aeronautical communication are used to provide cellular network service and solve the problems observed in satellite communication. With this A2G network, airplanes can connect to base stations placed in the ground and provide Internet access to their users [14]. However, the ground base implementation of base stations causes coverage issues during remote flights.

To solve the problems of the satellite and A2G networks, by establishing connections between the airplanes, a temporary air network has been proposed as a new effective technique called AANET. The AANET operates on the principle that one aircraft receives packets from another connected aircraft and routes them to a destination. Due to this ultra-dynamic architecture, the AANET is different from other vehicular networks. More specifically, the AANET experiences distinct challenges in data link, network, and transport layers due to its specific topology and characteristics. Correspondingly, in this chapter, we give our particular interest to the AANET by explaining its main features, topology, and open research challenges based on these layers.

1.2.1 Contributions of the chapter

As explained in the above, in this chapter, we focus on the AANET concept by explaining its main features, topological details, and open research problems. More specifically, we can summarize the main contributions of this chapter as follows:

- Exhaustive analysis of AANET technology
- Discuss open research challenges about AANET
- Present a holistic layered review on open research challenges

1.2.2 Organization of the chapter

The rest of the chapter is organized as follows: We start to explain the AANET topology. Then, we examine the effects of environment and mobility on AANET. The open research challenges for AANETs are analyzed after these according to a layered concept. Here, we investigate these challenges for the data link, network, and transport layers. Finally, we conclude the chapter by also giving the general concept of the next chapter.

1.3 Aeronautical Ad-Hoc Networks (AANETs)

The AANETs are created by establishing air-to-air (A2A) links between the aircraft in the sky without relying on a central node or entity [15]. The packets of a source aircraft are routed through these links until reaching the destination aircraft having Internet connectivity. Accordingly, A2A links are the crucial components for the AANETs. Generally, these links have line-of-sight (LOS) characteristics by utilizing ultra high frequency/very high frequency (U/VHF) band with 119-137 MHz spectrum, relatively high signal to noise ratio, and unrestricted battery power [16], [17]. Generally, the establishment of A2A links is executed based on the communication range between airplanes [18]. If the distance between two airplanes is smaller than the transmission range, then the A2A link is established among these planes based on omnidirectional transmission [19]. During the packet routing through these A2A links, each aircraft becomes a router in AANET. Also, the destination aircraft has an Internet connection via satellite or A2G connectivity. Accordingly, the advantages of both satellite connectivity and the A2G network are combined under the AANET structure. Therefore, the coverage problem of the A2G network is solved by the AANETs as they extend the coverage area of an A2G network by enabling Internet access to the aircraft, which cannot directly access the A2G infrastructure.

As shown in Figure 6.4, the AANETs have a three-layered topology [20]. In this layered topology, the top, middle, and bottom layers correspond to the satellite, aircraft, and ground layers, respectively. Each layer could interact with others using inter-layer links [21]. The satellite layer connects to the aircraft and ground layers through the satellite-to-air and satellite-to-ground links. Similarly, the A2G links are used to connect the aircraft to the A2G base stations [22]. Also, the A2A links

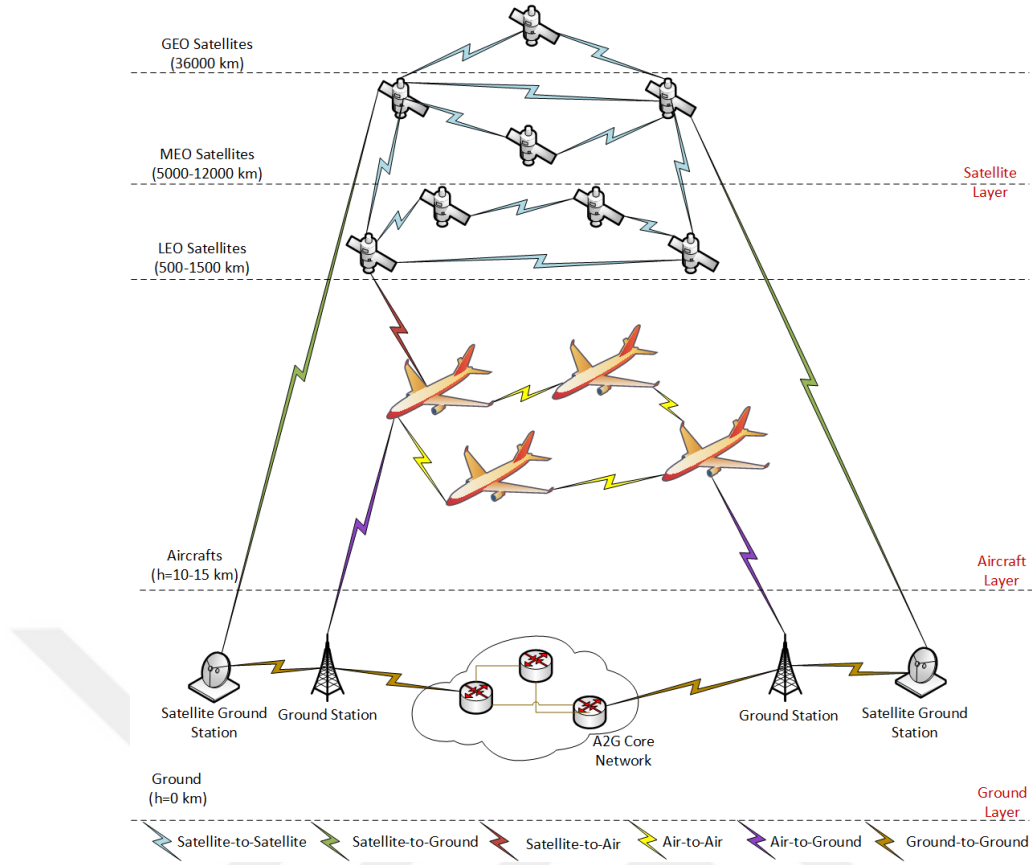


Figure 1.2 : The AANET topology.

are established between the airplanes to create an AANET in the aircraft layer, as explained paragraph above. The AANETs have an ultra-dynamic and unstructured ad-hoc topology with easily broken A2A links. The topological characteristics of AANETs lead to some research challenges, and we investigate these challenges from a layered aspect in Section 1.4. At first, we will investigate the effects of environment and mobility on AANET in the following subsection since these are the main reasons for AANET specific characteristics. We will also investigate AANET specific characteristics in Chapter 2.

Effects of environment and mobility on AANET

As explained above, the propagation in AANET has the LOS characteristic, and it could be modeled as free space loss. But, the propagation effects should be included in the design of these LOS systems [23]. Here, we investigate the attenuations due to atmospheric gases and hydrometeors. We consider that the oxygen absorption, rain, and cloud attenuations could be added to the free space path loss model of AANETs.

Table 1.1 : Frequency dependent oxygen loss.

Frequency (GHz)	0-52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67
Loss (dB/km)	0	1	2.2	4	6.6	9.7	12.6	14.6	15	14.6	14.3	10.5	6.8	3.9	1.9	1

The oxygen absorption should be regarded due to the significant propagation distance of aeronautical networks. The oxygen absorption loss model and the frequency-dependent oxygen loss values for the A2G networks are summarized in Table 1.1 [24]. We claim that these values could also be utilized for AANETs due to the free space loss model. The rainfall and atmospheric gaseous cause absorption and scattering for frequencies above 5 GHz. This situation increases the transmission losses by leading to high channel error rates [25], [26]. Additionally, the loss of signal strength and transmission power could be listed as the main attenuated factor caused by the rain. More specifically, in AANET, the A2A links between the aircraft could be easily broken due to the rain attenuation effect. This situation causes quick topology change, as observed in the mobility effect case. Therefore, the rain attenuation should be included in the propagation model for the frequencies above 5 GHz, as shown in Table 1.2. The water or ice particles in clouds also cause attenuation of transmitted signals through the A2A links between the aircraft. This effect becomes more important for the higher frequencies, as shown in Table 1.2 [27], [28]. One possible solution to reduce this effect is the optical link utilization during the en-route phase since clouds drop their performance in lower altitudes. The cloud and rain attenuations for different frequencies are shown in Table 1.2.

Table 1.2 : Attenuation by cloud and rain.

Frequency f(GHz)	Cloud Attenuation [dB]	Rain Attenuation [dB]
5	0.023	0.0313
10.7	0.106	0.249
15.4	0.217	0.528
23.8	0.507	1.114
31.4	0.859	1.574
90	4.74	3.17

As shown in Figure 4.3, the flight pattern of aircraft is modeled in seven phases taxiing, takeoff, climb, cruise/en-route, descent, approach, and landing [29]. During these phases, the airplanes fly at different altitudes, and these altitudes affect the aircraft modeling [30]. More generally, the aircraft fly in the lower altitudes of the stratosphere. Accordingly, the propagation in an AANET has LOS characteristics. It could be modeled as free space loss or linear uniform motion during cruise/en-route since buildings or objects could not block the links. In addition to these general models, the Poisson process and Improved Semi-Markov Smooth Mobility Model could be used for aircraft modeling [31], [32]. Here, the Poisson process is generally used during the takeoff and approach phases since we have an exact random timing for events with the known average time between events. Also, the Improved Semi-Markov Smooth Mobility Model aims to simulate aircraft mobility based on the physical law of airplane motion. For this reason, it could be used during the seven phases of aircraft mobility. The mobility characteristics of aircraft could be considered pseudo linear in AANET. This means that the aircraft can move with a relatively linear path without changing direction and motion parameters. Therefore, the nodes in AANET generally have a regular and predictable movement. However, the ultra-high speeds of aircraft limit this predictable movement, and time-varying link characteristics lead to rapidly changing network topology. Accordingly, the links between the aircraft could be quickly established and broken. Moreover, the frequent reorganization of the network also complicates the regular monitoring of the network. For these reasons, ultra-high-speed

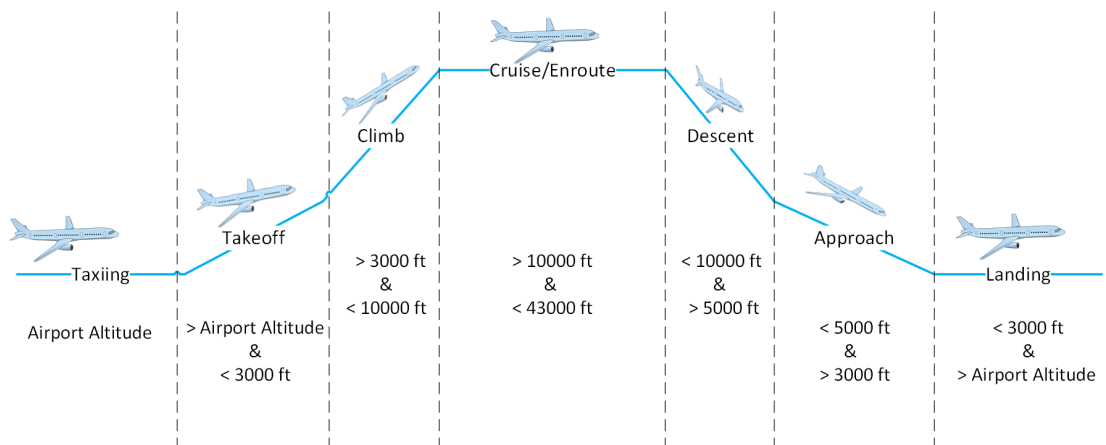


Figure 1.3 : The pattern of flight.

and three dimensional (3D) movement characteristics should be considered to establish more durable A2A links between the aircraft. The sustainability of the AANET topology is increased with these durable A2A links. Accordingly, the AANETs can observe fewer packet losses and drops with higher transmission success.

As explained in this part, the effects of environment and mobility cause specific characteristics for AANETs. Also, these specific characteristics lead different research challenges for AANETs. As explained above, we will investigate these AANET specific characteristics in Chapter 2. Before them, in the sequel part, we will investigate main research challenges by evaluating the state-of-the-art from a layered aspect.

1.4 Open Research Challenges for AANETs

As explained above, AANETs experience various research challenges in the Data Link, Network, and Transport layers. In this section, we explore these challenges by investigating the solutions in the state-of-the-art.

1.4.1 Data link layer issues

We investigate the data link layer issues in three parts as Link Stability, Link Connectivity, and Medium Access Control (MAC) requirements in AANET as shown in Figure 1.4. Here, the Link Stability and Connectivity Requirements are related to the logical link control (LLC) of layer two as explained follows:

1.4.1.1 Link stability requirement in AANET

The AANETs have a highly dynamic topology caused by ultra-high speeds of aircraft. This situation makes the AANET environment challenging for the data link layer since it is hard to manage the A2A and A2G links in these dynamic conditions. Additionally, the atmospheric effects can reduce the quality of these links through oxygen absorption, rain, and cloud attenuation. Therefore, one of the main aims in the data link layer is to enable link stability under these conditions. Here, the main effects on link stability in AANET could be listed as follows:

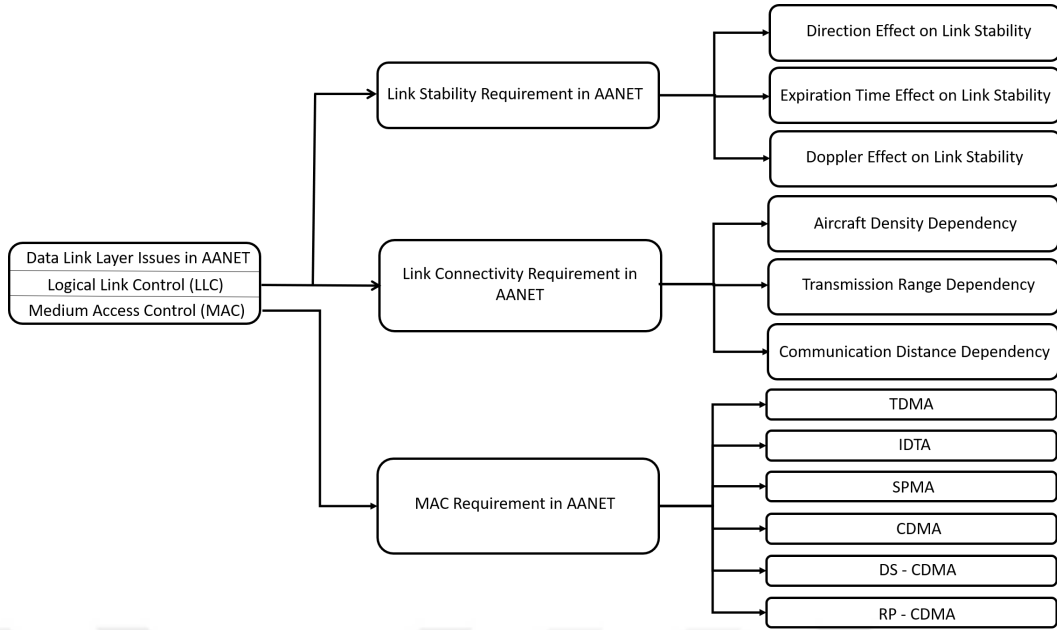


Figure 1.4 : Data link layer issues in AANET.

- *Direction Effect on Link Stability:* The links could be established between the aircraft flying in the same direction to enable link stability. The main reason for this consideration is that the links between the aircraft moving to the opposite directions are unreliable due to the Doppler shift effect. More clearly, the links between the aircraft in the same direction are likely to last longer than in opposite directions. Another possible solution is a two-phase transmission scheduling scheme. In the first phase, the horizontal transmission is executed between the aircraft at the same height until the packets reach the nearest plane to the destination. Then, the relay aircraft realizes the vertical transmission to the destination. In this way, the links are more efficiently utilized for packet transmission. Similarly, the available link probability could be detected to control the topology [33].
- *Expiration Time Effect on Link Stability:* The link and path expiration times are estimated based on the relative velocity and position of aircraft to increase the AANET link stability in [34]. Here, the relative velocities could be found using the global positioning system (GPS) and power or Doppler shift. Here, the relative movement of the transmitter and receiver leads to an apparent change in the frequency of transmitted electromagnetic signals, and this effect is called a

Doppler shift. The Doppler shift as a link stability metric is more efficient since the atmosphere and rain can attenuate the radio signals in GPS and power methods.

- *Doppler Effect on Link Stability:* The Doppler shift could be calculated by comparing the expected and received radio signal frequencies between the aircraft and satellites, and the Doppler shift of control packets could be utilized to calculate the link duration [35]. The aircraft can remain within the LOS of other aircraft if the positive Doppler values are calculated. Also, the stability of the link is increased with the smaller Doppler value. These situations will provide a more persistent connection between the aircraft by also reducing the total handover number.

Based on these main effects on link stability, we consider that the direction, expiration time, and Doppler Effect should be regarded during the formation of AANETs. More clearly, we should establish the A2A links between aircraft with similar flight characteristics on the same movement route to disable the direction effect with higher expiration time and more negligible Doppler Effect.

1.4.1.2 Link connectivity requirement in AANET

The effects of the environment and mobility reduce the link connectivity in AANET. The reduced connectivity leads to packet losses by increasing breakage rates [36]. On the other hand, we can reduce the packet losses by establishing more durable links between airplanes. At this point, the link connectivity depends on the *aircraft density* on flight path, *transmission range*, and *distance* between two airplanes detailed as follows [37]:

- *Aircraft Density Dependency:* The communication range for network connectivity decreases with increasing aircraft density, according to work in [38]. The Bernoulli experiment could be used to find the relationship between the node density and the probability of forming a network in AANETs. According to the Bernoulli and Poisson estimations, the likelihood of creating an AANET increases with the growing aircraft density. Also, the connectivity of AANET is restored by the movement of the relay nodes in [39]. They utilize an online optimization approach to control the activities of the relay nodes.

- *Transmission Range Dependency*: The necessary and sufficient transmission range to show the connectivity requirement in AANET is defined in [40]. The necessary transmission range is used to indicate the conditions based on the disconnection probability of AANET. In contrast, the sufficient transmission range represents the conditions to obtain a connected AANET with one possibility. The necessary and sufficient transmission ranges are defined as a function of the aircraft density, flight path length, and airspace separation. Here, the airspace separation divides the airspace into multiple height levels to define 2-dimensional AANETs. Additionally, the hop count between the nodes could affect the transmission range. If the communications are executed based on the single hop model, then the source node can directly communicate with the destination [41].
- *Communication Distance Dependency*: Communication distance plays a vital role in link connectivity. Generally, this distance is determined according to the Earth's curvature and the aircraft's flight level. The aircraft at low flight levels experience higher connectivity by creating longer communication distances. If we want to determine the stable number for communication distance, the A2A communication range could be taken as the 450 nautical mile (nmi) at 35000ft [42]. Also, the maximum distance between the aircraft is defined as 444 nautical miles at 32808ft [43], [44].

The AANETs could be more easily created in areas with higher aircraft density. The A2A link distances decrease with the increasing number of airplanes in these areas. This means that the dependency on the transmission range is satisfied by most airplanes, and accordingly, more long-lasting connections are established between them.

1.4.1.3 Medium access control (MAC) requirement in AANET

The MAC protocols in the terrestrial links could not be effectively utilized to satisfy the requirements of AANETs [45]. However, the management and allocation of these links are crucial factors to control the network performance for aeronautical networks [46]. One of the common considerations is to utilize the carrier sense multiple access

(CSMA) techniques in the AANET, but the high traffic load on the network increases the collision probability and delay. More specifically, the high number of waited packets in the aircraft queues grows delay and losses. For this reason, the single time division multiple access (TDMA) channel is aimed to use with AANET instead of CSMA in [47], [48]. The interference-based distributed TDMA algorithm (IDTA) is also proposed for AANET to diminish the problems of basic TDMA [49]. It can run both the sender and receiver of a link to reduce the computational load of the receiver node, but it observes the delay problems. As a solution to delay related problems, the statistical priority multiple access (SPMA) protocol is proposed in [50]. This priority access technique works based on the statistics of channel occupancy, and here, the congestion degree of the channel is compared with the channel accessing threshold of packets. This method reduces the waiting time observed for taking the channel control compared to the CSMA. Also, during a conflict, the nodes wait a random amount of time before re-transmission. Additionally, to solve the capacity-related problems, the code division multiple access (CDMA) could be utilized because of the higher capacity advantage [51]. This work also considers that the TDMA and also frequency division multiple access (FDMA) is inefficient due to the high load caused by high aircraft numbers. Clearly, the FDMA and TDMA are not efficient for the AANET due to bandwidth reduction and clock synchronization problems. Similar to this work, the CDMA is used to enable simultaneous communication opportunity to the aircraft [52]. As an extension to the CDMA, the direct sequence CDMA (DS-CDMA) is proposed as an allocation method in work [53]. The main reason for selecting the DS-CDMA is that it can allow multiple simultaneous transmissions without coordination among nodes. Similarly, the random packet code division multiple access (RP-CDMA) could be utilized by assigning randomly selected spreading codes to each transmission [54].

We can state that the AANETs do not utilize the terrestrial-based MAC protocols due to the ultra-dynamic and unstructured characteristics. Accordingly, we should design a new MAC protocol for AANET by considering its specific features. Generally, the designed protocol should work in an ad-hoc manner without a central entity or coordination according to AANET topology. Additionally, the design phase of the

MAC protocol should include the high packet load on the network and collision probability between aircraft.

1.4.2 Network layer issues

In this chapter, we investigate the network layer issues in four parts Internet gateway (IGW) selection, aircraft clustering, routing management, and handover management, as shown in Figure 1.5. The details of these issues could be explained in the following subsections:

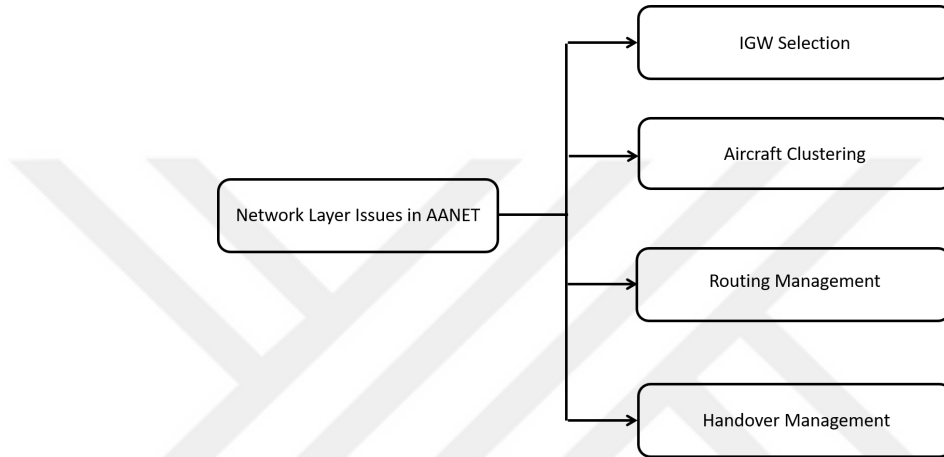


Figure 1.5 : Network layer issues in AANET.

1.4.2.1 IGW selection

The IGWs are deployed to connect the AANET to the Internet. Therefore, all packets are transferred through the IGWs during the operation of AANET, and the congestion probability is higher than other nodes. For these reasons, the IGW selection is essential for the success of the AANETs. This selection generally could be executed proactively or reactively. In the proactive gateway discovery, the gateways periodically send advertisement messages to announce their presence to a network. The node transfers gateway solicitation messages to the network for taking gateway advertisements in the reactive gateway discovery approach. Also, the IGW selection based on the distance in terms of hop count constitutes one of the possible methodology [55]. Here, the packet delays potentially increase with the distance between the mobile node and a gateway. The IGW selection based on the defined utilization metric is proposed as another possible method [56]. Here, the traffic carried by a gateway is divided by the

wireless interface capacity as a utilization metric. Accordingly, the traffic loads are shared between the IGWs by also maximizing connectivity [57].

Additionally, the delay-based IGW selection scheme could be utilized to increase the packet delivery ratio and fairness by decreasing the average packet delay. This delay is affected by the traffic on a path and gateway. This delay could be estimated by putting a time stamp on the gateway advertisement messages. Also, the variance between the delays of these successive gateway advertisements could be used as an IGW selection parameter [58].

The IGWs are the most loaded nodes on the AANET topology since it is an Internet connection point of the whole topology. For this reason, the load and connection density metrics should be considered during the IGW selection. Additionally, it should be reasonably close to the Internet access point due to the same reason. At that point, the shortest path or topological spanning algorithms can help determine the IGW according to its location. Here, the load and location-based methodologies could also be combined as a hybrid selection methodology.

1.4.2.2 Aircraft clustering

The main aim of the clustering is to collect the aircraft having similar direction, velocity, angle, and mobility attributes in a common set. Accordingly, we can obtain a stationary AANET topology with long-lasting A2A links. More specifically, the links between the aircraft having more similar attributes could be maintained for a long time and this situation increases the AANET topology's sustainability. Therefore, a multi-dimensional clustering model could be utilized for aircraft clustering by taking the position, velocity, or mobility as separate dimensions as shown in Figure 1.6 [59]. Two different clustering algorithms as dynamic Doppler velocity clustering (DDVC) and dynamic link duration clustering (DLDC) are proposed in [60].

In DDVC, the main clustering metric is the relative velocity between the nodes obtained through the Doppler value of packets. Accordingly, the DDVC is utilized if the position or velocity information could not be obtained directly. On the other hand, the DLDC utilizes the link expiration time estimated by using the position and velocity



Figure 1.6 : A possible AANET clustering.

parameters. Similarly, the 1-hop clustering algorithm is proposed based on three steps in work [61]. Here, the first step is the neighbor discovery with periodic hello messages that include current position and speed information. Then, the cluster head is determined by taking advantage of neighbor numbers and relative speed parameters. Finally, the stability structure clustering algorithm is used for merging the clusters if the two cluster heads move to the communication range of each other. Additionally, the honeycomb division-based clustering algorithm is proposed in [62]. In this algorithm, the whole area is divided into hexagonal regions, and a cluster consists of a different number of hexagons. Accordingly, the cluster head is selected from these hexagons and the spanning tree algorithms are used to prevent the overlaps between the hexagons. Also, the Doppler velocity clustering could be used, and in this clustering method, the backbone aircraft is selected as a cluster head. Here, the cluster head sends a beacon to the neighbors and checks the Doppler shift of the beacon replies to determine the aircraft to be connected. The aircraft can also become a member of different clusters at the same time, and it can enable communication between these clusters.

1.4.2.3 Routing management

In addition to the above issues, the specific characteristics of AANETs should be considered during the design of the routing algorithms [63]. For this aim, the Multipath Doppler Routing (MUDOR) is proposed, which considers the mobility and link duration as routing parameters [64]. The main aim of the MUDOR is to find a more stable path to transfer the data to the destination. Here, the Doppler value is used for estimating the quality and stability of routes as shown in Table 1.3. Also, thanks to the

multipath characteristics, the remote cluster or aircraft could participate in the routing procedure. In the Geographic Load Share Routing (GLSR) algorithm, the packets are forwarded to the geographically closest neighbor of the destination. Here, greedy forwarding is utilized to choose the best neighbor, which maximizes the advance of a packet. During this routing, queuing delay and link congestion probability are also aimed to reduce by enabling load sharing among the neighbors. The main reason for this is that the transferring packets can wait in the relaying aircraft queues, which possibly increases the end-to-end delay of the packet transfer in AANET [65], [66]. To reduce this queuing delay of a packet, the GLSR takes advantage of the join the shortest queue approach [67]. The Hierarchical Space Routing Protocol (HSRP) is proposed as an improvement of the Zone Routing Protocol which cannot be applied to the AANETs [68], [69]. The HSRP uses the flight flow rate, flight speed, and air vehicle density parameters to change the frequency of the HELLO beacons during the routing. The exchange of these HELLO messages is essential for detecting and maintaining links between two aircraft in the topology-based routing protocols. The Path Link Availability Routing Protocol (PLAR) uses the link stability for network topology control [70]. Also, multi-point relaying is the leading technology used in PLAR to reduce redundant transmission messages during the broadcast. The multi-point relay set includes the multi-point relay nodes. This routing protocol uses two different algorithms as the production way of the multi-point relay set. These algorithms are chosen according to aircraft density in the investigated area. The Ad-hoc Routing Protocol for Aeronautical Mobile Ad-Hoc Networks (ARPAM) is proposed based on the Ad hoc On-Demand Distance Vector (AODV) and Topology Dissemination Based on Reverse-Path Forwarding [71]. The main aim of ARPAM is to discover the shortest route by using different parameters like distance and the number of hops between nodes. In ARPAM, if an aircraft wants to communicate with another node, it sends a route request (RREQ) message through the omnidirectional link. The destination aircraft sends a route reply (RREP) message back to the source node to show the existence of a valid path.

The AeroRP takes the routing decisions per-hop basis without any knowledge about the end-to-end source to the destination route [72]. In AeroRP, the velocity-based

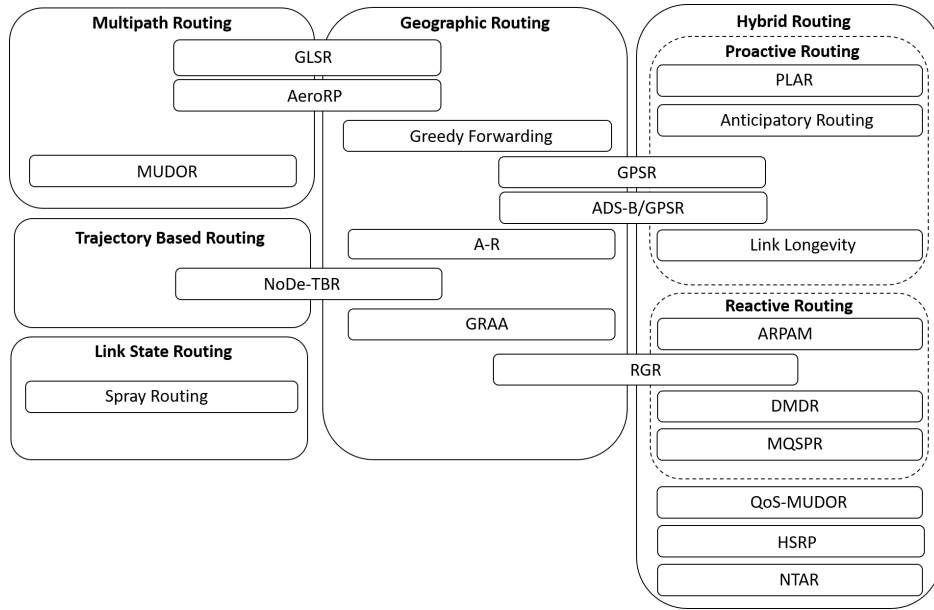


Figure 1.7 : Basis of AANET routing.

heuristics are calculated for each one-hop neighbor. Also, the time to intercept is the primary metric used during the routing decisions. With this parameter, the source node can have information about the duration when the potential neighbors are in the transmission range of the destination. Accordingly, this parameter is calculated by the source node for each neighbor. Also, the speed and coordinates are the main components for the time to intercept calculation. Additionally, the Secure AeroRP (SAeroRP) is proposed to increase the security of AeroRP by disabling active and passive attacks with the X.509 authentication [73], [74].

The Anticipatory Routing uses the past movement history of the endpoint to predict future locations [75]. More specifically, linear regression is utilized to predict endpoints' future locations and departure times. By estimating the trajectory, direction, and affiliation/departure, the location of endpoints is reached. Then, the traffic is routed to this new location before the movement. They claim that this situation improves routing performance compared to the reactive methods. The Spray Routing executes traffic multicasting in the vicinity of the last known location of the endpoint [76]. Here, the sprayed packet first unicast to a node close to the destination, then this packet is multicast to the multiple nodes around the destination. During this process, the width and depth become the main routing parameters. The width represents the neighbor level number to which the packets should be multicast. The depth indicates the hop

Table 1.3 : Routing protocols for AANETs.

Protocol Name	Ref.	Parameters	Path Selection Criteria
MUDOR	[64]	Mobility Link Duration	Doppler Value Stability
GLSR	[65], [66], [67]	Queue size Geographic Distance	Load Sharing
HSRP	[68], [69]	Flight Flow Rate Flight Speed Air Vehicle Density	HELLO Beacon Update Frequency
PLAR	[70]	Link Stability Aircraft Density Position Coordinates	Multi-Point Relaying
ARPAM	[71]	Velocity Vectors Distance	RREQ Messages
AeroRP	[72]	Time to Intercept	Velocity-Based Heuristics
Anticipatory Routing	[75]	Endpoint of Locations	Prediction of Future Locations Linear Regression
Spray Routing	[76]	Width and Depth	Vicinity of Last Known Location
Greedy Forwarding	[77], [78]	Distance to Destination Location	Locally Optimal Greedy Choice
RGR	[79]	Distance	RREQ and RREP messages
NoDe-TBR	[80]	Aircraft Density	Selected Geopath
GPSR	[81]	Location Distance	ADS-B Messages
ADS-B/GPSR	[82]	Security addition to GPSR	Hybrid hash function Cryptographic signature block
A-R	[83]	Distance Relative Velocity Position	ADS-B Messages
DMDR	[84]	Queuing delay Relative Velocity	Doppler Value
NTAR	[85]	Doppler Value Queue lengths Path Availability Period	Mobility and traffic loads of nodes
MQSPR	[86]	Available Path Load Capacity Path Latency	QoS Guarantee
QoS-MUDOR	[87]	Doppler Value QoS Parameters	FOBREQ Messages
GRAA	[88]	Topology Information Destination Position Unique Identification Number	Locally Optimal Greedy Choices
LLR	[89]	Aircraft Positions Velocities	Link Longevity Prediction

distance of the point where multicast starts to destination. In Greedy Forwarding algorithm, each sender aircraft marks the packet with the destination location. Then, each forwarding node decides locally to the next hop according to the relative location of the neighbor to the corresponding destination. For this reason, each aircraft should know its position and neighbors' positions. Therefore, the exchange of position information is executed only between the neighbors locally with the reduced overhead [77]. But here, the nodes should have a sufficient number of neighbors to apply the greedy forwarding mechanism [78].

The Reactive Greedy Reactive (RGR) protocol is proposed to combine reactive routing, and greedy geographic forwarding [79]. In this protocol, the source node

transmits the route request packets to the network for route discovery, similar to the AODV approach. After receiving a route response from the destination node, the route is established. But, the transmission of the route request and response packets causes overhead on the network. This overhead of the RGR protocol is aimed to reduce in two steps as RGR with scoped flooding and RGR with delayed route request [90]. Similarly, to reduce the route discovery overhead and packet dropping probability, the Modified-RGR is proposed in [91]. In Modified-RGR, the main aim is to keep all discovered paths in a table while only the primary path is used. Therefore, the number of the route discovery process is reduced with network overhead and delay. According to the Node Density Trajectory Based Routing (NoDe-TBR), the sender aircraft specifies both the packets' destination position and geographic path according to this destination position [80]. This routing algorithm consists of two main parts geopath computation and forwarding strategy. It is desired that the selected geopath is short, and the density of the aircraft on this path is high. Therefore, the actual aircraft densities are considered to maximize the packet delivery. The Greedy Perimeter Stateless Routing (GPSR) is proposed by taking advantage of Automatic Dependent Surveillance-Broadcast (ADS-B) [81]. The main aim is to reduce the overhead and collision probability causing the geographic routing mechanism. In geographic routing, each node requires neighbors' and destination positions. Accordingly, the stages of obtaining these parameters are the main reason for the overhead in geographic routing. In the ADS-B combined GPSR, the neighbor table is created and updated with the periodic state vector broadcasts in ADS-B messages to reduce the overhead. Similarly, the ADS-B Based Greedy Perimeter Stateless Routing (ADS-B/GPSR) is proposed to increase the security of the GPSR with the message integrity addition to ADS-B [82]. This message integrity is provided through a hybrid hash function/cryptographic signature block.

Additionally, the ADS-B Aided Geographic Routing (A-R) is proposed in [83]. Here, routing operations among aircraft are executed in three parts neighbor discovery, next-hop decision, and forwarding strategy. The Delay aware Multipath Doppler Routing (DMDR) uses the Doppler shift, expected queuing delay of packets, and relative velocities to select the stable and efficient paths for routing [84]. Also, with

these parameters, it is achieved that load sharing among all neighbors with reduced link congestion. As a very similar method to the DMDR, the Node Mobility and Traffic Load Aware Routing (NTAR) considers both mobility and traffic loads of nodes at the same time [85]. This routing protocol uses the Doppler value and transmission queue length as mobility and traffic load metrics. The Multiple QoS Parameters-based Routing (MQSPR) utilizes the path availability period, available path load capacity, and path latency metrics for route selection [86]. With these metrics, stable paths are selected, and the traffic is balanced between these A2G paths. Accordingly, they expect to observe reduced congestion, end-to-end delay, and packet loss rate during routing. Also, this work proposes to forward the best advertisement for the route discovery process. Here, they aim to prevent excessive advertisement flooding by only forwarding the best packets. The QoS Multipath Doppler Routing (QoS-MUDOR) uses the Doppler value and Quality of Service (QoS) parameters to select more stable paths [87]. Also, during this path selection, the RREQ messages are sent in the form of forward best request (FOBREQ). Here, only the best packets are forwarded, and others are discarded. The sender creates a packet by including geographic position information of destination and unique node identification number in Geographic Routing Protocol for Aircraft Ad Hoc Network (GRAA) [88]. Also, different from other geographic routing protocols, it can adapt the topology changes by using mobility information received from the ground station. The Link Longevity-Based Routing Protocol (LLR) proposes a method to predict the link longevity [89]. The aircraft positions, velocities, and signal to interference plus noise ratio (SINR) of the received signal from neighbor aircraft are used during the link longevity prediction. The maintenance of link and route is increased with reduced topology update overhead by predicting the link longevity. The main parameters and path selection methodologies of the above-explained routing algorithms are summarized in Table 1.3. Additionally, all of these AANET routing algorithms are built based on different ad-hoc routing protocols, as shown in Figure 1.7.

Although there are various routing algorithms in literature as detailed above, the artificial intelligence (AI)-based methodologies are not proposed in any work. On the other hand, we claim that the AI-based routing algorithms adapt to the dynamic

conditions of AANETs by considering the instant status of each aircraft. At that point, one of the possible solutions is to utilize reinforcement learning for routing management. Here, aircraft can take their own routing decision through exploration and exploitation without any guidance.

1.4.2.4 Handover management

As explained above, the mobility and atmospheric effects cause link breakages by reducing their qualities in AANET. Here, the significant propagation distance between aircraft increases the oxygen absorption effect, and this situation reduces the A2A link quality between them. Also, the absorption and scattering effects grow with the rain and cloud. These effects lead to frequent A2A link breakages by reducing signal strength. Similarly, the highly dynamic AANET environment caused by the ultra-high velocity of aircraft increases the link breakages by leading to frequent aircraft replacements. These broken links should be established again to the other aircraft as shown in Figure 1.8. This transferring is executed through the handover procedure as shown in Figure 1.6 [92].

The following works do different handover management algorithms and performance evaluations in the literature. Accordingly, a handover mechanism consisting of three phases: information collection, handover decision, and handover execution is proposed in [93]. In the information collection phase, the parameters like signal strength and

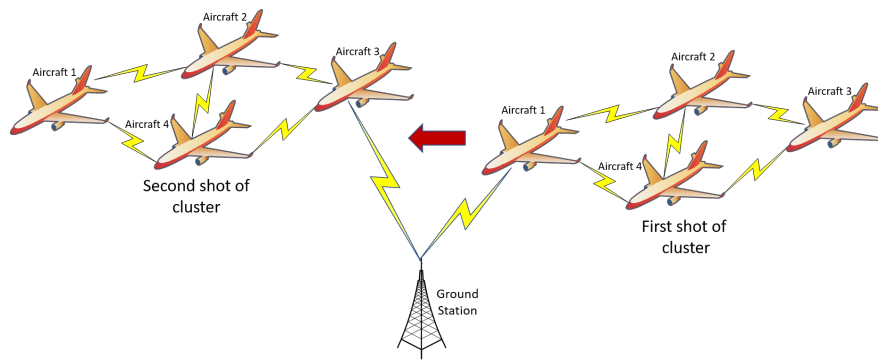


Figure 1.8 : Sample handover procedure on AANET.

bit error rate are continuously monitored and compared with the threshold values. According to the comparison results, the handover decision is taken in the second phase. Then, the handover is executed based on the Mobile Internet Protocol (IP)

and Resource Reservation Protocols (RSVP). With this handover mechanism, the geographic proximity and congestion parameters control the associated IGW of an aircraft. Therefore, all communications on an IGW are transferred to another gateway. Here, the congestion of an IGW is defined as the maximum transmission buffer from all links. One of the main aims of the handover procedure is to balance traffic load among IGWs according to the congestion parameter. This congestion-aware handover strategy will also increase the per-aircraft bandwidth [94]. Moreover, the handover performance in L-band Digital Aeronautical Communication System-1 (L-DACS 1) access network based on the Internet Protocol version 6 (IPv6) functionality is analyzed in [95]. Here, the ground station polls the received signal strengths of the neighboring cells through broadcast control information messages. If the received signal level of the neighbor station is greater than the current one, then the current cell triggers a handover to this cell. Then, the cell exit message is transferred to the current station, and the connection is switched to the channel of the selected station. In addition to these, the dual connectivity for the aircraft connections is proposed by utilizing VHF, and mobile user objective links [96]. Accordingly, the handover management is executed under these dual connectivity conditions. During the handover management, the queue backlog, user fairness, and resource constraints are also considered to reduce the delay. As explained above, the aircraft clusters can change continuously due to the ultra-dynamic characteristics of AANETs. Accordingly, the AANET experiences higher handover rates due to these continuous changes. As such, one of the possible solutions is to estimate the subsequent movements of aircraft to take precautions for the upcoming handovers. By estimating the next handovers, we can pre-determine the clusters that the aircraft will connect. Then, we can assign the aircraft to these pre-determined clusters to reduce the delays during the handover.

1.4.3 Transport layer issues

As explained above, the AANETs have specific features and requirements compared to the terrestrial networks. The current transport protocols cannot satisfy these requirements as in data link and network layers. As an example, the frequent retransmissions of lost packets in Transmission Control Protocol (TCP) reduce the

AANET performance by causing a high delay. Also, the highly asymmetric channels

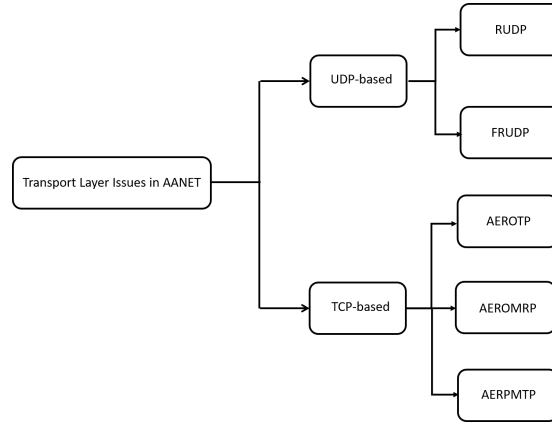


Figure 1.9 : Transport layer issues in AANET.

can cause congestion on the reverse link. This congestion could be assumed to be the main reason for the packet loss, which possibly reduces the throughput in AANET. On the other hand, if the User Datagram Protocol (UDP) is used as a transport protocol, the transmissions are executed more fastly with reduced reliability. For these reasons, as shown in Figure 1.9, different UDP and TCP-based solutions are proposed for AANET. Accordingly, by combining the TCP reliability and UDP low latency features, the Reliable User Datagram Protocol (RUDP) is presented in [97], [98]. But, the performance degradations caused by acknowledgments and retransmissions also affect the efficiency of RUDP.

The FRUDP is proposed by combining the RUDP and the fountain code schemes in [101]. The main aim of this protocol is to obtain a reliable and efficient data transfer protocol for AANETs. Furthermore, the AeroTP is proposed as a TCP-friendly transport protocol [102], [103]. The AeroTP includes the connection setup, management, transmission control, and error control functionalities. The AeroTP also has multiple transfer modes to support reliable, near-reliable, quasi-reliable, and unreliable connections. As another approach, the Aeronautical Multipath Reliable Protocol (AeroMRP) utilizes Raptor codes as a forward error correction mechanism to avoid and mitigate the retransmission and head-of-line blocking problems [104]. The head-of-line problems reduce the transport protocol performance if the different network conditions are valid for paths. But, the AeroMRP takes advantage of path diversity by using various aeronautical networks simultaneously. Similarly, a fountain

Table 1.4 : Summary of open research problems.

Layer	Aspect	AANET Requirement	Ref.	Recommendation
Data Link Layer	Logical Link Control (LLC)	Link Stability	[33]	Direction based link stability
			[99]	Free space loss based link stability
		Link Connectivity	[34]	Expiration time based link stability
			[34], [35]	Doppler Shift based link stability
			[38], [39]	Aircraft density based link connectivity
			[40], [40]	Transmission range based link connectivity
	Medium Access Control (MAC)	MAC	[42], [43], [44]	Communication distance based link connectivity
			[47], [48]	TDMA
			[51], [52]	CDMA
			[53]	DS-CDMA
			[54]	RP-CDMA
			[50], [100]	SPMA
Network Layer	Network Management	IGW Selection	[49]	IDTA
			[55]	Hop count based selection
			[56]	Utilization based selection
			[57]	Traffic load based selection
		Aircraft Clustering	[58]	Delay based selection
			[60]	Multi-dimensional clustering
			[61]	1-hop clustering
		Routing Management	[62]	Honeycomb division based clustering
			[64]- [89]	Summarized on Table 1.3
			[93], [94]	Congestion aware handover
Transport Layer	End-to-End Management	Handover Management	[95]	Signal strength based handover
			[96]	Dual connectivity based handover
		UDP Based	[97], [98]	RUDP
			[101]	FRUDP
			[102], [103]	AeroTP
			[104]	AeroMRP
		Based	[105]	AeroMTP

code-based multipath transport protocol (AeroMTP) effectively utilizes the available bandwidth, and path diversity [105]. The AeroMTP deploys as a TCP-friendly congestion control mechanism and uses fountain codes as forwarding error correction codes in data recovery.

As explained through this section, we investigate the open research problems of AANETs in a layered manner as data link, network, and transport layers. We summarize all of these open research problems and proposed methods in Table 1.4.

1.4.4 Additional issues

1.4.4.1 Effects of the aircraft antennas on connectivity

The IFC is enabled through the aircraft antennas implemented based on the utilized aeronautical network type. Accordingly, the effects of the antenna parameters, directions, and gains should be considered while enabling the IFC. During the antenna design, one of the important metrics is the antenna gain, and it should be higher to compensate for the path loss caused by a significant distance and high carrier frequency [106]. Additionally, the antenna array should be large in aircraft compared to the lower flying ad-hoc networks (FANETs) like unmanned aerial vehicles (UAVs). In addition to these, the communication generally is executed based on the LOS propagation. This leads to the antennas' directions and the ranges between the aircraft having a crucial effect on the aircraft connectivity. More specifically, if an aircraft is out of the coverage range of an A2G station, it is not connected to this network. Similarly, if the distance between two aircraft is more than the predetermined distance, they cannot connect through A2A links. At that point, aircraft antennas' directions and parameter settings (antenna gain, azimuth beam-width, polarization) are essential in determining these A2G and A2A link distances. Also, the characteristics of aeronautical networks are different from the terrestrial-based conditions. Here, the dynamic characteristic of aeronautical networks makes the location of the antennas critical. Accordingly, they should be optimally located to disable the interference with the other aircraft systems.

1.4.4.2 Effects of regulations

This part of our chapter summarizes more social and organizational issues observed during the IFC. At that point, the first issue is related to the network selection rules of airlines. The IFC is enabled through the airlines to the passengers. At that point, the airline can choose different aeronautical network types according to the position and capability of the aircraft. More clearly, if an aircraft moves through the ocean, it can connect to satellite or AANETs instead of an A2G network [107]. Similar to this example, there are various conditions, and the airlines determine their policies according to these conditions. The airline-specific policies affect the connectivities of all aircraft by changing network capacities and loads. Additionally, the second issue is related to the hardware supports of airlines that also affects their policies for IFC. Here, three aeronautical network types require different hardware equipment, and aircraft

connectivity is shaped according to the airlines' support. Another issue is related to the security precautions of airlines during the IFC. The IFC should be enabled to the aircraft without letting the malicious intruders since this risk is more observed with the increasing amount of data and system complexity. The final consideration is the collision risk between the aircraft. Here, the connectivity of aircraft should be established by also considering the connection status of others. Otherwise, the aircraft observe collision during the packet transfer.

1.5 Conclusion

With the increasing technology and passenger number, in-flight Internet connectivity becomes crucial during a flight. This connectivity also becomes an essential income for the airlines. For this reason, the IFC takes the attention of both industry and academia. At that point, the AANETs are started to be included in the literature as a new practical solution. In this chapter, we gave our specific interest to the AANETs by investigating its particular topology and open research challenges. More clearly, we examine the AANETs by giving topological details, environment and mobility effects, and open research challenges. Also, these open AANET challenges were examined from a layered aspect in with this chapter.

General concept for the next chapter

At that point, the intelligent frameworks could be utilized based on the AI to overcome the link, network, and transport-level management complexities of AANETs. The utilization of AI in wireless networks is common in the industry and academia. However, AI-driven AANETs are the new and unexplored research area. We claim that the AI-based supervised, unsupervised, and reinforcement learning methodologies can ease the management of AANETs. The ultra-dynamic characteristics of AANETs could be learned utilizing AI-based methods, and these experiences could be used during management decisions dynamically without any central node and entity.

Based on these, in the next chapter, we aim to investigate the utilization of AI-driven methodologies for the network layer issues of AANETs as examined in this chapter.

2. ARTIFICIAL INTELLIGENCE-BASED AANETs

2.1 Purpose

AANET has been considered a promising candidate to complete the vision of “Internet in the sky” by supporting high-speed broadband connections on airplanes. However, the specific characteristics of AANET restrict the applicability of conventional topology and routing management algorithms. Here, these conventional methodologies reduce the packet delivery success of AANET with higher transfer delay. At that point, the AI-based solutions have been adapted to AANET to provide intelligent frameworks and architectures to cope with the high complexity. The AI-based AANET can provide intelligent topology formation, sustainability, and routing management decisions in an automated fashion by considering its specific characteristics during the learning operations. More clearly, AI-based AANETs support intelligent management architectures, overcoming drawbacks of conventional methodologies. Although AI-based management approaches are widely used in terrestrial networks, there is a lack of a comprehensive study that supports AI-based solutions for AANETs. To this end, this chapter explores the possible utilization of primary AI methodologies on the road to AI-based AANET. Specifically, the chapter addresses unsupervised, supervised, and reinforcement learning as primary AI methodologies to enable intelligent AANET topology formation, sustainability, and routing management. Here, we identify the challenges and opportunities of these primary AI methodologies during the execution of AANET management.

2.2 Introduction

Satellite and A2G connectivity are the two pivotal methods to enable IFC [108], [109]. However, the cost and coverage restrictions of these methods reduce the quality of the end-user experience. The AANETs can enable Internet connectivity to the aircraft that cannot IFC directly access the terrestrial network, and accordingly, this extends IFC’s

coverage area with cost reduction [110]. For this reason, AANETs have attracted a lot of attention from both industry and academia as a potential candidate to improve the coverage area and end-user experience of IFC with cost-efficiency [111]. These advantages are achieved by establishing A2A links between airplanes that execute packet transfer, as shown in Figure 6.4. More specifically, airplanes' packets are transferred through A2A links in an ad hoc manner until reaching the destination node having Internet connectivity. Each plane acts as a router for packet transfer in this formed network without the assistance of any centralized terminal.

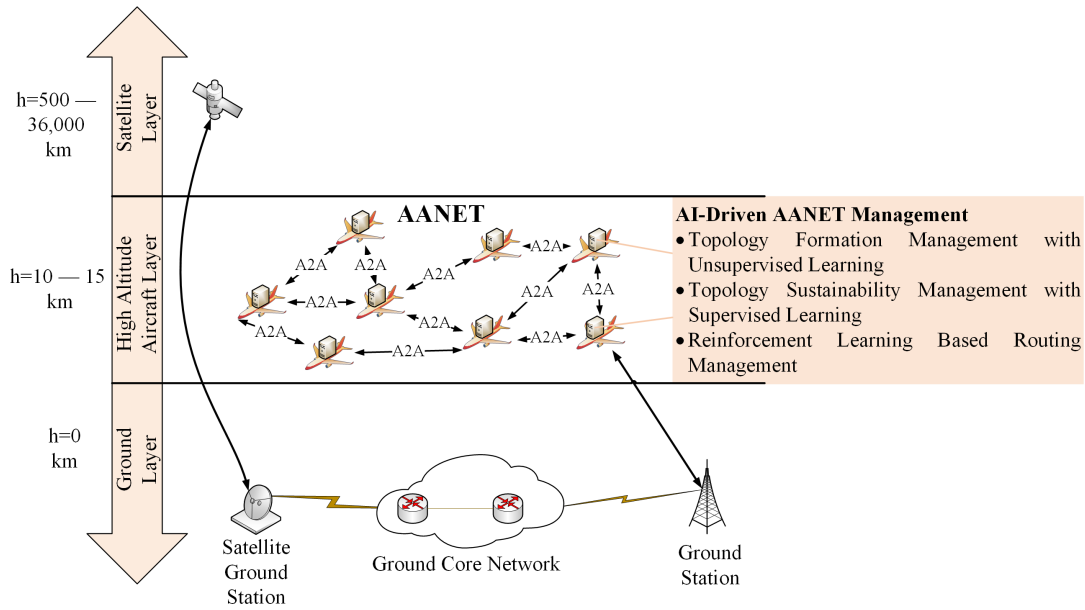


Figure 2.1 : An exemplary illustration of AANET topology.

The AANET operation is to form a topology and sustain it to enable packet routing through A2A links during IFC. However, at this point, the use of existing topology formation, sustainability, and routing management algorithms is challenging due to the specific characteristics of AANETs. Accordingly, the following section investigates the conventional solutions for topology formation, sustainability, and routing management challenges caused by the AANET-specific characteristics. Additionally, in this chapter, the specific characteristics of AANETs are considered as the high aircraft density in 3D space, unstructured ad hoc topology, ultra-dynamic network, and packet transmissions through the A2A links. We also provide the details of these specific characteristics in Section 2.3.

2.2.1 Contributions of the chapter

As explained in Chapter 1, there are various works in the literature proposing AANET topology formation, sustainability, and routing management algorithms in network layer. On the other hand, these algorithms cannot handle the AANET-specific characteristics simultaneously. Therefore, to overcome the AANET management challenges caused by its specific characteristics, we require intelligent frameworks and architectures specially designed for topology formation, sustainability, and routing procedures. This intelligence could be achieved by embedding AI-based solutions to the design of AANETs. The embedding of AI to wireless networks is well accomplished [112,113]. However, the utilization of AI in AANET is an unexplored and relatively new research area.

In this chapter, we investigate the possible AI methodologies to ease the management of AANETs. More specifically, as shown in Figure 6.4, we will analyze the unsupervised, supervised, and reinforcement learning-based methodologies for AANET topology formation, sustainability, and routing management. Accordingly, we will investigate the AI-based AANET management methodologies in three main categories. Additionally, we analyze the challenges and opportunities of these methodologies on the road ahead to AI-based AANETs. The main contributions of the chapter could be summarized as follows:

- Examination of AI for AANET management: The utilization of AI-based methodologies for AANET management is still an unexplored area. This is the first work to investigate possible AI methodologies for AANET management.
- Discovery of AANET's specific characteristics: We describe AANET's specific characteristics to highlight the necessity of AI for AANETs. These specific characteristics also become the starting point in AANET management problems.
- Discussion about AANET management problems in network layer: We discuss three main management problems for AANET caused by its specific characteristics.

- Matching of AI methodologies and AANET management problems: We give our specific attention to proposing AI-based methodologies in three main AANET management problems in network layer.

2.2.2 Organization of the chapter

The rest of the chapter is organized as follows: We first explore the necessity of AI for AANET by discussing its specific characteristics. Then, we investigate the AI-based AANET management in terms of three main management-level challenges. Finally, we conclude the chapter by also giving the general concept of the next chapter.

2.3 Necessity of AI for AANETs

Unlike other ad hoc networks, the AANETs have specific characteristics, which are the root causes for different management-level challenges. Before giving the details of these challenges, we first investigate the specific characteristics of AANETs by matching them with AI-based methodologies to underline the necessity of AI-based solutions.

- High Aircraft Density in 3D Space: One of the main characteristics of AANET is the movement of a large number of aircraft in the 3D environment, which produces a massive amount of data on heterogeneous characteristics (airplanes' locations, speeds, and angles). Accordingly, this enormous amount of heterogeneous data overwhelms the management procedures of AANETs. AI has become the leading solution to learn and analyze these data for AANET management. Therefore, we can make estimations, create clusters, and recognize patterns in AANET by analyzing the collected information through AI.
- Unstructured Ad Hoc Topology: As the name suggests, the AANETs have an ad hoc characteristic that is independent of central guidance. Airplanes in the network can directly communicate with each other, and during this communication, each aircraft behaves as a router. This ad hoc infrastructure makes topology formation a challenging issue coupled with the ultra-dynamic network characteristic. Here, we can handle the topology formation of AANET through AI. At this point, we propose

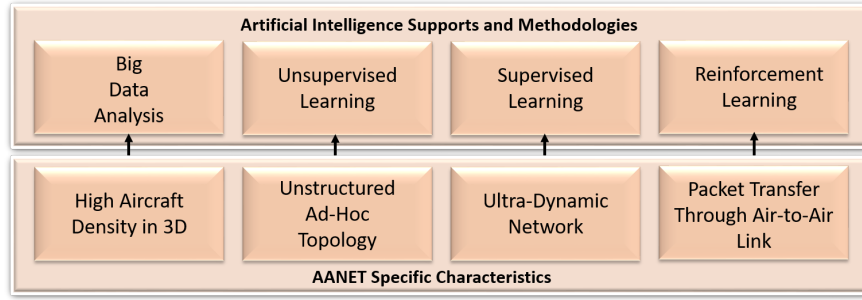


Figure 2.2 : AI supports and methodologies for AANET-specific characteristics.

to utilize unsupervised learning due to its unlabeled data supporting character. Clearly, we can form a topology by trying to search for meaningful structures in an unstructured topology through unsupervised learning, as shown in Figure 2.2.

- **Ultra-Dynamic Network:** The ultra-dynamism is one of the essential AANET characteristics resulting from airplanes' high speed. This over-speed makes the formed network ultra-dynamic by increasing the breakage rate of A2A links between airplanes. Accordingly, the AANET topology with aircraft nodes and A2A links changes continuously. However, continuous topology creation becomes time-consuming in this highly dynamic network. Instead, the initially formed topology could be utilized through supervised learning, as shown in Figure 2.2. To achieve this, we can consider the formed topology as training data for mapping the new instances by handling heterogeneity.
- **Packet Transfer through A2A Link:** The packets of the source aircraft are transferred through A2A links until reaching the destination to enable Internet connectivity. However, due to the ad hoc behavior and ultra-dynamic network characteristics, the determination of the next plane becomes challenging [114]. More specifically, the next plane should be determined dynamically according to the

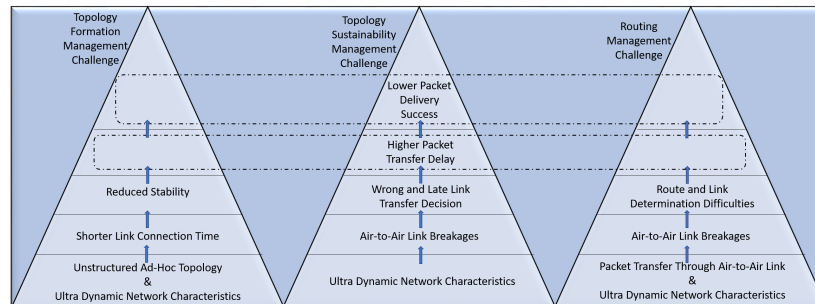


Figure 2.3 : The root causes chains of AANET management challenges.

current situation of the network topology during packet transfer since the connected airplanes and links of an aircraft change continuously with their movements. Reinforcement learning could be one of the possible AI-based solutions to handle this dynamism by also increasing packet transfer success, as shown in Figure 2.2. Here, the airplanes can dynamically find their optimal packet transfer path through exploration and exploitation procedures without guidance.

These specific characteristics become the origin of management level challenges, as illustrated in Figure 2.3. To overcome the AANET management challenges, we require AI-based intelligent frameworks and architectures. Therefore, they also underline the necessity of AI for AANETs. The following section of the chapter will detail each of these management-level challenges and suggest corresponding AI-based solutions based on AANET's specific characteristics.

Moreover, in literature, various works propose AANET algorithms without utilizing AI. However, these works only consider the general aspects of AANET instead of focusing on its management. Additionally, these works do not consider the effects of the AANET-specific characteristics simultaneously. On the other hand, we should handle these AANET-specific characteristics simultaneously to manage the challenging AANET environment. At that point, we should utilize AI-based solutions to overcome the problems observed during AANET management caused by its specific characteristics.

2.4 AI-Driven AANET Management

In the previous section, we explained the necessity of AI for AANETs by matching the critical AI methodologies with AANET-specific characteristics. In what follows, we investigate the details of three management challenges caused by AANET's specific characteristics. We aim to handle each of these management challenges with AI-based methodologies. At that point, the AI-based methodologies enable fast adaptation of airplanes to the dynamic conditions of AANETs [115]. More specifically, the aircraft can form and sustain the AANET topology by finding routes according to the instant status of AANETs. In this way, the dynamic conditions of AANETs could

be learned by airplanes thanks to AI-based methodologies. The aircraft could then make the management decisions dynamically without any guidance. As explained in the previous section, we divide the management issues of AANETs into three main categories: topology formation, topology sustainability, and routing. In the rest of the chapter, we aim to investigate and propose corresponding AI-based methodologies for these management challenges.

2.4.1 Topology formation management with unsupervised learning

As explained in Section 2.3, the unstructured ad hoc topology is one of the main AANET characteristics. This infrastructure complicates the formation of AANET topology when combined with ultra dynamism. These two characteristics reduce the stability of the AANET topology by shortening the connection life of the links between two airplanes. This unstable network with short-lived links deteriorates the packet delivery success by increasing transfer delay, as illustrated in Figure 2.3. We can handle the unstructured and ultra-dynamic characteristics of AANETs if we support topology formation management with AI. Accordingly, we can improve the packet delivery success with a shorter delay.

Unsupervised learning is the most relevant AI concept for AANET topology formation management. Here, the main aim of unsupervised learning is to find previously unknown patterns in data. Accordingly, it aims to explore the data to find meaningful structures and define classes. If we think about AANETs, we have independently located airplanes with unstructured characteristics before forming the topology. They could be considered as the unlabeled training data for unsupervised learning.

Accordingly, our main aim is to group the airplanes with similar features under one cluster to form the AANET topology. Therefore, we will investigate the role of two main unsupervised learning methods in topology formation: clustering and self-organizing maps.

2.4.1.1 Clustering-enabled AANET topology formation

The main aim of the clustering-enabled AANET topology formation is to create different aircraft groups according to their similarities. In AANETs, we utilize the

clustering for grouping airplanes according to their similarities in direction, distance, angle, altitude, and mobility characteristics as detailed in Chapter 1. Therefore, the airplanes in the same cluster can establish more long-lasting A2A links to increase the stability of AANET. In this chapter, we investigate three clustering methods that are mainly utilized for AANETs in terms of their advantages and disadvantages, as explained below:

- **Prototype-based Clustering:** It assigns the instances to the nearest prototypes, and the K-means clustering constitutes one of the primary examples. It can easily be implemented and effectively work with large data sets. This enables high-speed performance for the ultra-dense and 3D AANET environment. However, the cluster number should be assigned before the algorithm runs, which contradicts the ultra-dynamic characteristic of AANET as summarized in Table 2.1.
- **Hierarchical Clustering:** The hierarchical clustering aims to create clusters based on the previously established ones. During this creation, two different algorithms, agglomerative and divisive, could be utilized. The divisive approach starts with one initial cluster and then splits this into child clusters. Conversely, the agglomerative clustering starts from smaller groups to form larger clusters according to the bottom-up approach. In this clustering, we do not specify the number of clusters initially, which is the main advantage of the dynamic environment of AANETs, as summarized in Table 2.1. On the other hand, the algorithm does not return from the previously completed operation, which contradicts the unstructured ad hoc topology of AANETs since the clusters need to be updated according to the formed topologies.
- **Density-based Clustering:** Density-based clustering can discover non-linear clusters with arbitrary shapes. This feature increases the clustering performance in the unstructured topology of AANETs. Similar to hierarchical clustering, this method also does not require the defined cluster number [116]. However, the AANETs have high and changing aircraft density, while the algorithm does not efficiently work on the varying density clusters, as summarized in Table 2.1.

2.4.1.2 Self-organizing maps-enabled topology formation

Clustering is mainly utilized to form the AANET topology, as explained above. However, other unsupervised learning methodologies could also be used for AANET topology formation. Here, we introduce the Self-Organizing Maps (SOMs), an artificial neural network based on unsupervised learning. The SOMs use competitive learning to cluster the distinct patterns from the input data into output sets. SOMs can handle the previously unknown inputs, and this strongly matches the ultra-dynamic characteristic of the AANETs, as summarized in Table 2.1. Therefore, SOMs could be utilized to create the unstructured AANET topology in the form of clusters. The clustering performance of the SOMs is not affected by the unstructured AANET environment since it keeps the relative distances steady in a cluster. Accordingly, the SOMs can handle the newly included aircraft to the AANET for cluster assignment.

Table 2.1 : AANET related opportunities and challenges of AI-driven methods.

Methods	Advantages for AANE	Disadvantages for AANET
Prototype-based	◊ Effectively works for large data sets	◊ Previously defined cluster number
Clustering	◊ Supports high aircraft density in 3D	◊ Contradicts ultra-dynamic network
Hierarchical	◊ Undefined cluster number	◊ Unchanged priori operation
Clustering	◊ Supports ultra-dynamic network	◊ Contradicts unstructured topology
	◊ Undefined cluster number	
Density-based	◊ Supports ultra-dynamic network	◊ Failure on the varying density clusters
Clustering	◊ Non-linear clusters with arbitrary shape	◊ Contradicts high aircraft density in 3D
	◊ Supports unstructured ad-hoc topology	
	◊ Topology protection without changing relative distances	
Self-Organizing	◊ Supports unstructured ad-hoc topology	◊ Difficulty on optimal map size decision
Maps	◊ Processing of previously unknown inputs	◊ Contradicts high aircraft density in 3D
	◊ Supports ultra-dynamic network	
	◊ Efficient pattern classification	
Learning Vector	◊ Supports ultra-dynamic network	◊ Lower success on high dimensional data
Quantization	◊ Efficiency for multi-class classification	◊ Contradicts high aircraft density in 3D
	◊ Supports unstructured ad-hoc topology	
	◊ The ability to work in changing areas	
Q-Learning	◊ Supports ultra-dynamic network	◊ High dimensional state and action space
	◊ Sequential decision-making	◊ Contradicts high aircraft density in 3D
	◊ Supports transfer through A2A link	

After explaining the main reasons for selecting the SOM for topology formation, we aim to adapt it for AANETs. During this adaptation, each aircraft in AANET corresponds to the input vector in SOMs. These input nodes are directly connected to the output without any hidden layer, and here, each connection is associated with a weight value. In the output layer, the AANET clusters become the output vectors,

and the number of these represents the maximum cluster number of the AANET [117]. These clusters are connected through intra-layer links without weight values. Additionally, in the output layer, the topologies could be mapped in the form of a cylinder or toroid instead of the hexagonal structure due to the 3D environment of AANETs. Additionally, the Euclidean distances could be utilized by determining the similarities between input vectors to execute this mapping. Thanks to these features, we can map the 3D space of the AANET onto a plane with SOMs by preserving the underlying structure. However, determining the optimal map size is already challenging in SOMs. This decision will be more difficult for AANET due to the high and changing aircraft density in the 3D environment, as summarized in Table 2.1.

2.4.2 Topology sustainability management with supervised learning

The ultra-dynamic network constitutes one of the main characteristics of the AANET caused by the high speed of airplanes. As detailed in Section 2.4.1, this ultra-dynamism complicates the formation of a stable topology. The sustainability of this formed topology also becomes challenging due to ultra-dynamism. More specifically, the formed topology is continuously disrupted due to the breakages of A2A links between airplanes. On the other hand, to enable the sustainability of the AANET topology, the broken links should be established again with different airplanes. Here, the wrong and late link transfer decisions again reduce the packet delivery success with higher delay, as illustrated in Figure 2.3. These effects make the sustainability management of an AANET challenging.

We can utilize the initially formed topology to enable the sustainability of AANET instead of creating it continuously. At that point, we can take advantage of *Supervised learning* to ensure the sustainability of AANET based on the initial topology. More specifically, we can use the initially formed topology as a training set to determine its upcoming states due to supervised learning can produce output by utilizing previous experience [118]. Moreover, thanks to the classification category of supervised learning, we can facilitate the assignments of new airplanes to clusters identified with unsupervised learning. Based on these arguments, in the following parts, we will exemplify a supervised learning method for AANET topology sustainability as

learning vector quantization by also giving its opportunities and challenges during this operation.

LVQ-based AANET topology sustainability

The learning vector quantization (LVQ) could be considered the supervised version of the SOM. Similarly, the LVQ classifies the patterns into output units that denote classes. More clearly, the initially given network and output classes are utilized to classify an input vector [119]. This working principle strongly matches the topology sustainability requirement of ultra-dynamic AANETs. Therefore, we can enable the sustainability of the initially formed AANET topology by using the pattern classification feature of this methodology since it is more effective to ensure the sustainability of ultra-dynamic AANET topology rather than reforming it continuously. Additionally, it can be easily implemented for multiclass classification problems, which increases the efficiency of grouping in the unstructured environment of AANETs. Accordingly, it allows for the separate grouping of airplanes with different characteristics. The newly included aircraft could also be placed in a previously formed suitable group of AANET without reclustering, thanks to LVQ.

Therefore, we can utilize the LVQ for topology sustainability management in AANETs. Accordingly, our main aim is to find the best matching cluster for an aircraft observing A2A link breakage. Here, we take advantage of this AI-based approach to prevent continuous topology formation. Instead, we utilize the initially formed topology in pattern classification by utilizing the aircraft clusters in the formed AANET topology as *pre-classified training data* in *multidimensional space* of AANET. Here, the *weight vectors* are the airplanes in the formed AANET cluster. Additionally, the aircraft with a broken A2A link corresponds to the *input vector*. Accordingly, each known aircraft cluster becomes *output*. We aim to sustain the AANET topology instead of continuous topology formation by utilizing LVQ. Although LVQ is critical for enabling topology sustainability, it can achieve lower success on high-dimensional data. This situation contradicts the high aircraft density

characteristic of AANETs in 3D, as summarized in Table 2.1. The high aircraft density of AANETs increases the computational cost of applying LVQ [120].

2.4.3 Reinforcement learning-based routing management

As explained in Section 2.3, the AANETs work based on the principle of packet transfer over A2A links. The routing management creates an inevitable need to deliver the packets of source aircraft to the destination by determining A2A links [121]. On the other hand, the routing protocols used in terrestrial networks do not satisfy the requirements of AANETs due to ultra-dynamism. More specifically, ultra-dynamic characteristics of AANET increase the breakage rates of A2A links between airplanes, as explained in Section 2.3. This situation also makes the route and link determination difficult during routing, as illustrated in Figure 2.3. The difficulty in making a route decision reduces the packet delivery success with higher delay. Therefore, routing management in AANET becomes another challenging issue.

Reinforcement learning enables us to make routing decisions in the ultra-dynamic environment of AANET by also maximizing packet delivery success [122]. Clearly, reinforcement learning makes immediate routing decisions by exploring A2A links through experience [123]. Thus, each aircraft can sequentially determine its routing path with reinforcement learning in two different ways: model-based and model-free. The model-based reinforcement learning makes decisions based on the model of the environment, and it is not suitable for the AANETs due to its dynamic characteristic. On the other hand, model-free reinforcement learning can only make decisions with instant exploration without any environment model, which makes it very applicable for AANETs. We will examine the implementation details of Q-learning, one of the main model-free reinforcement learning techniques, in the upcoming section.

Q-learning for AANET routing management

Q-learning is a model-free reinforcement learning method working based on action and reward mechanism [124]. This flexibility makes it easy to adapt to the dynamic and unstructured AANET environment. Here, the aircraft can explore different A2A links to choose the one with high packet delivery success as a reward. Clearly, the Q-learning

considers the dynamism of the AANET environment, and the previous decisions affect the subsequent actions [125]. Accordingly, this sequential decision-making facilitates path selection during routing with higher packet rates. Therefore, in Q-Learning, we assume each aircraft as an agent with a state, then this agent takes action, earns a reward, and switches to a new state, as shown in Figure 2.4. More clearly, Figure 2.4 summarizes the mapping of Q-learning concepts to the AANET scenario. In this mapping, the AANET creates the *Environment* of Q-learning, and each airplane that performs packet transfer through the A2A link in this environment becomes an *Agent*, as shown in Figure 2.4. Accordingly, the packet transfers executed through the A2A links could be defined as an action. These actions are included by the action set changing according to the connections of airplanes, as shown in Figure 2.4. Based on these, the delivery success of the A2A packet transfer denotes the *Reward*. Therefore, a reward is obtained according to the result of each action. The obtained rewards are used to fill the *Reward Table*. Here, each cluster in an AANET is represented with a reward table. This table shows the airplanes in that cluster with rows and columns. Accordingly, the rewards for the corresponding links are placed in the cells of this table. After obtaining rewards and filling the reward table, we can observe the new states, as shown in Figure 2.4. The *State* represents the status of an aircraft according to its packet transfers. These determined states and rewards help us to update the Q-Table that enables aircraft to learn the environment as a result of experiences. Then, each aircraft can determine the actions in the next step. At this point, the Bellman Equation is utilized to fill the Q-table. We will give the detail of the Bellman Equation and Q-learning-based AANET routing in Chapter 6. Although Q-learning is directly applicable to dynamic environments, its high-dimensional state and action space can increase the computational load when combined with the high aircraft density of AANETs in 3D, as summarized in Table 2.1. More specifically, the memory and time requirement of the algorithm is increased with a growing Q-table. At that point, the *Deep Q-Learning* could be utilized to solve the performance defects caused by the complex action, and state-space [126,127]. In deep Q-learning, the neural network is used to approximate the Q-value function instead of table usage, as shown in Figure 2.4. The neural network outputs the Q-value of all possible actions when it

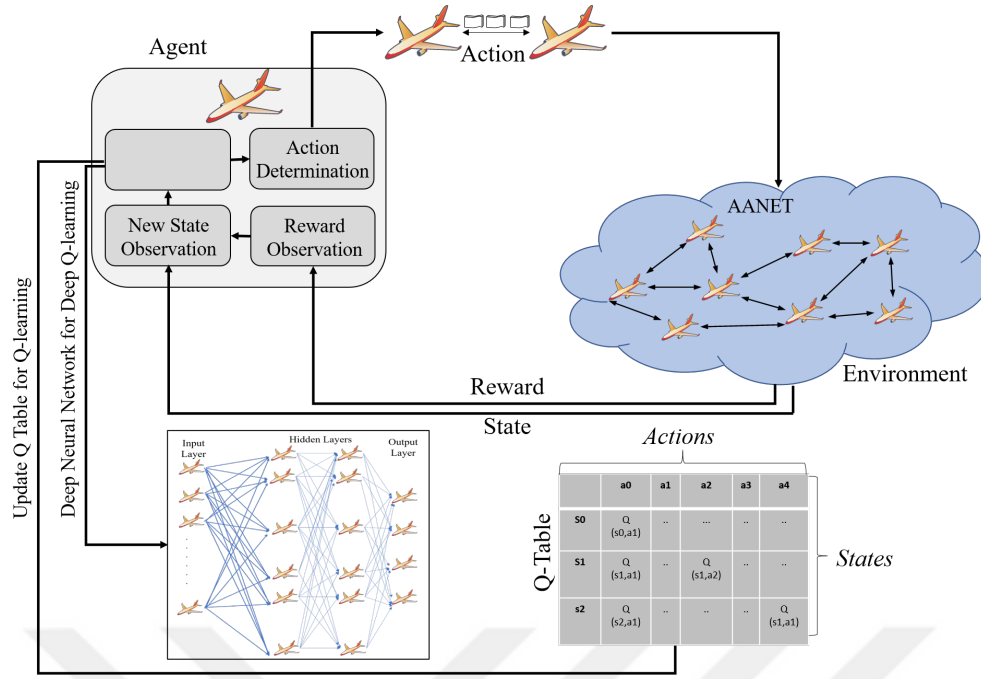


Figure 2.4 : Q-learning and deep Q-learning for AANETs

receives the initial value. Therefore, deep Q learning can become a more scalable solution for the environments that require more complex action and state schemes.

2.5 Conclusion

In this chapter, we propose the using of AI-based methodologies to overcome the AANET management challenges caused by its specific characteristics. The chapter investigates these management challenges in three main aspects: topology formation, sustainability, and routing management. Accordingly, the AI-based methodologies enable us to obtain intelligent frameworks and architectures for these management issues. For this aim, we first underline the necessity of AI by matching its primary methodologies with AANET-specific characteristics. Here, we considered unsupervised, supervised, and reinforcement learning as primary methodologies for these management challenges on the road ahead to AI-based AANET. We reviewed the challenges and opportunities of different approaches from these methodologies to automatically make topology formation, sustainability, and routing management decisions on AANET. As a result of these, we aim to provide an insight into the road ahead toward AI-based AANET management.

General concept for the next chapter

In this chapter, we aim to solve three main management problems on AANETs by utilizing AI-based methodologies. In the first step, we should create the AANET topology in the form of clusters. Then, we should enable its sustainability, and finally, we can route the packet from the source aircraft to the destination. Accordingly, this chapter aims to solve these management issues through the unsupervised, supervised, and reinforcement learning methodologies.

Based on these, in the next chapters, we will investigate the topology formation, topology sustainability, and routing management issues in AANETs by proposing above-explained AI-based algorithms for them. In the upcoming chapter, we start this investigation by proposing a clustering algorithm for AANET topology formation.



3. TOPOLOGY FORMATION MANAGEMENT FOR AANETS

3.1 Purpose

IFC has become one of the crucial needs of passengers with technological improvements. The AANETs have been proposed by establishing A2A links between aircraft to satisfy this need. However, the unstructured aircraft topology caused by their ultra-dynamic characteristics reduces the longevity of A2A links in AANET. This shorter longevity decreases the stability of AANET by accelerating the connection establishment/termination procedures between aircraft. Additionally, the shorter link longevity affects the packet transfer delay and the delivery ratio of AANET. To the best of our knowledge, these three challenges are not investigated simultaneously under one topology management model. This chapter proposes a three-phased topology formation model for AANETs to increase the stability and packet delivery ratio in AANET with a shorter packet transfer delay. The first phase corresponds to the aircraft clustering formation, and here, we aim to increase the AANET stability by creating spatially correlated clusters. The second phase consists of the A2A link determination for reducing the packet transfer delay. Finally, the cluster head selection increases the packet delivery ratio in AANET. According to our simulations, we see that the stability and packet delivery ratio of AANET topology are roughly improved 35% and 31% with a 28% delay reduction compared to the methods in the literature.

3.2 Introduction

The evolving technology and increasing number of passengers leading to Internet connectivity during a flight have become crucial requirements. The passengers want to access the high-quality Internet regardless of their location. This situation makes the IFC one of the most critical airline selection criteria for passengers. As given in Chapter 1, approximately 75% of passengers are ready to change the airway they use to get faster and uninterrupted Internet access, while 20% have changed the airway they

use. Also, the IFC also becomes a vital income source for airlines. More generally, the industry is expected that the IFC will create 130\$ billion global markets up to 2035. For these reasons, the aeronautical networks take both industry and academia's attention to enable high-quality IFC to passengers.

The AANETs are created by establishing links between airplanes to satisfy passengers' IFC requirements. In AANETs, a packet from one aircraft is transmitted through these links until it reaches the destination, enabling an Internet connection through the Ground Station (GS). On the other hand, the unstructured and ultra-dynamic characteristics of aircraft make AANET management compelling. In this chapter, we investigate these challenges in terms of topology formation management as detailed in the following subsection.

3.2.1 Topology formation challenges in AANETs

As explained above, unstructured and ultra-dynamic characteristics of aircraft make topology formation challenging in AANET [128]. In this chapter, we investigate these topology formation challenges from three different perspectives as detailed follows:

- *Stability Challenge:* The main reason for the stability problem is the shorter longevity of A2A links, which originates from the unstructured and ultra-dynamic characteristics AANETs. Here, we can define longevity as the connection life of the links between two aircraft, which is shortened due to the AANET's specific features. Accordingly, in an AANET having shorter link longevity, we observe

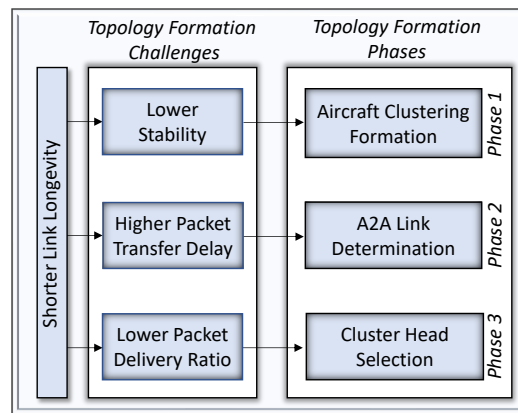


Figure 3.1 : The topology formation challenges and phases of chapter.

lower stability due to frequent connection establishment and termination procedures between airplanes.

- *Packet Transfer Delay Challenge:* The other effects of the shorter link longevity are related to the packet transfer efficiency of AANET. The unstructured and ultra-dynamic AANET with short-lived A2A links makes packet transfer difficult. Its packets must be transferred to a destination with Internet connectivity to provide Internet access to an aircraft. However, an optimal route may not be chosen during this transfer since the connections are constantly changing. In this situation, the packet transfer with lower delay becomes challenging.
- *Packet Delivery Ratio Challenge:* As explained above, the AANET with shorter link longevity increases packet transfer delay. The higher packet delay also supports the lower delivery ratio because of the growing queue waiting times. The main reason for this situation is that the growing waiting time increases the packets' drop rate from the queue. Additionally, the constantly changing connections also support this increase in packet drops.

Considering the leading causes of these three challenges, we can define link longevity as our main problematic parameter. It leads to three challenges: lower stability, higher packet transfer delay, and lower packet delivery ratio, as shown in Figure 3.1. In this chapter, we aim to investigate these three challenges in three main phases as Aircraft Clustering Formation, A2A Link Determination, and Cluster Head Selection as summarized in Figure 3.1.

3.2.2 Related work

Three challenges given in above will be solved by deploying different topology formation phases. If we divide the topology formation into three phases and perform each of them using the corresponding algorithms as shown in Figure 3.1, we can ensure that these phases solve these three challenges. Here, one of the conventional approaches to creating stable AANET topology with higher link longevity is the clustering of airplanes, as explained in Chapter 1 and Chapter 2. The aircraft clustering aims to collect the airplanes with similar features in a common set [129]. Thanks to

the grouping of airplanes with similar characteristics, the more stable topology with higher link longevity could be obtained. Also, the determination of heads in these clusters plays an essential role in increasing the success of packet delivery ratio while the correctly positioned A2A links reduce the packet transfer delay, as shown in Figure 3.1.

We investigated the clustering algorithms in Chapter 1. In addition to the clustering algorithms, different link stability algorithms are proposed for AANETs in the literature. The A2A links between the airplanes flying in the same direction are established by detecting available link probability on the topology [130]. Similarly, the A2A links are managed by calculating the duration of these links with the Doppler shift. There are also different studies proposing cluster head selection methodologies. The Random Competition-based Clustering (RCC) scheme based on the broadcasting mechanism is proposed in [131]. Here, a node initiates the cluster formation through packet broadcasting and defining itself as a head. Then, the neighbor nodes accepting broadcast become a member of this cluster. However, these broadcast messages could be created continuously, which can increase the delay during cluster determination. A two-step distributed clustering algorithm called Mobility-based clustering (MOBIC) is proposed in [132]. In this work, all nodes transfer/receive hello messages to/from their neighbors to calculate the aggregate relative mobility metric. The node having the smallest aggregate relative mobility metric is defined as cluster head. However, if cluster heads move into others' range, re-clustering should be executed according to the MOBIC. Similarly, the connectivity for the initialization of cluster heads is utilized by proposing Lease Cluster Change (LCC) in [133]. On the other hand, the head may not prevail when it moves into a cluster. But, this situation is not a problem for the non-cluster head nodes.

Although numerous studies suggest clustering, A2A link management, and cluster head selection algorithms, these three main management issues are not investigated simultaneously by the literature. Clearly, each of these three phases is tried to be solved independently in a different study. However, if we can collect these three management issues under one topology formation mechanism, then the lower stability, higher packet transfer delay, and lower packet delivery ratio challenges could be

solved synchronously. More specifically, these three problems are directly related to the aircraft cluster formation, A2A link determination, and cluster head selection topological issues, as shown in Figure 3.1. Accordingly, to obtain a complete AANET topology solving these three main problems, we consider all of the above-listed issues under one management model. The general contributions of this topology formation model are listed in the following subsection.

3.2.3 Contributions of the chapter

In this chapter, we propose a three-phased topology formation for AANETs. Here, we aim to form the topology of an AANET in three phases as Aircraft Clustering Formation, A2A Link Determination, and Cluster Head Selection. Our main objective is to divide the topology formation into three phases and perform them using corresponding algorithms under one management model. The main contributions of the chapter could be listed as follows:

- The first phase corresponds to the Aircraft Clustering Formation, and here, we aim to form a more stable AANET topology by creating correlated clusters with higher link longevity.
- The second phase corresponds to the A2A Link Determination, and here we aim to reduce the packet transfer delay by determining A2A links having shorter distance and delay.
- The third phase corresponds to the Cluster Head Selection, and here, we aim to increase the packet delivery ratio by selecting highly connected and less loaded cluster heads, as shown in Figure 3.1.

3.2.4 Organization of the chapter

The rest of the chapter is organized as follows: This chapter first presents our proposed topology formation model by detailing the AANET topology and topology formation framework. Then, the performance of the proposed model is evaluated. Finally, we conclude the chapter.

3.3 Three-Phased Clustered Topology Formation Managment for AANETs

3.3.1 AANET topology

Based on the AANET topology given in Chapter 1, this chapter considers a three-layered AANET topology consisting of ground, aircraft, and satellite layers, as shown in Figure 6.4. The ground and satellite layers consist of GSs and satellites, respectively, to enable Internet connectivity. Accordingly, an aircraft in the aircraft layer can satisfy the Internet connectivity requirement through satellite or GS. The aircraft layer consists of a high number of aircraft which are connected through A2A links. In this chapter, our primary focus is on the topological aspects of AANETs. For this reason, we use an idealized communication link model with an omnidirectional transmission range. More specifically, these links have LOS characteristics with a relatively high SINR since the available transmit power or battery power are not a limitation for aircraft. The Earth's curvature limits the theoretical maximum transmission range of this LOS communication among aircraft and aircraft's flight level [134]. Additionally, the proposed AANET topology formation framework is embedded in aircraft, and it constitutes the most critical component of this system, as illustrated in Figure 6.4.

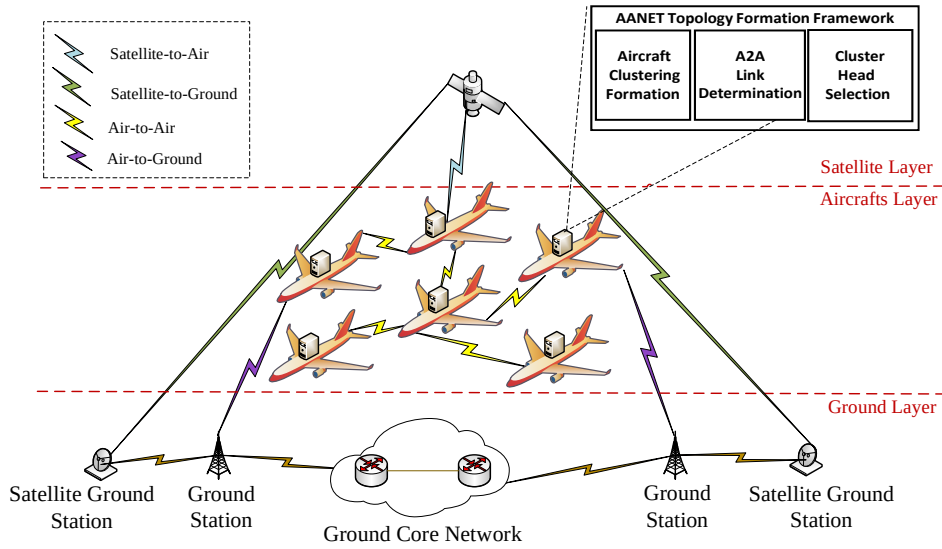
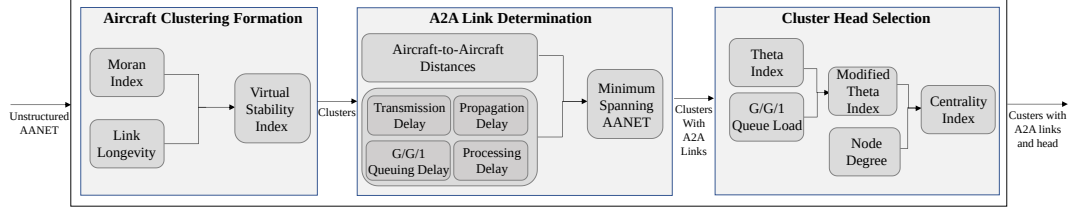
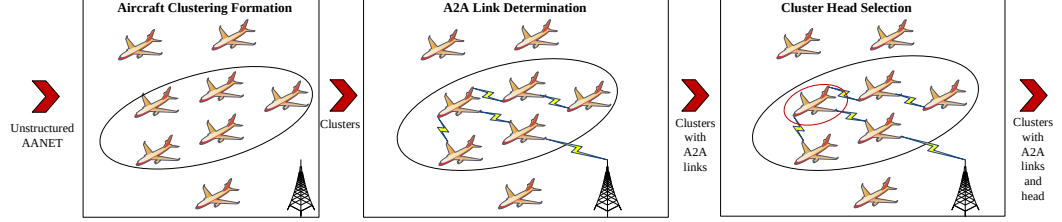


Figure 3.2 : The system model of AANET with proposed framework.



(a) Topology formation framework.



(b) Illustration of three-phased topology formation.

Figure 3.3 : Topology formation framework and its illustration.

3.3.2 Topology formation framework

In this chapter, our main aim is to form an AANET topology having higher stability, lower packet transfer delay, and a higher packet delivery ratio. To achieve this aim, we divide the AANET topology formation into three phases, and each of them focuses on solving one of these challenges under one management model. Correspondingly, we propose a three-phased topology formation framework for AANETs, as shown in Figure 3.3(a). This framework consists of three main phases as Aircraft Clustering Formation, A2A Link Determination, and Cluster Head Selection as detailed follows:

3.3.2.1 Aircraft clustering formation

The first phase corresponds to the *Aircraft Clustering Formation*, and here, we aim to create clusters consisting of spatially correlated aircraft to increase the stability of AANET topology. We obtain a virtual stability index during the cluster formation by exploiting the Moran Index and link longevity parameters in spatial autocorrelation, as shown in Figure 3.3(a). This index can collect the correlated aircraft under one cluster to form a more stable topology. Therefore, this phase shapes the unstructured AANET topology in the form of clusters, as illustrated in Figure 4.3.

We start to explain the clustering formation procedures by explaining link longevity. The connection life of the links between two aircraft could be defined as the link longevity, and it could be formulated with equation 3.1 in LLR as explained in Chapter 1. In this formula, $\vec{D}$ and $\vec{V}$ show the relative position and velocity vector between the neighbor aircraft j and corresponding ones i . Additionally, the R offers the transmission range of an airplane. Here, the shorter longevity of A2A links is the main topological problem of an AANET since this situation reduces the stability of the AANET topology. For this reason, in the first phase, our main aim is to increase the stability of the AANET topology by improving the longevity of A2A links. To achieve this, we should collect the spatially correlated aircraft under common clusters.

$$T_{Lij} = \frac{-\vec{D}_{ij}\vec{V}_{ij} + \sqrt{(\vec{D}_{ij}\vec{V}_{ij})^2 + V_{ij}^2(R^2 - D_{ij}^2)}}{V_{ij}^2} \quad (3.1)$$

During this clustering, we utilize spatial autocorrelation to determine airplanes' dependence since it can calculate the spatial correlation between the same variable's values in different places in space. With this methodology, if spatial autocorrelation is positive between neighboring aircraft, we can group them in the same cluster. On the other hand, if we have negative spatial autocorrelation, the nearby locations should exist in different geographic clusters. Here, the Moran Index helps us to calculate the spatial autocorrelation because it can work in space in two or more dimensions with high flexibility [135]. However, the Moran Index works by calculating the distance metric between two points. This makes it unpractical for AANETs since the distance between the two aircraft can change continuously in this environment. Instead, we estimate the link longevities between aircraft and use it with correlation; we can form a more stable topology by considering AANET specific characteristics. For this reason, in this chapter, we modify the Moran Index by employing link longevity (T_L) given with equation 3.1 in autocorrelation instead of distances to increase stability as formulated with equation 3.2. Then, we call this parameter a virtual stability index (V_S) because it helps us improve the stability of AANET.

In equation 3.2, W_{ij} represents the connectivity matrix of an aircraft in $n \times n$ dimensions. The aircraft receives the neighbor aircraft information through an ADS-B

system to find this matrix in an AANETs.

$$V_S = \frac{n}{\sum_{i=1}^{i=n} \sum_{j=1}^{j=n} W_{ij} \frac{\sum_{i=1}^{i=n} \sum_{j=1}^{j=n} W_{ij} (T_{L_i} - \bar{T}_L)(T_{L_j} - \bar{T}_L)}{\sum_{i=1}^{i=n} (T_{L_i} - \bar{T}_L)^2}} \quad (3.2)$$

While creating this matrix, the existent and nonexistent connections are represented with 1 and 0, respectively. Here, n represents the neighbor aircraft number. Additionally, $\bar{T}_L$ represents the arithmetic mean of all link longevities. However, the V_S given with equation 3.2 is an inferential statistic, and it does not mean anything alone. Therefore, we should assess its statistical significance by calculating the z-value. Here, to estimate the statistical significance of V_S , we utilize the randomization hypothesis. Accordingly, we first calculate the variance of this index ($Var(V_S)$) according to the Randomization hypothesis as given with equation 3.3.

$$\begin{aligned} Var(V_S) &= \frac{((n[(n^2 - 3n + 3)S_1 - nS_2 + 3S_0^2]) - (k[(n^2 - n)S_1 - 2nS_2 + 6S_0^2]))}{(n-1)(n-2)(n-3)S_0^2} - E(V_S^2) \\ E(V_S) &= \frac{-1}{n-1} \\ S_0 &= \sum_{i=1}^{i=n} \sum_{j=1}^{j=n} W_{ij} \\ S_1 &= \frac{\sum_{i=1}^{i=n} \sum_{j=1}^{j=n} (W_{ij} + W_{ji})^2}{2} \\ S_2 &= \sum_{i=1}^{i=n} (W_i \cdot + W \cdot i)^2 \\ k &= \frac{\frac{\sum_{i=1}^{i=n} (T_{L_i} - \bar{T}_L)^4}{n}}{(\frac{\sum_{i=1}^{i=n} (T_{L_i} - \bar{T}_L)^2}{n})^2} \end{aligned} \quad (3.3)$$

In equation 3.3, the S_0 equals the sum of the connectivity matrix (W_{ij}). Similarly, the S_1 and S_2 are also related with the connectivity matrix (W_{ij}) and they are calculated as given with equation 3.3. More specifically, the $E(V_S)$, S_0 , S_1 , S_2 , and k are the intermediate stages to find the variance $Var(V_S)$ of the virtual stability index (V_S). Also, in this equation, k is a constant value. After finding the $Var(V_S)$ and $E(V_S)$, we can determine the z-score of V_S as formulated in equation 3.4.

$$Z_{V_S} = \frac{V_S - E(V_S)}{\sqrt{Var(V_S)}} \quad (3.4)$$

In equation 3.4, the calculated Z_{V_S} is used to determine the virtual stability index's statistical significance since it is an inferential statistic, and its significance should be evaluated with hypothesis testing. In this hypothesis testing, the Null hypothesis (H_0) and Alternate hypothesis (H_1) represent the spatial autocorrelation and no spatial autocorrelation between neighbors, respectively. Accordingly, the high positive z-score shows a significant correlation, which means that the aircraft could be clustered by accepting H_0 . However, the low negative z-score represents a statistically significant spatial outlier. Therefore, if we can accept H_1 , then the aircraft should not be clustered together. Finally, the z-value near zero indicates randomly or independently distributed aircraft. To determine this statistical significance, we should compare the Z_{V_S} with the corresponding range according to the confidence level as given with Algorithm 4. (For example, z-score < -1.96 or z-score > 1.96 for 95% confidence level.) As shown in Algorithm 4, if the Z_{V_S} is out of the confidence range, we should accept the H_0 and put the corresponding aircraft into the correlated cluster. Here, the time complexity of the Algorithm 4 is calculated as $O(n^2)$. Also, after executing the Algorithm 4 for all aircraft, we can shape the unstructured AANET in the form of correlated clusters as shown in Figure 3.3.

Therefore, the communication complexity of this algorithm is determined by the number of aircraft in the unstructured AANET and the number of formed correlated clusters.

At the end of the first phase, we decided which planes should be included by the same cluster without thinking about the connections between them. Therefore, generating correlated clusters from the unstructured AANET and forwarding them become the trigger for the second module to start processing. In the following subsection, we will examine the processes that begin with this triggering in our second module. More specifically, we will determine the A2A links between the aircraft in the same cluster through *A2A Link Determination*.

Algorithm 1 Aircraft Clustering Formation

Data: Unstructured AANET**Result:** Aircraft Clusters**Initialization:** n : number of neighbor aircraft**while** $i \leq n$ **do** Calculate T_{L_i} **while** $j \leq n$ **do** Calculate T_{L_j} Calculate connectivity matrix (W_{ij}) of aircraft i Calculate V_{S_i} , $E(V_{S_i})$, and $Var(V_{S_i})$ Find $Z_{V_{S_i}}$ **if** $Z_{V_{S_i}} \geq \text{confidence_range}$ **then** Accept H_0 and Put aircraft i to correlated cluster **else** Accept H_1 **end** **end****end**

3.3.2.2 A2A link determination

After forming the clusters, we should determine the links between the aircraft in each cluster. Correspondingly, the second phase consists of the *A2A Link Determination*,

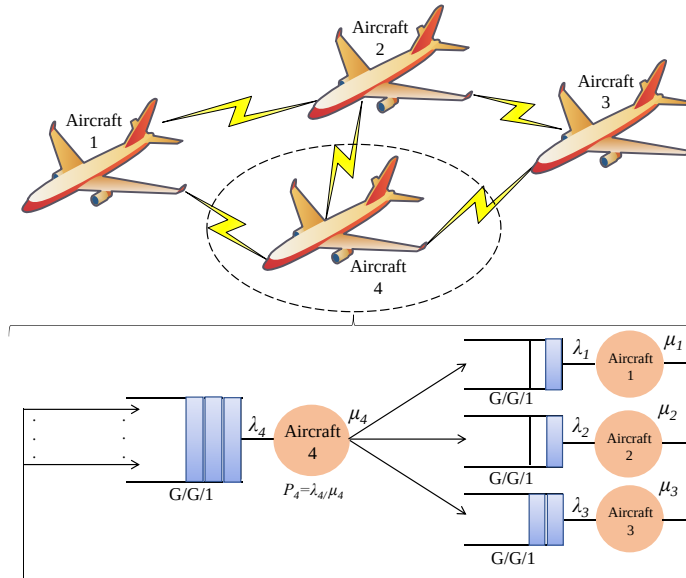


Figure 3.4 : G/G/1 queuing model of an aircraft in AANET.

and here we aim to determine the A2A links to reduce the packet transfer delay of

AANET. To get it, we modify the delay bounded minimum spanning tree (MST) algorithm with our proposed delay definition, including the G/G/1 system-based queuing delay, as shown in Figure 3.3(a). By obtaining the minimum spanning AANET, we can minimize the lengths of links between the aircraft on the network by considering delay bound. More specifically, we can observe lower packet transfer delay if we create the shorter A2A links by considering delay bounds. Therefore, we can obtain clusters with A2A links at the end of this phase by utilizing the outputs obtained from phase 1, as shown in Figure 4.3.

Before detailing the delay bounded-MST algorithm modification, we should explain our packet transfer delay model for AANETs since we update this algorithm by arranging the delay definition according to the packet transfer delay of AANETs. For this aim, as shown in Figure 6.7, we first model each aircraft as G/G/1 queues since the packets come as unexpected arrivals with high variability.

This model will allow us to find the queuing delay (T_{q_j}) of aircraft j , which is a crucial component of the total delay (T_{ij}) between aircraft i and j . Accordingly, λ_j and μ_j are the mean arrival and service rates of the packets for aircraft j with $\rho_j = \lambda_j/\mu_j$, ($j \in 1, 2, \dots, N$) as illustrated on Figure 6.7 [136]. Here, N shows the maximum aircraft number. After obtaining the queuing model of AANET, we can calculate the queue waiting time of aircraft j as formulated with equation (3.5). Also, in equation (3.5), the $\sigma_{a_j}^2$ and $\sigma_{s_j}^2$ denote the variance of inter-arrival and service times of aircraft j . After calculating queuing delay, we can write the total delay as given with equation (6.6).

According to equation (6.6), the delay equals a total of $T_{r_{ij}}$, $T_{p_{ij}}$, T_{pr_j} , and T_{q_j} , which denote the transmission, propagation, processing, and queuing delays of the packet transfer, respectively. Here, the transmission delay is calculated with $T_{r_{ij}} = \frac{L}{B}$. In this equation, L and B show the packet length and bandwidth, respectively. The processing delay equals to $T_{pr_j} = \frac{1}{\lambda_j}$ according to the G/G/1 system as given with Figure 6.7. The propagation delay is found as $T_{p_{ij}} = \frac{d_{ij}}{S}$. Here, S shows the transmission speed while d_{ij} is the distance between aircraft i and j . Finally, the queuing delay (T_{q_j}) is found by utilizing equation (3.5).

$$\begin{aligned}
T_{q_j} &\approx \frac{\rho_j(1+C_{s_j}^2)(C_{a_j}^2+\rho^2C_{s_j}^2)}{2(1-\rho_j)(1+\rho_j^2C_{s_j}^2)\lambda_j} \\
C_{a_j}^2 &= \frac{\sigma_{a_j}^2}{(1/\lambda_j)^2} \\
C_{s_j}^2 &= \frac{\sigma_{s_j}^2}{(1/\mu_j)^2}
\end{aligned} \tag{3.5}$$

$$T_{ij} = T_{rij} + T_{pij} + T_{prj} + T_{qj} \tag{3.6}$$

After defining the total delay (T_{ij}), we can determine the A2A links by modifying the delay bounded-MST algorithm according to this definition [137]. Before deciding the A2A links, we should also choose the IGW aircraft connected to the GS. Here, we select the plane that minimizes the distance (d) with the GS as IGW aircraft as shown in Algorithm 2. However, a cluster may not have an IGW due to the coverage range (r_{GS}) of the GS, as expressed in Algorithm 2. If even the aircraft closest to the GS cannot exceed this range, an IGW is not assigned to that cluster. This coverage range depends on the antenna characteristics, signal quality, and GS's signal level. Accordingly, if the cluster does not have an IGW, the IFC requirement is provided through the head node by communicating with other clusters. After choosing the IGW for clusters, we should determine the A2A links inside the clusters. Here, we first select an aircraft that minimizes distance with current ones and then calculate the total delay.

If this link delay is less than the cluster delay bound (Δ), we can establish a link between aircraft i and j . If a cluster only has the IGW to GS connection as an established link, we take it as a Δ . Conversely, if a cluster already has the established A2A links, we take the arithmetic mean to calculate the Δ . Here, the time complexity of the Algorithm 2 is calculated as $O(n)$. Also, after applying this procedure for all nodes, we can obtain clusters with A2A links in the form of minimum spanning AANETs, as summarized in Figure 3.3. Therefore, the communication complexity of this algorithm is determined by the number of accepted clusters and the number of minimum spanning AANETs with A2A links.

Algorithm 2 Minimum Spanning AANET

Data: Aircraft cluster (Received from *Phase 1*)

Result: Aircraft cluster with A2A links

Initialization:

N : total number of aircraft in cluster

$aircraft_index$: number of aircraft establishing link

T : total delay

r_{GS} : range of GS

Set lists for each aircraft consisting of all airplanes in cluster

Select GS as source node

Find *current aircraft* that minimizes distance(d) with GS

if $d \leq r_{GS}$ **then**

 Create link between *current aircraft* and GS

 Set $aircraft_index \leftarrow 1$

 \\ A2A Link creation

while $aircraft_index \leq N$ **do**

 Select aircraft minimizing distance with current aircraft

 Delete the selected aircraft from the list of current aircraft

 Calculate T between selected and current aircraft

if $T \leq \Delta$ **then**

 Create link between selected and current aircraft

 Set selected aircraft as current aircraft

 Set $aircraft_index \leftarrow aircraft_index + 1$ **else**

 Continue

end

end

else

 Use A2A link creation without GS connection

 Select cluster head

end

After executing the second phase, we determined the clusters with A2A links and in the form of minimum spanning AANET. Here, the accepted clusters from phase 1 become a trigger for this module to start processing. Additionally, the generated clusters with A2A links trigger the operations of our third module. In the upcoming subsection, we will detail the procedures that begin with the triggering of the third module. Clearly, we will select the heads of these formed clusters through *Cluster Head Selection*.

3.3.2.3 Cluster head selection

After the determination of clusters and A2A links, we should select a head for each cluster. Accordingly, the third phase consists of the *Cluster Head Selection*, and

here, we aim to choose a head for each cluster to increase the packet delivery ratio of AANET. To enable this, we choose the highly connected and less loaded node as a cluster head. During this selection, we modify the theta index with a G/G/1 system-based queuing load and combine it with node degree to obtain a centrality index, as shown in Figure 3.3(a). At the end of this final phase, we can get clusters having A2A links and heads as output, as shown in Figure 4.3.

We start the head selection with the definition of the centrality index (C_I). It gives information about an aircraft's centrality and load by utilizing theta index (θ_I) and G/G/1 queuing load (L_q). In our AANET topology, we have two types of clusters as with IGW and without IGW. To control the clusters without IGW, we should determine the head for these clusters. This head becomes the contact point between the clusters to enable communication. More specifically, the head of a cluster without IGW communicates with the head of other clusters. Additionally, the head should communicate with as many nodes as possible from its cluster. Therefore, the connectivity of the cluster head must be higher than other aircraft. On the other hand, the communication complexity of the cluster head increases the load on it. This overloading also results in a reduced packet delivery ratio. Therefore, we should deal with the head's load to increase the packet delivery ratio in AANET. For these reasons, our main objective is to select an aircraft that maximizes the C_I as cluster head since it consists of theta index (θ_I) and G/G/1 queuing load (L_q) parameters as shown in Algorithm 2. Accordingly, to calculate the centrality index, we first require the queuing load for each aircraft, and these loads (L_{q_j}) are calculated by using equation 6.2 based on the G/G/1 model given in Figure 6.7.

$$L_{q_j} \approx \frac{\rho_j(1 + C_{s_j}^2)(C_{a_j}^2 + \rho_j^2 C_{s_j}^2)}{2(1 - \rho_j)(1 + \rho_j^2 C_{s_j}^2)} \quad (3.7)$$

In equation 6.2, j represents the elements in a cluster. Also, the $C_{a_j}^2$ and $C_{s_j}^2$ correspond to the squared coefficient variation of arrival times and service times to aircraft j , calculated as formulated in equation 3.5. After calculating queuing loads with equation 6.2, we utilize the estimated L_{q_j} to update the theta index (θ_I) as $\theta_{I_{Modified}} = L_{q_j}/N$. Here, N shows the maximum aircraft number in the investigated cluster. Also, instead

of traffic load, queuing load in the theta index increases the packet transfer efficiency by preventing the drops from the queue. Accordingly, by calculating $\theta_{I_{Modified}}$ with queuing load, we do not overwhelm an aircraft that has already loaded with extra communication traffic complexity of cluster head.

Algorithm 3 Cluster Head Selection

Data: Clusters with A2A links (Received from *Phase 2*)

Result: Head of cluster

Initialization:

N : total number of aircraft in cluster

i : corresponding aircraft

$head$: head of cluster

$cluster$: list keeping centrality indexes of cluster

Set $i \leftarrow 1$

while $i \leq N$ **do**

 Calculate queuing load (L_{q_i}) of aircraft i

 Calculate modified theta index ($\theta_{I_{Modified_i}}$) of aircraft i

 Calculate degree (d_i) of aircraft i according to delay bounded MST

 Find centrality index as $C_{I_i} = \theta_{I_{Modified_i}} d_i$

 Add C_{I_i} to list $cluster$

end

$head \leftarrow \max(cluster)$

Additionally, to satisfy the head's centrality characteristic, we add the node degree parameter (d_j) of aircraft j to this formula according to the degree centrality. To find the degree of each aircraft, we use delay bounded-MST obtained from the previous phase. Finally, we calculate the centrality index (C_I) of aircraft j as given with equation 3.8. Here, the time complexity of the Algorithm 3 is calculated as $O(n)$. As a result of these operations, as illustrated in Figure 4.3, we can shape the unstructured AANET topology in the form of clusters having A2A links and heads. The communication complexity of this algorithm is calculated by the accepted number of minimum spanning AANETs with A2A links.

$$C_{I_j} = \frac{L_{q_j} d_j}{N} = \theta_{I_{Modified_j}} d_j \quad (3.8)$$

Additionally, these three phases work sequentially as a topology formation framework. The first phase receives the unstructured AANET and gives clusters as output. These clusters are taken by the second phase and given to the third module by adding

A2A links. Finally, the third module selects a head for each cluster and gives this information as output. Therefore, the unstructured AANET, generated clusters, and clusters with A2A links become the trigger for the first, second, and third modules, respectively.

3.4 Performance Evaluation

3.4.1 Simulation environment

In this chapter, we use the OMNET++ framework for the realistic air traffic simulations, and to obtain real air traffic data, we utilize the flight radar databases, as illustrated in Figure 3.5(a) [138]. Here, we consider 1200×2000 nautical mile (NMI ≈ 1852 m) air traffic area, as shown in Figure 3.5(b). We placed four GSs at each corner of the simulation area, as shown in Figure 3.5(b).

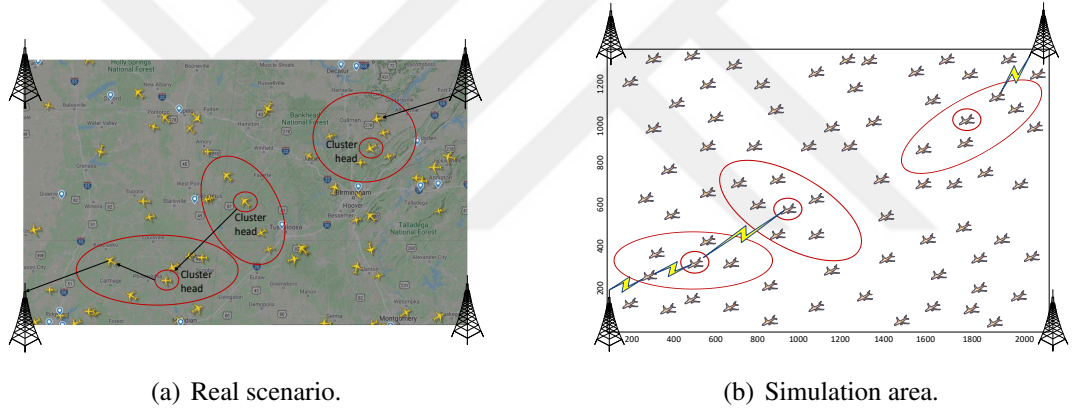


Figure 3.5 : Simulation details.

The wireless channel capacity of each GS is 1 Mbit/sec [139]. This channel capacity gives the bound for the reliable packet transfer rate over the A2A links. We take the distance between aircraft and GS as 200 NMI to establish a connection between these. Accordingly, if the distance between an aircraft and GS is less than 200 NMI (r_{GS}), then we can establish a connection between them as explained with Algorithm 2. Here, if a cluster does not have an IGW, it enables IFC by connecting other cluster heads, as shown in Figure 3.5(a). Moreover, the receiver sensitivity and transmit radio frequency power are taken as -24.67 dBm and 125 W, respectively. Here, the receiver sensitivity shows the minimum signal received power required at the receiver to achieve a certain

quality [140]. On the other hand, the transmit radio frequency power gives the impact of transmitters.

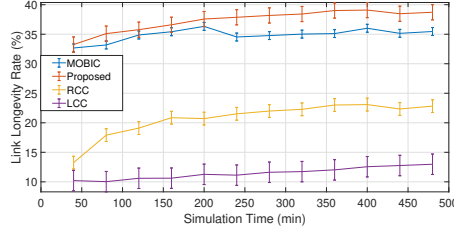
In addition to these, we consider that the A2A communication range is 450 NMI at 35000ft flight altitude as explained in Chapter 1. Accordingly, if the distance between two aircraft is less than 450 NMI, we can say that there is a link between them. However, the direction, angle, and velocity alterations of airplanes disrupt this distance. Accordingly, the established links could be easily broken due to the dynamic environment of AANET. More clearly, the failure to maintain this distance constitutes the main problem in terms of link longevity. For this reason, this is the minimum value for the A2A link establishment, and the continuity of links also depends on the mobility pattern of an aircraft. Moreover, the bandwidth values of these links are taken as 10 Mbps.

Also, we assume that the airplanes can produce up to 4 packets per second, and the size of each packet is set to 512 kB. The airplanes continue their movements typically during the simulation based on the real air traffic data. We model each aircraft according to the G/G/1 queuing system, as shown in Figure 6.7. Additionally, to evaluate the z-score related to the virtual stability index, we use a 95% confidence level.

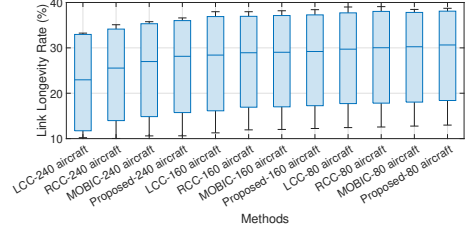
3.4.2 Simulation results

After determining the simulation parameters, we investigate the link longevity rate, packet transfer delay, and packet delivery ratio metrics to evaluate the proposed topology formation mechanism's performance. Here, to evaluate these metrics, we mainly utilize the given heuristics as Algorithm 4, Algorithm 2, and Algorithm 3. Also, we compare the proposed mechanism with the existing works, LCC, RCC, and MOBIC, as detailed in Section 3.2.2. Furthermore, before explaining the simulation results, we illustrate the movement phases of airplanes, as shown in Figure 1.3 since we select three different movement types for our simulations as *Climb*, *Cruise*, and *Descent*.

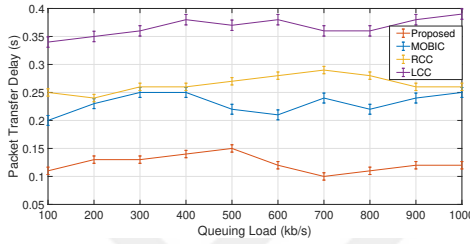
- **Climb Phase:** We first simulated the airplanes in the climb phase during our evaluations. The main selection reason for the climb phase is that the airplanes increase their altitudes during this phase. Here, we utilize the link longevity rate, packet transfer delay, and packet delivery ratio parameters for simulations.



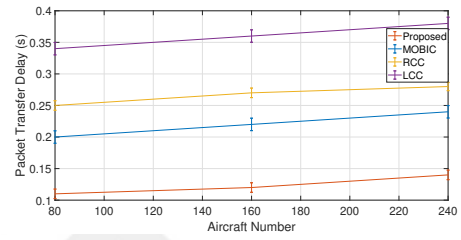
(a) Link longevity vs. Simulation time



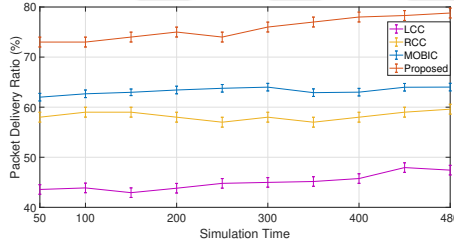
(b) Link longevity vs. Aircraft number



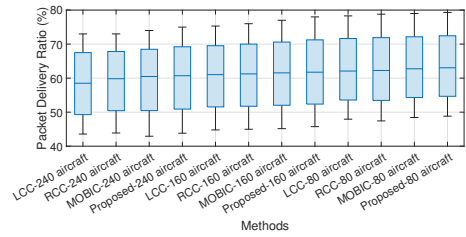
(c) Packet transfer delay vs. Queuing load



(d) Packet transfer delay vs. Aircraft number



(e) Packet delivery ratio vs. Simulation time



(f) Packet delivery ratio vs. Aircraft Number

Figure 3.6 : Evaluations for the climb phase.

Link Longevity Rate: The results of the link longevity rate are shown in Figure 3.6(a) and Figure 3.6(b). To obtain these results, we randomly select a cluster, and then we analyze the link longevitys of the airplanes in this cluster by utilizing equation 3.2. However, the airplanes can change their clusters throughout the whole movement, and clusters cannot have a stable aircraft number. For this reason, in Figure 3.6(a), each different method's given results include the average link longevitys of the airplanes in this cluster in the time interval of 480 min. More specifically, due to the ultra-dynamic characteristic of AANETs, we take the average of results over 20 min periods. These assumptions are also valid for the given results in Figure 3.6(e), Figure 3.7(a), Figure 3.7(e), and Figure 3.8(a).

Here, the LCC and RCC are not compatible with the dynamic re-clustering since the cluster head can only be replaced when two cluster head moves to a range of others in these algorithms. The MOBIC works based on the local mobilities of neighbors. But, this mobility could be affected by the ultra-high velocities of airplanes and atmospheric effects. These conditions reduce the link longevity rate of these algorithms. In our proposed clustering mechanism, we aim to increase the link longevity rate by deploying it in the virtual stability index given with equation 3.2. Thanks to our methodology, we can link longevity at the 40% level while others fall below 35%. In the second step, we evaluate the link longevity rate according to the increasing aircraft number. Here, we select 80, 160, and 240 airplanes according to the available realistic data in the time interval of 480 min. These airplanes have different movement directions, angles, and characteristics from the firstly considered 30 airplanes. The increase in aircraft number results in the degradation of link longevity in each method since many aircraft cause a rapid change of clusters. However, our proposed method achieves more stable link longevity at the 30% level than the other three methodologies thanks to the advantage of spatially correlated clustering as given Algorithm 4.

Packet Transfer Delay: The queuing load of an AANET has an essential effect on the packet transfer delay. For this reason, we investigate the delay of the proposed approach according to different queuing load levels. Here, the queuing load is determined according to the arrival and service rates of the airplanes, as formulated in equation 6.2. Although we have a constant packet size, the accepted and transferred packets are different for each airplanes. Therefore, by changing the service and arrival rates of airplanes, we can obtain different queuing loads, as shown in Figure 3.6(c) for the analysis. Here, the y-axis belonging to the queuing load of 100 kb/s represents the average packet transfer delay of 30 airplanes with this load as an example. This packet transfer delay is calculated by using the total delay equation as given in equation 6.6. Then, we take the average of these delays observed during the whole movement of 30 airplanes. Accordingly, as shown in Figure 3.6(c), we can reduce the packet transfer delay, roughly 34% thanks to the Algorithm 2, compared to the other three methods. In the second step, we evaluate

the packet transfer delay according to the increasing aircraft number. Here, we again select 80, 160, and 240 airplanes according to the available realistic data in the time interval of 480 min. We take the average of delay results over 20 min periods for all available airplanes from a global perspective. For example, we take the average of results belonging to 80 airplanes over 20 min period according to the LCC method to find a delay. Accordingly, as we expected, we obtained 28% lower packet transfer delay compared to the other three methods thanks to the queuing delay (T_q) consideration in Algorithm 2 as shown in Figure 3.6(d). Here, the broadcast and hello messages constitute the main reasons for the delay increase for the RCC, MOBIC, and LCC mechanisms.

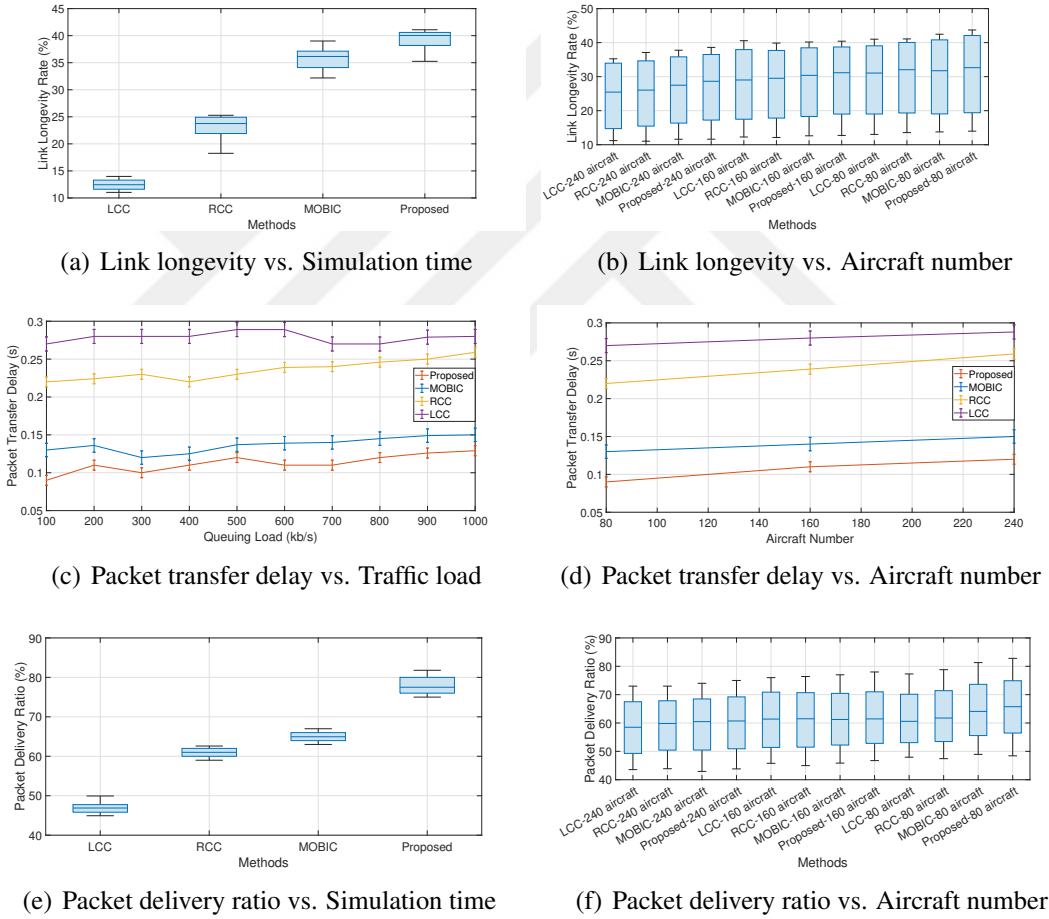


Figure 3.7 : Evaluations for the cruise phase.

Packet Delivery Ratio: The packet delivery ratio shows the number of successfully received packets among the total sent packets. Here, we first investigate the packet delivery ratios of four methods according to simulation time. Here, we again

consider a randomly selected cluster with an unstable aircraft number. For this reason, we take the average of results over 20 min periods. According to our simulations, we can achieve a packet delivery ratio at the 80% level while others fall below 65%, as shown in Figure 3.6(e). Additionally, as shown in Figure 3.6(f), we can achieve a more stable packet delivery at the level of 65% according to the increased aircraft number thanks to the equation 3.8 consideration in Algorithm 3. Here, the total delay increase constitutes the leading cause of the reduced packet delivery in the other three methods.

- **Cruise Phase:** After evaluating the climb phase, we analyze the proposed model's performance for the airplanes in the enroute phase. It constitutes the central part of an aircraft flight by starting at the end of the climb and ending at the beginning of the descent. We utilize the same parameters and measurements used in the climb phase to evaluate the cruise phase's performance. Here, we can achieve higher link longevity at the 42% level while others remain below 40%, as shown in Figure 3.7(a). We also can achieve higher link longevity here compared to the climb phase. Due to the aircraft's altitude of the airplanes remains more stable here, different from the climb phase, significantly reducing the link breakages by increasing their longevity. Our proposed method can observe more stable link longevity, as shown in Figure 3.7(b), even if the aircraft number increases. The link longevity consideration mainly causes this in equation 3.2. Additionally, compared to the climb phase, we observe higher packet delivery ratios, as shown in Figure 3.7(c) and Figure 3.7(d). Here, the RCC, MOBIC, and LCC again observe higher packet transfer delays due to the transmission of broadcasting and hello messages during the cluster determination. However, we can watch a lower packet transfer delay than the climb since the clusters' characteristics become more stable in the cruise phase. Additionally, we again observe the shortest delay in our method thanks to the A2A link determination in the form of minimum spanning AANET given in Algorithm 2. This delay reduction also improves the packet delivery ratios of all methodologies, as shown in Figure 3.7(e) and Figure 3.7(f). The queuing drops could be prevented thanks to the delay reduction in the cruise phase. Accordingly, this situation plays a vital role in packet transfer success. Moreover, we can

experience the packet delivery ratio at the 83% level in our proposed approach, thanks to the load consideration in equation 3.8.

- **Descent Phase:** Finally, we evaluate the proposed model's performance during the descent phase since the aircraft decreases its altitude in this phase, contrary to the climb phase. Here, we obtain similar results with the climb phase, which are again worse than the cruise phase. Due to we cannot observe the cluster stability of the cruise phase in the descent phase. In the descent phase, we first evaluate the average link longevity of a cluster during the simulation time. We observe the link longevity at the 37% level in our proposed model, and this is better than the other three methods, as shown in Figure 3.8(a). Then, we calculate the average packet transfer

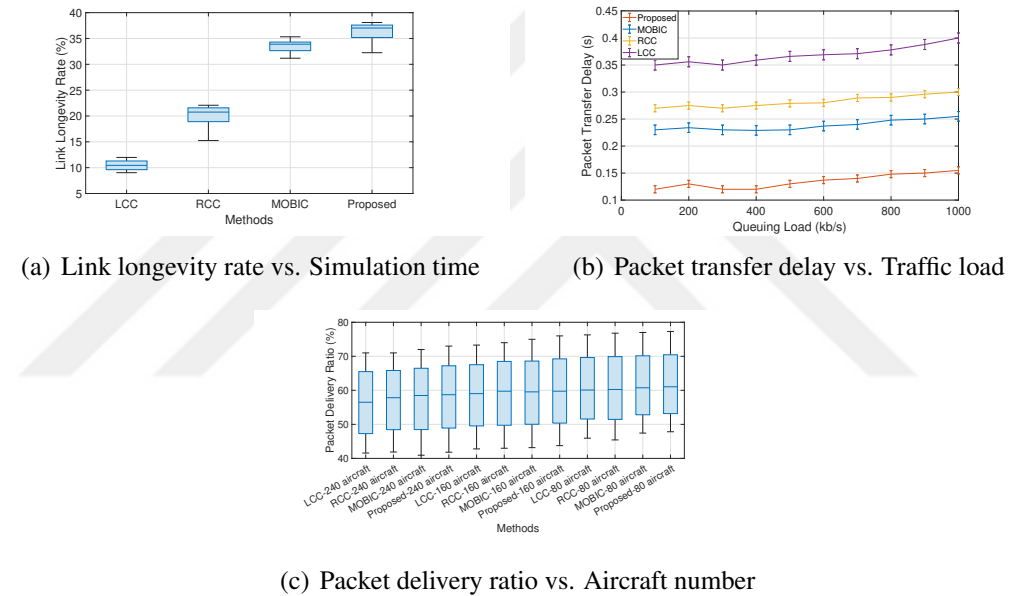


Figure 3.8 : Evaluations for the descent phase.

delay for different queuing load levels by utilizing the equation 6.6 and equation 6.2. According to simulation results, we can achieve a 30% lower packet transfer delay compared to the others, as shown in Figure 3.8(b) since we do not require any message exchange mechanism during the formation of AANET topology. This delay reduction also enables us to obtain a higher and more stable packet delivery ratio, as shown in Figure 3.8(c). Furthermore, we experience a very low rate of deterioration in evaluation results compared to the climb phase. We consider that the altitude reduction has a more significant impact on the alteration of clusters.

3.5 Conclusion

In this chapter, we propose three-phased dynamic topology formation for AANETs. Firstly, in the aircraft clustering formation phase, we utilize Moran Index and link longevity metrics in spatial autocorrelation. The second phase modifies the delay bounded minimum spanning tree to determine the A2A links. In the third phase, we modify the theta index by employing the G/G/1 queuing load, which is combined with the node degree. As shown in simulation results, the proposed model outperforms the LCC, RCC, and MOBIC methods' performance with roughly 35%, 31%, and 28% better stability, packet transfer delay, and packet delivery ratio parameters. The highly dynamic environment of AANET is a significant factor in the poor performance of the other three methods. Conversely, we can form an AANET topology having higher stability and packet delivery ratio with reduced packet transfer delay thanks to the proposed model.

General concept for the next chapter

In this chapter, we propose a solution to create the AANET topology in the form of clusters with higher stability. Here, our main aim is to manage the AANET topology formation procedures. However, it is not sufficient to create the AANET topology. Additionally, we should enable the sustainability of this formed topology in the form of clusters to conduct the routing management procedures.

Based on these, in the next chapter, we will propose a solution to manage the AANET topology sustainability procedures.

4. TOPOLOGY SUSTAINABILITY MANAGEMENT FOR AANETS

4.1 Purpose

The AANETs dramatically increase the Internet access rates of aircraft by widening the coverage area thanks to the A2A links established. However, the mobility and atmospheric effects on AANET increase the A2A link breakages, leading to frequent aircraft replacement and reducing link quality. These broken A2A links should be transferred to other aircraft to enable the sustainability of the AANET topology. At that point, the wrong and late transfer decisions of broken links make topology sustainability challenging by reducing packet transfer success and increasing end-to-end latency, respectively. Despite these challenges, to do best of our knowledge, A2A link transfers between aircraft in AANET are not investigated by any study to enable the topology sustainability. This chapter proposes to utilize the Learning Vector Quantization (LVQ) based methodology in three phases as winning cluster selection, intra-cluster link determination, and attribute update to enable the sustainability of AANET topology, which is in the form of aircraft clusters. In winning cluster selection, we consider each cluster in the topology as a pattern. Then, we aim to find the best matching cluster of an aircraft observing A2A link breakage through pattern classification. Then, we take airplanes in a cluster pattern as weight vectors with location and queuing delay attributes to determine the intra-cluster links of newly assigned aircraft. Here, the aircraft is modeled according to G/G/1 queuing system for delay attribute calculations. Finally, according to the freshly established intra-cluster links, we update weight vectors' attributes using the throughput rate as the learning rate. We simulate the proposed system by utilizing OMNET++ and Weka tools with realistic air traffic data obtained from flight radar databases. In these simulations, we can reduce the end-to-end latency of AANET topology by 25% and increasing the packet transfer rate by 31% compared to the methodologies in the literature.

4.2 Introduction

In the last decade, IFC becomes a crucial need for passengers with the tremendous improvement of machine-to-machine communications. On the other hand, long-haul flights are likely to pass through oceans and deserts where it isn't easy to connect the communication infrastructure. At that point, the AANETs attract both industry and academia's attention since it can provide a wide coverage area at a lower cost compared to its alternatives. More specifically, the AANETs satisfy the in-flight connectivity requirements of passengers establishing connections between aircrafts. This particular type of aeronautical network creates A2A links between aircraft to execute packet transfer, and during this procedure, all aircrafts act as a router. An aircraft's packets are continuously routed through these links until reaching an aircraft having Internet connectivity. A plane can access the Internet thanks to the AANET, even if it is not directly connected to the GS or satellite. Accordingly, this infrastructure results in the extended coverage area at a lower cost for in-flight connectivity.

4.2.1 Topology sustainability challenges in AANET

Although AANETs provide a wide connection network to aircraft, maintaining the sustainability of its topology becomes more difficult. The main reason for this difficulty is based on the higher breakage rate of A2A links causing by AANET specific effects. In this chapter, we investigate these effects from two different aspects as detailed follows:

- **Atmospheric Effects:** The atmospheric effects of the AANETs could be considered in oxygen absorption, rain, and cloud attenuations. Here, the large propagation distance between two aircraft causes the degradation of A2A link quality by increasing the effect of oxygen absorption. Similarly, the rain and cloud lead to frequent A2A link breakages since it can reduce the signal strength by increasing the absorption and scattering. Therefore, these three atmospheric effects significantly increase the A2A link breakages by reducing their qualities, as shown with *1a* in Figure 4.1.

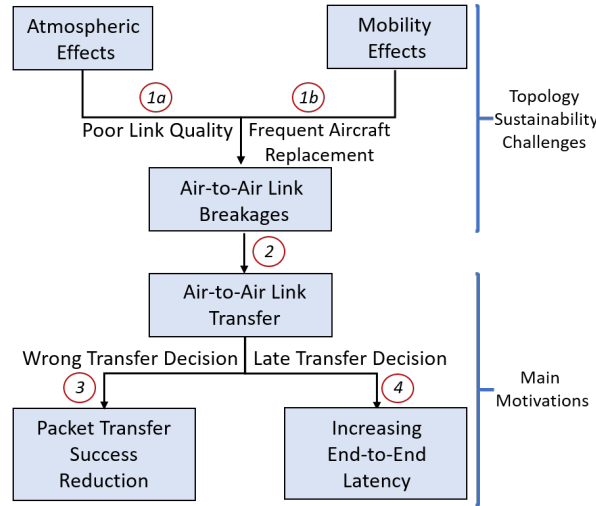


Figure 4.1 : Topology sustainability challenges in AANET.

- **Mobility Effects:** The mobility effects of the AANETs are based on the ultra-high-speed of aircraft. Here, this over-speed forms a highly dynamic AANET environment when it combines with the 3D movement characteristics. This dynamic environment experiences numerous A2A link breakages due to frequent aircraft replacements, as illustrated with *1b* in Figure 4.1.

When atmospheric and mobility effects come together, the breakage rate of A2A links in the AANET increases, these increased A2A link breakages make the sustainability of the AANET topology challenging, as shown in Figure 4.1. Here, the broken links, resulting from atmospheric and mobility effects, should be established again with other aircraft to ensure AANET topology's sustainability, as shown with 2 in Figure 4.1. On the other hand, determining the aircraft to which the link will be transferred is challenging due to the atmospheric and mobility effects. At that point, as shown with 3 the wrong transfer decisions reduce the packet transfer success, while the late transfer decisions increase the end-to-end latency, as demonstrated with 4 in Figure 4.1. Here, the late transfer decision means the aircraft's tardy determination to which the A2A links will be transferred. This situation significantly increases the end-to-end latency of A2A link transfer. The wrong transfer decision also represents an aircraft that moves an A2A link to another that should not be connected. The packet transfer success of AANET reduces with these wrong decisions since transmissions over this inaccurate connection will not reach the destination.

The reduced packet transfer success and increased end-to-end latency in transferring broken A2A links to ensure the AANET topology sustainability constitute this chapter's primary motivation. More specifically, in this chapter, our main aim is to transfer the broken A2A links to new aircraft to enable AANET topology sustainability with higher packet transfer success and lower end-to-end delay.

4.2.2 Related work

There are not numerous algorithms about AANET topology sustainability and performance evaluations in the literature. Some works enable this sustainability through handover management. The handover management algorithm consisting of three phases as information collection, handover decision, and handover execution is proposed in [141]. The information collection phase monitors the handover parameters like signal strength and compares these with the threshold values. The results of these comparisons are utilized in the handover decision phase to determine the handover, and the handover is executed based on Mobile IP and Resource Reservation Protocol. The energy efficiency, signal intensity, delay, and bandwidth are first defined to establish a utility function in [142]. They then take advantage of this utility function and multi-criteria utility approach to design an energy-efficient network selection approach. Additionally, the details about energy harvesting, efficiency, and transference concepts could be found in [143]. The congestion-aware handover strategy is proposed in [144]. They aim to control which aircraft is associated with IGW at any time, based on geographic proximity and a measure of IGW congestion.

The studies have investigated the establishment and termination procedures of links, mainly in terms of the physical and link layers, without considering topology sustainability from a management aspect. Also, although there are different handover algorithms, the sustainability of AANET topology is not viewed by any work at the network management level. On the other hand, the formation of A2A links is not sufficient for AANET topology; the transfers of these links also need to be arranged through sustainability management in case of possible breakage. Moreover, the effects of mobility and atmosphere are not considered simultaneously by these given works.

4.2.3 Contributions of the chapter

This chapter proposes a LVQ-based topology sustainability for AANETs to increase the packet transfer success and reduce end-to-end latency encountered during the transfer of broken A2A links. Here, the LVQ can utilize the formed AANET topology as training data for mapping the broken A2A links. Therefore, the LVQ can reduce the end-to-end latency observed during the A2A link transfer by preventing the execution of topology formation continuously. Also, the timely and correct transfer decisions of the LVQ can contribute to the packet transfer success of AANET with lower end-to-end latency. Correspondingly, the benefits of LVQ for AANETs strongly matches the main motivations of this chapter, and the main contributions of the chapter could be summarized as follows to utilize these benefits:

- We define an LVQ-based topology sustainability framework in three phases: Winning Cluster Selection, Intra-Cluster Link Determination, and Attribute Update.
- We consider the aircraft as clusters rather than separately. Accordingly, we can consider each cluster in the AANET topology as a pattern. Then, we aim to find the best matching cluster of an aircraft observing A2A link breakages through pattern classification in the winning cluster selection phase. These aircrafts are considered as the input vector to achieve this.
- To find the best matching cluster through pattern classification, we utilize the link longevity metric. Accordingly, we can find the most similar cluster to the input vector: the cluster that it can connect to for a longer time.
- We define each aircraft belonging to a cluster as a weight vector with location and queuing delay attributes. These attributes are used to determine the intra-cluster links between input and weight vectors after the cluster selection. Also, each aircraft is modeled according to G/G/1 queuing system to specify the delay attribute.
- We update the weight vectors' attribute after intra-cluster link determination by utilizing the throughput rate instead of the learning rate. With the throughput rate utilization, we can also take the effects of successfully delivered packets on queuing delay into consideration.

Additionally, the AI-based methodologies are not utilized by any work to manage the transfer of links for topology sustainability. However, the AI-based methodologies can overcome the effects of mobility and atmosphere on AANET that are the root causes of topology sustainability challenges as explained in Chapter 2. More specifically, we can handle the A2A link transfers by disabling the wrong and late decisions thanks to the learning methodologies in AI.

4.2.4 Organization of the chapter

The rest of the chapter is organized as follows: We first give the LVQ-based topology sustainability management for clustered-AANETs. Then, the performance of the proposed model is evaluated. Finally, the chapter is concluded.

4.3 LVQ-Based Topology Sustainability Management for AANETs

4.3.1 AANET topology

The considered AANET topology mainly consists of ground and aircraft layers. The ground layer includes the ground stations to enable Internet connectivity to aircraft through a direct A2G link. The aircraft layer consists of the aircraft connected through an A2A link to form an AANET. In this formed AANET, it is sufficient to have only one airplane with Internet connectivity through the GS, and other aircraft behave as routers to transfer the packets to it via established A2A links as shown in Figure 4.2. More specifically, aircraft do not need to be directly connected to the airplane with Internet connectivity. Instead, this connection could be realized using other aircraft connected to the airplane with Internet connectivity as routers. Additionally, as shown in Figure 4.2, we create AANETs in the form of clusters based on the formation model given in Chapter 2. Moreover, during the packet transfer through the A2A links, each aircraft behaves as a router with its queue. In this chapter, we model these queues according to the G/G/1 system with λ_i arrival rate, μ_i service rate, and $\rho_i = \frac{\lambda_i}{\mu_i}, i = 1, 2, \dots, M$ utilization as shown in Figure 4.2. This system operates in a self-organized manner instead of a centralized structure to facilitate the operation of AANET.

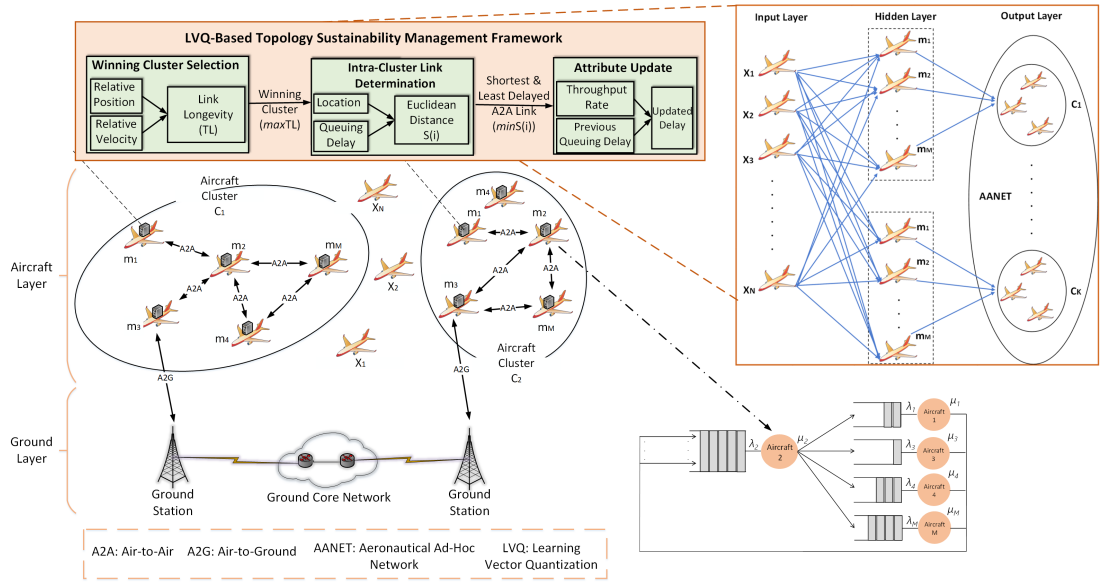


Figure 4.2 : AANET topology and system model.

After creating the topology of AANET in clusters, we should enable its sustainability without repeating the formation procedures continuously. Instead, we can select a formed cluster for a corresponding airplane if an A2A link is broken. In this chapter, we utilize the LVQ during this selection. The main reason for choosing the LVQ is that we already have an AANET topology before the A2A link breakage, and we can use it in supervised learning for training. More specifically, we can first consider the clusters in AANET topology as a pattern; then, we can find the best matching cluster of an aircraft observing A2A link breakages through pattern classification.

First of all, we should map the LVQ with the AANET, and here, the LVQ requires a multidimensional space containing pre-classified training data. Accordingly, the aircraft clusters consist of this pre-classified training data, while the AANET becomes a multidimensional space. In this mapping, there are three different layers: input, hidden, and output layers, as shown in Figure 4.2. Correspondingly, the input layer consists of the aircraft with broken A2A links. Also, the aircraft that are included by different clusters are located in the hidden layer. Finally, the output layer consists of the formed clusters, and these clusters together create the AANET. Here, each aircraft in the input layer is associated with aircraft in the hidden layer to determine the correct cluster for A2A link transfer. The other parameters of the LVQ and AANET mapping could be listed as follows:

- The $x_n, n = 1, 2, \dots, N$ is the *input vector*, and each input vector denotes an aircraft having broken A2A links at a corresponding time, as shown in Figure 4.2. Here, N shows the maximum number of broken A2A links at the appropriate time.
- As illustrated in Figure 4.2, the aircrafts in an AANET clusters become the *weight vectors* ($m_i, i = 1, 2, \dots, M$) in LVQ. According to the formed topology, these weight vectors can have two different attributes; location and queuing delay. These weight vectors with these attributes could be represented as $m_{ij}, i = 1, 2, \dots, M, j = 1, 2$. Here, M denotes the maximum aircraft number in a cluster. Additionally, the j represents the attribute number of the system.
- Each output represents a known aircraft cluster as $C_k, k = 1, 2, \dots, K$. Here, K shows the maximum cluster number in an AANET.

After establishing the topology and system model to apply the LVQ for AANETs, we can continue with the proposed framework's module details.

4.3.2 Topology sustainability framework

As explained above, we utilize the LVQ for topology sustainability management based on supervised learning. Here, we first consider each cluster in the AANET topology as a pattern, then we can find the best matching cluster of an aircraft observing A2A link breakage through pattern classification with LVQ. Therefore, in this method, aircraft attributes in clusters are evaluated to find the best matching cluster for an aircraft having a broken A2A link. After determining the new cluster, the links of that aircraft should be assigned as a second phase. Finally, the attributes of aircraft according to the newly assigned links should be updated automatically explained as follows:

4.3.2.1 Winning cluster selection

We first aim to find the best matching cluster of the corresponding input vector, and we call it a winning cluster. Here, we consider the clusters as a pattern and find the best matching cluster of input vector through pattern classification, as shown in Figure 4.3. To find it, we take advantage of the link longevity (T_L) metric between the input and weight vectors in clusters as given with equation 4.1. By utilizing the link longevity,

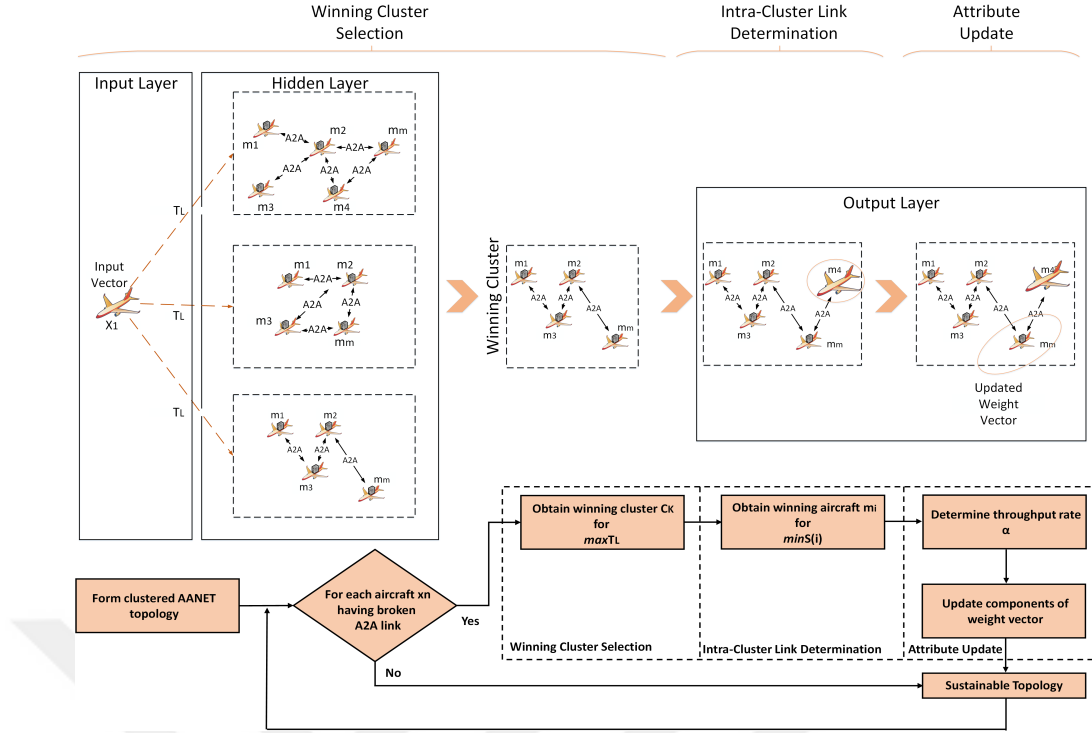


Figure 4.3 : The modules and flowchart of topology sustainability framework.

we can find the most similar cluster to the input vector: the cluster that it can connect to for a longer time. In this formula, $\vec{D}$ and $\vec{V}$ show the relative position and velocity vector between the input vector (x_n) and weight vectors (m_i) according to the LLR, as given in Chapter 2. Also, the transmission range of an aircraft is shown at R . After investigating the link longevities between an input vector and weight vectors in clusters, we should determine the winning cluster of the input vector that maximizes T_L as shown in Algorithm 4.

According to the link longevity metric, we can find the best matching pattern of the input vector with the winning cluster selection, as shown in Figure 4.3. Then, we should determine the links of this input vector in the winning cluster. Therefore, we can evaluate the aircraft that this plane will establish A2A links in the selected cluster.

$$T_L = \sum_{i=1}^M \frac{-\vec{D}_{x_n m_i} \vec{V}_{x_n m_i} + \sqrt{(\vec{D}_{x_n m_i} \vec{V}_{x_n m_i})^2 + V_{x_n m_i}^2 (R^2 - D_{x_n m_i}^2)}}{V_{x_n m_i}^2}, \quad (4.1)$$

$$n = 1, 2, \dots, N$$

$$i = 1, 2, \dots, M$$

4.3.2.2 Intra-cluster link determination

After winning cluster selection, we use the Euclidean distance measure to specify the weight vector in that cluster for A2A link establishment. More specifically, the input vector can establish an A2A link with this specified weight vector. This Euclidean distance ($S(i)$) is calculated by taking the sum of squared differences between an input vector x_{nj} (aircraft observing link breakage) and weight vector attributes ($m_{ij}, i = 1, 2, \dots, M, j = 1, 2$) in that cluster as given with equation 4.2.

$$S(i) = \sqrt{\sum_{j=1}^2 \sum_{i=1}^M (x_{nj} - m_{ij})^2}, \quad n = 1, 2, \dots, N \quad (4.2)$$

In equation 4.2, locations and queuing delays of aircraft are used as attributes of weight vectors. Here, we aim to establish the shortest A2A links between the aircraft by utilizing the locations. The location attribute is obtained through the ADS-B system to calculate the distance between aircraft. Moreover, the risk of packet losses over A2A links could be reduced with the queuing delay consideration as an attribute. We model the aircraft according to G/G/1 system as shown in Figure 4.2 to find the queuing delay attribute. Based on this model, the queuing delay is calculated as given with equation 6.6. Here, the $\sigma_{a_m}^2$ and $\sigma_{s_m}^2$ denote the variance of inter-arrival and service times of aircraft m .

$$T_{q_m} = \frac{\rho_m(1 + \frac{\sigma_{s_m}^2}{(1/\mu_m)^2})(\frac{\sigma_{a_m}^2}{(1/\lambda_m)^2} + \rho_m^2 \frac{\sigma_{s_m}^2}{(1/\mu_m)^2})}{2(1 - \rho_m)(1 + \rho_m^2 \frac{\sigma_{s_m}^2}{(1/\mu_m)^2})\lambda_m}, \quad m = 1, 2, \dots, M \quad (4.3)$$

Therefore, we can calculate the distance between aircraft thanks to the location attribute through the ADS-B system while the delay attribute is estimated with equation 6.6. By using these two attributes in equation 4.2, we can determine the Euclidean distance between the input vector and weight vectors of the winning cluster. Here, the input vector should select a weight vector to establish the shortest and least delayed A2A link. To achieve this, we should find aircraft m_i that minimize the $S(i)$ as shown in Algorithm 4. We can then establish an A2A link between the input vector x_n and weight vector m_i as demonstrated in Figure 4.3.

4.3.2.3 Attribute update

After link determination, we should update the attributes of the weight vector according to newly established links. However, one of the weight vector attributes is the location, and it is updated automatically through movement. Accordingly, we only require to update the delay attribute of weight vector m_i as $m_{ij}(new) = m_{ij}(old) + \alpha(x_{nj} - m_{ij}(old))$, $n = 1, 2, \dots, N, j = 2$. Here, $m_{i2}(old)$ and $m_{i2}(new)$ denote the previous and updated values of the delay attribute of weight vector m_i . Additionally, α shows the learning rate, and in this chapter, we take it as a *throughput rate* that represents the ratio of successful packet delivery over an established A2A link since the ratio of the successfully accepted packets directly affects the number of packets in the queue of the weight vector m_i . This means that the queuing delay is directly affected by the throughput rate. Therefore, we can update the attributes of the weight vector that is newly linked to the input vector as given with Algorithm 4.

4.4 Performance Evaluation

4.4.1 Simulation environment

Algorithm 4 LVQ-Based Topology Sustainability

Data: Input Vectors

Result: Sustainable Topology

Initialization:

N : maximum number of aircraft having broken A2A link

while $i \leq N$ **do**

 Calculate T_L between x_n and m_i

 Find winning cluster C_k for $\max T_L$ (*Winning Cluster Selection*)

while $j = 1 \leq 2$ **do**

while $i = 1 \leq M$ **do**

$S(i) \leftarrow S(i) + x_{nj} - m_{ij}$

end

 Find m_i for $\min S(i)$ (*Intra-Cluster Link Determination*)

$\alpha \leftarrow \text{throughput rate}$

$m_{ij}(new) = m_{ij}(old) + \alpha(x_{nj} - m_{ij}(old))$ (*Attribute Update*)

end

end

To establish the AANET in clusters and simulate the proposed sustainability management system, we utilize the *OMNET++* framework with realistic air traffic

data. For these simulations, we create a $500 * 1000$ NMI air traffic area. In our simulations, we assume that aircraft can only establish an Internet connection with ground base stations to create an AANET, and we embed four stations at each corner of the simulation area. We take each GS's channel capacity as 2 Mbit/sec with a 200 coverage area. This means that the minimum distance is 200 NMI to establish a connection between aircraft and GS (A2G link).

Moreover, the receiver sensitivity and transmit radio frequency power are taken as -24.67 dBm and 125 w, respectively. Additionally, to create an AANET, we take the coverage area of aircraft as 450 NMI at 35000 ft with 10 Mbps A2A link bandwidth. The aircraft produce three packets per second, and the maximum size of these packets could be 512 Kbytes.

After setting the topology establishment parameters, we also require an additional tool to implement the LVQ. This chapter utilizes *Weka* (3-9-4 Azul's 64-bit OpenJDK Java VM 11 version) application with a self-organizing map 1.0.3 addition.

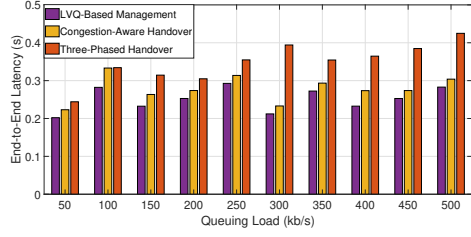
After creating the topology of AANET in clusters, we can create the training data according to this formed topology. This data is given as input to *Weka* in a comma-separated format. In this data set, each aircraft is described by five attributes as *<aircraft id, cluster id, connected aircraft ids, ADS-B locations, and queuing state>*. More specifically, the values of these attributes belonging to aircraft on formed topology constitute the training data. We utilize this training data to determine the clusters of aircrafts observing A2A link breakages. Clearly, we train our model to learn about already formed clusters and aircraft in these clusters.

Therefore, we can assign the aircraft observing A2A link breakages according to obtained models and results instead of forming the topology from the beginning. Here, the time taken to test the model on training data is 0.01 seconds, with 257 instances for each aircraft. Additionally, we take advantage of *Phyton* as a programming language to speed up the simulations of the proposed model [145].

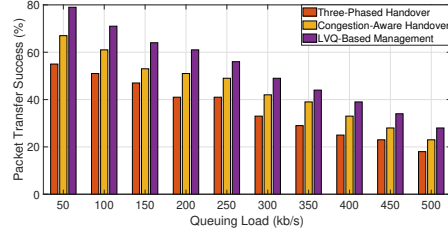
4.4.2 Simulation results

According to end-to-end delay and packet transfer success parameters, we evaluate the performance of the proposed LVQ-based topology sustainability management. To evaluate these parameters, we compare the proposed method with the three-phased and congestion-aware handover management algorithms given in Section 6.2.2.

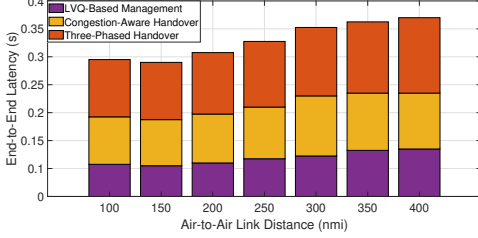
- **End-to-End Latency Analysis:** In this chapter, one of the main objectives is to reduce the end-to-end latency causing by the late transfer of broken A2A links. The end-to-end latency indicates the time elapsed between the breakage and reestablishment of an A2A link. To evaluate the proposed topology sustainability management's end-to-end latency, we utilize the queuing load, A2A link distance, and aircraft number parameters. Here, the queuing load is a crucial parameter for the system because as it rises, the waiting time for incoming requests also increases. Similarly, the A2A distance is an essential parameter for evaluating system performance as A2A links will break more as the distance increases. Additionally, the aircraft number is utilized to explore the success of the methods when the number of aircraft increases. As shown in Figure 4.4(a), we first evaluate the end-to-end latency of the proposed model according to increasing queuing load levels. Here, the simulations are performed over 100 min periods for each queuing load level, and we take an average of results over 10 min periods due to the ultra-dynamic characteristics of AANETs. Also, we utilize 100 aircraft for each load level and take the average of these aircrafts' end-to-end latencies. Accordingly, we can achieve a 24% lower end-to-end latency in our proposed method than other methodologies, as shown in Figure 4.4(a). We can obtain this delay efficiency by reducing late transfer decisions with equation 4.1. More specifically, the consideration of link longevity in pattern classification to execute the A2A link transfers reduces the late transfer decisions. The parameter monitoring of three-phased approach, and the congestion measurement of congestion-aware approach could be listed as the main reasons for latency growth in other methods. Then, as shown in Figure 4.4(c), we investigate the end-to-end latency of the proposed model according to the increasing A2A distance levels. Here, we again perform the simulations over 100 min periods for each different distance and take the average of these results over 10 min periods by utilizing the 100 aircraft.



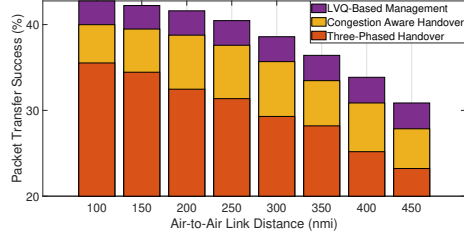
(a) End-to-end latency vs. Queuing Load.



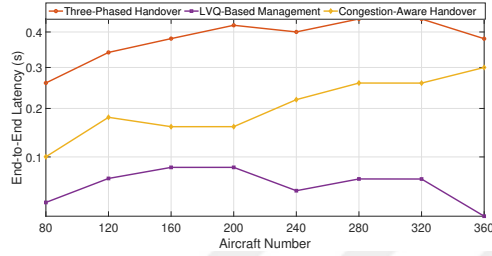
(b) Packet Transfer Success vs. Queuing Load.



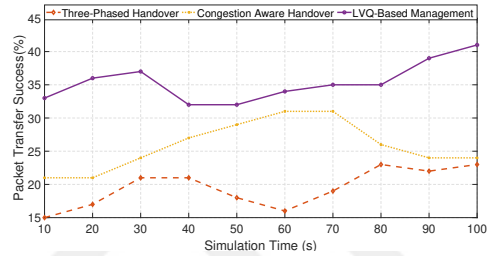
(c) End-to-end latency vs. A2A Link Distances.



(d) Packet Transfer Success vs. A2A Link Distances



(e) End-to-end latency vs. Aircraft Number.

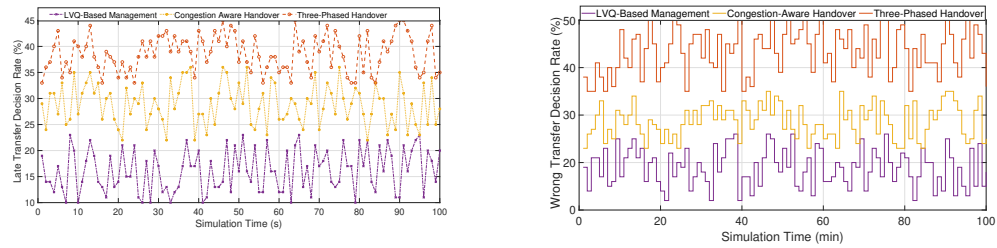


(f) Packet Transfer Success vs. Simulation Time.

Figure 4.4 : Simulation results.

Correspondingly, we monitor 28% reduced end-to-end latency in our proposed method in terms of A2A link distances, as shown in Figure 4.4(c). We can provide this reduction by including a location attribute to the weight vector during Euclidean distance calculations as given with equation 4.2. Finally, we analyze the proposed model's end-to-end latency according to increased aircraft number over 100 min. Here, we observe 25% reduced latency in our proposed method compared to the other two methodologies, as shown in Figure 4.4(e). The fact that LVQ is not affected by the number of aircraft is an essential factor in this success, as given with Algorithm 4. On the other hand, the handover preparation and geographic proximity calculations could also be listed as the main reasons for the latency growth in other methodologies (three-phased and congestion-aware approaches). Moreover, the absence of AI in other methods allows us to get better results.

- **Packet Transfer Success Analysis:** The wrong transfer decisions of A2A links could reduce the packet transfer success of AANET. For this reason, our second objective is to increase the packet transfer success. Here, the packet transfer success shows the successfully accepted packets by the destination aircraft overall transfers. Like the end-to-end latency, we utilize the queuing load, A2A link distances, and simulation time to evaluate the proposed topology sustainability management's packet transfer success. Unlike the end-to-end latency, we also assess the methods' success over a period thanks to the simulation time parameter. As shown with Figure 4.4(b), we first evaluate the proposed model's packet transfer success according to the increasing queuing load levels. Here, we also utilize the same simulation conditions explained in end-to-end latency analysis. Accordingly, we experience a packet transfer success at the level of 80% compared to the selected handover heuristics according to the queuing load parameter, as shown in Figure 4.4(b). In our method, we can achieve this packet transfer success by utilizing ρ_{m_i} , $\sigma_{a_{m_i}}^2$, and $\sigma_{s_{m_i}}^2$ parameters during link determination. These parameters enable us to consider the queue status of aircraft, which increases the packet transfer success by preventing queuing drops. Here, the congestion-aware handover heuristic can achieve better results than the three-phased strategy thanks to the measure of IGW congestion. However, these two methodologies' high latencies play an important role in reducing packet transfer success. Then, as shown in Figure 4.4(d), we investigate the packet transfer success of these methodologies according to the increasing A2A distance levels. Here, we again utilize the same simulation



(a) Late transfer decisions vs. Simulation time. (b) Wrong transfer decisions vs. Simulation time.

Figure 4.5 : Late and wrong transfer decision results.

conditions, but differently, 200 aircraft are deployed. Correspondingly, we monitor packet transfer success at the level of 47% in our proposed method, as shown in Figure 4.4(d). Here, the role of location attribute during the selection of winning

aircraft constitutes an important part of our success as given with Algorithm 4. Finally, we evaluate the packet transfer success of three methods according to the simulation time. During this analysis, we again deploy 200 aircraft and take the average of their packet transfer success for each time interval. Therefore, we can achieve 31% better packet transfer success than the other methodologies, as shown in Figure 4.4(f). The failure to take advantage of AI and preparation stages of other methodologies could be listed as the main causes of the reduced packet transfer success.

- We also evaluate the late and wrong transfer decisions of A2A links. Here, we assess late and wrong transfer decisions according to the simulation time. Similar to Figure 4.4(f), we use 200 aircraft by taking the average of their wrong and late transfer decisions for each time interval under the same simulation conditions. Accordingly, we can obtain a 22% reduction on the late transfer decisions in LVQ-based management compared to other methodologies as shown in Figure 4.5(a). The three-phased operation as information collection, handover decision, and handover execution could be listed as a primary reason for the late transfer decisions in three-phased approach. Also, the congestion determination of Internet gateways is the primary reason for the late transfer decision in congestion-aware approach. Moreover, we observe 24% fewer wrong transfer decisions in LVQ-based management according to other methods as shown in Figure 4.5(b). The sudden changes in the signal strength become the main reason for large numbers of wrong transfer decisions in three-phased approach. On the other hand, this is due to the geographic proximity-based calculations in congestion-aware approach.

4.5 Conclusion

In this chapter, we proposed an LVQ-based topology sustainability management for AANET. We first find the best matching cluster of input vectors through pattern classification based on the link longevity parameter. Then, we determine the intra-cluster links between input and weight vectors by utilizing location and queuing delay attributes. Finally, by using throughput rate as a learning parameter, we update the weight vector attributes. The simulation results show that the proposed

sustainability management mechanism reduces the end-to-end latency 25% by increasing packet transfer rate 31% according to the compared handover management heuristic.

General concept for the next chapter In this chapter, we propose a LVQ-based management to enable the sustainability of the clustered AANET topology. Then, airplanes can route the packet on this formed and sustained topology. However, before the routing, we require an additional step to conclude the topology related procedures. More clearly, after enabling the sustainability of AANET topology, we should continuously map the current cluster topology to airplanes at each timestamp. In this way, we can notify the airplanes about the sustained AANET topology.

Based on these, in the next chapter, we will propose a management model to map the current topology to AANETs after each sustainability procedure.



5. TOPOLOGY MAPPING MANAGEMENT FOR AANETS

5.1 Purpose

AANET is a fairly new concept that connects airplanes via wireless A2A links, allowing passengers to access the Internet during a flight. The unstable A2A link characteristics and ultra-dynamic topology become the main differences between AANETs and usual ad-hoc architectures. To handle these differences, the AANET topology could be created in the form of clusters by collecting airplanes having similar features under the same set. However, it is also difficult to sustain the cluster topologies since ultra-dynamic characteristics still affect them. Therefore, the current cluster topology must be continuously mapped to the airplanes to notify them as a part of sustainability. If we do not ensure the sustainability of the clusters, the packet transfer success of AANET is decreased with higher end-to-end latency. At that point, to solve this aircraft notification problem and map the current cluster topology to them at each timestamp, in this chapter, we propose a Gated Recurrent Unit (GRU)-based topology mapping mechanism for AANETs. Here, the GRU can continuously notify the airplanes at each timestamp about topology changes. Therefore, airplanes can forget the old topology when it changes. Otherwise, the topology taken from the previous timestamp is continuously remembered if it does not change. Finally, the performance evaluations reveal that the GRU-aided topology mapping can achieve roughly 42% higher packet delivery ratio with 34% reduced end-to-end latency.

5.2 Introduction

AANET is a relatively new technology that enables passengers to access Internet services during the flight. In this novel concept, airplanes are connected in an ad-hoc manner via wireless A2A links. Each aircraft on this AANET architecture becomes responsible for routing packets to the destination. Here, the destination is also an airplane with an Internet connection via satellite and ground station. Therefore,

AANET requires only one aircraft linked to the satellite and ground station instead of per-aircraft connectivity. In this way, the packets are transferred via airplanes without any central node or entity to provide Internet connectivity as detailed in the previous sections.

5.2.1 Topology mapping challenges in AANET

Although AANET is an innovative concept, it experiences various challenges due to its specific characteristics as explained in Chapter 1. Unlike other ad-hoc networks, the AANETs consist of airplanes, which becomes the root cause of these specific characteristics. This chapter investigates these characteristics from two main aspects: unstable A2A link characteristics and ultra-dynamic AANET topology. The high speed of airplanes causes frequent breakages of A2A links, and this situation changes the AANET topology continuously compared to the usual deployments, as summarized in Figure 5.1. One of the common approaches is to create the AANET topology in the form of clusters to handle this ultra-dynamic environment with the reduction of A2A link breakages as explained in Chapter 3.

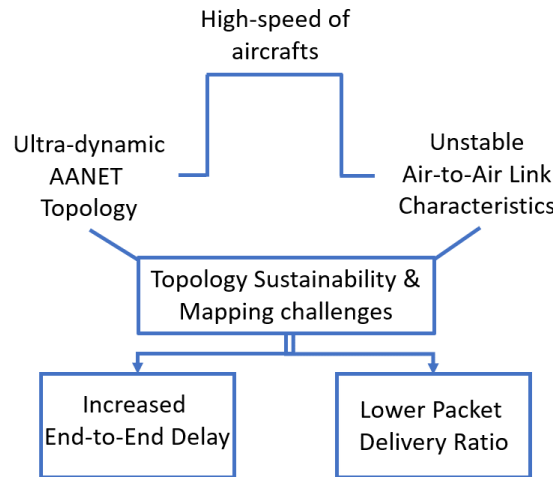


Figure 5.1 : AANET specific characteristics and challenges.

Even if AANET's specific characteristics are handled by acquiring aircraft with similar features under the same cluster, the topology sustainability problems may persist as explained in previous chapters. It isn't easy to sustain the cluster topology since the ultra-dynamic characteristic still affects them. As a solution to this problem, we

proposed a LVQ-based topology sustainability for clustered-AANETs in Chapter 4. This method ensures the sustainability of the clustered topology dynamically; however, the airplanes must continuously notified about the current topology through mapping. Otherwise, the packet transfer success of AANET is decreased with higher end-to-end latency, as shown in Figure 5.1. To solve this notification problem and map the current cluster topology to airplanes at each timestamp, in this chapter, we aim to propose a Gated Recurrent Unit (GRU)-based topology mapping model for AANETs.

5.2.2 Related work

There are various works in the literature proposing solutions to different AANET topology management problems as investigated in Chapter 1. The in-air monitoring system is proposed to deal with the network partitions of AANET in [146]. Here, multi-dimensional clustering is used to identify the network partitions by considering movement direction and congestion parameters. To analyse the AANET capacity, an aeronautical broadband wireless access methodology is proposed in [147]. Here, their main aim is to find the number of concurrent successful transmissions in AANETs and for this aim, they mainly utilize the throughput parameter.

Although there are various topology-related concepts in literature, topology mapping is a novel concept as a part of sustainability management. Also, these works do not utilize AI based concepts to handle the ultra-dynamic characteristics of AANETs as explained in Chapter 2. To the best of our knowledge, this is the first work proposing a topology mapping model for AANETs. The main aim of this work is to map the current cluster topology to airplanes at each timestamp to notify them continuously. Here, we take advantage of the GRUs, a kind of AI-based methodology to handle the ultra-dynamic and unstructured characteristics of AANETs.

5.2.3 Contributions of the chapter

GRUs are a special kind of recurrent neural networks (RNNs) with long-term learning capabilities. Therefore, they can remember information over long periods. This capability strongly matches the specific characteristics of AANETs. Also, the AANETs may include airplanes performing long-haul flights. These long flights

increase the amount of information aircraft need to remember. At that point, GRU allows information to be continuously transferred to the next timestamp.

In this chapter, this information becomes the topologies of AANET clusters. More clearly, we utilize the GRU to notice airplanes about topology changes at each timestamp. Accordingly, GRU allows airplanes to forget the old ones when the topology changes and remember if it does not change. For this aim, in this chapter, we propose a GRU-aided intra-cluster topology mapping for AANETs. The main contributions of the chapter could be summarized as follows:

- The hidden state representation in GRU allows an airplane to carry its own topology in terms of its links.
- Thanks to the forward propagation mechanism in GRU, the airplanes in a cluster can remember the old topology they carried unless it changes.
- The gate mechanism in GRU allows quick mapping of new cluster topology to airplanes.

5.2.4 Organization of the chapter

The rest of the chapter is organized as follows: we first give our proposed GRU-Aided Intra-Cluster Topology Mapping model. Then, we evaluate the performance of the proposed GRU-Aided Intra-Cluster Topology Mapping model. Finally, we conclude the chapter.

5.3 GRU-Aided Intra-Cluster Topology Mapping Management for AANETs

5.3.1 AANET topology

Similar to the previous sections, we consider two-layered AANET topology with aircraft and ground layers. In aircraft layer, airplanes are connected to each other in a clustered manner to form an AANET as shown in Figure 6.3(b). Also, there are GSs in ground layer and it is sufficient to only have one airplane with GS connectivity from each cluster.

As shown in Figure 6.3(b), in this topology, we aim to implement a GRU-aided topology mapping framework with two main parts as forget and update gates. The details of these framework and its main parts will be explained in the following sequel.

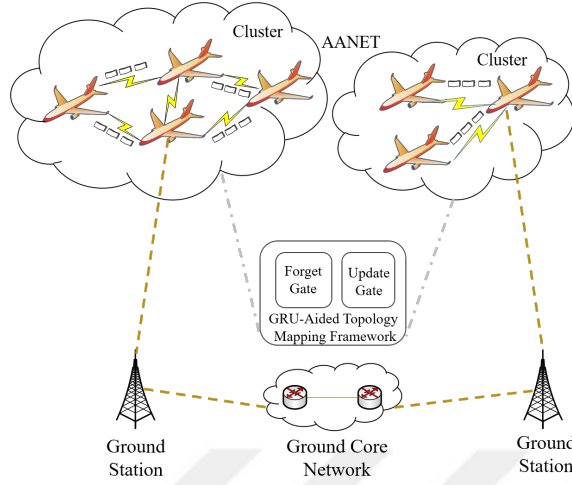


Figure 5.2 : The AANET topology.

5.3.2 Topology mapping framework

As explained in the above section, in this chapter, we propose a GRU-aided intra-cluster topology mapping model for AANETs. To apply GRU to the AANET, we consider each airplane as a cell. These cells are updated at each timestamp according to the current situation of topology. As shown in Figure 5.3, each cell accepts two main inputs: input and previous hidden state vectors. The details of cell inputs could be detailed as follows:

- **Input Vector (I_t):** In this chapter, we propose a topology mapping solution for clustered-AANETs. For this reason, before the mapping, the AANET topology should be created in the form of clusters as given in Chapter 3. In this architecture, each cluster consists of a different number of airplanes to create the AANET. Accordingly, each aircraft only knows the topology of the cluster it belongs. Based on these, we transfer the list of airplanes that this plane should connect to in the cluster as a vector. Therefore, the input vector corresponds to the list of aircraft to which the corresponding plane will connect in the cluster. This vector is represented as $I_t = [i_1, i_2, \dots, i_n]$ and here, n corresponds to the aircraft number of the cluster. Accordingly, each aircraft in a cluster is shown in this vector with 1 or

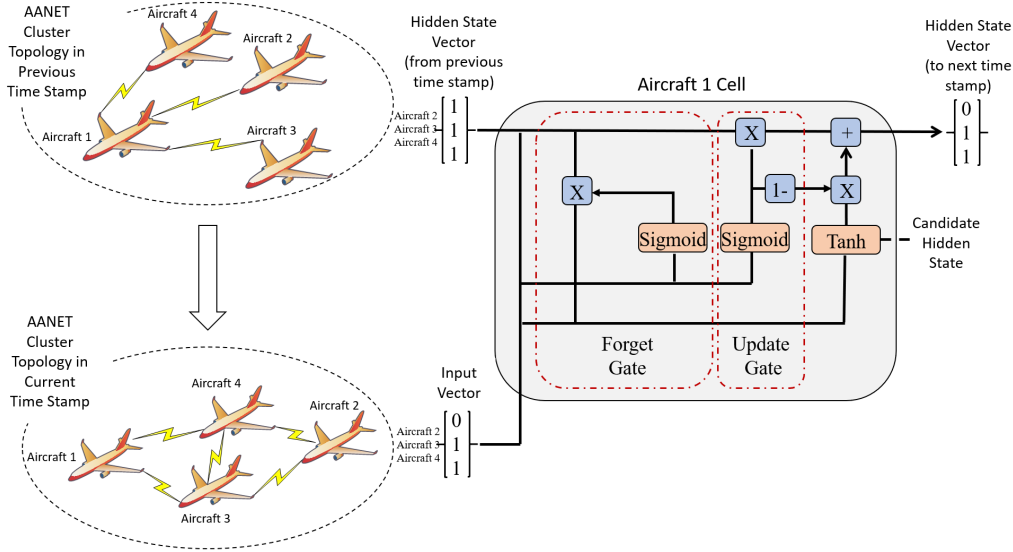


Figure 5.3 : GRU-aided intra-cluster topology mapping model.

0 values. Here, 1 means connection should be established with the airplane in the corresponding line of the input vector, while 0 represents no connection.

- **Previous Hidden State Vector ($H_{(t-1)}$):** The hidden state is taken from the previous timestamp to the current one. This vector gives the topology of the corresponding airplane in the previous timestamp in terms of its links. More specifically, the prior topology of an aircraft is transferred to the current timestamp in the form of a hidden state vector. Similar to the input vector, it is represented as $H_{(t-1)} = [h_1, h_2, \dots, h_n]$ and n again shows the aircraft number of the related cluster. Also, each vector element has a value of 1 or 0, as explained above.

In addition to these inputs, the GRU-aided intra-cluster topology mapping module consists of two main components: Forget Gate and Update Gate, as shown in Figure 5.3. We can detail these gates as follows:

5.3.2.1 Forget gate

The main aim of the forget gate is to determine how much of the AANET cluster topology is forgotten and is not passed to the next timestamp. To achieve this, the forget gate utilizes the input (I_t) and previous hidden state vectors ($H_{(t-1)}$) as shown in Figure 5.3. During these procedures, the input vector (I_t) is accepted by the cell, and

it is multiplied by its own weight (W_{IF}). Similarly, the previous hidden state vector ($H_{(t-1)}$) and its weight (W_{HF}) are multiplied as shown in Figure 5.3. Then, the sigmoid function (σ) is applied to the sum of these results as given in equation 5.1. Here, the sigmoid function keeps the values between 1 and 0. Accordingly, the older topology information is filtered, and only the updated current one will be transferred to the next timestamp as summarized in Figure 5.4. Therefore, all of these procedures correspond to the forget gate definition (F_t) of GRU as given in equation 5.1.

$$F_t = \sigma(I_t * W_{IF} + H_{(t-1)} * W_{HF}) \quad (5.1)$$

In equation 5.1, W_{IF} and W_{HF} represent the weight values of input and previous hidden state vectors for the forget gate. In this chapter, we define these weights based on the connectivity degree of a corresponding aircraft. More specifically, W_{IF} corresponds to the connectivity degree of the input vector (I_t) while W_{HF} is based on the previous hidden state vector.

5.3.2.2 Update gate

The update gate aims to output the information from the current timestamp to the next. Similar to the forget gate, the update gate also uses the input (I_t) and previous hidden

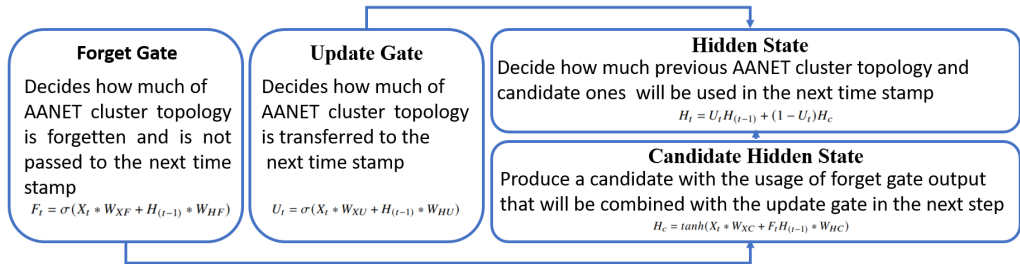


Figure 5.4 : Flow of the GRU-aided intra-cluster topology mapping model.

state vectors ($H_{(t-1)}$) as shown in Figure 5.3. The definition of the update gate (U_t) is obtained in the same way as forget gate as given with equation 5.2.

$$U_t = \sigma(I_t * W_{IU} + H_{(t-1)} * W_{HU}) \quad (5.2)$$

Similar to the forget gate, W_{IU} and W_{HU} correspond to the weight values of the input and previous hidden state vectors for the update gate. Here, we utilize the same weight values as explained in the forget gate. Again, the sigmoid function aims to keep the update gate values between 1 and 0.

After reaching the results of forget and update gates, we should find a candidate hidden state. To reach it, we also use the result of the forget gate in addition to the current input and previous hidden state vectors. Here, the forget gate is added to see the impact rate of the previous state on the current ones. Accordingly, the formulation of the candidate hidden state is given with equation 5.3. In this equation, we use the tanh function instead of the sigmoid based on the main structure of the GRU model. Unlike the sigmoid function, the tanh function keeps the values between -1 and 1. Also, we again take the weight values the same as in the forget and update gates.

$$H_c = \tanh(I_t * W_{IC} + F_t H_{(t-1)} * W_{HC}) \quad (5.3)$$

However, equation 5.3 gives the candidate state, and it requires additional operations to reach the final exact value (current hidden state). To achieve this, we also use the result of the update gate (U_t) with the candidate hidden state as given with equation 5.4. Accordingly, we can decide how much previous AANET cluster topology and candidate ones will be used in the next timestamp. According to equation 5.4, if U_t approaches 1, the old topology is remembered. Otherwise, the candidate hidden state (H_c) has an important role in determining the next state, as summarized in Figure 5.4.

$$H_t = U_t H_{(t-1)} + (1 - U_t) H_c \quad (5.4)$$

As a result of this methodology, aircraft could quickly be informed about the new cluster topology when it changes. Additionally, unchanging topology could be easily transferred to the upcoming timestamps.

5.4 Performance Evaluation

5.4.1 Simulation environment

The performance of the proposed GRU-aided intra-cluster topology mapping model is evaluated by using the Python 3.9.7 language. The proposed model is implemented and tested on a station with Windows 10 Enterprise, 16GB RAM, and i7-1185G7 (4 cores). We utilize the Spyder and Visual Studio Code as a python IDE and code editor. Here, we use TensorFlow and PyTorch libraries to include machine and deep learning functionalities in the proposed model. The Numpy library is used to ease complex mathematical operations. Also, the panda and sklearn libraries are utilized to execute all of the clustering-related functions in AANET.

After establishing the simulation environment, we can evaluate the performance of the proposed model. At that point, we utilize the packet delivery ratio and end-to-end latency parameters for performance evaluation. The main selection reason for these is that they are the main parameters affected by topology changes of AANETs, as explained in Section 7.1. The details of these parameters could be summarized as follows:

- **Packet Delivery Ratio:** This parameter corresponds to the ratio of successfully accepted packets by destination aircraft to all transferred packets.
- **End-to-End Delay:** It is the total amount of delay the packet encounters until it leaves the source aircraft and reaches the destination. In this chapter, we calculate this delay in terms of the waiting time of packets in the queues of airplanes. To calculate this delay, we model airplanes according to G/G/1 queuing system due to their ad-hoc and unstructured characteristics. Accordingly, the waiting time (T_{q_j}) of the packet in only one airplane (j) queue is calculated as $T_{q_j} = \frac{\rho_j(1+C_{S_j}^2)(C_{A_j}^2+\rho_j^2C_{S_j}^2)}{2(1-\rho_j)(1+\rho_j^2C_{S_j}^2)\lambda_j}$, $\rho_j = \frac{\lambda_j}{\mu_j}$, $C_{S_j}^2 = \frac{\sigma_{S_j}^2}{(1/\mu_j)^2}$. Here, C_{A_j} and C_{S_j} are the squared coefficient of variation arrival and service rates.

Also, to evaluate the performance of the proposed mapping model, we compare it with the usual AANET topology creation mechanism that does not include sustainability and mapping procedures. We will give the details of these evaluations in the following subsection.

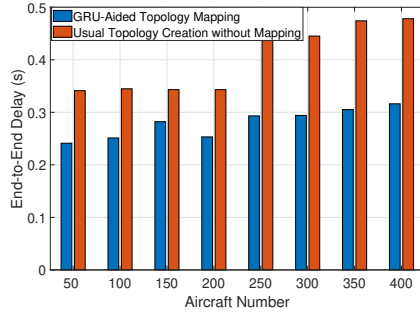
5.4.2 Simulation results

As explained in the above sections, we observe a higher end-to-end delay and lower packet delivery ratio if topology sustainability is not provided and airplanes are not notified about topology. For these reasons, the end-to-end delay and packet delivery ratio constitute our main simulation parameters. We also evaluate the topology mapping success of the proposed model for different numbers of aircraft.

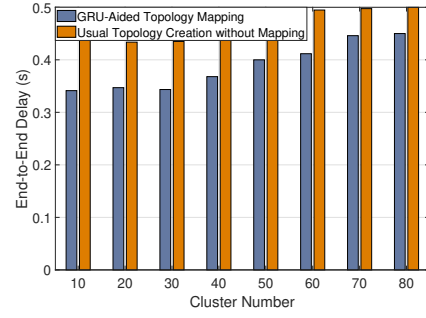
We first evaluate the end-to-end delay of the proposed GRU-aided topology mapping model according to the aircraft and cluster numbers parameters as shown in Figure 7.7(b) and Figure 7.7(a). Here, we conduct our simulations over 10 min periods for each aircraft and cluster number. Then, we take an average of these simulation results to find a delay value for each aircraft and cluster number value. As shown in Figure 7.7(b), we start with 50 airplanes and increase them by 50 at each stage until reaching 400 planes. Here, we can achieve a delay of less than 0.3s for each aircraft number. Additionally, we can experience roughly 40% more minor end-to-end delay with our proposed model than the usual topology creation without mapping. Similarly, as shown in Figure 7.7(a), we start our simulations with 10 clusters, and they are increased by ten at each stage until 80 clusters. Here, we can observe end-to-end delays around 0.4s levels.

Also, the usual topology creation without mapping follows roughly 30% higher end-to-end delay than our proposed GRU-aided topology mapping model. The main reason for these results is that all airplanes in each timestamp have information about the current topology. Thus, each aircraft automatically knows the planes they need to connect at each timestamp. In this way, the connection establishment and termination procedures are executed quickly. Accordingly, this reduces the waiting times of packages in airplane queues, and we can observe a lower end-to-end delay.

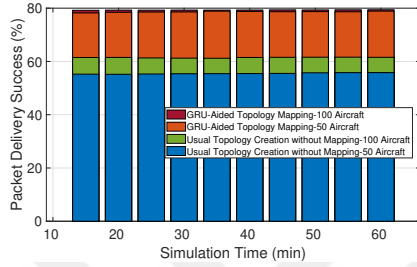
We also evaluate the packet delivery success of the proposed model by comparing it with the usual topology creation model without mapping, as shown in Figure 7.7(c). Here, we again conducted our simulations over 10 min periods for 50 and 100 airplanes. Then, we can take the average of obtained results to find a packet delivery success value for each stamp of simulation time. Correspondingly, we can



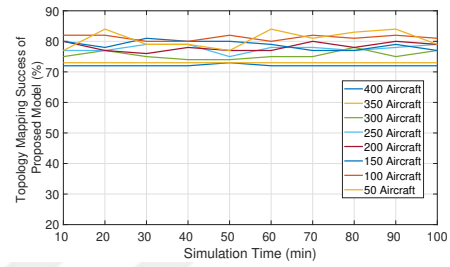
(a) End-to-end delay vs. aircraft number.



(b) End-to-end delay vs. cluster number.



(c) Packet delivery ratio vs. aircraft number.



(d) Topology mapping success of the proposed model.

Figure 5.5 : Performance evaluation results.

achieve a packet delivery success of roughly 80% levels, and this rate can drop to 50% levels for the usual topology creation model without mapping, as shown in Figure 7.7(c). Additionally, we can see that an increase in the number of aircraft could enhance the packet transmission efficiency in an AANET. In our proposed model, the aircraft notification challenge has been solved through mapping as a part of topology sustainability management. This situation increases the packet transmission success by enhancing the adaptation of aircraft to changing topologies.

Finally, we calculate the mapping success of our proposed model in the ultra-dynamic AANET environment. Here, we take an average of mapping results for each stamp of simulation time. Accordingly, we can successfully map the current topology to airplanes at roughly 80% level since a deep learning-based model helps us to handle the ultra-dynamic AANET environment, as shown in Figure 5.5(d). More clearly, the unstructured and ultra-dynamic characteristics of AANET do not affect the mapping performance of the GRU-aided model.

5.5 Conclusion

It is very challenging for AANET topology sustainability that each aircraft in an AANET cluster should be constantly aware of the instant topology. For this reason, in this chapter, we propose a GRU-aided topology mapping model for AANETs as a solution to the aircraft notification problem. Accordingly, we aim to map the current cluster topology to airplanes to notify them as a part of topology sustainability. According to this proposed model, the GRU can map the current topology to airplanes as input vectors or help airplanes remember the old unchanged topology as hidden vectors. Accordingly, airplanes can remember the old topology and forget it as soon as it is updated. Our simulation results support this model by achieving a roughly 42% higher packet delivery ratio with 34% lower end-to-end latency than the usual topology creation without mapping.

General concept for the next chapter

We formed the AANET topology, enabled the sustainability of it, and finally, continuously mapped the formed and sustained AANET topology to airplanes up to this point. Accordingly, we can start to route the packets on AANET without encountering any topology related problems. On the other hand, AANET does not utilize the traditional routing protocols designed for terrestrial networks due to its specific characteristics.

Based on these, in the upcoming chapter, we aim to propose a routing management model for ultra-dynamic and unstable AANETs.

6. ROUTING MANAGEMENT FOR AANETs

6.1 Purpose

AANET is one of the critical methodologies to satisfy the Internet connectivity requirement of airplanes during their flights. However, the ultra-dynamic topology and unstable A2A link characteristics increase the need for AANETs to a particular routing algorithm compared to the terrestrial networks. This need is mainly since these AANET specific characteristics increase the delays, packet losses, network load and reduce accuracy by continuously changing topology and breaking A2A links during routing. This article proposes an updated Layered Hidden Markov (updated-LHMM) estimation-based Q-learning (QLR) routing for AANETs to solve the delay, packet loss, network load, and accuracy problems. Our main aim is to let the airplanes find their routing path through exploration and exploitation by mapping the AANET environment to QLR. This enables the routing algorithm can adapt to the dynamic conditions of AANETs. For this aim, we adapt the Bellman equation to the AANET environment by proposing different methodologies for its related QLR components. Here, we find the maximum-state action determination component by using an updated-LHMM that divide the routing path of an aircraft into layers and find the best aircraft for each layer. Also, we model the AANET environment as G/G/1 queues to determine the aircraft states while the decomposition-based two-parameter characterization model defines of A2A link states in the updated-LHMM and QLR. Finally, we define a quadratic reward function with A2A link quality and distance metrics for the dynamic reward determination component of QLR.

6.2 Introduction

According to the IATA expectations, there will be a massive increase in passenger numbers, and it will reach 7.8 million in 2036, as shown in Figure 6.1 [148]. The enormous increase in passenger numbers with developing technology transforms the

IFC into an essential requirement instead of a luxury. The passengers want to access the Internet in a high quality similar to that they receive on a terrestrial connection during flight without observing interruption. These changes have increased the number of people who need and use IFC. Correspondingly, the importance of IFC for passengers increases airlines' motivations to satisfy the passenger demand, attract and hold passengers by generating revenue. As explained in Chapter 1, it is expected that the total revenue obtained from IFC will increase from \$700 million in 2015 to nearly \$5.4 billion by 2025 with a 23% CAGR over the ten years, as shown in Figure 6.1. These reasons have made IFC an essential field of research, and different technologies have been proposed to enable its infrastructure. Although the satellite and A2G connectivities constitute the leading technologies for IFC, the high cost, latency, and reduced coverage generally deteriorate these key enablers' efficiency.

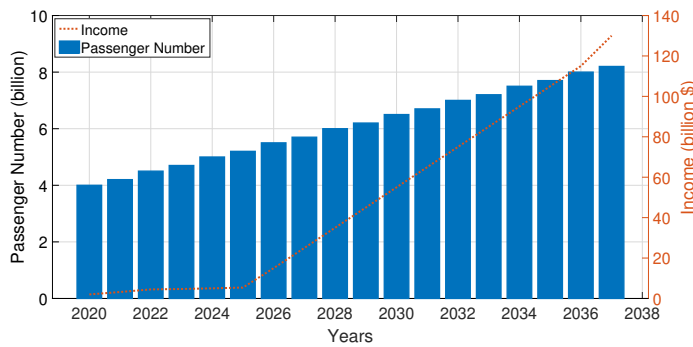


Figure 6.1 : Passenger and income changes by year.

One of the novel solutions is the AANETs to satisfy the IFC's huge demand by creating A2A links and transmission of packets over these connections to enable IFC for airplanes [149]. The advantage of routing protocols and management models are taken to transfer the packets over AANET. However, traditional routing protocols designed for stationary and terrestrial networks do not satisfy the requirements for AANETs.

6.2.1 Routing management challenges in AANETs

In AANET, the aircraft's packets are routed through A2A links until reaching the destination aircraft that have Internet connectivity, and each aircraft could be considered as a router during this procedure, as shown in Figure 6.2. However, the requirements of AANETs are not satisfied by the traditional routing protocols designed

for terrestrial networks [150], [151]. The specific characteristics of AANETs constitute the main reasons for its special routing management requirement [152]. This chapter investigates these characteristics from two different aspects as ultra-dynamic AANET topology and unstable A2A link characteristics. These two characteristics also lead to further problems during AANET routing management, as explained follows:

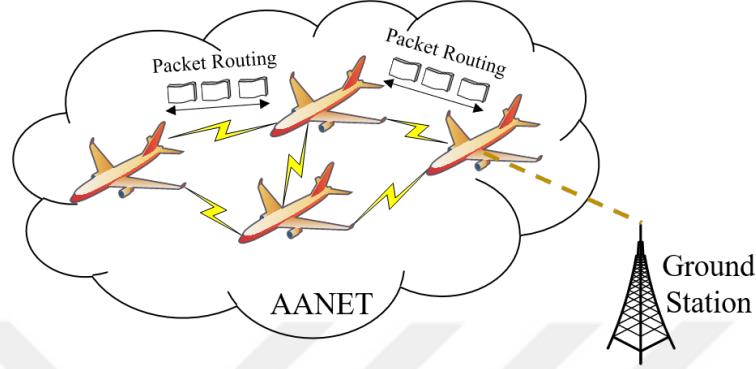


Figure 6.2 : AANET routing mechanism.

- **Ultra-dynamic AANET topology:** The high speed of aircraft makes the AANET environment ultra-dynamic compared to the terrestrial vehicular networks, as shown in Figure 6.3(a). Due to this dynamic environment, the AANET topology changes continuously. Accordingly, the dynamic characteristic of AANET topology makes it challenging to determine the route and increases the routing delay and network load as given in Figure 6.3(a). More specifically, a new route must be found for each aircraft during continuous topology changes. This situation increases the cost of routing in terms of delays, as shown in Figure 6.3(b).
- **Unstable A2A Link Characteristics:** The main reason for the unstable A2A link characteristics is based on the high speed of airplanes in AANET topology, as shown in Figure 6.3(a). Here, the high-velocities cause frequent breakages of A2A links by continuously changing the connected aircraft and links as given in Figure 6.3(a). These frequent breakages reduce the accuracy of routing operation, and this situation also increases the packet losses during routing, as shown in Figure 6.3(b).

Based on these two characteristics and their possible problems, the routing management model determination constitutes the critical design criteria to increase

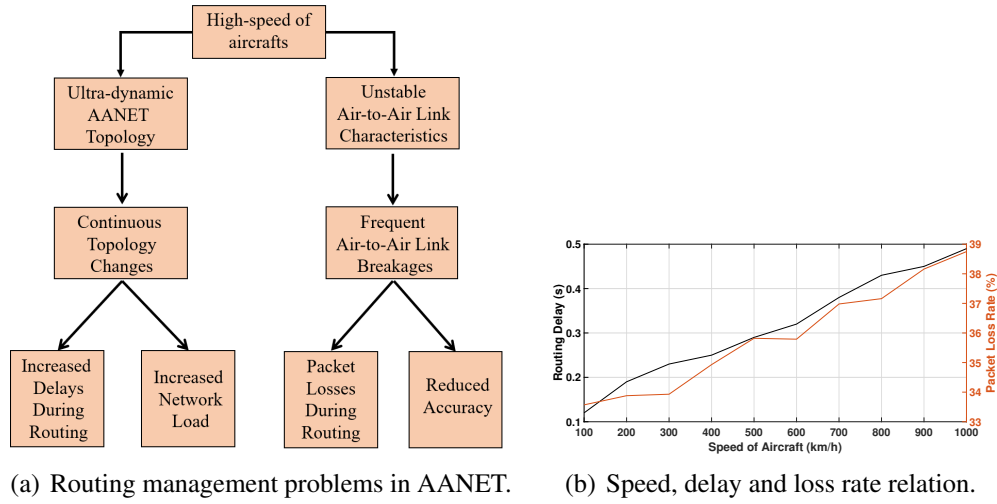


Figure 6.3 : Routing management challenges in AANET.

the packet delivery success and accuracy in an AANET by reducing routing latency with lower network load.

Table 6.1 : Existing techniques for AANET routing management

Algorithm	Parameters	AI Addition	Performance Criteria	Methodology
GLSR	Queue Size Geographic Distance	None	Network Throughput	Greedy Forwarding
			Packet Delivery Ratio	
			End-to-End Delay	
MUDOR	Doppler Value	None	Link Down	Route Request
			Number of Handoff	Route Reply
PLAR	Node Density Link Availability	None	Path Availability Rate	Multipoint Relaying
AeroRP	Time to Intercept Speed	None	Average Accuracy	Speed-based Heuristic Geographic Routing
			Average Delay	
			Average Overhead	
HSRP	Air Vehicle Density Flight Flow Rate Aircraft Velocity	None	Delivery Ratio	HELLO Beacon
			End-to-End Delay	
			Routing Overhead	
ARPAM	Distance Number of Hops	None	Routing Overhead	Route Request
				Route Reply
DMDR	Relative Velocity Expected Queuing Delay	None	Throughput	Doppler Shift
			Routing Overhead	
MQSPR	Path Availability Period Residual Path Load Capacity Path Latency	None	End-to-End Delay	IGW Advertisements Forward Best Advertisement
			Packet Delivery Ratio	
			Handoff per Hour	
Node-TBR	Delay Reachability	None	Average Delay	Geopath Computation Algorithm Forwarding Algorithm
			Normalized Reachability	
			Aircraft Density	
RGR	Location End-to-End Paths	None	Delivery Ratio	Greedy Geographic Forwarding Reactive Routing
			Average Delay	
			Average Routing Packets	

6.2.2 Related work

There are numerous studies proposing routing management algorithms for AANETs in the literature. We extensively investigated these algorithms in Chapter 1. In this chapter, we summarize some of them in Table 6.1 by investigating their different

characteristics. We will also select four of these algorithms to use in our simulations and evaluate the performance of our proposed algorithms.

6.2.3 Contributions of the chapter

Although there are numerous AANET routing algorithms in the literature, as explained in Section 6.2.2, the ultra-dynamic environment and unstable A2A link characteristics are not considered simultaneously by any work. More specifically, the packet routes should be calculated dynamically according to the current status of network topology due to these two AANET-specific characteristics. However, this dynamism has not been handled dramatically to reduce packet loss, delay, and network load with increased accuracy. Moreover, if we utilize AI-based solutions, we can adapt to the dynamic conditions of AANETs by finding routes according to the instant status of AANETs. However, the AI-driven methods for AANET routing management are not considered in these works, as shown in Table 6.1. At that point, reinforcement learning enables airplanes to make their own routing decisions sequentially through exploration and exploitation different from the usual AANET routing mechanism as given with Figure 6.2. More specifically, we can consider each aircraft as an agent having a state. This agent takes action (route packet), gets a reward, and switches to a new state during the routing procedure as detailed in upcoming section. Furthermore, the reinforcement learning-based routing algorithms in literature (i.e. [153], [154], [155], [156]) do not satisfy the two main requirements of AANETs as explained in Section 6.2.1.

This chapter proposes a routing management model for AANETs by utilizing Q-learning, one of the main reinforcement learning approaches. The main aim of the reinforcement learning utilization is to let the airplanes find their own routing paths in the dynamic AANET environment through exploration and exploitation without any central node or entity. Also, another selection reason for reinforcement learning is that the decision-making is sequential in AANET routing, and this situation strongly matches the principal characteristic of reinforcement learning. For these reasons, we adapt the Bellman equation by proposing different methodologies for each of its elements according to AANETs. The main contributions could be listed as follows:

- We utilize the Bellman equation by proposing different methodologies, each of its related QLR components according to the AANET environment.
- We updated-LHMM for the maximum state-action determination component in QLR. Here, we aim to divide the routing path of an aircraft into layers and find the best airplanes for layers that give the maximum possible states.
- We use G/G/1 system-based queuing load to determine the aircraft states in the updated LHMM. We also utilize the decomposition-based two-parameter characterization to determine A2A link states that correspond to the state-transition probabilities in the updated-LHMM. Accordingly, we can divide the packet transfer procedure between two aircraft into three phases: superposition, queuing, and splitting by considering accepted, queued, and transferred packet streams.
- We define a quadratic reward function by considering the A2A link quality and distance parameters for the dynamic reward determination component. Here, we utilize the link longevity and G/G/1 system-based queue waiting time metrics for the link quality. Also, we use Yen's N-shortest path algorithm for the distance parameter.

6.2.4 Organization of the chapter

The rest of the chapter is organized as follows: We first describe the QLR for AANETs. Then, we give the proposed QLR-driven AANET routing management framework. After these, the performance of the proposed method is evaluated and finally, the chapter is concluded.

6.3 Q-Learning (QLR)-Driven Routing Management for AANETs

6.3.1 AANET topology

In this chapter, we consider a two-layered AANET topology consisting of ground and aircraft planes, as shown in Figure 6.4. In this topology, we aim to implement a QLR-driven AANET routing management framework given in Figure 6.4. The details of these layers and the proposed framework could be summarized as follows:

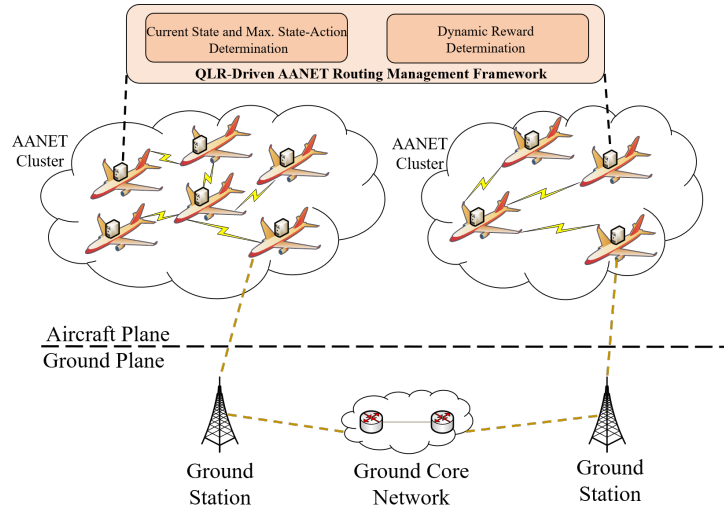


Figure 6.4 : The AANET topology.

- **Ground Plane:** In this plane, we aim to deploy the special GSs on terrestrial areas to utilize cellular communication, as shown in Figure 6.4. Correspondingly, we can establish a direct A2G link between aircraft and the closest ground station to enable broadband connectivity. More clearly, we can take advantage of the cellular communication model thanks to these ground stations to satisfy the Internet connectivity requirement of airplanes.
- **Aircraft Plane:** This plane consists of the airplanes connected through A2A links to form an AANET, as shown in Figure 6.4. Here, the packets of an aircraft are transferred through other airplanes and A2A links to reach the aircraft having Internet connectivity. Accordingly, airplanes without ground station connectivities can utilize the AANET to access the Internet. In this chapter, our primary focus is routing the packets over an already established AANET. For this reason, we do not focus on the physical characteristics and channel models of A2A links in this layer in the rest of the chapter.
- **QLR-Driven AANET Routing Management Framework:** This framework is implemented on airplanes to take their own routing decisions based on the Q-learning without any central node or entity. This framework is composed of two main parts current state& maximum state-action determination and dynamic reward determination, as shown in Figure 6.4. The details of this framework will be explained in Section 6.3.2.

As explained above, in this chapter, we adapt the Q-Learning, which is one of the basic reinforcement algorithms for routing management [157]. Here, we consider that each aircraft as an agent having a state, then this agent takes action (route packet), gets a reward, and switches to a new state during the routing as shown in Figure 6.5. Based on this, we map our AANET routing scenario to the reinforcement learning concept, and the main components of this system could be explained as follows:

Environment: The AANET in the aircraft layer constitutes the environment for Q-learning. Here, we specifically create the topology of AANETs in the form of clusters since the airplanes in the same cluster have the same movement and position characteristics. However, AANET is an ultra-dynamic and unstable environment, as detailed above. For this reason, the common clustering algorithms do not satisfy the requirements of the AANET environment. Therefore, we should utilize a clustering algorithm that considers the AANET-specific characteristics. At that point, we use our already proposed three-phased clustering algorithm for topology formation since this is a highly complex procedure. This algorithm considers the cluster creation, A2A link determination, and head selection procedures at the same time to create more stable clusters by also considering their aircraft-specific characteristics. The details about the three-phased topology formation algorithm can be found in Chapter 3. Therefore, in the rest of the chapter, we consider that our AANET topology is created in the form of more stable clusters based on the three-phased formation algorithm.

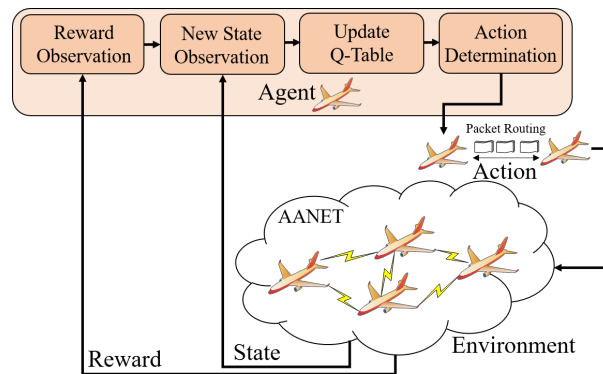


Figure 6.5 : Q-learning for AANET routing.

- *Agent:* We define each aircraft that routes the packet in an AANET as an agent. Therefore, each cluster includes agent $i, i = 1, 2, \dots, N$. Here, N shows the maximum

aircraft number in the corresponding cluster. Also, each cluster in the AANET can have a different aircraft number.

- *Action (a)*: Every packet transfer done by an aircraft through A2A links is defined as an action. The action set includes these actions, and this set is different for each aircraft depending on its A2A links. Also, each action causes a change in the current state of airplanes by affecting their queuing status.
- *State (s)*: We model each cluster in an AANET according to the G/G/1 queuing system and decomposition-based two-parameter characterization. Here, we represent the states of airplanes with G/G/1 system-based queuing load. Also, the states of A2A links are represented with decomposition-based two-parameter characterization by considering the accepted, waited, and transferred packet streams. Accordingly, each A2A link state is represented with two parameters as the arrival rate (λ_i) and squared coefficient of variation of arrival requests (C_{Ai}^2) to aircraft i in the form as $s_i(\lambda_i, C_{Ai}^2)$, $i = 1, 2, \dots, N$. These queuing-based models increase the suitability of the Markovian characteristic of reinforcement learning to the AANET environment.

Table 6.2 : Mapping of QLR components to AANET.

QLR Component	AANET Mapping
Environment	AANET
Agent	Aircraft
Action	Packet Transfer
State	G/G/1 based queuing load for aircraft states Decomposition-based two parameter characterization for A2A link states
Reward	A2A link quality A2A link distances
Reward Table	Table of A2A link rewards
Q-Table	Table of routing decisions

- *Reward (R)*: The aircraft accepts a reward for each packet transfer, and it is used for measuring the success of this transfer. This chapter utilizes the A2A link quality and distance metrics for reward definition. Here, the link quality is defined by the G/G/1-based queue waiting time and A2A link longevity parameters.

- *Reward Table*: The rows and columns of this table show the airplanes in a cluster. Also, the value of each cell denotes the reward of the corresponding A2A link. Therefore, each aircraft fills a row in the reward table with associated reward values belonging own links.
- *Q-Table*: The Q-table is filled through an iterative process by using Bellman equation as given with equation 6.1 for routing decisions. The aircraft is learned through the experience without any guide or controller by filling this table. We also summarize these mappings in Table 6.2.

Additionally, we define the aircraft having Internet connectivity through a ground station as IGW in the rest of the chapter.

6.3.2 Routing management framework

Table 6.3 : Elements of QLR-driven AANET routing management.

Element	Aim	Proposed Methodology
$Q_{t-1}(s, a)$	Current state-action determination	G/G/1-Based Queuing Model
α	Learning rate	Constant
$R(s, a)$	Dynamic reward determination	Polynomial function definition based on A2A link qualities and distances
γ	Discount factor	Constant
$\max_{a'} Q(s', a')$	Maximum state-action determination	G/G/1-Based Queuing Model Updated Layered-Hidden Markov Model (Updated-LHMM) Decomposition-based two parameter characterization

As explained above section, we determine the A2A links in a cluster through the three-phased clustering. According to its characteristics, we can establish more long-lasting links between airplanes. Therefore, the longevity of A2A links allows the clustered AANET topology could be taken as stable. Accordingly, we can apply the QLR-driven routing to these stable clusters. In this QLR-based methodology, we aim to enable airplanes in an AANET with memory, and accordingly, they can determine their own routing path dynamically. Thus, we study in a clustered topology whose links do not change as much as possible, and we handle dynamism on it with QLR. To achieve this, we adapt the Bellman equation as given with equation 6.1. With equation 6.1, the aircraft in a particular state considers all the possible routing directions and states of airplanes in these directions to select one of them, maximizing reward value.

$$Q_t(s, a) = Q_{t-1}(s, a) + \alpha(R(s, a) + \gamma \max_{a'} Q(s', a') - Q_{t-1}(s, a)) \quad (6.1)$$

In equation 6.1, s and a show a particular state of an aircraft and an action performed in that state, respectively. The $R(s, a)$ is a reward function outputting a reward value by taking action a at state s . Also, the $Q_{t-1}(s, a)$ equals the value of state-action pair and the s' is the state coming after from s by taking action a . Additionally, the a' is the action that could be taken at state s' . Here, γ is a discount factor that shows the importance of upcoming states with α learning rate. In this chapter, we adapt equation 6.1 to AANETs by determining its components with proposed methods as given in Table 6.3. The main aim of the component adaptations is to make QLR compatible with the ultra-dynamic and unstable characteristics of the AANET environment. Each component of the Bellman equation is executed through QLR-driven AANET routing management framework as given with Figure 6.4. We detail this framework and its components in the following subsections.

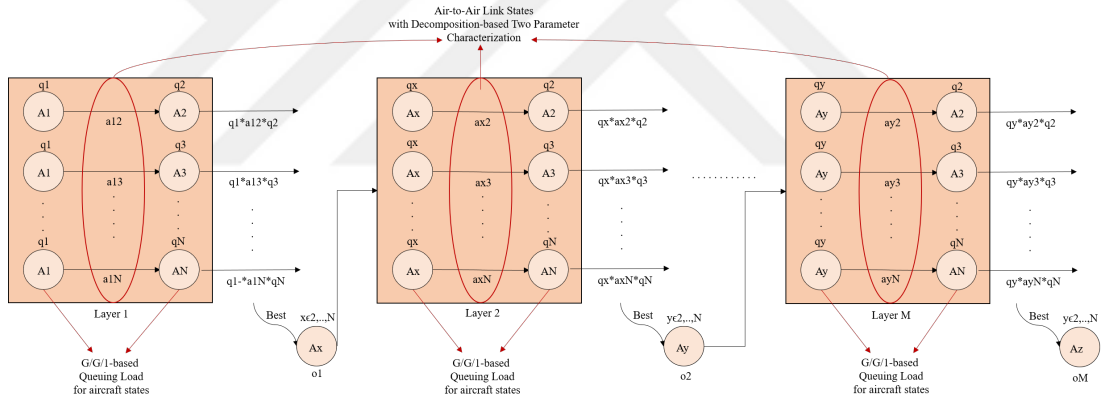


Figure 6.6 : Updated-LHMM for AANET routing management.

6.3.2.1 $(Q_{t-1}(s, a))$ and $\max_{a'} Q(s', a')$ determination

In this chapter, we take advantage of the $G/G/1$ system to find the current state-actions $(Q_{t-1}(s, a))$ of airplanes. Accordingly, we represent each aircraft with a $G/G/1$ queue due to the general distribution of arrival and service times. In this chapter, we relate the states of airplanes with the load conditions. For this reason, the queuing load parameter as given with equation 6.2 belonging to the $G/G/1$ system becomes the $Q_{t-1}(s, a)$ parameter for aircraft i during the current state determination.

$$L_{qi} = \frac{\rho_i^2(1 + C_{S_i}^2)(C_{A_i}^2 + \rho_i^2 C_{S_i}^2)}{2(1 - \rho_i)(1 + \rho_i^2 C_{S_i}^2)}, i = 1, 2, \dots, N \quad (6.2)$$

In the queuing load definition given with equation 6.2, $C_{A_i}^2$ and $C_{S_i}^2$ correspond to the squared coefficient of variation of arrival and service times of aircraft i . Here, $C_{A_i}^2 = \sigma_{A_i}^2 / \frac{1}{\lambda_i}, i = 1, 2, \dots, N$ is the equation for the squared coefficient of variation of inter-arrival time. Similarly, $C_{S_i}^2 = \sigma_{S_i}^2 / \frac{1}{\mu_i}, i = 1, 2, \dots, N$ gives the squared coefficient of variation of service time. In these equations, $\sigma_{A_i}^2$ and $\sigma_{S_i}^2$ correspond to the variance of the inter-arrival and service time, respectively. Also, λ_i becomes the mean arrival rate while μ_i represents the service rate on aircraft i . Here, these two parameters are also used to calculate the ρ_i in equation 6.2 as λ_i / μ_i . This ρ_i shows the utilization of single aircraft according to the mean arrival and service rates.

The current state-action pair is calculated as given with equation 6.2. However, the it is not sufficient to determine the routing path according to the QLR as given with equation 6.1. At that point, we update the Layered Hidden Markov Model to shape the packet routing process of airplanes and determine the $\max_{a'} Q(s', a')$. Accordingly, we can find the $\max_{a'} Q(s', a')$ for each step of the routing path from a source aircraft to IGW. Here, we conduct the next aircraft determinations in one step after from a current layer. Therefore, the result of current layer gives $\max_{a'} Q(s', a')$ observation to the next layer as shown in Figure 6.6. By applying the same procedures with updated-LHMM, we can find $\max_{a'} Q(s', a')$ for each step of the routing path to fill the Q-table.

Therefore, we divide the whole routing path into layers through the updated-LHMM and calculate possible aircraft probabilities in these layers to find the $\max_{a'} Q(s', a')$. Here, each layer corresponds to the $\max_{a'} Q(s', a')$ decision (that is also the next aircraft) and gives this decision as an output for the following layer, as shown in Figure 6.6. The row number of a layer in updated-LHMM depends on the neighbor number of an aircraft having a packet for transfer. Also, each row consists of two columns that represent the current and neighbor airplanes, as shown in Figure 6.6. The neighbor airplanes become hidden states, while output observation of a layer is called the observable state. The observations correspond to the $\max_{a'} Q(s', a')$ for that layer and could be represented as $O = o1, o2, \dots, oX, \dots, oM$ for M layers. Also, the hidden

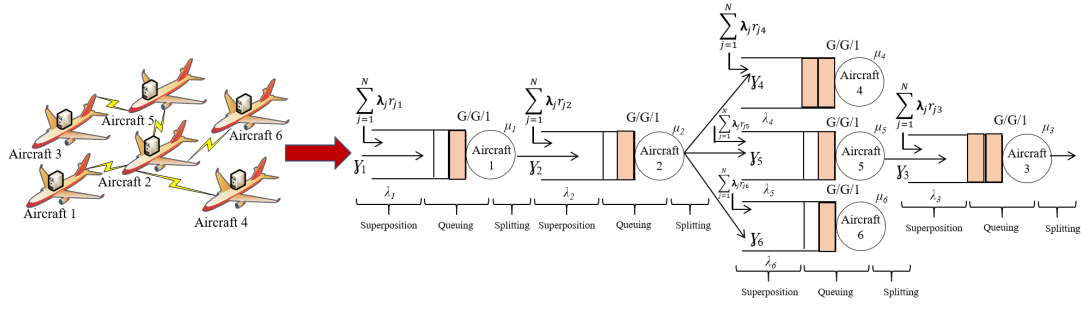


Figure 6.7 : Decomposition-based two parameter characterization-aided queuing model of a sample AANET topology.

states are not directly used, and they only show the possible route for the corresponding aircraft. Additionally, current states of airplanes are shown with $q1, q2, \dots, qx, \dots, qN$ as shown in Figure 6.6 and here, $q1 = L_{q1}, q2 = L_{q2}, \dots, qx = L_{qx}, qN = L_{qN}$ as given with equation 6.2. The transition probabilities between two airplanes in a layer are shown with $a_{ij}, i, j = 1, 2, \dots, N$. We will find these transition probabilities by utilizing the decomposition-based two-parameter characterization as will be explained in the upcoming part. Therefore, we utilize G/G/1 system-based queuing load and decomposition-based two-parameter characterization to find the observations for different neighbor airplanes in a corresponding layer. Then, we can select the aircraft having the best observation among them with a greater value ($\max_{a'} Q(s', a')$) and transfer it to the next layer to find the maximum state-action pair for that layer.

We apply these procedures through the M layer and fill Q- and reward tables by considering the next steps during the movements. With the updated-LHMM concept, we can consider all of the possible routing probabilities for each step to find the $\max_{a'} Q(s', a')$. Also, by utilizing updated-LHMM, we aim to represent the routing probability distributions in AANET over sequences of different airplanes, as shown in Figure 6.6. Additionally, as explained above, we find the transition probabilities between two airplanes through the decomposition-based two-parameter characterization. We will detail this concept in the following part of the chapter.

Decomposition-based two parameter characterization

To determine the transition probabilities in updated-LHMM, we take advantage of the G/G/1 queuing system with decomposition-based two-parameter characterization

[158], [159], [160]. At that point, we consider each aircraft as a router that executes packet transfer through the existed links in an AANET. Here, the accepted, waited, and transferred packets affect the transition probabilities between airplanes. Accordingly, in this chapter, we take characteristics of accepted, waited, and transferred streams into consideration to determine the transition probabilities between airplanes. On the other hand, the ultra-dynamic environment constitutes one of the critical AANET characteristics, which makes the modeling of AANET challenging. More specifically, we cannot reach the direct analytical results for these streams, and at that point, the approximation-based models help us determine the transition probability metric. This chapter uses decomposition-based two-parameter characterization as an approximation method that divides the network into smaller parts and analyzes these parts as separate systems. This type of decomposition strongly supports the AANET environment, where the airplanes are the elements of topological clusters. Then, we can investigate the A2A links in aircraft clusters separately.

As explained above, our main aim is to define a relation between the accepted, queued, and transferred packets of airplanes to approximate the transition probabilities in two-parameter characterization. Here, we divide the packet transfer procedures of airplanes into three phases: superposition, queuing, and splitting. Accordingly, we can establish a relationship with the accepted, queued, and transferred packets of airplanes for transition probability approximation. In these relations, our main aim is to determine the arrival rate (λ_i) and the squared coefficient of variation of the inter-arrival times (C_{Ai}^2) to aircraft i . Here, we aim to find more precise results for C_{Ai}^2 different from the $Q_{t-1}(s, a)$ since we should reach the A2A link states instead of a single aircraft. To execute these targets, two-parameter characterization starts with the superposition phase. In the superposition phase, the packets that come into an aircraft i are first combined as a total arrival stream to find C_{Ai}^2 as given with equation 6.3.

$$C_{Ai}^2 = \frac{\gamma_i}{\lambda_i} C_{0i}^2 + \frac{1}{\lambda_i} \sum_{j=1}^k \lambda_j C_{ji}^2, i = 1, 2, \dots, k \quad (6.3)$$

In equation 6.3, C_{0i}^2 and C_{ji}^2 are the squared coefficient of variation of the inter-arrival times for the accepted packets from the directly connected neighbors and other

airplanes, respectively. As shown in Figure 6.7, each aircraft can receive packets from its directly connected neighbors, or these packets could be originated by others (aircraft j , $j = 1, \dots, N$). Accordingly, the total arrival rate to an aircraft could be formulated as $\lambda_i = \gamma_i + \sum_{j=1}^N \lambda_j r_{ji}$, $i, j = 1, \dots, N$. In this formula, the γ_i represents the arrivals of aircraft i from the directly connected neighbors, and it could be found as $1/E[A_i]$. Here, A_i represents the arrival time of packets to aircraft i . Also, r_{ji} is the transfer probability of packets from aircraft j to i , and the arrival rate of aircraft j is the λ_j as shown in Figure 6.7.

After determining the behavior of accepted requests, we should investigate their subsequent paths. Here, the arrival packets wait in the queue of airplanes, and then they are transferred to other A2A links as shown in Figure 6.7. Based on this path of packets, we should consider the queuing effects first. For this aim, we define a relation between the arrived packets into an aircraft and the left packets from the queue as given with equation 6.4. This relation is defined by calculating the squared coefficient of variation between the departures of the aircraft queue (C_{Di}^2) as given with equation 6.4. In this equation, $\rho_i = \lambda_i / \mu_i$ is the utilization rate of the aircraft i as explained in Section 6.3.2.1. Also, C_{Ai}^2 is obtained from equation 6.3. Finally, C_{Si}^2 is the squared coefficient of variation of the service distribution for the aircraft i , and it is calculated as explained in Section 6.3.2.1. Therefore, in the first step, we highlight the input behaviors of an aircraft queue by defining equation 6.3. Then, to underline the departures from this queue, we define a relation as given with equation 6.4. Thus, in the last step, we should consider the departures forwarded to other queues from the current one, and this forwarding corresponds to the splitting procedure as shown in Figure 6.7. More clearly, the queued packets are transferred to other airplanes, and for this reason, the third phase corresponds to the splitting procedures to digitize these transfers. Accordingly, the splitting phase establishes a relation between the transfers to other airplanes and queuing departures as given with equation 6.5. In this relation, C_{Di}^2 is obtained from equation 6.4 and r_{ji} is the transfer probability of packets from aircraft j to i as explained above.

$$C_{Di}^2 = \rho_i^2 C_{Si}^2 + (1 - \rho_i^2) C_{Ai}^2, i = 1, 2, \dots, k \quad (6.4)$$

$$C_{ji}^2 = r_{ji}C_{Di}^2(1 - r_{ji}), i = 1, 2, \dots, k \quad (6.5)$$

Additionally, transfers from a queue (C_{ji}^2) mean input for another queue of the aircraft to which it is directly attached, as seen on equation 6.3 and equation 6.5. For this reason, the accepted, queued, and transferred packets should be reflected simultaneously for all airplanes, as shown in Figure 6.7. In this way, we can obtain a complete result for each A2A link. Based on this, we should consider equation 6.3, equation 6.4, and equation 6.5 under one formulation as given with equation 6.6.

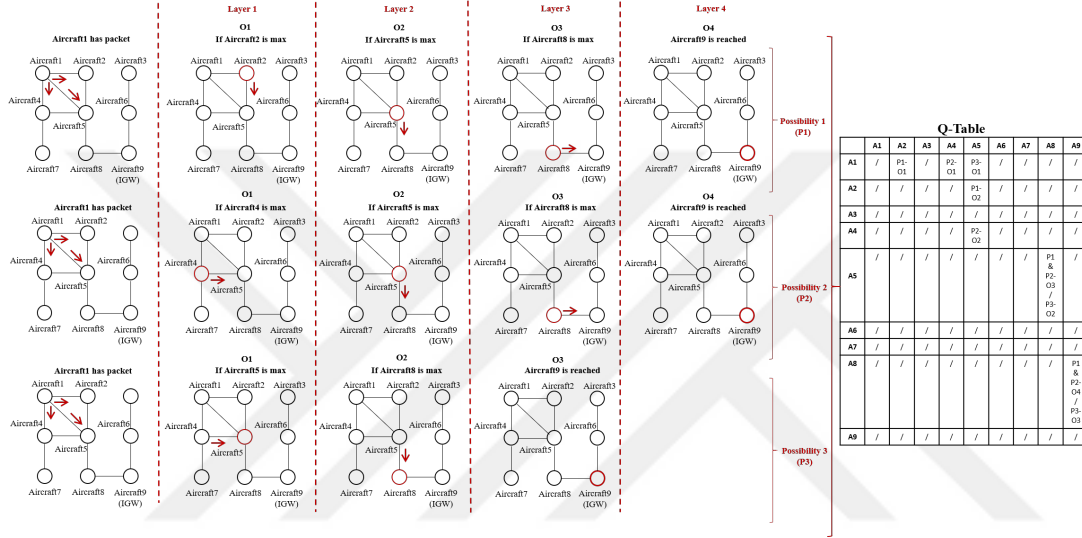


Figure 6.8 : Illustrations of a sample packet transfer using the updated-LHMM strategy.

$$C_{Ai}^2 = \frac{\gamma_i}{\lambda_i} C_{0i}^2 + \sum_{j=1}^k \frac{\lambda_j}{\lambda_i} r_{ji} (r_{ji} [\rho_j^2 C_{Sj}^2 + (1 - \rho_j^2) C_{Aj}^2] + 1 - r_{ji}) \quad (6.6)$$

6.3.2.2 Dynamic reward ($R(s, a)$) determination

As explained in above sections, we have a Reward Table to show the possible rewards of all A2A links. The rows and columns of this table correspond to airplanes in a cluster. The value of each cell denotes the reward of related A2A links between corresponding airplanes. Accordingly, this table can also illustrate the topology of a cluster. More specifically, if there is no connection between two airplanes, we should assign zero to the value of this link on the Reward Table. However, we cannot assign

fixed values to the existed A2A links since the packet transfer does not take place with the same quality on every A2A link in an AANET. Also, the quality of a specific A2A link does not always remain at the same value due to the ultra-dynamic AANET environment. For these reasons, in this chapter, we aim to determine the rewards dynamically instead of assigning fixed values. Clearly, we aim to set a reward value to an A2A link by considering the *quality* of packet transfer executed through this link with its *distance* value.

In this chapter, we define two main parameters to highlight the A2A link *quality*: stability (link longevity) and queue waiting time. Here, we assume that the *quality* of an A2A link firstly depends on its stability since the ultra-high velocities of airplanes in AANET reduce this stability. Here, the longevity of links becomes the crucial criteria for digitizing link stability. More specifically, this parameter gives how long an A2A link lasts between two aircraft. Accordingly, the longevity of a link between aircraft i and j is calculated with $T_{L_{ij}} = \frac{-\vec{D}_{ij}\vec{V}_{ij} + \sqrt{(\vec{D}_{ij}\vec{V}_{ij})^2 + V_{ij}^2(R^2 - D_{ij}^2)}}{V_{ij}^2}$. In this formula, $\vec{D}$ and $\vec{V}$ show the relative position and velocity vector between the next aircraft j and current ones i as explained in previous chapters. Additionally, the R offers the transmission range of an aircraft.

The second parameter for the *quality* becomes the waiting times of the transferred packets in the queue of corresponding airplanes. More specifically, the packets of an aircraft can wait longer in the queue of the nearest neighbor compared to the farther one. For this reason, we also add the queue waiting time parameter to the reward determination. To find the queue waiting time of airplanes, we again utilize the proposed G/G/1 based queuing model as explained in Section 6.3.2.1. According to G/G/1 model, the queue waiting time of airplanes (T_{q_i}) could be found by utilizing equation 6.7. In this equation, $C_{A_i}^2$ and $C_{S_i}^2$ correspond to the squared coefficient of variation of arrival and service times of aircraft i . The definitions of other parameters are the same with equation 6.2

$$T_{q_i} \approx \frac{\rho_i(1 + C_{S_i}^2)(C_{A_i}^2 + \rho_i^2 C_{S_i}^2)}{2(1 - \rho_i)(1 + \rho_i^2 C_{S_i}^2)\lambda_i}, i = 1, 2, \dots, N \quad (6.7)$$

In addition to the *quality*, the *distances* between airplanes affect the reward of the links. Clearly, the packets that are transferred to the destination aircraft through the shortest paths can receive higher rewards compared to the longer ones. This situation leads that *distance* of the aircraft to IGW through different A2A links is an important criterion during reward assignment based on the *quality*. Therefore, we should find the shortest distance from the aircraft to IGW without edge repetition. On the other hand, we have many state-action pairs in the QLR concept to select the most optimal one with a higher reward value, as shown in Figure 6.8. This situation increases the number of possible routing paths for reward evaluation, and finding only one shortest path would not be a sufficient assessment. For this reason, we should find additional shortest path options to fill the reward and Q-table by considering all possibilities. At that point, we utilize the *Yen's Algorithm* to find the N possible shortest path on AANET [161].

Algorithm 5 Yen's Algorithm for N-Shortest Path

Data: AANET topology, Source aircraft s , IGW, N

Result: N-Shortest paths from source aircraft to IGW

Initialize A and B

1. Determine A^1 from s to IGW with Dijkstra
 2. Store $n - 1$ paths ($n = 1, 2, \dots, N$) in A and candidates in B
 3. To find A^k , get A^{k-1} with path $\langle s, a_1^{k-1}, a_2^{k-1}, \dots, a_l^{k-1}, \text{IGW} \rangle$
 Set of vertices $P = \langle s, a_1^{k-1}, a_2^{k-1}, \dots, a_l^{k-1} \rangle$ for analysis
 4. **for** s to a_l^{k-1} **in** P **do**
 - if** $A^i \in A$ with $\langle s, a_1^{k-1}, a_2^{k-1}, \dots, a \rangle$ **then**
 - Set edge weight from a to its immediate neighbor to ∞ for A^i
 - Set $R^k = \langle s, a_1^{k-1}, a_2^{k-1}, \dots, a_l^{k-1} \rangle$ as root path
 - Set the path from a_l^{k-1} to IGW as S^k spur path
 - Remove vertices of R^k from AANET topology
 - Find shortest path from a to IGW with Dijkstra
 - if** Path is found and returned **then**
 - Add R^k and S^k to form candidate path
 - Add candidate path into B
 - end**
 - end**
 5. Choose path from B with shortest distance, add into A
 6. Go to step 3 and continue until N shortest path determined
-

Here, we aim to find the N shortest path from source aircraft to IGW with N iterations as $A^1, A^2, \dots, A^N$ and keep them in container A. Here, the first shortest path A^1 is found

by using Dijkstra's algorithm and other potential paths are held in container B. Then, a path with a lower cost is selected from container B to insert A. The pseudo-code of Yen's algorithm is given with Algorithm 5. As utilized in Algorithm 5, the path $\langle s, a_1^{k-1}, a_2^{k-1}, \dots, a \rangle$ from $\langle s, a_1^{k-1}, a_2^{k-1}, \dots, a_l^{k-1}, IGW \rangle$ is called as root path while the a_l^{k-1} to IGW is the spur path.

After conducting the *quality* and *distance* based operations, we should combine them to determine the final reward values. Here, by defining the link longevity and queue waiting time parameters for each shortest path separately, we obtain a reward function between aircraft j and i ($R_{ji}(s, a)$) in the polynomial form as given with equation 6.8. In this function, if there is a link between two airplanes and this link is located on the shortest paths, then the value of X is taken as X_1 . Conversely, if the existed link between the aircraft i and j is not on shortest path, in this case the X equals to X_2 ($X_1 > X_2$). Finally, the value of X is taken as 0 if there is no A2A link between aircraft i and j . Moreover, the reward of a packet transfer through an A2A link connecting an aircraft to an IGW should be greater than other links given with equation 6.8 since the goal node has a higher reward according to the QLR principle. Additionally, in equation 6.8, $T_{L_{ij}}$ and T_{q_j} are found with equation 6.2 and equation 6.7. Here, $T_{L_{ij}}$ could be directly utilized while T_{q_j} at the denominator since we prefer links having higher longevity with a lower queue waiting times. More specifically, we can obtain a higher reward for an aircraft having a lower queue waiting time (T_{q_j}). Therefore, we can fill the reward table by considering all of the state-action pair and shortest path possibilities correspondingly, as shown in Figure 6.8.

$$R_{ji}(s, a) = T_{L_{ji}}X^2 + \frac{1}{T_{q_i}}X + c, \quad i, j \in 1, 2, \dots, N$$

$$c = \begin{cases} 100, & \text{if } i = IGW \& x \neq 0 \\ 0, & \text{otherwise} \end{cases} \quad (6.8)$$

Finally, the learning rate (α) shows how often we modify our estimations. The number of iterations to reach the solution could be increased if this rate is selected as too small. Additionally, we cannot control the dynamic changes in the environment if it is taken as a more excellent value. Also, the discount factor (γ) shows the importance of future rewards, and we relate it with the change rate of the cluster. Clearly, if there are too

many agent replacements in a cluster, then the immediate rewards become essential for us since the reward table should be updated for every change. These values should be selected from the range $0 < \alpha, \gamma < 1$ according to the QLR principles. By utilizing the proposed methodologies with equation 6.1, each aircraft can fill the Q-learning table according to its state and maximum state-action pair determinations as given with equation 6.8.

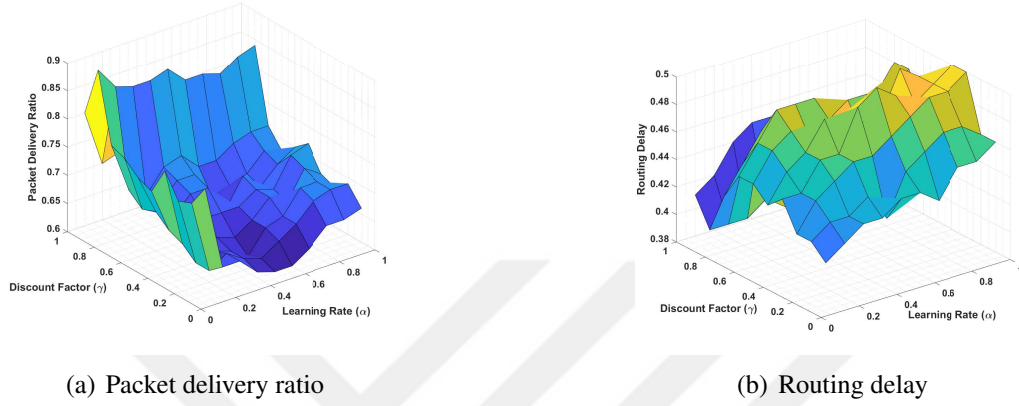


Figure 6.9 : The effects of α and γ on (a) packet delivery ratio and (b) routing delay.

6.4 Performance Evaluation

This part of the chapter aims to evaluate our proposed approach's performance and compare it with some routing algorithms in the literature. For this aim, we first detail our simulation environment and utilized parameters. Then, we evaluate the results belonging to these simulations.

6.4.1 Simulation environment

We utilize the *OMNET++* framework to create the AANET in the form of clusters to simulate the routing management procedures. We create a $1000 * 1000$ NMI air traffic area with four ground stations at each corner of this area in these simulations. Here, NMI is selected since it is a unit of length used in air and space navigation. As explained in above sections, we define the airplanes connected to these ground stations as IGW, and the Internet connectivity is enabled thanks to these IGWs through an established AANET. Here, the distance between aircraft and ground station is taken as 200 NMI for connection establishment [162]. If this length is exceeded, then the

connection cannot be established as in terrestrial cellular networks. Also, the channel capacity of each ground station is taken as 2 Mbit/sec with 125 w transmit radio frequency power. These are the generally utilized values during A2G connectivity for extended performance. Moreover, the A2A links have LOS characteristics and a relatively high Signal to Noise Ratio due to the battery power is no limitation for the airplanes. For this reason, we take the minimum distance to establish A2A as 450 NMI at 35000ft flight altitude with equal capacities and 10 Mbps A2A link bandwidth as detailed in previous chapters. Here, we normalize the packet intensity of links according to the given capacities of them. Also, we use an idealized communication link model with an omnidirectional transmission range. This omnidirectional antennas can execute the signal transmission in all directions; for this reason, it could be considered more suitable for AANETs. Additionally, we consider each aircraft as a router with the capacity to receive, queue and route packets to its neighbors according to the AANET concept and we model these queues according to the G/G/1 system with buffer size of 30 packets. This maximum size is selected to reduce the drops caused by queuing overflow. Nevertheless, the received packets could be dropped if the queues of airplanes are full. Also, the packets are generated at random airplanes according to the general distribution and we limit the packet production rate of airplanes to 4 per second with 512 Kbytes. These values are also selected according to the queuing capacities to reduce the drops caused by drops. Additionally, we consider 10 traffic intensity levels on OMNET++ ranging from 10% to 100% of total network capacity. Correspondingly, we normalize the network traffic intensity according to the capacity. Here, we also generate 100 traffic matrices for each traffic intensity according to the general distribution to obtain more exact results. Moreover, the error bars will be given in Section 6.4.2 indicates the 95% confidence interval.

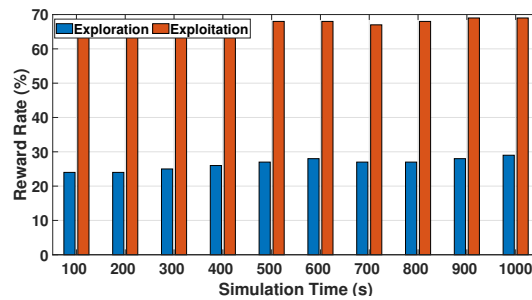


Figure 6.10 : Reward performance of exploration-exploitation phases.

In these simulations, we also utilize the TensorFlow as a backend with Python 2.7 to implement the QLR-driven AANET routing management framework of our topology. Here, the discount factor of future rewards is selected as 0.9 with a 0.1 learning rate. We determine the discount factor and learning rate values according to their effects on routing delay and packet delivery ratio. Accordingly, we select $\alpha = 0.1$ and $\gamma = 0.9$ to enable a good trade-off between the routing delay and packet delivery ratio, as shown in Figure 6.9. Also, to utilize Yen's N-shortest path algorithm, we use the Dijkstra Algorithm in the first step as given with Algorithm 5.

Moreover, we evaluate the reward metric to tackle the success of the exploration and exploitation phases. In this chapter, the exploration and exploitation phases follow a greedy approach, as shown in Figure 6.8. More clearly, the airplanes do not initially know the current topology and routing paths. For this reason, the aircraft explores the AANET cluster to improve the current information and next rewards in the exploration phase. On the other hand, in exploitation, the currently estimated knowledge is utilized to get a higher reward based on the Greedy approach.

In this chapter, we highlight the rewards of the exploration and exploitation phases based on simulation time, as shown in Figure 6.10. Here, we first execute the exploration phase to explore the clustered AANET topology. Accordingly, we aim to improve the following rewards by continuously exploring the environment. For this reason, as shown in Figure 6.10, we can perform the proposed methodology with a lower reward compared to the exploitation. More clearly, the exploitation phase utilizes the obtained knowledge based on the Greedy model to get higher rewards. For this reason, as shown in Figure 6.10, we can receive roughly 50% better reward compared to the exploration case. Based on these, we evaluate our performance evaluation metrics by considering the exploitation phase. The details of these metrics will be explained in the upcoming part.

Furthermore, we evaluate the performance of the proposed method in terms of routing delay, packet loss rate, accuracy, and network load. The details of these metrics could be summarized as follows:

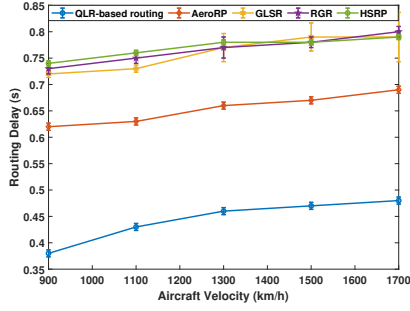
- *Routing Delay* is computed in terms of the waiting time of the routing packets in the queues of the airplanes . Accordingly, it is calculated by using the equation 6.7. This metric is crucial as it indicates the operation speed of the routing process.
- *Packet Loss Rate* shows the ratio of packets that are not successfully accepted by the destination aircraft to the total sent. This metric is vital in showing the rate at which packets are successfully delivered.
- *Accuracy* is found by dividing the number of received packets by the airplanes by the number of packets sent at the MAC layer. This metric gives information about the accuracy of the determined route by the routing algorithm in the ultra-dynamic AANET environment.
- *Network Load Rate* is calculated by dividing the number of packets in the queues of airplanes by the total sum of maximum sizes of all the queues. Here, loads of the queues are calculated with equation 6.2. This metric is essential to show that packets are not waiting in queues of airplanes and are being routed.

In the following part of the chapter, we will give the results of performance evaluations based on these metrics and the above-explained simulation environment.

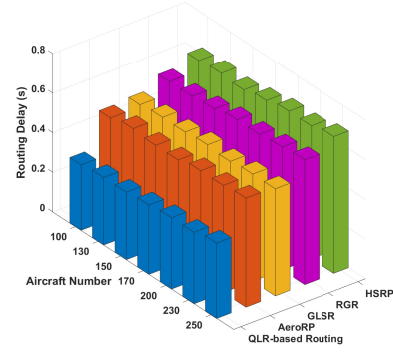
6.4.2 Simulation results

We evaluate the performance of the proposed QLR-based routing method by comparing it with the GLSR, AeroRP, HSRP, RGR algorithms as detailed in Chapter 1 and Section 6.2.2. The main reason for choosing these algorithms is that their purpose and performance criteria are almost the same as our routing methodology. Also, the details of these routing methodologies are given in Section 6.2.2.

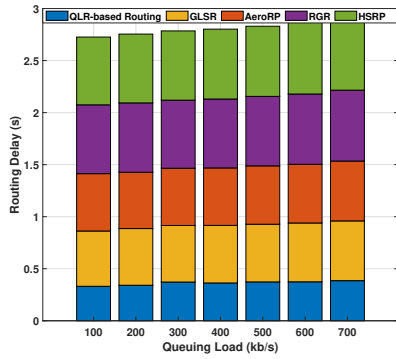
- **Routing Delay:** To calculate the routing delay, in our simulations, we train each methodology 100 times for each different x-axis value, and then, we take the average routing delay of each step as shown in Figure 6.11. These average delays are summed to obtain a total routing delay for each x-axis episode. Here, the x-axis values correspond to the aircraft velocity, number, queuing load, and network traffic intensity, as shown in Figure 6.11. In all methodologies, the routing delay tends to



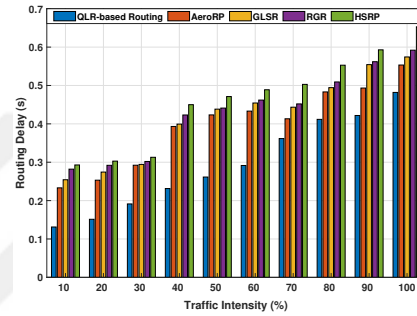
(a)



(b)



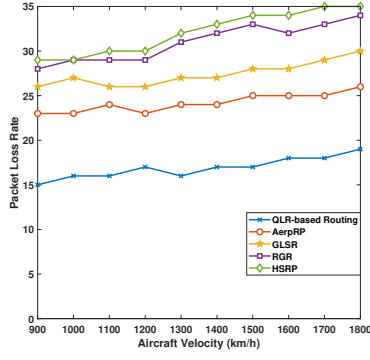
(c)



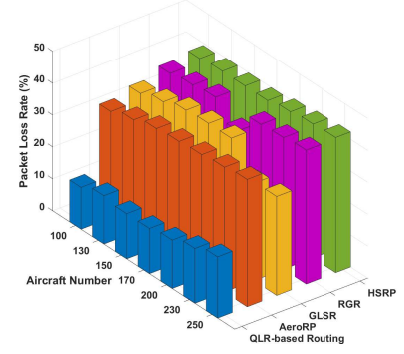
(d)

Figure 6.11 : Routing Delay performance of AeroRP, GLSR, HSRP, RGR, and QLR-based routing methodologies according to the (a) Velocity of airplanes (b) Number of airplanes (c) Queuing load of airplanes (d) Network traffic intensity

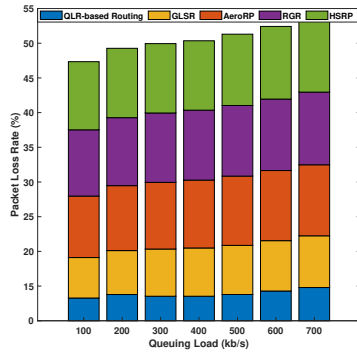
increase with the growing x-axis values. Here, the AeroRP, GLSR, RGR, and HSRP observe the routing delay greater than 0.55s. The performance of the AeroRP is greater than the other three methodologies in literature as shown in Figure 6.11(a), Figure 6.11(b) and Figure 6.11(d). The AeroRP takes time to intercept as a primary metric for routing decisions by including speed and relative velocity components. Accordingly, the source node in AeroRP can have an idea about the potential neighbors in the transmission range. This situation reduces the routing delay of the AeroRP compared to the other methodologies in the literature. However, as shown in Figure 6.11(c), the GLSR improves its performance compared to the other three methodologies in terms of queuing load. The main reason for this is that the GLSR also forwards the packets to the geographically closest neighbor with the additional speed of advance metric. Therefore, it can route the packet by considering the queuing load and waiting time situations thanks to this metric. Unlike the AeroRP



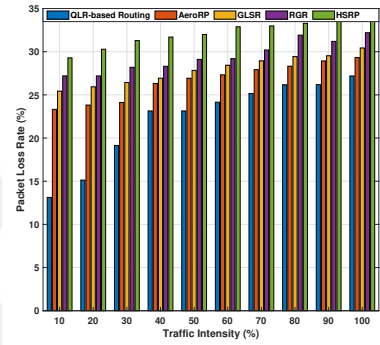
(a)



(b)



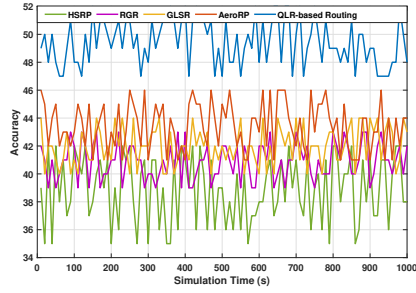
(c)



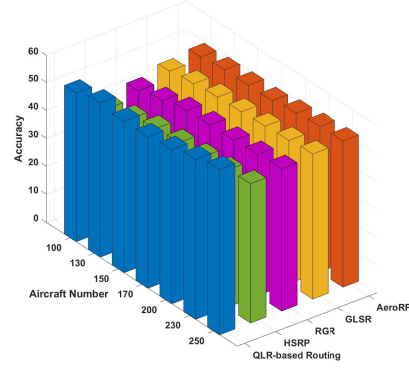
(d)

Figure 6.12 : Packet loss rate performance of AeroRP, GLSR, HSRP, RGR, and QLR-based routing methodologies according to the (a) Velocity of airplanes (b) Number of airplanes (c) Queuing load of airplanes (d) Network traffic intensity

and GLSR, the RGR utilizes the request/response messages to obtain the location information of neighbors. These periodic message exchanging procedures can increase the routing delay of RGR compared to the other methodologies. Moreover, the HSRP works based on increasing or decreasing the spatial radius by observing the change of flight flow rate. More specifically, the update rate of HSRP is increased if the aircraft moves faster, aircraft number and flight flow rate are larger. The increased update frequency has a significant effect on the increase in routing delay of HSRP. Unlike, our proposed QLR-based routing strategy outperforms the other methodologies thanks to the delay consideration during reward determination as given with equation 6.8. By deploying the queuing delays of airplanes as provided with equation 6.7 in reward determination, we can enable airplanes to choose routes where their packets will wait less in the queue. Here, we can achieve the routing delay of less than 0.43s as shown in Figure 6.11. Accordingly,



(a)



(b)

Figure 6.13 : Accuracy performance of AeroRP, GLSR, HSRP, RGR, and QLR-based routing methodologies according to the
(a) Simulation time (b) Number of airplanes

we can reduce the routing delay by roughly 30% compared to the investigated methodologies in the literature.

- **Packet Loss Rate:** The average packet loss is expressed as the average of all losses observed at a certain point of the x-axis. Like the routing delay, we train each methodology 100 times for each certain point of x-axis values. Also, as in the routing delay, we select the x-axis values of packet loss metric as aircraft velocity, number, queuing load, and network traffic intensity as shown in Figure 6.12. Again in all methodologies, the packet loss tends to increase with the growing x-axis values. Here, the AeroRP, GLSR, RGR, and HSRP observe packet loss rates higher than the 24% level. The AeroRP has a slightly better performance compared to the other three methodologies. It adds location information to each per-hop packet instead of periodic beacon messages. All neighboring node nodes in the transmission area can receive this location information from the packet header and keep it on the neighbor table to utilize during routing. Accordingly, the AeroRP can transfer packets with accurate routes, which affects reducing packet losses

compared to the HSRP, GLSR, and RGR. Also, the HSRP, GLSR, and RGR have approximately the same performance as shown in Figure 6.12. The change of spatial radius cannot effectively increase the packet delivery ratio of HSRP compared to the other two methodologies, as shown in Figure 6.12. The requests, replies, and hello messages carry source location, destination location, and neighbor location

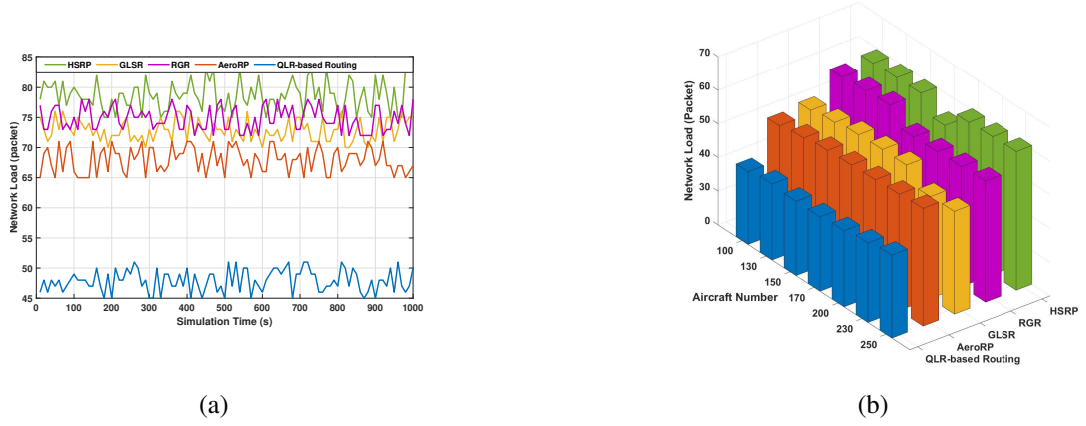


Figure 6.14 : Network load performance of AeroRP, GLSR, HSRP, RGR, and QLR-based routing methodologies according to the (a) Simulation time (b) Number of airplanes

information, respectively, in RGR. These could be utilized to update the node tables for reducing the packet losses compared to the HSRP. However, the GLSR has a reduced packet loss in terms of queuing load compared to the HSRP, RGR, and also airplanes, as shown in Figure 6.12(c). The main reason for this situation is that the GLSR can perform load sharing between the neighbors, thanks to the speed of advance metric. Unlikely, as shown in Figure 6.12, we observe less packet loss in our proposed QLR-based routing strategy thanks to the updated-LHMM utilization. In the updated-LHMM, we consider the corresponding and neighbor aircraft states in terms of queuing load as given with equation 6.2. Also, we can consider the link states by observing accepted, queued, and transferred packets as given in equation 6.6. Therefore, we can route the packets by considering less loaded airplanes and links during $\max_{a'} Q(s', a')$ determination as summarized in Figure 6.6. Accordingly, we can achieve packet losses of less than 16%, and we can reduce the packet losses roughly 33% compared to the investigated methodologies in the literature.

- **Accuracy:** We evaluate the accuracy of investigated methodologies according to the simulation time and number of aircraft parameters as shown in Figure 6.13. To analyze the accuracy according to the simulation time, we take average accuracies of 100 aircraft in 100s intervals and sum them for each corresponding episode as shown in Figure 6.13(a). Here, the AeroRP, GLSR, and RGR observe an accuracy of less than 46% with very close accuracy values due to the consideration of

geographic effects during routing. The AeroRP uses a prediction mechanism by utilizing the last known distance and velocity parameters to determine the neighbors in the transmission range of an aircraft. Clearly, the aircraft can accept hello beacons with coordinate and velocity data to select the time-to-intercept for each neighbor. Similarly, in GLSR, the packets are forwarded to the geographically closest neighbor to the destination. Also, the RGR uses both reactive and greedy approaches during the routing of data packets. Differently, the HSRP updates the neighbor node information table based on flight flow rate. Accordingly, they change the update frequency of hello beacons according to the flight flow rate, and this reduces the accuracy of HSRP by roughly 10% compared to the other three methodologies. However, our proposed QLR-based routing methodology outperforms these works by enabling an accuracy of higher than 50%. In our method, the airplanes can take more precise routing decisions based on their learned experience, thanks to equation 6.1 and Q-table as given in Figure 6.8. Also, we utilize the link longevity parameter during reward determination as provided with 6.8. This parameter increases accuracy by enabling us to select more stable paths during routing. Similarly, we can improve the accuracy 26% compared to the investigated works in literature according to the increased aircraft number as shown in Figure 6.13(b).

- **Network Load:** Like the accuracy metric, we evaluate the load on the network according to the simulation time and aircraft number parameters as shown in Figure 6.14. Again, we calculate the load on the network according to the simulation time by taking the average of results for 100 aircraft in 100s intervals and sum them for each corresponding episode as shown in Figure 6.14(a). Also, we measure the load on the network by increasing aircraft number 100 to 250 as shown in Figure 6.14(b). Here, we take the average network loads of airplanes for each corresponding aircraft number and sum them. The high load on the network shows that routing is not executed effectively, and the number of packets waiting in the queue is growing. The investigated four routing methodologies experience higher than 65% load on the network as given with Figure 6.14(a). The AeroRP makes its routing decisions per-hop basis based on speed-based heuristics different from the other geographic

routing protocols as GLSR and RGR. This situation reduces its network load by roughly 10% compared to the other two algorithms. The GLSR and RGR work based on greedy geographic forwarding, and this means that they aim to find the nearest neighbor during routing. Here, the GLSR utilizes the speed of advance metric to consider the congestion and load balancing issues during routing. Also, the RGR keeps the location information of each neighbor and updated them via periodic hello messages. Accordingly, these situations increase the network load on these algorithms. The HSRP observes higher network load due to the dynamic correction of spatial radius. Unlikely, our proposed QLR-based routing strategy outperforms these four methodologies achieving network load less than 50% as given with Figure 6.14(a). In the proposed QLR-based routing, we aim to select the most optimal paths by considering the states of airplanes and links through updated-LHMM as shown with Figure 6.6. To determine these states, we consider the load of the airplanes and links as given with equation 6.2 and equation 6.6. Also, we can achieve these determinations for all neighbors at the same hop to determine the optimal one as given with Figure 6.8. Additionally, as shown in Figure 6.13(b), we can reduce the network load 24% compared to the investigated works in literature according to the increased aircraft number.

6.5 Conclusion

This chapter proposed QLR-driven routing management for AANETs to reduce the delays, packet losses, and network load with higher accuracy. In this methodology, our main aim was to adapt the Bellman equation in QLR to the AANET environment by proposing different methods for each of its elements. To obtain the maximum state-action determination component of the Bellman equation, we updated the layered hidden Markov model. In the updated-LHMM, we determined the states of airplanes by modeling them according to G/G/1 queuing system. Also, we utilized the decomposition-based two-parameter characterization to obtain the A2A link states in updated-LHMM. The decomposition-based two-parameter characterization takes characteristics of accepted, waited, and transferred streams into consideration to determine the A2A link states. Finally, to obtain the reward value in the Bellman

equation, we defined a quadratic reward function based on the A2A link quality and distance metrics. Here, we used the link longevity and G/G/1-based queue waiting time parameters to define the link quality. Also, we used Yen's N-shortest path algorithm to reach the distance metric in the reward function definition. According to the simulation results, the proposed strategy reduces the routing delay and packet losses by 30% and 33% compared to the works in the literature. Also, we can reduce the network load by 24% with 26% higher accuracy.

General concept for the next chapter

Up to this point, we proposed different management approaches for the topology-related issues and routing procedures on AANETs. More clearly, we proposed AANET topology formation, sustainability, and mapping management models with a routing approach.

Based on these, in the following chapter, we aim to propose a digital twin based approach to conduct these management procedures through a more proactive and real-time approach. In this way, we aim to improve the compatibility of these management models with the ultra-dynamic and unstable AANET environment.

7. DIGITAL TWIN FOR AANET MANAGEMENT

7.1 Purpose

The airplanes were top of the places that disrupted the seamless connectivity requirement of next generation networks. The AANETs take the attention of both industry and academia to satisfy this connectivity requirement with the low cost, easy deployment, and continuous coverage features. On the other hand, the ultra-dynamic characteristics of AANET with unstructured topology make its environment hard-to-follow. Here, AI-based methodologies have an essential role in handling the management complexity of this hard-to-follow environment as detailed in the previous chapters. However, these methodologies increase the computational complexity of aircraft due to the continuous update, convergence time, and scalability issues. At that point, we propose the utilization of the Digital Twin (DT) technology to handle the computational complexities of AI-based methodologies on AANET. The DT can virtually replicate the physical AANET components through closed-loop feedback in real-time. Therefore, this work introduces the utilization of DT technology for the AANET orchestration and, accordingly, proposes a DT-enabled AANET management (DT-AANET) framework. This framework consists of the Physical AANET Twin and Controller, including Digital AANET Twin with Operational Module. Here, the Digital AANET Twin virtually represents the physical environment. Also, the operational module executes the AI-based computations. We specifically illustrate the working of operational module on the unsupervised learning-based topology formation and supervised learning-based topology sustainability management. Here, we present a case study by proposing a customized K-means algorithm for unsupervised learning-based topology formation instead of our proposed three-phased formation. In these case studies, we aim to show the usability of the proposed DT-AANET framework and support this through evaluation results.

7.2 Introduction

One of the principal requirements for the next generation networks is to stay connected anytime, anywhere, and under any circumstance. Aircraft were at the top of the places that disrupted this situation. However, the IFC becomes one of the main requirements for passengers instead of luxury with the evolving technology [163]. More specifically, 54% of passengers agree to pay extra fees for the IFC services, 75% of them are ready to change the airline they use to get seamless and faster Internet connectivity, while 20% already changed the current airlines as explained in Chapter 1. These key figures reveal the need for new technologies to provide seamless Internet access for aircraft.

7.2.1 Management challenges for AANETs

AANETs become an innovative solution to enable seamless IFC with higher coverage and lower cost. The AANETs work based on the transmission of packets through A2A links by considering aircraft as routers to enable IFC. In this architecture, each aircraft act as a router and transfer packets to others in an ad-hoc manner without any central node or entity. Although AANET is an emerging technology, it brings the following challenging characteristics that make the AANET environment hard-to-follow.

- **Ultra-Dynamic Network:** The aircraft are the main components of the AANET structure, and they make the AANET environment ultra-dynamic. More precisely, one of the important, challenging characteristics of AANETs is the ultra-dynamism, and this originates from the ultra-high-speed of aircraft as shown in Figure 7.1.
- **Unstructured Ad-Hoc Topology:** The unstructured ad-hoc topology is mainly caused by the ultra-dynamic network characteristic of AANETs. The ultra-dynamic characteristic increases the breakage rate of A2A links between aircraft, and these short-lived A2A links create an unstructured ad-hoc topology with continuous changes as shown in Figure 7.1. These challenges reduce the packet transfer efficiency of AANET by increasing the packet transfer delay and packet drop rates.

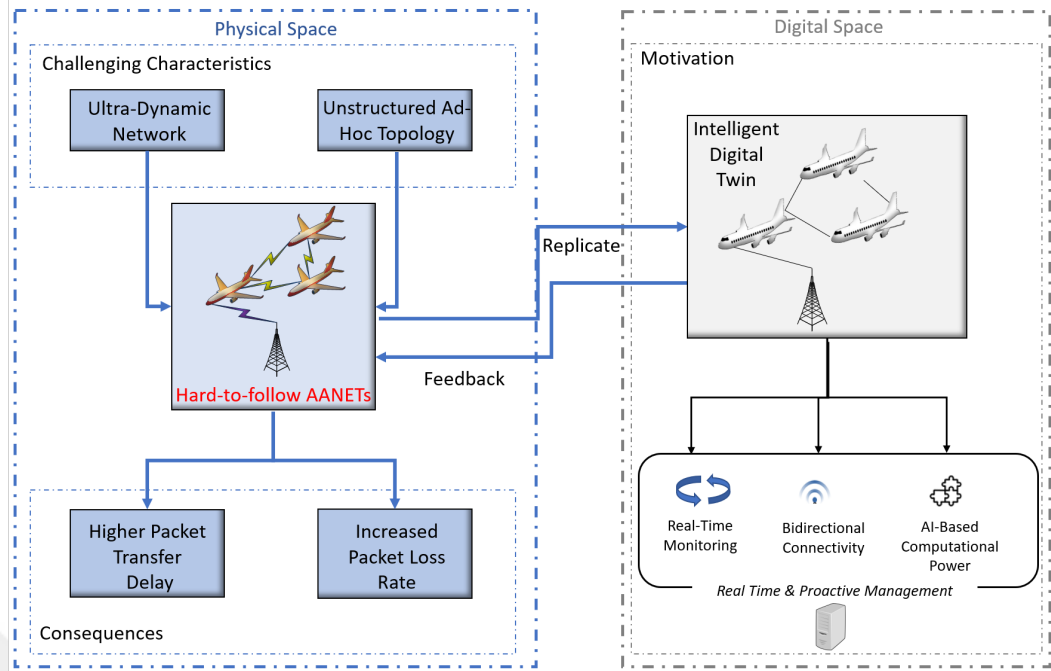


Figure 7.1 : Challenging characteristics of hard-to-follow AANETs and DT Evolution.

The AANET management becomes complex due to the above-explained challenging characteristics, and it requires AI-based methodologies to overcome them. However, these methodologies increase the computational complexity on aircraft due to the continuous update, convergence time, and scalability issues. Therefore, we should handle this computational complexity by also solving these issues. For this aim, in the following subsection, we will introduce a novel and powerful concept called *Digital Twin (DT)*.

7.2.2 The digital twin evolution for hard-to-follow AANETs

As a solution to the challenging characteristics of hard-to-follow AANETs, we explore the concept of DT that can virtually replicate all components of a physical system in a digital space as shown in Figure 7.1. Here, there are various approaches of DT technology as pre-DT, DT, adaptive DT, and intelligent DT [164]. In this chapter, as shown in Figure 7.1, we consider an intelligent DT approach to support the real-time and proactive management of the physical AANET topology by also enabling additional features summarized as follows:

- **Real-Time Monitoring:** Thanks to the real-time monitoring feature, there is a continuous relationship between the physical space and its digital component. Clearly, DT permanently monitors the physical space through received real-time data. Here, by utilizing this real-time data, the DT replicates a digital mirror in the form of present twin if it accepts an error or trigger message from physical space. Otherwise, the received data could be deleted or kept in an edge-cloud for further utilization in DT as a past twin. As seen in this process, one of the main support for real-time monitoring is the bidirectional connectivities established between physical and digital spaces to enable continuous data transfer.
- **Bidirectional Connectivity:** One of the essential features of the intelligent DT is the bidirectional connectivity between physical space and its virtual mirror. Since the collected data is replicated in the digital space and results are transferred to the physical world again in closed-loop feedback thanks to these bidirectional connectivities. More specifically, the DT uses continuous bidirectional interactions between the physical world and its digital space to improve network management performance [165]. Also, this becomes the main helpful difference between the simulation environment and DT [166].
- **AI-Based Computational Power:** The intelligent DT can exhibit the AI-based methodologies during network management to increase performance. These methodologies support the computational operations in DT to create future twins. More specifically, the DT could replicate the historical and real-time data in the form of past and present twins based on the physical environment as explained above. By applying AI-based methodologies to these past and present twins, the future twins could be trained or predicted with unsupervised and supervised learning-based methodologies. These future twins reveal the structure that will emerge when the virtually tried methodologies are implemented. Accordingly, this improves network performance by preventing time-wasting by wrong methods.

As explained above, the intelligent DT can support the above-explained features to increase network management performance. Accordingly, we can obtain a real-time and proactive management opportunity in a digital space for the AANETs.

Table 7.1 : Role of digital twin on network management.

Methodology	Environment	Enabling Technology for DT	DT Task
Intelligent UAV Swarms [167]	UAV Swarms	Deep Learning	UAV Mission Control
Intelligent DT-based Software Defined Vehicular Network [168]	Vehicular Networks	SDN	Temporal Graph Routing
DT for 5G and beyond [169]	5G	Artificial Intelligence	Automation, Validation, and Optimization of 5G Predictive Maintenance
Block-chain for DT [170]	Industry 4.0	Block-chain	Improved Production Collaborative Engineering
DT for CPSs [171]	Industry 4.0	Artificial Intelligence	Movement Prediction Fault Management
DT for Optical Communication [172]	Optical Networks	Deep Learning	Hardware Configuration Transmission Simulation
DT-enabled 6G [173]	6G	Machine Learning Models	Internet of Everything
DT for B5G Smart Grids [174]	B5G Smart Grid	Machine Learning Models Blockchain	Smart Grid Deployments

7.2.3 Related work

There are various works on literature that utilize the DT for different types of network management issues due to the above explained advantages. A DT-based intelligent cooperation framework for UAV swarm is proposed in [167]. This work aims to provide the optimal decisions to UAV swarms in real-time since their cooperation is a highly complex task with high computation requirements. An intelligent DT-based software-defined vehicular network to execute the prediction, verification, and validation procedures before applying them to the real environment in [168]. They utilize this concept for the temporal graph routing as a case study. The utilization of DT technology for 5G and beyond is proposed in [169]. They aim to speed up the deployment of new technologies for 5G with highly flexible and available testbed facilities. The utilization of DT for industrial application with blockchain support is proposed in [170]. They aim to take advantage of blockchain's transparency, trust, and security characteristics during DT deployment in industrial applications. The role of artificial intelligence to solve the DT-related problems of Industry 4.0 is investigated in [171]. For this aim, they also demonstrate the applicability of AI during movement prediction with the help of collected real-time data through DT. A DT framework is proposed in [172] for efficient and reliable management of optical communication

systems. This framework includes the intelligent fault management model, flexible hardware configuration model, and dynamic transmission simulation model with the deep learning additions for the optical communication system management. The design requirements of the DT to enable 6G are proposed in [173]. This work provides a comparative description of various twins and architectural components of DT. The context awareness and fault detection capabilities of the DT are discussed for the smart grids in beyond fifth generation networks in [174]. They aim to develop an autonomous and self learning agents with DT technology by also taking advantages of blockchain and machine learning methodologies. We also summarize the main concepts of these works in Table 7.1 to give the differences between them in a more compact form.

Although there are various works in literature utilizing the DT technology for the different network management issues as summarized in Table 7.1, to do the best of our knowledge, this is the first work to investigate the DT technology for the management of hard-to-follow AANET environment. In this chapter, we design a DT-enabled management framework, particularly for AANETs. We will detail our contributions in the upcoming subsection.

7.2.4 Contributions of the chapter

This chapter aims to introduce the use of DT technology for the AANET orchestration and with this aim, proposes a DT-enabled AANET management (DT-AANET) framework. In DT-AANET, we take advantage of the intelligent-DT for real-time and proactive AANET topology management as shown in Figure 7.1. Since the intelligent-DT supports the AI-based computational power, bidirectional connectivity, and real-time monitoring to ease the control of the hard-to-follow AANETs as explained in Sec. 7.2.2. The main contributions of chapter could be summarized as follows:

- We aim to provide the key benefits of utilizing DT for the management of AANETs and accordingly, propose a DT-enabled management framework to overcome the challenging characteristics of hard-to-follow AANETs.

- We offer to utilize our proposed AI-based management models of AANETs in the DT-AANET framework. Here, we utilize the unsupervised learning-based topology formation and supervised learning-based topology sustainability management as our AI-based AANET management use cases.
- We present a case study, customized K-means, on the utilization of DT-enabled model with AANET topology formation management use case to underline its full potential on this challenging environment. Also, the LVQ-based topology sustainability is taken as our second case study.

7.2.5 Organization of the chapter

The rest of the chapter is organized as follows: We first give the digital twin-enabled AANET management framework. Then, the proposed digital twin-enabled AANET management framework is illustrated with case studies. Finally, the performance of the case studies is evaluated and the chapter is concluded.

7.3 Digital Twin-Enabled Management for AANETs (DT-AANET)

In this chapter, our main aim is to virtually represent the physical AANET in the DT for the real-time and proactive management of this hard-to-follow environment. To enable this, our Digital Twin-Enabled AANET Management (DT-AANET) framework consists of two main parts: Physical AANET Twin and Controller. The Physical AANET Twin mainly includes the aircraft that communicate through A2A links, while the Controller consists of the Digital Twin-based AANET and Operational Module as detailed in the following subsections.

7.3.1 DT-AANET framework

7.3.1.1 Physical AANET twin

The physical AANET twin is the real environment consisting of three main parts as detailed follows:

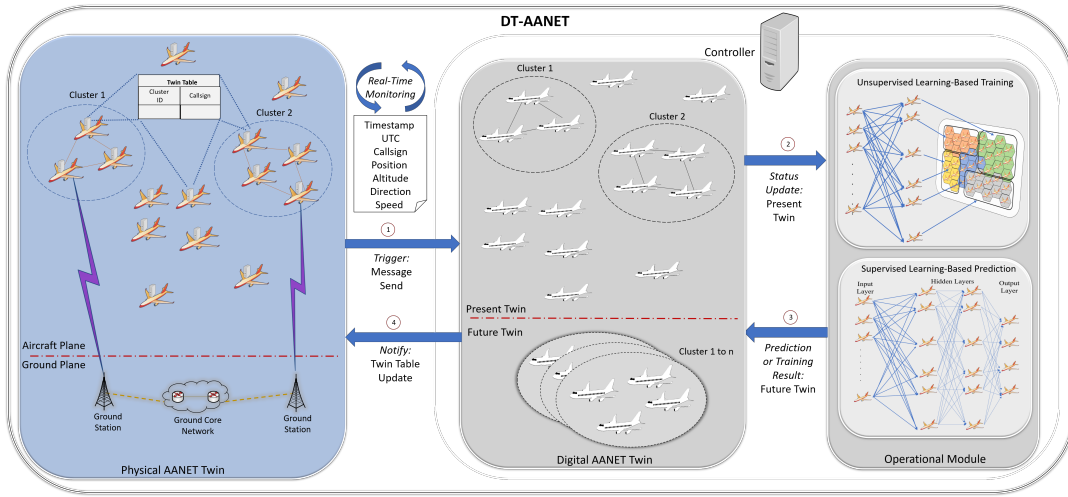


Figure 7.2 : DT-AANET topology management framework.

- **Ground Plane:** The ground plane supports the IFC through A2G connectivity, and for this aim, special ground stations are deployed on terrestrial areas. An aircraft can establish a connection with the ground station if it falls in the range of them. Otherwise, the aircraft should be included by an AANET cluster as detailed in the following item.
- **Aircraft Plane:** The AANET is the main component of the aircraft plane, and the main aim of it is to route the packets through the aircraft until reaching the destination that has A2G connectivity as shown in Figure 7.2.
- **Twin Tables:** We also embed twin tables to each aircraft to notify them about the future twin results. These twin tables could be considered as built-in tables consisting of two main fields: AANET Cluster ID and Callsign, as shown in Figure 7.2. The AANET Cluster-ID gives information about the cluster to which the aircraft will be connected, while Callsign specifies the intra-cluster connection detail between two aircraft.

As shown in Figure 7.2, these main components of Physical AANET Twin are replicated in the Digital AANET Twin in the form of present twin with the acceptance of trigger message. We will explain the details of these procedures in the upcoming part.

7.3.1.2 Controller

The Controller utilizes the cloudlet and edge servers to conduct the proactive and real-time management of Physical AANET Twin topology as shown in Figure 7.2. Therefore, we take advantage of the edge-cloud-based DT to consider both the latency and computational power requirements of AANETs during proactive and real-time topology management. For these purposes, the Controller consists of two main parts as detailed follows:

- **Digital AANET Twin:** It decouples the physical AANET twin to the digitized virtual format for the proactive and real-time management of this hard-to-follow environment. For this aim, the *real-time monitoring* is executed continuously by collecting specific aircraft data, and this continuous transfer of real-time data is one of the main characteristics of the DT concept different from the traditional simulations. This aircraft data consists of seven main fields: Timestamp, UTC, Callsign, Position, Altitude, Direction, and Speed as shown in Figure 7.2. The Digital AANET Twin establishes a present twin if it accepts a *Trigger* message from the Physical AANET Twin as shown with 1 in Figure 7.2. This *Trigger* could be in the form of a "create" or "sustain" message. Here, the "create" message activates the unsupervised learning-based topology formation while "sustain" establishes the supervised learning-based topology sustainability when the Digital AANET Twin transfers them to the Operational Module. This transfer is executed as *Status Update* including present twin and trigger message as shown with 2 in Figure 7.2. According to the *Formation and Sustainability Results* obtained from the operational module, the future twin is created in the form of clusters in the Digital AANET Twin as shown with 3 in Figure 7.2. The future twins could be considered as virtual AANET clusters with intra-cluster A2A links. Then, the Digital AANET Twin transfer these future twin details to the Physical AANET Twin as a *Notify* messages to fill the twin tables of aircraft as shown with 4 in Figure 7.2.
- **Operational Module:** The Operational Module could be considered as the brain of the Controller. It mainly utilizes the AI methodologies for proactive and real-time management of the Physical AANET Twin topology. According to

our previous chapters, the Operational Module consists of two main parts as unsupervised learning-based topology formation and supervised learning-based topology sustainability management. These parts work based on the requirements of the Physical AANET twin. More clearly, if the physical twin does not have AANET clusters and requires to establish them, it transfers a "create" message as shown in Figure 7.2. This message activates the unsupervised learning-based topology formation in the Operational Module. On the other hand, if the physical twin has AANET clusters with only some unlinked aircraft, it transfers a "sustain" message. In this case, the main aim is to enable the sustainability of the already clustered AANET topology as shown in Figure 7.2. Additionally, we use our two main AI-based AANET management models in here to keep the DT-AANET framework simple. We can also add the topology mapping and routing management modules to the DT-AANET framework. The details of unsupervised learning-based topology formation and supervised learning-based topology sustainability parts could be summarized as follows:

- Unsupervised Learning-Based Topology Formation: The main aim of this part is to create the topology of the AANET in the form of clusters. Since before creating an AANET, the aircraft were independently located with unstructured characteristics, and these could be considered unlabeled training data for unsupervised learning. Here, our main focus is to collect the aircraft that show similar features under one cluster. These features could be a position, direction, speed, altitude, or combination of them. We can select different unsupervised learning-based methodologies for this clustering. To underline this situation, in here, we aim to propose a customized K-means-based clustering as an unsupervised learning-based topology formation management model in operational module instead of three-phased formation as given in Chapter 3. We will give the details of this model in the upcoming section as a case study of operational module for the unsupervised learning-based topology formation.
- Supervised Learning-Based Topology Sustainability: The main aim of this part is to enable the sustainability of the clustered AANET topology. More

specifically, this part uses the already established clustered AANET topology instead of creating it continuously, as shown in Figure 7.2. At that point, we use the previously created AANET topology as training data and execute a supervised learning-based methodology to make predictions based on this training data. Clearly, we can achieve the assignments of new aircraft to the clusters already established with unsupervised learning thanks to the classification feature of supervised learning. Here, we aim to utilize the LVQ as given in Chapter 4. Therefore, we take it as a case study of operational module for the supervised learning-based topology sustainability.

The results of the unsupervised learning-based topology formation or supervised learning-based topology sustainability are given to the Digital AANET Twin as a future twin as illustrated in Figure 7.2. These future twins could be considered as virtual AANET clusters with intra-cluster A2A links, as explained above. Again, these future twins are transferred to the Physical AANET Twin from the Digital Twin as a *Notify* for twin table update.

7.3.2 Case studies for DT-AANET framework

7.3.2.1 Unsupervised learning-based topology formation management

In the first step, we propose a customized K-means-based clustering as a topology formation methodology. Here, we propose this clustering algorithm to show that we can select different unsupervised learning-based methodologies for AANET clustering. The proposed customized K-means-based clustering formation system consists of three main parts: flight data obtainment, data model creation, and clustering.

Flight data obtainment and data model creation

The flight data is obtained from the FlightRadar24.com as explained in Chapter 1. This data consists of seven fields: Timestamp, UTC, Callsign, Position, Altitude, Speed, and Direction as shown in Figure 7.2. The position is represented with the longitude and latitude, while the direction is given with the angle degree in the range (0-359). We utilize the North Atlantic airspace to obtain the real-time aircraft data [175]. We can

get the aircraft in the en-route phase on this space, and this space also gives us dense flight traffic.

After obtaining the data, we first drop the UTC and speed columns that are unnecessary for our operations. We also remove the data below 30000 feet since we focus on aircraft finishing their taking off and entering a more stable flight path. We limit the region by (66.5000, 40.000) and (-75.000, 00.000) latitudes by longitudes to remove the outside flight areas. We map the flights by their timeslot and sample by 15-minute intervals. Also, we group and label them by their Callsign to address every data by itself as shown in Figure 7.2. Additionally, we divide the location parameter into two columns as latitude and longitude. After the data collection, the Pandas framework is used to work on the data frames. Here, the dataframe() function creates the data frame of the airplane based on the dataset that is regulated according to our aims.

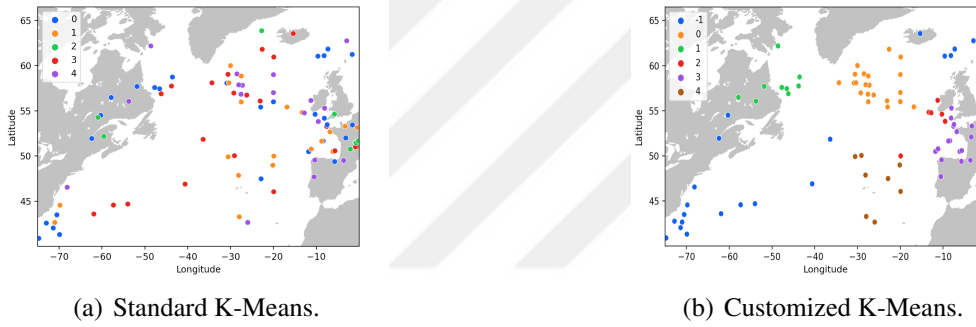


Figure 7.3 : Comparison of standard K-means and customized K-means.

Clustering

After obtaining the data, we can execute the machine learning part and we start to clustering processing with the standard K-means [176]. However, the standard K-means do not give efficient results in terms of aircraft clustering. More clearly, the aircrafts having higher distances are also included by the same cluster with the standard K-means algorithm as shown in Figure 7.3(a). For this reason, we propose a customized K-means algorithm according to our data and metrics. This algorithm works based on the following conditions. The aircraft iteratively check that they fit generated clusters. At the first iteration, we assign random centroids (number of k) that are the cluster heads. Then, the location of each centroid is shifted to the geographical mean of the cluster members at each iteration. After the determination of the cluster head, other aircraft within the communication range is defined as the

cluster candidates. We can achieve it by calculating the distance between aircraft and defined cluster head. If an aircraft is not in the communication range, it is not assigned to the corresponding cluster and labeled as -1. This means that this airplane is not in the communication range of the cluster. To calculate the communication range, we use the equation given in Figure 7.4(a). Here, the R corresponds to the Earth radius that is 3443.92 NMI, and the h is taken as 30000 feet (4.94 NMI) according to our limitations. Therefore, we calculate the communication range as 385 NMI in our implementations.

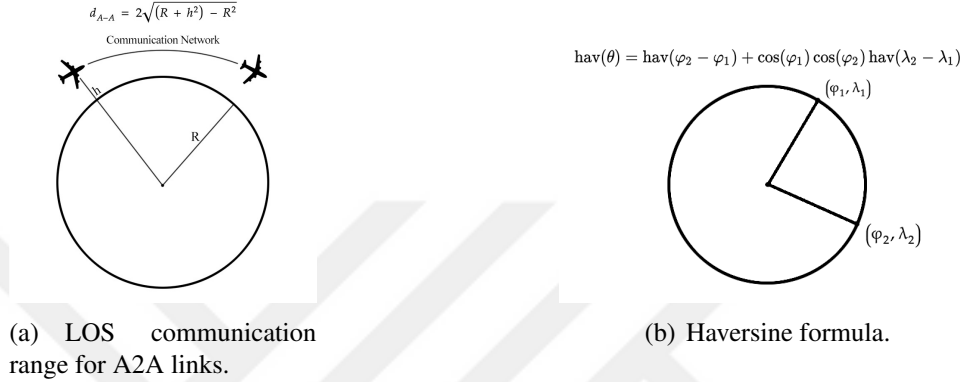


Figure 7.4 : Communication range-related calculations.

Also, to calculate the distance between aircraft, we use the Haversine formula instead of Euclidean distance. This is one of the main differences between the customized and standard K-means algorithms. The Haversine formula as shown in Figure 7.4(b) is used to calculate the angular distance between the latitude and longitude of two points on the surface of the sphere [177]. Therefore, we can calculate the communication between two nodes in terms of longitude and latitude instead of a kilometer. As shown in Figure 7.4(b), the first coordinate could be defined as a radius while the second is longitude.

After checking the communication range, we also control the direction of the candidate aircraft to check that their flight route is similar to the direction of the cluster head or not. To achieve this, we linearly compare the direction angles of cluster head and candidate aircraft. If the flight route of the cluster head and candidate aircraft are correspondents, then we define the candidate as a member of that cluster and label it with 1. With this methodology, we can obtain a better clustering performance than standard K-means as shown in Figure 7.3(b). Then, result of these clusters are given

to the Digital AANET Twin from the operational module as a future twin based on the "create" message.

7.3.2.2 Supervised learning-based topology sustainability management

After obtaining the clusters on AANETs with unsupervised learning, we aim to map the newly arrived and unlabeled aircraft to the labeled clusters to improve the clustering performance in the second step. Accordingly, we can assign the new aircraft to the most appropriate cluster with the LVQ as explained in Chapter 4. For the mapping with LVQ, we use the labels accepted from unsupervised learning. As explained in Chapter 4, the main aim of the LVQ-based topology sustainability is to enable clustered-AANET sustainability in three phases: winning cluster selection, intra-cluster link determination, and attribute update. In the winning cluster selection phase, we aim to find the best matching cluster of an unlinked aircraft through pattern classification by considering clustered-AANET topology as a pattern at the beginning. Then, the aircraft in a cluster is considered as a weight vector with location and delay attributes. These attributes are used to determine the intra-cluster links of clusters. Finally, the attributes of weight vectors are updated based on the established A2A intra-cluster links as shown in Figure 7.5.

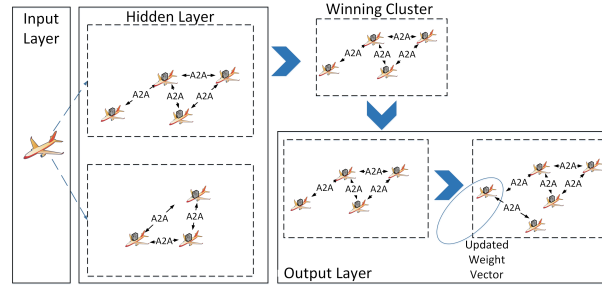


Figure 7.5 : Summary of LVQ-based topology sustainability.

Although the customized K-means-based clustering and LVQ-based topology sustainability can achieve end-to-end latency reduction with a higher transfer rate, we require some improvements due to the computational complexity issues. These requirements could be listed as follows:

label=() The customized K-means and LVQ should continuously adapt to the ultra-dynamic AANET environment with the parameter, aircraft, and data set updates.

lbbel=() The customized K-means and LVQ should also adapt to the larger AANET clusters in terms of scalability and convergence time without performance defects.

These two requirements reduce the efficiency of AI-based AANET management methodologies, and at that point, the DT fits perfectly as a solution to these problems, as explained in Section 7.2.2. Accordingly, we aim to form the AANET topology and enable the sustainability of it through our proposed DT-AANET framework as our case studies. Compared to previous chapters, we aim to observe a lower *end-to-end latency* with higher *packet transfer success* since the proposed methodology handles the above-explained ultra-dynamism requirement. Additionally, we aim to reduce the *convergence time* of our algorithm by satisfying the *scalability* needs.

To execute the above-given aims, we utilize the DT-AANET framework with the customized K-means and LVQ for the formation and sustainability of the AANET topology. Accordingly, we first create the topology of the AANETs in the form of clusters. Then, we utilize the supervised learning part in the Operational Module for the topology sustainability aim. Here, the aircraft in Physical AANET Twin starts by checking the existence of a ground station in its connectivity range. As explained above, we create the AANET topology in the form of clusters and, this aircraft may be newly joined to physical space or disconnected from a cluster. Accordingly, if there is not any ground station deployment in its range, the aircraft wants to connect an AANET and *triggers* the Digital AANET Twin with a "sustain" message as shown in Figure 7.2. After this message, the Digital AANET Twin first creates a present twin that is the virtual replicate of the currently deployed physical topology and then transfers this message to the Operational Module with the real-time data. As explained in Section 7.2.2, this real-time data consists of seven main fields, and we should crop the unnecessary columns from this set. We should also filter this data with altitude parameters to obtain more stable flight values belonging to the en-route

phase. Additionally, by filtering longitude and latitude values as shown in Figure 7.2, we restrict the flight path to the North Atlantic corridor to obtain dense air traffic. This preprocessed real-time aircraft data and "sustain" message transfer is executed as *Status Update* with present twin. An important point here is that this data preprocessing is performed when there is no cluster in AANET, an aircraft breaks its link, or newly joins a physical topology.

The supervised learning-based topology sustainability works based on the LVQ when *Status Update* is received by the Operational Module. The LVQ works using the conventional algorithm as shown in Figure 7.5. Then, the results are transferred to the Digital AANET Twin with *Prediction Result* as a future twin. Finally, according to the clusters and intra-cluster A2A links in this future twin, the Physical AANET Twin is *notified* through twin tables. Therefore, in this DT-AANET workflow, (i) the Digital AANET Twin is automatically triggered by the physical topology with the parameter, aircraft, and data set updates. Also, (ii) the aircraft density and size of the AANET clusters do not significantly affect the formation or sustainability performance.

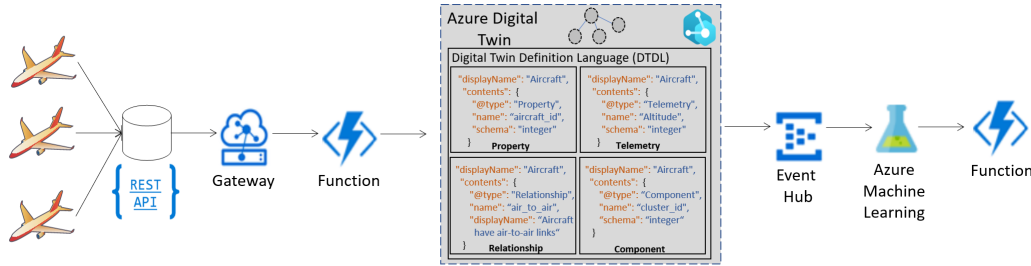


Figure 7.6 : Simulation architecture based Microsoft-ADT.

7.4 Performance Evaluation

7.4.1 Simulation environment

To evaluate the DT-AANET framework, we utilize the Microsoft Azure Digital Twin (ADT) [178]. The Microsoft-ADT is a platform as a service tool that provides DT graphs of physical topology by using the Digital Twins Definition Language (DTDL) based on JSON-LD (stands for JavaScript Object Notation for Linked Data). Accordingly, in the DT-AANET framework, the aircraft becomes the physical input

data source, as explained in the above sections. The collected real-time aircraft data is transferred with the Representational State Transfer application programming interfaces and cloud gateway to the ADT as shown in Figure 7.6. Here, the function operation triggers the ADT according to the real-time data.

In ADT, the Physical AANET Twin is replicated via the DTDL, as explained above. In DTDL, the entities in the physical twin are represented as models, and in our DT-AANET model, each aircraft becomes a model. Additionally, in the model definition, the interface becomes a top-level code item with four main fields: Property, Telemetry, Relationship, and Component. Here, the property shows the entity's state while measurements or events are represented with telemetry. Also, the relationship specifies the representation of our DT among others, and the components enable us to build our model interface as an assembly of others. A few examples belong to these main field definitions of our aircraft interface are summarized in Figure 7.6.

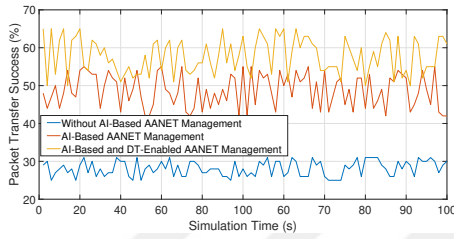
The data is transmitted to the event hub from the ADT for further processing. We also utilize the Azure machine learning during these further processes since we use unsupervised and supervised learning-based methodologies according to our DT-AANET model as shown in Figure 7.6. Finally, the function triggers the applying of obtained results and the upcoming operations.

7.4.2 Simulation results

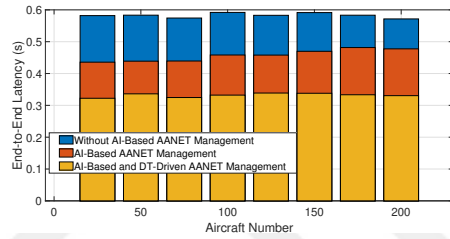
In these simulations, we assume that we first form the AANET topology according to our topology formation case study. Then, we utilize the sustainability case for the newly arrived and broken airplanes. Here, we consider three main steps for simulations. In the first step, the topology formation and sustainability of AANET is conducted without utilizing any AI-based management model. In the second step, we execute these steps by utilizing AI-based models according to the previous chapters of this thesis and as explained in case studies. Finally, in the last step, we conduct these AI-based management models through the DT-AANET framework as shown in Figure 7.7.

Based on these steps, we evaluate the performance in terms of packet transfer success rate, end-to-end latency, and convergence time parameters as summarized follow:

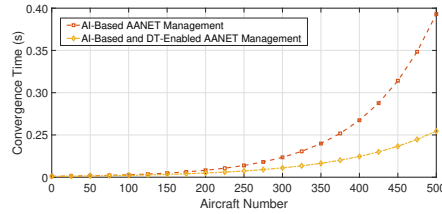
- *Packet Transfer Success Rate*: shows the ratio of packets that are successfully accepted by the destination aircraft.
- *End-to-End Latency*: represents the time it takes for the transferred packet from the source to the destination.
- *Convergence Time*: is an indication of how long it takes the AI-based model to converge. Accordingly, this parameter enables us to compare the convergence time of the DT-AANET-based and only AI-based models according to the increasing aircraft number.



(a) End-to-end delay vs. Aircraft Number.



(b) Packet Transfer Success vs. Simulation Time.



(c) Convergence Time vs. Aircraft Number.

Figure 7.7 : Simulation environment and results.

As shown in Figure 7.7(a), we first evaluate the end-to-end latency according to the increasing aircraft number. Here, we run each methodology 100 times for each aircraft number in the x-axis and calculate latency for each iteration. Then, we take the average of calculated latencies to find a final value for each specific aircraft number in the x-axis. Generally, in three methodologies, the end-to-end latency tends to increase in small amounts with rising aircraft number as shown in Figure 7.7(a). Here, the AANETs without an AI-based management mechanism observe the end-to-end

latency greater than 0.5s. The main reason for this performance defect is that it continuously creates the AANET topology when a new aircraft arrives instead of using the previously established ones. The AI-based AANET management utilizes the previously established topology as a pattern and executes pattern recognition to assign the newly arrived aircraft. However, the adaptation requirement of the AI-based models to the ultra-dynamic AANET environment in terms of parameter, aircraft, and data set updates increases its latency compared to the DT-AANET framework as shown in Figure 7.7(a). The DT-AANET framework observes the latency lower than 0.3s by eliminating this need for adaptation. Similar comments are also valid for the packet transfer success evaluation. As shown in Figure 7.7(b), the AANETs without an AI-based management experience the packet transfer success lower than 30%. Also, the DT-AANET framework achieves the packet transfer success roughly 70% while the AI-based AANET management observes it at the level of 50% as shown in Figure 7.7(b). Here, the management is executed in a real-time and proactive manner in DT-AANET compared to the AI-based management model. Finally, we evaluate the convergence time of the AI-based management model as shown in Figure 7.7(c). Here, the AI-based management model experiences performance defects with increasing aircraft numbers due to its scalability problem. However, the aircraft density and size of the AANET clusters do not significantly affect the sustainability performance of the DT-AANET framework. This situation reduces the converge time of the DT-AANET-based management, as shown in Figure 7.7(c).

7.5 Conclusion

This chapter proposes a DT-AANET to carry the AI-based AANET management models to the DT technology. With this framework, we can virtually replicate the physical components of AANET topology to a digital space. Here, the monitoring, bidirectional connectivity, and AI-based computational power features of intelligent DT enable real-time and proactive AANET management to our proposed framework. In the proposed DT-AANET framework, the Digital AANET Twin replicates the physical topology to Controller. At the same time, the Operational Module supports the AI-based computational power through unsupervised learning-based topology

formation and supervised learning-based topology sustainability according to our previous chapters. Finally, we illustrate the customized K-means-based topology formation and LVQ-based AANET topology sustainability as case studies to show the usability of the DT-AANET and support this through evaluation results.

General concept for the next chapter

Up to this point, we proposed an AI-based complete management model for the topology formation, sustainability, mapping, and routing issues in AANETs. With the end of this chapter, we carry this AI-based complete management model consisting of four main parts to DT environment. More clearly, the AI-based complete management model is conducted through a Digital Twin-Enabled AANET Topology Management Framework, DT-AANET. In the upcoming chapter, we will conclude our thesis by summarizing each part of the proposed AI-based and DT-enabled AANET management model.

8. CONCLUSIONS

In next-generation networks, one of the essential needs is to stay connected independent of the location, time, and situation. Airplanes are one of the top places that have problems in meeting this need. On the other hand, the IFC turns into a crucial need instead of a luxury with technological advances. The passengers want to access the high-quality Internet service during flight as they received from the terrestrial network. To satisfy this connectivity requirement, the satellite and A2G networks are commonly utilized by airplanes. However, the high installation/equipment cost of satellites and coverage problems of A2G networks reduce the efficiency of these solutions. At that point, the AANETs become a novel solution to enable IFC for airplanes and solve the main problems of satellites and A2G networks.

On the other hand, the AANETs have different characteristics compared to the terrestrial networks. These characteristics are mainly caused by the mobility and atmospheric effects on AANETs. The ultra-high velocities of airplanes reduce their link qualities and increase the A2A link breakages. More clearly, these ultra-dynamic link characteristics of high-density airplanes create an unstructured and unstable topology in three-dimensional space for AANETs. These specific characteristics increase the management complexity of AANETs. Accordingly, they reduce the applicability of conventional topology and routing management algorithms to AANETs.

At that point, to solve the management complexity challenge, we propose the utilization of AI-based methodologies on AANETs. More clearly, we manage the topology formation, sustainability, mapping, and routing operations on AANET through AI-based methodologies. Also, to solve the computational complexity problems of these methodologies, we utilize the DT technology. Therefore, we first propose AI-based management methodologies for topology formation, sustainability,

mapping, and routing issues in AANET. Then, we combine them under the DT technology to solve their computational complexity challenges.

Therefore, in this thesis, we propose an AI-based and DT-enabled management for AANETs. In the first step, this management system creates the AANET topology in the form of clusters through the topology formation management. Then, the sustainability of the formed topology is enabled with the topology sustainability management. Also, the formed and sustained AANET topology is continuously mapped to the airplanes with the topology mapping management to notify them about topological changes. Finally, airplanes can transfer their packets on this formed, sustained, and continuously mapped AANET topology through routing management. After executing these procedures through the AI-based models, we combine them under the proposed DT-enabled AANET (DT-AANET) management model. According to our simulation results, we can achieve roughly 48% higher packet delivery success with 35% lower end-to-end latency with the AI-based and DT-enabled AANET management system.

This thesis is the first work to create an AI-based and DT-enabled management system for AANETs and we believe that it opens several interesting avenues for future research. The management system in this thesis specifically focuses on the network layer issues on AANET. In addition to these network-level management issues, as explained in Chapter 1, the AANETs can experience different problems in terms of the data link and MAC layers due to their specific characteristics. In future work, we are planning to focus the management issues on the data link and MAC layers of AANETs. Accordingly, we are working to propose AI-based and DT-enabled management models for the issues in these layers.

REFERENCES

- [1] **IATA** (2018). Passenger numbers to hit 8.2bn by 2037 (Technical Report).
- [2] **Gupta, N. and Aggarwal, A.** (2017). Airborne Internet-The Internet in the air, *7th International Conference on Cloud Computing, Data Science Engineering - Confluence*, pp.441–444.
- [3] **Dinc, E., Vondra, M., Hofmann, S., Schupke, D., Prytz, M., ... and Cavdar, C.** (2017). In-Flight Broadband Connectivity: Architectures and Business Models for High Capacity Air-to-Ground Communications, *IEEE Communications Magazine*, 55(9), 142–149.
- [4] **Kabashkin, I.** (2016). Resilient communication network of Air Traffic Management system, *Advances in Wireless and Optical Communications (RTUWO)*, pp.156–160.
- [5] **Deloitte** (2018). Fasten your seatbelts:in-flight connectivity takes off (Technical Report).
- [6] **Honeywell** (2016). Survey: Airlines risk losing passengers due to poor wi-fi (Technical Report).
- [7] **EUROCONTROL** (2013). Challenges of growth, task 4:european air traffic in 2035 (Technical Report).
- [8] **Euroconsult** (2016). Passenger connectivity services to surpass \$5 billion by 2025 (Technical Report).
- [9] **ATAG** (2011). Air Transport Action Group, The economic social benefits of air transport (Technical Report).
- [10] **SES** (2016). Sky high connectivity (Technical Report) (Technical Report).
- [11] **Inmarsat** (2017). Sky high economics (Technical Report) (Technical Report).
- [12] **Leipold, F., Tassetto, D. and Bovelli, S.** (2013). Wireless in-cabin communication for aircraft infrastructure, *Telecommunication Systems*, 52(2), 1211–1232.
- [13] **Jun-lin, W. and Chun-sheng, L.** (2011). Development and Application of INMARSAT Satellite Communication System, *First International Conference on Instrumentation, Measurement, Computer, Communication and Control*, pp.619–621.

- [14] **Beljour, H., Hoffmann, R., Michael, G., Shields, J., ... and Willson, A.** (2009). Concept for an all-digital satellite communications earth terminal, *MILCOM*, pp.1–5.
- [15] **Hoffmann, F., Medina, D. and Wolisz, A.** (2010). Protocol architecture analysis for Internet connectivity in aeronautical ad hoc networks, *29th Digital Avionics Systems Conference*, pp.3.C.4–1–3.C.4–12.
- [16] **Shi, C., Ren, Q., Zheng, B. and Liu, Y.** (2008). Analysis of Node Density and Probability of Forming a Network in Military Aeronautical Ad Hoc Networks, *International Conference on Microwave and Millimeter Wave Technology*, volume 4, pp.1980–1982.
- [17] **Hoffmann, F., Medina, D. and Wolisz, A.** (2009). Two-step delay based Internet gateway selection scheme for aeronautical ad hoc networks, *IEEE 20th International Symposium on Personal, Indoor and Mobile Radio Communications*, pp.2638–2642.
- [18] **Li, J., Gong, E., Sun, Z., Li, L. and Xie, H.** (2014). Fault-tolerant topology control in aeronautical ad hoc networks, *IEEE International Conference on Mechatronics and Automation*, pp.368–372.
- [19] **Ben Mahmoud, M.S., Guerber, C., Larrieu, N., Pirovano, A. and Radzik, J.** (2014). *Aeronautical Air–Ground Data Link Communications*.
- [20] **Rana, A. and Kumar, V.** (2019). Aircraft Ad-hoc Network (AANET), *International Journal of Innovative Research in Computer and Communication Engineering*, 3, 6679–6684.
- [21] **Sakhaee, E., Jamalipour, A. and Kato, N.** (2006). Aeronautical ad hoc networks, *IEEE Wireless Communications and Networking Conference*, volume 1, pp.246–251.
- [22] **Vondra, M., Dinc, E., Prytz, M., Frodigh, M., Schupke, D., Hofmann, S. and Cavdar, C.** (2017). Performance Study on Seamless DA2GC for Aircraft Passengers toward 5G, *IEEE Communications Magazine*, PP.
- [23] **Liu, Y., Yang, Y., Yang, L.L. and Hanzo, L.** (2021). Physical Layer Security of Spatially Modulated Sparse-Code Multiple Access in Aeronautical Networking, *IEEE Transactions on Vehicular Technology*, 70(3), 2436–2447.
- [24] **ETSI** (2017). Study on channel model for frequency spectrum above 6 GHz (Technical Report).
- [25] **Kesavan, U., Islam, M.R., Abdullah, K. and Tharek, A.R.** (2014). Rain Attenuation Prediction for Higher Frequencies in Microwave Communication Using Frequency Scaling Technique, *International Conference on Computer and Communication Engineering*, pp.217–219.

- [26] **ITU-R** (2017). Propagation data and prediction methods required for the design of terrestrial line-of-sight systems (Technical Report).
- [27] **ITU-R** (2012). Recommendation ITU-R P.840-5, Attenuation due to clouds and fog (Technical Report).
- [28] **ALAWADI, T.A.** (2012). *Investigation of the Effects of Cloud Attenuation on Satellite Communication Systems*, Cranfield University, School of Engineering, Department of Engineering Physics.
- [29] **Alam, S., Abbass, H. and Barlow, M.** (2008). ATOMS: Air Traffic Operations and Management Simulator, *IEEE Transactions on Intelligent Transportation Systems*, 9, 209–225.
- [30] **Zhang, J., Xiang, L., Liu, D., Cui, J., Ng, S.X., ... and Hanzo, L.** (2021). Semi-Stochastic Aircraft Mobility Modelling for Aeronautical Networks: An Australian Case-Study Based on Real Flight Data, *IEEE Transactions on Vehicular Technology*, 70(10), 10763–10779.
- [31] **Yan, J., Hua, C., Chen, C. and Guan, X.** (2014). The Capacity of Aeronautical Ad-Hoc Networks, *Wirel. Netw.*, 20(7), 2123–2130.
- [32] **Li, J., Lei, L., Liu, W., Shen, Y. and Zhu, G.** (2012). An Improved Semi-Markov Smooth Mobility Model for Aeronautical Ad Hoc Networks, *8th International Conference on Wireless Communications, Networking and Mobile Computing*, pp.1–4.
- [33] **Zhong, D., Zhu, Y., You, T. and Kong, J.** (2014). Topology Control Mechanism Based on Link Available Probability in Aeronautical Ad Hoc Network, *Journal of Networks*, 9.
- [34] **Sakhaee, E. and Jamalipour, A.** (2006). The Global In-Flight Internet, *IEEE Journal on Selected Areas in Communications*, 24, 1748–1757.
- [35] **Sakhaee, E. and Jamalipour, A.** (2005). Aerouter™ - A graphical simulation tool for routing in aeronautical systems, volume 4, pp.2506–2511.
- [36] **Ghosh, S. and Nayak, A.** (2016). ACPM: An associative connectivity prediction model for AANET, *Eighth International Conference on Ubiquitous and Future Networks (ICUFN)*, pp.605–610.
- [37] **Nguyen Thi Xuan My, Yoshikazu Miyanaga and Chaiyachet Saivichit** (2010). Connectivity analytical modelling for a single flight path ad hoc aeronautical network, *International Confernce on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology*, pp.51–55.
- [38] **Zhang, H., Chen, X., Zheng, B. and Wang, Y.** (2011). Analysis of connectivity requirement for aeronautical Ad hoc networks, *Proceedings of International Conference on Electronic Mechanical Engineering and Information Technology*, volume 8, pp.3943–3946.

- [39] **Li, H., Yang, B., Chen, C. and Guan, X.** (2010). Connectivity of Aeronautical Ad hoc Networks, *IEEE Globecom Workshops*, pp.1788–1792.
- [40] **Yan, J., Song, G., Li, H., Hua, C., Chen, C. and Guan, X.** (2012). Critical transmission range for connectivity in aeronautical ad-hoc networks, *Proceedings of the 10th World Congress on Intelligent Control and Automation*, pp.4446–4451.
- [41] **Wang, Y., Ertürk, M.C., Liu, J., Ra, I., Sankar, R. and Morgera, S.** (2015). Throughput and delay of single-hop and two-hop aeronautical communication networks, *Journal of Communications and Networks*, 17(1), 58–66.
- [42] **Medina, D., Hoffmann, F., Rossetto, F. and Rokitansky, C.** (2010). Routing in the Airborne Internet, *Integrated Communications, Navigation, and Surveillance Conference Proceedings*, pp.A7–1–A7–10.
- [43] **Hoffmann, F., Medina, D. and Wolisz, A.** (2011). Optimization of routing and gateway allocation in aeronautical ad hoc networks using genetic algorithms, *7th International Wireless Communications and Mobile Computing Conference*, pp.1391–1396.
- [44] **Hoffmann, F., Medina, D. and Wolisz, A.** (2013). Joint Routing and Scheduling in Mobile Aeronautical Ad Hoc Networks, *IEEE Transactions on Vehicular Technology*, 62(6), 2700–2712.
- [45] **Tang, J., Wang, Y., Dong, S., Sun, Q. and Huang, G.** (2018). A feedback-retransmission based asynchronous frequency hopping MAC protocol for military aeronautical ad hoc networks, *Chinese Journal of Aeronautics*, 31(5), 1130–1140.
- [46] **Kato, N., Fadlullah, Z.M., Tang, F., Mao, B., Tani, S., Okamura, A. and Liu, J.** (2019). Optimizing Space-Air-Ground Integrated Networks by Artificial Intelligence, *IEEE Wireless Communications*, 26(4), 140–147.
- [47] **Tu, H.D. and Shimamoto, S.** (2009). A Proposal of Relaying Data in Aeronautical Communication for Oceanic Flight Routes Employing Mobile Ad-Hoc Network, *First Asian Conference on Intelligent Information and Database Systems*, pp.436–441.
- [48] **Gronkvist, J., Karlsson, J., Sterner, U., Nilsson, J. and Hansson, A.** (2012). Adaptation delay and its impact on application performance for TDMA ad hoc networks, *The 11th Annual Mediterranean Ad Hoc Networking Workshop (Med-Hoc-Net)*, pp.55–60.
- [49] **Li, J., Gong, E., Sun, Z., Li, L. and Xie, H.** (2013). An Interference-Based Distributed TDMA Scheduling Algorithm for Aeronautical Ad Hoc Networks, *International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery*, pp.453–460.

- [50] **Qiu, Q., Fang, Z. and Gong, C.** (2015). Study on Key Techniques of Aeronautical Ad Hoc Network MAC and Network Layer, *Procedia Engineering*, 99, 280–291, asia-Pacific International Symposium on Aerospace Technology.
- [51] **Besse, F., Pirovano, A., Garcia, F. and Radzik, J.** (2011). Interference Estimation in an Aeronautical Ad Hoc Network, *AIAA/IEEE Digital Avionics Systems Conference - Proceedings*.
- [52] **Besse, F., Garcia, F., Pirovano, A. and Radzik, J.** (2010). Wireless Ad Hoc Networks Access For Aeronautical Communications.
- [53] **Indra** (2013). Antares communication standard design definition file, IRIS Technical Note (Technical Report).
- [54] **Vey, Q., Pirovano, A. and Radzik, J.** (2016). Performance Analysis of Routing Algorithms in AANET with Realistic Access Layer, *Communication Technologies for Vehicles*, Springer International Publishing, pp.175–186.
- [55] **Sun, Y., Belding, E. and Perkins, C.** (2002). Internet Connectivity for Ad Hoc Mobile Networks, *International Journal of Wireless Information Networks*, 2.
- [56] **Huang, C.F., Lee, H.W. and Tseng, Y.C.** (2003). A two-tier heterogeneous mobile Ad Hoc network architecture and its load-balance routing problem, volume 9, pp.2163–2167.
- [57] **Buchter, K.** (2017). Availability of aeronautical ad-hoc network in different global air transport fleet scenarios, *General Assembly and Scientific Symposium of the International Union of Radio Science*, pp.1–4.
- [58] **Brannstrom, R., Ahlund, C. and Zaslavsky, A.** (2005). Maintaining Gateway Connectivity in Multi-hop Ad hoc Networks, *The IEEE Conference on Local Computer Networks 30th Anniversary*, pp.682–689.
- [59] **Flightradar24**. Live flight tracker - real-time flight tracker map, Accessed: 27 January 2022, <https://www.flightradar24.com>.
- [60] **Sakhaee, E. and Jamalipour, A.** (2008). Stable Clustering and Communications in Pseudolinear Highly Mobile Ad Hoc Networks, *IEEE Transactions on Vehicular Technology*, 57(6), 3769–3777.
- [61] **Royer, M., Garcia, F. and Pirovano, A.** (2015). An enhanced 1-hop clustering algorithm for Publish / Subscribe systems in AANETS, *IEEE/AIAA 34th Digital Avionics Systems Conference (DASC)*, pp.2D2–1–2D2–6.
- [62] **Shi, F., Shi, Y. and Lai, L.** (2011). A clustering algorithm of Ad-Hoc network based on honeycomb division, *IEEE International Conference on Granular Computing*, pp.863–866.
- [63] **Fan, X., Cai, W. and Lin, J.** (2017). A survey of routing protocols for highly dynamic mobile ad hoc networks, *IEEE 17th International Conference on Communication Technology (ICCT)*, pp.1412–1417.

- [64] **Vey, Q., Pirovano, A., Radzik, J. and Garcia, F.** (2014). Aeronautical Ad Hoc Network for Civil Aviation.
- [65] **Wang, Y., Ertürk, M.C., Arslan, H., Sankar, R., Ra, I. and Morgera, S.** (2012). Throughput and delay analysis in Aeronautical Data Networks, *International Conference on Computing, Networking and Communications (ICNC)*, pp.771–775.
- [66] **Jiang, J., Zhang, P., Zhang, R., Chen, S. and Hanzo, L.** (2016). Aperture Selection for ACO-OFDM in Free-Space Optical Turbulence Channel, *IEEE Transactions on Vehicular Technology*, 65(8), 6089–6100.
- [67] **Medina, D., Hoffmann, F., Rossetto, F. and Rokitansky, C.** (2011). North Atlantic Inflight Internet Connectivity via Airborne Mesh Networking, *IEEE Vehicular Technology Conference*, pp.1–5.
- [68] **Zhong, D., Wang, Y., Zhu, Y. and You, T.** (2016). An Aeronautical Ad Hoc Network Routing Protocol Based on Air Vehicles Movement Features, *22nd International Conference on Applied Electromagnetics and Communications*, pp.1–6.
- [69] **Durresi, A., Paruchuri, V., Barolli, L. and Jain, R.** (2006). Air to air communication protocol, *IEEE Aerospace Conference*, pp.8 pp.–.
- [70] **Zhong Dong, Zhu Yian, You Tao, Duan Junhua and Kong Jie** (2014). A new data transmission mechanism in aeronautical ad hoc network, *International Conference on Big Data and Smart Computing (BIGCOMP)*, pp.255–260.
- [71] **Iordanakis, M., Yannis, D., Karras, K., Bogdos, G., Dilintas, G., ... and Baiotti, S.** (2006). Ad-hoc Routing Protocol for Aeronautical Mobile Ad-Hoc Networks.
- [72] **Peters, K., Jabbar, A., Cetinkaya, E.K. and Sterbenz, J.P.G.** (2011). A geographical routing protocol for highly-dynamic aeronautical networks, *IEEE Wireless Communications and Networking Conference*, pp.492–497.
- [73] **Swidan, A., Khattab, S., Abouelseoud, Y. and Elkamchouchi, H.** (2015). A secure geographical routing protocol for highly-dynamic aeronautical networks, *IEEE Military Communications Conference*, pp.708–713.
- [74] **Hoang, T.M., van Chien, T., van Luong, T., Chatzinotas, S., Ottersten, B. and Hanzo, L.** (2022). Detection of Spoofing Attacks in Aeronautical Ad-Hoc Networks Using Deep Autoencoders, *IEEE Transactions on Information Forensics and Security*, 17, 1010–1023.
- [75] **Tchakountio, F. and Ramanathan, R.** (2004). Anticipatory routing for highly mobile endpoints, *IEEE Workshop on Mobile Computing Systems and Applications*, pp.94–100.

- [76] **Tchakountio, F. and Ramanathan, R.** (2001). Tracking Highly Mobile Endpoints, *Proceedings of the 4th ACM International Workshop on Wireless Mobile Multimedia*, p.83–94.
- [77] **Mauve, M., Widmer, J. and Hartenstein, H.** (2001). A Survey on Position-Based Routing in Mobile Ad Hoc Networks, *IEEE Network*, 15(6), 30–39.
- [78] **Shirani, R., St-Hilaire, M., Kunz, T., Zhou, Y., Li, J. and Lamont, L.** (2011). The Performance of Greedy Geographic Forwarding in Unmanned Aeronautical Ad-Hoc Networks, *Ninth Annual Communication Networks and Services Research Conference*, pp.161–166.
- [79] **Shirani, R., St-Hilaire, M., Kunz, T., Zhou, Y., Li, J. and Lamont, L.** (2012). Combined Reactive-Geographic routing for Unmanned Aeronautical Ad-hoc Networks, *8th International Wireless Communications and Mobile Computing Conference*, pp.820–826.
- [80] **Vey, Q., Puechmorel, S., Pirovano, A. and Radzik, J.** (2016). Routing in aeronautical ad-hoc networks, *IEEE/AIAA 35th Digital Avionics Systems Conference (DASC)*, pp.1–10.
- [81] **Dong-Woo Seo, Sok-Hyong Kim and Young-Joo Suh** (2008). System integration of GPSR and ADS-B for aeronautical ad hoc networks, *IEEE Military Communications Conference*, pp.1–6.
- [82] **Mahmoud, M.S.B. and Larrieu, N.** (2013). An ADS-B Based Secure Geographical Routing Protocol for Aeronautical Ad Hoc Networks, *IEEE 37th Annual Computer Software and Applications Conference Workshops*, pp.556–562.
- [83] **Zhou, Q., Gu, W., Li, J., Sun, Q. and Yang, F.** (2012). A Topology Aware Routing Protocol Based ADS-B System for Aeronautical Ad Hoc Networks, *International Conference on Wireless Communications, Networking and Mobile Computing*, pp.1–4.
- [84] **Gu, W., Li, J., Lv, M., Sun, Q. and Yang, F.** (2011). Delay Aware Multipath Doppler Routing in Aeronautical Ad hoc Networks, *IEEE International Conference on Computational Science and Engineering*, pp.251–255.
- [85] **Jinhua Zhou, Lei Lei, Weikang Liu and Jiamin Tian** (2012). A simulation analysis of nodes mobility and traffic load aware routing strategy in aeronautical ad hoc networks, *Proceedings of 9th International Bhurban Conference on Applied Sciences Technology*, pp.423–426.
- [86] **Luo, Q. and Wang, J.** (2017). Multiple QoS Parameters-Based Routing for Civil Aeronautical Ad Hoc Networks, *IEEE Internet of Things Journal*, 4(3), 804–814

- [87] **Sakhaee, E., Jamalipour, A. and Kato, N.** (2006). Multipath Doppler Routing with QoS Support in Pseudo-linear Highly Mobile Ad Hoc Networks, *IEEE International Conference on Communications*, volume 8, pp.3566–3571.
- [88] **SeUng Hyeon, Ki-II Kim and SangWoo Yang** (2010). A new geographic routing protocol for aircraft ad hoc networks, *29th Digital Avionics Systems Conference*.
- [89] **Nguyen Thi Xuan My, Miyanaga, Y. and Saivichit, C.** (2011). Link longevity-based routing mechanisms for aviation ad hoc network, *IEEE 16th International Workshop on Computer Aided Modeling and Design of Communication Links and Networks (CAMAD)*, pp.46–50.
- [90] **Li, Y., St-Hilaire, M. and Kunz, T.** (2012). Enhancements to Reduce the Overhead of the Reactive-Greedy-Reactive Routing Protocol for Unmanned Aeronautical Ad-Hoc Networks, *8th International Conference on Wireless Communications, Networking and Mobile Computing*, pp.1–4.
- [91] **Kardoust, M., Khayyambashi, M.R. and Bohlooli, A.** (2017). Introducing a method for improving the performance of routing algorithms in unmanned aeronautical ad-hoc networks, *9th International Conference on Information and Knowledge Technology (IKT)*, pp.85–92.
- [92] **Bilen, T., Canberk, B. and Chowdhury, K.R.** (2017). Handover Management in Software-Defined Ultra-Dense 5G Networks, *IEEE Network*, 31(4), 49–55.
- [93] **Che, H., Chan, P.M.L. and Hu, Y.F.** (2005). Design and Analysis of QoS-Enabled Handover Algorithm for Aeronautical Communication Systems, *2nd International Symposium on Wireless Communication Systems*, pp.724–728.
- [94] **Zhang, J., Chen, T., Zhong, S., Wang, J., ..., Maunder, R.G. and Hanzo, L.** (2019). Aeronautical Ad Hoc Networking for the Internet Above the Clouds, *Proceedings of the IEEE*, 107(5), 868–911.
- [95] **Ayaz, S., Hoffmann, F., Sommer, C., German, R. and Dressler, F.** (2010). Performance Evaluation of Network Mobility Handover over Future Aeronautical Data Link, *IEEE Global Telecommunications Conference*, pp.1–6.
- [96] **Wang, D., Wang, Y., Dong, S., Huang, G., Liu, J. and Gao, W.** (2019). Exploiting Dual Connectivity for Handover Management in Heterogeneous Aeronautical Network, *IEEE Access*, 7, 62938–62949.
- [97] **Muhammad, M. and Berioli, M.** (2011). Transport Protocol for Future Aeronautics.

- [98] **Fang, J. and Liu, M.** (2011). Design and Implementation of Embedded RUDP, *Second International Conference on Networking and Distributed Computing*, pp.7–9.
- [99] **Besse, F., Pirovano, A., Garcia, F. and Radzik, J.** (2010). Aeronautical ad hoc networks : a new datalink for ATM.
- [100] **Qiu, Q., Fang, Z. and Gong, C.** (2015). Study on Key Techniques of Aeronautical Ad Hoc Network MAC and Network Layer, *Procedia Engineering*, 99.
- [101] **Luo, Q. and Wang, J.** (2018). FRUDP: A Reliable Data Transport Protocol for Aeronautical Ad Hoc Networks, *IEEE Journal on Selected Areas in Communications*, 36(2), 257–267.
- [102] **Pathapati, K., Nguyen, T., Rohrer, J. and Sterbenz, J.** (2011). Performance Analysis of the AeroTP Transport Protocol for Highly-Dynamic Airborne Telemetry Networks.
- [103] **Alenazi, M., Gogi, S., Zhang, D., Cetinkaya, E., Rohrer, J. and Sterbenz, J.** (2011). ANTP Protocol Suite Software Implementation Architecture in Python.
- [104] **Luo, Q., Wang, J. and Liu, S.** (2019). AeroMRP: A Multipath Reliable Transport Protocol for Aeronautical Ad Hoc Networks, *IEEE Internet of Things Journal*, 6(2), 3399–3410.
- [105] **Li, J., Gong, E., Sun, Z., Liu, W. and Xie, H.** (2015). AeroMTP: A fountain code-based multipath transport protocol for airborne networks, *Chinese Journal of Aeronautics*, 28(4), 1147–1162.
- [106] **LIU, L.** (2017). *Performance evaluation of direct air-to-ground communication using new radio (5G)*, KTH Electrical Engineering.
- [107] **Liu, D., Zhang, J., Cui, J., Ng, S.X., Maunder, R.G. and Hanzo, L.** (2022). Deep Learning Aided Routing for Space-Air-Ground Integrated Networks Relying on Real Satellite, Flight, and Shipping Data, *IEEE Wireless Communications*, 29(2), 177–184.
- [108] **Plass, S.** (2012). Seamless networking for aeronautical communications: One major aspect of the SANDRA concept, *IEEE Aerospace and Electronic Systems Magazine*, 27(9), 21–27.
- [109] **Du, B., Xue, R., Zhao, L. and Leung, V.C.** (2020). Coalitional Graph Game for Air-to-Air and Air-to-Ground Cognitive Spectrum Sharing, *IEEE Transactions on Aerospace and Electronic Systems*, 56(4), 2959–2977.
- [110] **Cui, J., Liu, D., Zhang, J., Yetgin, H., Ng, S.X., Maunder, R.G. and Hanzo, L.** (2021). Minimum-Delay Routing for Integrated Aeronautical Ad Hoc Networks Relying on Real Flight Data in the North-Atlantic Region, *IEEE Open Journal of Vehicular Technology*, 2, 310–320.

- [111] **Du, B., Di, X., Liu, D. and Zhang, H.** (2021). Dynamic Graph Optimization and Performance Evaluation for Delay-Tolerant Aeronautical Ad Hoc Network, *IEEE Transactions on Communications*, 69(9), 6018–6036.
- [112] **Zhang, C., Patras, P. and Haddadi, H.** (2019). Deep Learning in Mobile and Wireless Networking: A Survey, 21(3), 2224–2287.
- [113] **Cote, D.** (2018). Using machine learning in communication networks [Invited], *IEEE J. Opt. Commun. Netw.*, 10(10), D100–D109.
- [114] **Hoffmann, F., Medina, D. and Wolisz, A.** (2013). Joint Routing and Scheduling in Mobile Aeronautical Ad Hoc Networks, 62, 2700–2712.
- [115] **Bilen, T., Erel-Özçevik, M., Yaslan, Y. and Oktug, S.F.** (2018). A Smart City Application: Business Location Estimator Using Machine Learning Techniques, *IEEE 16th International Conference on Smart City*, pp.1314–1321.
- [116] **Sander, J.** (2010). Density-Based Clustering, Springer US, Boston, MA, pp.270–273.
- [117] **Asan, U. and Ercan, S.** (2012). An Introduction to Self-Organizing Maps, Atlantis Press, Paris, pp.295–315.
- [118] **Sun, Y., Peng, M., Zhou, Y., Huang, Y. and Mao, S.** (2019). Application of Machine Learning in Wireless Networks: Key Techniques and Open Issues, 21(4), 3072–3108.
- [119] **Zhang, F., Wu, T.Y., Wang, Y., Xiong, R., ... and Liu, L.** (2020). Application of Quantum Genetic Optimization of LVQ Neural Network in Smart City Traffic Network Prediction, *IEEE Access*, 8, 104555–104564.
- [120] **Nova, D. and Estevez, P.A.** (2014). A review of learning vector quantization classifiers, *Neural Comput. Appl.*, 25.
- [121] **Arafat, M.Y. and Moh, S.** (2019). Routing Protocols for Unmanned Aerial Vehicle Networks: A Survey, *IEEE Access*, 7, 99694–99720.
- [122] **Liu, D., Cui, J., Zhang, J., Yang, C. and Hanzo, L.** (2021). Deep Reinforcement Learning Aided Packet-Routing for Aeronautical Ad-Hoc Networks Formed by Passenger Planes, *IEEE Transactions on Vehicular Technology*, 70(5), 5166–5171.
- [123] **Luong, N.C., Hoang, D.T., Gong, S., ... and Kim, D.I.** (2019). Applications of Deep Reinforcement Learning in Communications and Networking: A Survey, 21(4), 3133–3174.
- [124] **Elsayed, M. and Erol-Kantarci, M.** (2019). AI-Enabled Future Wireless Networks: Challenges, Opportunities, and Open Issues, 14(3), 70–77.

- [125] **van Otterlo, M. and Wiering, M.** (2012). Reinforcement Learning and Markov Decision Processes, Springer Berlin Heidelberg, Berlin, Heidelberg, pp.3–42.
- [126] **Nasir, Y.S. and Guo, D.** (2019). Multi-Agent Deep Reinforcement Learning for Dynamic Power Allocation in Wireless Networks, *37*(10), 2239–2250.
- [127] **Liu, D., Zhang, J., Cui, J., Ng, S.X., Maunder, R.G. and Hanzo, L.** (2022). Deep-Learning-Aided Packet Routing in Aeronautical Ad Hoc Networks Relying on Real Flight Data: From Single-Objective to Near-Pareto Multiobjective Optimization, *IEEE Internet of Things Journal*, *9*(6), 4598–4614.
- [128] **Du, B., Di, X., Liu, D. and Ning, H.** (2022). Coalitional Dynamic Graph Game for Aeronautical Ad Hoc Network Formation, *IEEE Internet of Things Journal*, *9*(8), 5773–5784.
- [129] **Tian, Y., Ma, L., Ru, L., Tang, H. and Song, B.** (2014). A Signal Characteristic Based Cluster Scheme for Aeronautical Ad Hoc Networks, *KSII Transactions on Internet and Information Systems*, *8*, 3439–3457.
- [130] **Oliveira, R., Luís, M., Bernardo, L. and Dinis, R.** (2012). Towards Reliable Broadcast in ad hoc Networks, *IEEE Communications Letters*, *16*, 314–317.
- [131] **Kaixin Xu and Gerla, M.** (2002). A heterogeneous routing protocol based on a new stable clustering scheme, *IEEE MILCOM Proceedings*, volume 2, pp.838–843 vol.2.
- [132] **Basu, P., Khan, N. and Little, T.D.C.** (2001). A mobility based metric for clustering in mobile ad hoc networks, *Proceedings IEEE 21st International Conference on Distributed Computing Systems Workshops*, pp.413–418.
- [133] **Ching-Chuan Chiang and Gerla, M.** (1997). Routing and multicast in multihop, mobile wireless networks, *6th International Conference on Universal Personal Communications*, volume 2, pp.546–551 vol.2.
- [134] **Medina, D., Hoffmann, F., Ayaz, S. and Rokitansky, C.** (2008). Feasibility of an Aeronautical Mobile Ad Hoc Network Over the North Atlantic Corridor, *5th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks*, pp.109–116.
- [135] **Das, M. and Ghosh, S.** (2016). A cost-efficient approach for measuring Moran’s index of spatial autocorrelation in geostationary satellite data, *IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, pp.5913–5916.
- [136] **Gross, D., Shortle, J.F., Thompson, J.M. and Harris, C.M.** (2008). *Fundamentals of Queueing Theory*, Wiley-Interscience, USA, 4th edition.

- [137] **Berlakovich, M., Ruthmair, M. and Raidl, G.R.** (2012). A Multilevel Heuristic for the Rooted Delay-Constrained Minimum Spanning Tree Problem, *Computer Aided Systems Theory- EUROCAST 2011*, Springer Berlin Heidelberg, pp.256–263.
- [138] **GEE** (2015). Connected Aircraft: Impact on Operations Management and Analysis (Technical Report).
- [139] **Zhong, D., Wang, Y. and Zhu, Y.** (2017). A new data transmission mechanism between air vehicles, *International Journal of Distributed Sensor Networks*, 13.
- [140] **Hui, R. and O’Sullivan, M.,** (2009). Optical System Performance Measurements, Fiber Optic Measurement Techniques, Academic Press, Boston, pp.481 – 630.
- [141] **Yang, X., Deng, L., Wei, P., Li, H. and Liu, J.** (2020). Multi-Agent Autonomous Operations in Urban Air Mobility with Communication Constraints.
- [142] **Jiang, D., Huo, L., Lv, Z., Song, H. and Qin, W.** (2018). A Joint Multi-Criteria Utility-Based Network Selection Approach for Vehicle-to-Infrastructure Networking, *IEEE Transactions on Intelligent Transportation Systems*, 19(10), 3305–3319.
- [143] **Sodhro, A.H., Gurtov, A., Zahid, N., Pirbhulal, S., ... and Abbasi, Q.H.** (2020). Towards Convergence of AI and IoT for Energy Efficient Communication in Smart Homes, *IEEE Internet of Things Journal*, 1–1.
- [144] **Medina, D., Hoffmann, F., Rossetto, F. and Rokitansky, C.** (2012). A Geographic Routing Strategy for North Atlantic In-Flight Internet Access Via Airborne Mesh Networking, *IEEE/ACM Transactions on Networking*, 20(4), 1231–1244.
- [145] **Lin, Y., Jin, X., Chen, J., Sodhro, A.H. and Pan, Z.** (2019). An analytic computation-driven algorithm for Decentralized Multicore Systems, *Future Generation Computer Systems*, 96, 101–110.
- [146] **Ghosh, S. and Nayak, A.** (2015). Multi-dimensional clustering and network monitoring system for aeronautical ad hoc networks, *Seventh International Conference on Ubiquitous and Future Networks*, pp.772–777.
- [147] **Wang, Y., Ertürk, M.C., Arslan, H., Sankar, R. and Morgera, S.** (2011). Throughput analysis in Aeronautical Data Networks, *WAMICON 2011 Conference Proceedings*, pp.1–5.
- [148] **IATA** (2017). 2036 forecast reveals air passengers will nearly double to 7.8 billion (Technical Report).
- [149] **Lei, L., Wang, D., Zhou, L., Chen, X. and Cai, S.** (2014). Link availability estimation based reliable routing for aeronautical ad hoc networks, *Ad Hoc Networks*, 20, 53–63.

- [150] **Yang, J., Sun, K., He, H., Jiang, X. and Chen, S.** (2022). Dynamic Virtual Topology Aided Networking and Routing for Aeronautical Ad-hoc Networks, *IEEE Transactions on Communications*, 1–1.
- [151] **Numani, A., Gulfam, S.M., Javed, M.A., ... and Nawaz, S.J.** (2022). Network Architecture and Wireless Ad Hoc Routing for Airborne Internet Services, *Wirel. Pers. Commun.*, 122(4), 3529–3543.
- [152] **Gurumekala, T. and Gandhi, S.** (2022). A Cross Layered Routing Approach for Civil AANET, *Wireless Personal Communications*.
- [153] **Wu, J., Fang, M., Li, H. and Li, X.** (2020). RSU-Assisted Traffic-Aware Routing Based on Reinforcement Learning for Urban Vanets, *IEEE Access*, 8, 5733–5748.
- [154] **Nazib, R.A. and Moh, S.** (2021). Reinforcement Learning-Based Routing Protocols for Vehicular Ad Hoc Networks: A Comparative Survey, *IEEE Access*, 9, 27552–27587.
- [155] **Guo, X., Lin, H., Li, Z. and Peng, M.** (2020). Deep-Reinforcement-Learning-Based QoS-Aware Secure Routing for SDN-IoT, *IEEE Internet of Things Journal*, 7(7), 6242–6251.
- [156] **Di Valerio, V., Lo Presti, F., Petrioli, C., Picari, L., Spaccini, D. and Basagni, S.** (2019). CARMA: Channel-Aware Reinforcement Learning-Based Multi-Path Adaptive Routing for Underwater Wireless Sensor Networks, *IEEE Journal on Selected Areas in Communications*, 37(11), 2634–2647.
- [157] **Chen, Q., Meng, W., Han, S., Li, C. and Chen, H.H.** (2021). Reinforcement Learning-Based Energy-Efficient Data Access for Airborne Users in Civil Aircrafts-Enabled SAGIN, *IEEE Transactions on Green Communications and Networking*, 5(2), 934–949.
- [158] **Bilen, T. and Canberk, B.** (2019). Deliver the content over multiple surrogates: A request routing model for high bandwidth requests, *Computer Communications*, 146, 39–47.
- [159] **Bilen, T., Kurnaz, D.S., Sevim, S. and Canberk, B.** (2018). Parametric-Decomposition Based Request Routing in Content Delivery Networks, *Wired/Wireless Internet Communications*, Springer International Publishing, Cham, pp.323–335.
- [160] **Bilen, T. and Canberk, B.** (2019). Handover-Aware Content Replication for Mobile-CDN, *IEEE Networking Letters*, 1(1), 10–13.
- [161] **Zhang, X., Li, P., Hu, J., Liu, M., Wang, G., Qiu, J. and Chan, K.W.** (2019). Yen's Algorithm-Based Charging Facility Planning Considering Congestion in Coupled Transportation and Power Systems, *IEEE Transactions on Transportation Electrification*, 5(4), 1134–1144.

- [162] **Jacob, P., Sirigina, R.P., Madhukumar, A.S. and Prasad, V.A.** (2016). Cognitive Radio for Aeronautical Communications: A Survey, *IEEE Access*, 4, 3417–3443.
- [163] **Gurumekala, T. and Gandhi, S.** (2022). Toward in-flight Wi-Fi: a neuro-fuzzy based routing approach for Civil Aeronautical Ad hoc Network, *Soft Computing*, 1–22.
- [164] **Madni, A.M., Madni, C.C. and Lucero, S.D.** (2019). Leveraging Digital Twin Technology in Model-Based Systems Engineering, *Systems*, 7(1).
- [165] **Jones, D., Snider, C., Nassehi, A., Yon, J. and Hicks, B.** (2020). Characterising the Digital Twin: A systematic literature review, *CIRP Journal of Manufacturing Science and Technology*, 29, 36–52.
- [166] **Moyne, J., Qamsane, Y., Balta, E.C., Kovalenko, I., Faris, J., Barton, K. and Tilbury, D.M.** (2020). A Requirements Driven Digital Twin Framework: Specification and Opportunities, *IEEE Access*, 8, 107781–107801.
- [167] **Lei, L., Shen, G., Zhang, L. and Li, Z.** (2021). Toward Intelligent Cooperation of UAV Swarms: When Machine Learning Meets Digital Twin, *IEEE Network*, 35(1), 386–392.
- [168] **Zhao, L., Han, G., Li, Z. and Shu, L.** (2020). Intelligent Digital Twin-Based Software-Defined Vehicular Networks, *IEEE Network*, 34(5), 178–184.
- [169] **Nguyen, H., Trestian, R., To, D. and Tatipamula, M.** (2021). Digital Twin for 5G and Beyond, *IEEE Communications Magazine*, 59, 10–15.
- [170] **Yaqoob, I., Salah, K., Uddin, M., Jayaraman, R., Omar, M. and Imran, M.** (2020). Blockchain for Digital Twins: Recent Advances and Future Research Challenges, *IEEE Network*, 34(5), 290–298.
- [171] **Groshev, M., Guimarães, C., Martín-Pérez, J. and de la Oliva, A.** (2021). Toward Intelligent Cyber-Physical Systems: Digital Twin Meets Artificial Intelligence, *IEEE Communications Magazine*, 59(8), 14–20.
- [172] **Wang, D., Zhang, Z., Zhang, M., Fu, M., ... and Chen, X.** (2021). The Role of Digital Twin in Optical Communication: Fault Management, Hardware Configuration, and Transmission Simulation, *IEEE Communications Magazine*, 59(1), 133–139.
- [173] **Khan, L.U., Saad, W., Niyato, D., Han, Z. and Hong, C.S.** (2021). *Digital-Twin-Enabled 6G: Vision, Architectural Trends, and Future Directions*.
- [174] **Lopez, J., Rubio, J.E. and Alcaraz, C.** (2021). Digital Twins for Intelligent Authorization in the B5G-Enabled Smart Grid, *IEEE Wireless Communications*, 28(2), 48–55.

- [175] **Cui, J., Yetgin, H., Liu, D., Zhang, J., Ng, S.X. and Hanzo, L.** (2021). Twin-Component Near-Pareto Routing Optimization for AANETs in the North-Atlantic Region Relying on Real Flight Statistics, *IEEE Open Journal of Vehicular Technology*, 2, 346–364.
- [176] **Liu, L., Du, J. and Guo, D.** (2016). Error beacon filtering algorithm based on K-means clustering for underwater Wireless Sensor Networks, *8th IEEE International Conference on Communication Software and Networks (ICCSN)*, pp.435–438.
- [177] **Alam, C.N., Manaf, K., Atmadja, A.R. and Aurum, D.K.** (2016). Implementation of haversine formula for counting event visitor in the radius based on Android application, *4th International Conference on Cyber and IT Service Management*, pp.1–6.
- [178] **Microsoft.** Azure Digital Twin, Accessed: 20 September 2021, <https://docs.microsoft.com/en-us/azure/digital-twins/>.



CURRICULUM VITAE

Name SURNAME: Tuğçe BİLEN

EDUCATION:

- **M.Sc.:** 2017, Istanbul Technical University, Faculty of Computer and Informatics, Computer Engineering Department
- **B.Sc.:** 2015, Istanbul Technical University, Faculty of Computer and Informatics, Computer Engineering Department

PROFESSIONAL EXPERIENCE:

- 2021- Innovation Development Specialist - Innovation, Product& Service Development Directorship, Turk Telekom
- 2018-2021 Research and Teaching Assistant - Department of Computer Engineering, Istanbul Technical University
- 2022-present, IEEE Open Journal of Vehicular Technology
- 2021-present, IEEE Transactions on Cloud Computing
- 2021-present, IEEE Network Magazine
- 2021-present, IEEE Communications Letters
- 2021-present, IEEE Wireless Communications Magazine
- 2020-present, IEEE Transactions on Network and Service Management
- 2018-present, IEEE Wireless Communications Letters
- 2018-present, IEEE Communications Letters
- 2017-present, Elsevier Ad Hoc Networks
- 2016-present, IEEE Transactions in Vehicular Technology
- 2016-present, Elsevier Computer Networks
- 2016-present, Elsevier Computer Communications

PUBLICATIONS, PRESENTATIONS AND PATENTS ON THE THESIS:

Journals

7. **T. Bilen**, B. Canberk, T.Q. Duong, “*Digital Twin Evolution for Hard-to-Follow Aeronautical Ad-Hoc Networks*,” Accepted, IEEE Communications Standards Magazine, 2022, Scopus Journal Quartile:Q1.
6. **T. Bilen**, B. Canberk, “*Q-Learning Driven Routing for Aeronautical Ad-Hoc Networks*,” Elsevier Pervasive and Mobile Computing, vol. 87, Nov. 2022, Scopus Journal Quartile:Q1.
5. **T. Bilen**, E. Ak, B. Bal, B. Canberk, “*A Proof of Concept on Digital Twin-Controlled WiFi Core Network Selection for In-Flight Connectivity*,” IEEE Communications Standards Magazine, vol. 6, no. 3, pp. 60-68, September 2022, Scopus Journal Quartile:Q1.
4. **T. Bilen**, B. Canberk, V. Sharma, M. Fahim, T.Q. Duong, “*AI-Driven Aeronautical Ad-Hoc Networks: Challenges, Opportunities, and the Road ahead*,” MDPI Sensors, vol. 22, May. 2022, Scopus Journal Quartile:Q1.
3. **T. Bilen**, T.Q. Duong, H. Ahmadi, B. Canberk, “*Aeronautical Networks for In-Flight Connectivity: A Tutorial of the State-of-the-Art and Survey of Research Challenges*,” IEEE Access, vol. 10, pp. 20053-20079, Feb. 2022, Scopus Journal Quartile:Q1.
2. **T. Bilen**, B. Canberk, “*Three-Phased Clustered Topology Formation for Aeronautical Ad-Hoc Networks*,” Elsevier Pervasive and Mobile Computing, vol. 79, Jan. 2022, Scopus Journal Quartile:Q1.
1. **T. Bilen**, B. Canberk, “*Learning Vector Quantization Based Topology Sustainability for Clustered-AANETs*,” IEEE Network Magazine, vol. 35, no. 4, pp. 120-128, July/August 2021, Scopus Journal Quartile:Q1.

Conference Papers

2. **T. Bilen**, B. Canberk, “*GRU-Aided Intra-Cluster Topology Mapping for Aeronautical Ad-Hoc Networks*,” in IEEE Global Communications Conference (GLOBECOM), Virtual, 4-8 December 2022.
1. **T. Bilen**, P.J. Aydemir, A. E. Konu, B. Canberk, “*Customized K-Means Based Topology Clustering for Aeronautical Ad-hoc Networks*,” in IEEE International Workshop on Computer Aided Modeling and Design of Communication Links and Networks (CAMAD), Virtual, 25-27 October 2021.

Patents

3. **T. Bilen** “*Geçici Havasal Ağlar için Temel Bileşen Analizi Tabanlı Veri Kümesi Azaltma*” applied on 2022.

2. **T. Bilen** “Geçici Havasal Ağlarda Kapılı Tekrarlayan Hücre Tabanlı Topoloji Atama” applied on 2022.
1. **T. Bilen** “Geçici Havasal Ağlarda Pekiştirmeli Öğrenme Tabanlı Yönlendirme” applied on 2022.

OTHER PUBLICATIONS, PRESENTATIONS AND PATENTS:

Journals

6. **T. Bilen**, B. Canberk, “Content Delivery From the Sky: Drone-Aided Load Balancing for Mobile-CDN,” EAI Endorsed Transactions on Industrial Networks and Intelligent Systems, vol. 9, March 2022, Scopus Journal Quartile:Q4.
5. **T. Bilen**, B.Canberk, “Overcoming 5G Ultra-Density with Game Theory: Alpha-Beta Pruning Aided Conflict Detection,” Pervasive and Mobile Computing (Elsevier), vol. 63, March 2020, Scopus Journal Quartile:Q1.
4. **T. Bilen**, B.Canberk, “Deliver the Content over Multiple Surrogates: A Request Routing Model for High Bandwidth Requests,” Computer Communications (Elsevier), vol. 146, pp. 39-47, July 2019, Scopus Journal Quartile:Q1.
3. **T. Bilen**, B.Canberk, “Handover-Aware Content Replication for Mobile-CDN,” IEEE Networking Letters, Vol. 1, pp. 10-13, March 2019.
2. **T. Bilen**, K. Ayvaz, B. Canberk, “QoS-based Distributed Flow Management in Software Defined Ultra-Dense Networks,” Ad-hoc Networks Journal (Elsevier), vol. 78, pp.24-31, September 2018, Scopus Journal Quartile:Q1.
1. **T. Bilen**, B. Canberk, K. R. Chowdhury, “Handover Management in Software Defined Ultra-Dense 5G Networks,” IEEE Network Magazine, vol. 31, pp.49-55, August 2017, Scopus Journal Quartile:Q1.

Conference Papers

5. B. B. Ödevci, M. Özdem, E. Emsen and **T. Bilen**, “A Machine Learning Model for Predicting Performance of Gamified Software Test Specialist,” in International Conference on INnovations in Intelligent SysTems and Applications (INISTA), Biarritz, France, 8-12 August 2022.
4. **T. Bilen**, D.S. Kurnaz, S. Sevim and B. Canberk, “Parametric-Decomposition Based Request Routing in Content Delivery Networks,” in The 16th International Conference on Wired/Wireless Internet Communications (IFIP WWIC), Boston, MA, USA, June 2018.
3. **T. Bilen**, M. E. Ozcevik, Y. Yaslan, S. F. Oktug, “A Smart City Application: Business Location Estimator using Machine Learning Techniques,” in The 16th IEEE International Conference on Smart City (SmartCity-2018), Exeter-UK, June 2018.

2. **T. Bilen**, T. Duong, B. Canberk, “*Optimal eNodeB Estimation for 5G Intra-Macrocell Handover Management*,” in 12th ACM International Symposium on QoS and Security for Wireless and Mobile Networks (ACM Q2SWiNET), Valletta-MALTA, November 2016.
1. **T. Bilen**, B. Canberk, “*Binary Context Tree based Middleware for Next Generation Context Aware Networks*,” in Proceedings of IEEE FiCloud, Rome-ITALY, August 2015.

Patents

1. **T. Bilen**, B. Canberk, S. Sevim, “*Parametric Parsing Based Routing System In Content Delivery Networks*,” US Patent App. 17/273,018, 2021.

