CLUSTERING ALGORITHM FOR D2D COMMUNICATION IN NEXT GENERATION CELLULAR NETWORKS

THESIS PRESENTED IN PARTIAL FULFILMENT OF

THE

REQUIREMENTS FOR THE DEGREE OF

DOCTOR OF PHILOSOPHY

IN

ENGINEERING

MASSEY UNIVERSITY, AUCKLAND, NEW ZEALAND.

Saad Aslam

April 2021

ABSTRACT

Next generation cellular networks will support many complex services for smartphones, vehicles, and other devices. To accommodate such services, cellular networks need to go beyond the capabilities of their previous generations. Device-to-Device communication (D2D) is a key technology that can help fulfil some of the requirements of future networks.

The telecommunication industry expects a significant increase in the density of mobile devices which puts more pressure on centralized schemes and poses risk in terms of outages, poor spectral efficiencies, and low data rates. Recent studies have shown that a large part of the cellular traffic pertains to sharing popular contents. This highlights the need for decentralized and distributive approaches to managing multimedia traffic.

Content-sharing via D2D clustered networks has emerged as a popular approach for alleviating the burden on the cellular network. Different studies have established that D2D communication in clusters can improve spectral and energy efficiency, achieve low latency while increasing the capacity of the network. To achieve effective content-sharing among users, appropriate clustering strategies are required. Therefore, the aim is to design and compare clustering approaches for D2D communication targeting content-sharing applications. Currently, most of researched and implemented clustering schemes are centralized or predominantly dependent on Evolved Node B (eNB). This thesis proposes a distributed architecture that supports clustering approaches to incorporate multimedia traffic. A content-sharing network is presented where some D2D User Equipment (DUE) function as content distributors for nearby devices. Two promising techniques are utilized, namely, Content-Centric Networking and Network Virtualization, to propose a distributed architecture, that supports efficient content delivery.

We propose to use clustering at the user level for content-distribution. A weighted multi-factor clustering algorithm is proposed for grouping the DUEs sharing a common interest. Various performance parameters such as energy consumption, area spectral efficiency, and throughput have been considered for evaluating the proposed algorithm. The effect of number of clusters on the performance parameters is also discussed. The proposed algorithm has been further modified to allow for a trade-off between fairness and other performance parameters. A comprehensive simulation study is presented that demonstrates that the proposed clustering algorithm is more flexible and outperforms several well-known and state-of-the-art algorithms.

The clustering process is subsequently evaluated from an individual user's perspective for further performance improvement. We believe that some users, sharing common interests, are better off with the eNB rather than being in the clusters. We utilize machine learning algorithms namely, Deep Neural Network, Random Forest, and Support Vector Machine, to identify the users that are better served by the eNB and form clusters for the rest of the users. This proposed user segregation scheme can be used in conjunction with most clustering algorithms including the proposed multi-factor scheme. A comprehensive simulation study demonstrates that with such novel user segregation, the performance of individual users, as well as the whole network, can be significantly improved for throughput, energy consumption, and fairness.

Keywords: 5G and Beyond, Cellular Networks, Content-Sharing, Clustering Algorithms, Deep Neural Network, Device-to-Device (D2D) Communication, Machine Learning, Random Forest, Social-Aware Networks, Support Vector Machine.

Author's Declaration

I, Saad Aslam, declare that this thesis and the work presented in it are my own and has been generated as a result of my original research.

Acknowledgments

The underlying thesis would not have been possible to complete without receiving the support and help of many people. It's truly a lifetime experience that I will never forget for the rest of my life.

First and foremost, I am greatly indebted to my supervisor, Associate Professor Dr. Fakhrul Alam, not only for his careful guidance and support in my academic research but also for his serving as my life mentor to teach me a lot of skills in academic research and life in general. Working with him has been a very enjoyable and rewarding experience. I would like to truly thank him for always being there to advise, support, and encourage me.

I would like to thank my co-supervisors, Dr. Mohammad Rashid and Dr. Syed Faraz Hasan for their encouragement and valuable comments that not only helped me to greatly improve the quality of this thesis but also inspired me to widen the area of my future research.

Most importantly, I would like to acknowledge my late mother, who passed away during my Ph.D. She did not let me know that she was ill, so I would not lose my concentration in Ph.D. Her prayers and unmatched support have always been with me.

Last but not the least, I would like to thank my family. During my Ph.D. pursuit, they always provided unconditional love and encouragement. Especially, I owe a debt of gratitude to my wife for her understanding, encouragement, and support in these years.

Table of Contents

List of Figures

List of Tables

List of Publications

The work presented in this thesis has been published in the following peer-reviewed Journal/Conference articles:

[P1] Aslam, S., Alam, F., Hasan, S. F., & Rashid, M. A. (2021). A Machine Learning Approach to Enhance the Performance of D2D-Enabled Clustered Networks. IEEE Access, 9, 16114-16132. **[Q1, IF: 3.75]**

[Related Chapters: 5,6]

[P2] Aslam, S., Alam, F., Hasan, S. F., & Rashid, M. A. (2020). A Novel Weighted Clustering Algorithm Supported by a Distributed Architecture for D2D Enabled Content-Centric Networks. *Sensors*, *20*, 5509.

[Related Chapters: 3,4]

[Q1, IF: 3.275]

*[P3] Aslam, S., Alam, F., Hasan, S. F., & Rashid, M. A., "Performance Analysis of Clustering Algorithms for Content-sharing Based D2D Enabled 5G Networks," 2019 29th International Telecommunication Networks and Applications Conference (ITNAC), Auckland, New Zealand, 2019, pp. 1-7. **[Related Chapters: 3,4]**

* **This paper received the highly commended research paper award at the 2019 29th ITNAC conference.**

[P4] Aslam, S., Alam, F., Hasan, S. F., & Rashid, M. A., "Decentralized Interference Mitigation Technique for D2D Networks Using Game Theory Optimization," 2019 29th International Telecommunication Networks and Applications Conference (ITNAC), Auckland, New Zealand, 2019, pp. 1-7.

Book Chapter

[P5] Aslam, S., Alam, F., Pajooh, H.H., and Rashid, M. A. (2021). Machine Learning Applications for Heterogeneous Networks. In: Real-Time Intelligence for Heterogeneous Networks. Springer Nature, Cham, Switzerland. **[Accepted] [Related Chapter: 2]**

Abbreviations

1G – First Generation

2G – Second Generation

3G – Third Generation

3GPP – 3rd Generation Partnership Project

4G – Fourth Generation

5G – Fifth Generation

6G – Sixth Generation

AI – Artificial Intelligence

ASE – Area Spectral Efficiency

BS – Base Station

Cal-Har – Calinski–Harabasz (Criterion)

CCN – Content-Centric Network

*CSN – Content Sharing Network

*(Throughout the thesis CSN is used interchangeably with content sharing framework and content sharing architecture)

CH – Cluster Head

CM – Cluster Member

CN – Core Network

CO – Cellular Operator

CSI – Channel State Information

CUE – Cellular User Equipment

D2D – Device-to-Device Communication

DUE – D2D User Equipment

DNN – Deep Neural Network

EBC – Entropy of Betweenness Centrality (Algorithm)

eNB – Evolved Node B (used interchangeably with BS throughout the thesis)

FCM – Fuzzy C-Means

GA – Genetic Algorithm

IoT – Internet of Things

LTE-A – Long Term Evolution – Advanced

MAC – Medium Access Control

MF – Multi-Factor (Algorithm)

ML – Machine Learning

mmWaves – Millimeter Waves

NA – Network Assisted

NNA – Non-Network Assisted

NV – Network Virtualization

PDR – Peer Discovery Resource

QoE – Quality of Experience

QoS – Quality of Service

RAN – Radio Access Network

RF – Random Forest

RRM – Radio Resource Management

SNR – Signal-to-Noise Ratio

SINR – Signal-to-Noise Plus Interference Ratio

SVM – Support Vector Machines

TDMA – Time Division Multiple Access

UE – User Equipment

V2V – Vehicle-to-Vehicle

V2X – Vehicle-to-Everything

VC – Virtualization Controller

VPN – Virtual Private Network

CHAPTER 1

INTRODUCTION

1.1 Background

Mobile communication has rapidly evolved from First Generation (1G) to Sixth Generation (6G) over the last few decades. Mobile Generation (G), in general, represents a significant change in technology, data rates, latency, capacity, frequency, and applications. Figure. 1.1 shows the evolution of mobile generations.

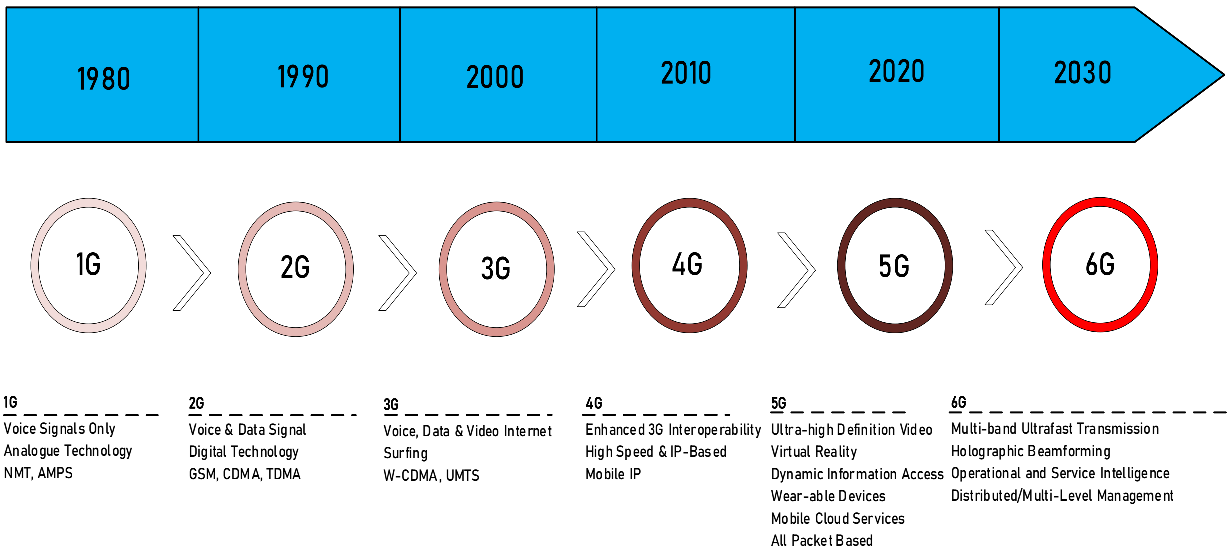

Figure. 1.1 The Evolution of Cellular Networks: From 1G to 6G.

1G was analog in nature which supported voice calls only. The Second Generation (2G) introduced digital technology and the concept of text messaging. Cellular systems experienced a major upgrade for supporting data services. Before the next big leap from 2G to Third Generation (3G), interim standards 2.5G and 2.75G [1, 2] introduced packet switching technology. 3G delivered increased capacity, higher transmission rate, and multimedia support. The Fourth Generation (4G) brought integration with fixed internet to support wireless mobile internet and in doing so, it overcame the limitations of 3G [1].

The Fifth Generation (5G) represents a new revolution in the cellular field. Cellular wireless networks based on 5G technology can provide unprecedented services with higher data rates, enhanced Quality of Experience (QoE), and lower energy consumption [2]. The upcoming 6G intends to leverage various new technologies such as Artificial Intelligence (AI), terahertz communication, three-dimensional networking, quantum communications, and big data analytics.

The various emerging applications of next generation cellular networks can be realized if the networks are intelligent and dynamic, providing ultra-low latency and high-speed data transmission. 6G is expected to deliver data rates of 1Tbps with less than 1ms end-to-end latency [3]. One of the technologies that are considered essential for the realization of modern cellular networks, such as 5G and 6G, is Device to Device (D2D) communication [4-6].

Direct communication between devices in proximity can take place using the concept of D2D communication. This type of communication does not need to traverse through the core network. It is different from the conventional network as the conventional network requires all the communication to go via the core network irrespective of the proximity of devices [7, 8]. The D2D concept is shown in Figure. 1.2.

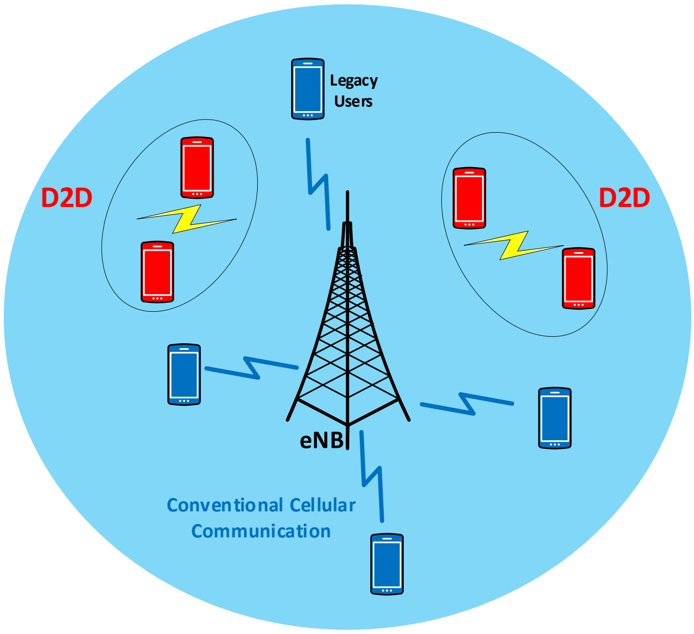

Figure. 1.2 A D2D Operated Network. The red-colored devices are communicating directly in D2D mode whereas the blue-colored devices are the conventional cellular users.

It is important to note that conventional communication is suitable for services such as low data rate voice calls and text messages. In contrast to this, today's mobile users demand high data rate services like video sharing, online gaming, social-aware, and context-aware networking. Users close to each other demanding such services are very well suited to be served by D2D communication. D2D has two major advantages; it increases the spectral efficiency of the network by avoiding unnecessary transmissions and, achieve higher data rates by leveraging the proximity of users. However, these are not the only benefits of D2D communication. The concept of D2D communication has initiated numerous research works to assist cellular networks. The on-going research in this field not only consists of proposals for relaying the cellular traffic, multicasting options, offloading cellular traffic, and content distribution [8], but also a complete architecture based on D2D communication to work in tandem with cellular-based services supporting new types of applications [9].

1.2 Applications of D2D Communication

Many scenarios require users in close proximity to exchange information. Therefore, several applications exist for D2D communication. Some of these applications are depicted in Figure. 1.3 and described in the following sub-section.

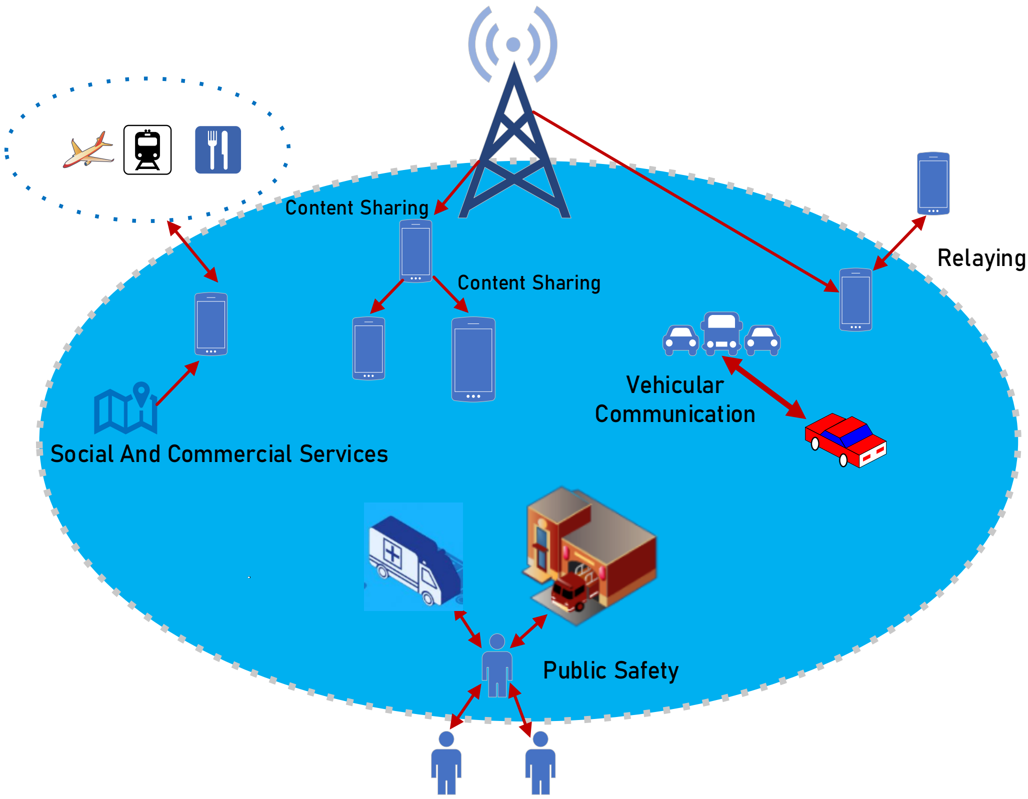

Figure. 1.3 Various applications of D2D communication.

1 - Cellular Offloading

Cellular traffic offloading is an incentive for deploying the D2D communication. Any device having good connectivity with the central controller can be made responsible for data offloading or data caching from the eNB so that devices nearby can download this data using direct links [10]. It can help a computational resource constrained user to offload the task to a nearby capable DUE. Various offloading techniques have been explored by different researchers [11].

2 - Coverage Extension

D2D communication is an ideal candidate for coverage extension via relays. D2D relays can be used for coverage extension by enabling multi-hop communication. Cooperative diversity is usually exploited to enhance the signal strength at the receiver by relaying it via multiple paths [12].

3 - Machine-to-Machine (M2M) communication

D2D promises to offer real-time responses that can be leveraged for setting up the M2M communication for Internet of Things (IoT) based services [13].

4 - Public Safety

D2D communication has the potential to support emergency services for public safety [14]. In the aftermath of a natural disaster (such as an earthquake), network failure may occur. In such scenario, safety organizations, such as police and rescue services, can rely on D2D communication to form communication links and convey important information to rescuers.

5 - Vehicular Networks

Vehicular Networks is another application of D2D communication that facilitates communication of a vehicle to other vehicles, devices, and infrastructure [15]. D2D links can help to share information among the vehicles and other entities. Moreover, D2D can also help realize Intelligent Transportation Systems (ITS) by meeting its strict delay requirements [16].

6 - Content Multicasting

One of the most promising applications of D2D communication is content multicasting. D2D communication can efficiently leverage local data services via different techniques such as unicast, broadcast, or group communication [17]. DUEs can transfer audio, video, and other files at higher data rates and most importantly with less energy and better power consumption. In such scenarios, group casting/multicasting can be used to convey the information to users in a group. The formation of clusters can facilitate such applications and services.

1.3 Motivation

Among the several D2D applications mentioned in the previous subsection, the focus of this thesis is on the multicasting/content-sharing scenarios. Multicasting is reliant on user cooperation and true potentials of user cooperation are exploited by D2D communication. It paves the path for cellular users to aid operators in developing techniques for cooperative communication which has the potential to enhance resource utilization efficiency [18]. One of the significant applications of D2D communication can be found in densely populated areas where many devices, confined in a small area, are demanding network resources. One such example is a concert, where many users are trying to download the same multimedia/social media contents. In such a scenario, the users can share common contents of interest. Multicasting via D2D communication can be exploited where a user is responsible for delivering the contents, received from the eNB, to different users. A multicasting scenario can be set up through the formation of clusters to deliver the contents. Clusters are composed of Cluster Members (CMs) that demand the content whereas a Cluster Head (CH) is selected to deliver the contents to its CMs. Clustering organizes the users with social ties i.e. sharing similar interests and restructures the network topology to achieve network and cluster level performance improvements [7]. As a result of clustering, network performance can be improved in terms of throughput, spectral efficiency, power, and energy consumption [19]. Given the potential advantages of clustering for D2D-enabled networks, designing and optimizing clustering algorithms is of utmost importance and therefore investigated in this thesis. We focus on the design, implementation, and comparative study of clustering algorithms targeting content-sharing applications. A distributed architecture supporting clustering is proposed as well. Machine Learning (ML) is subsequently employed as well to optimize the clustering algorithm and further improve the network performance.

1.4 Thesis Layout

Chapter Two presents the relevant literature review. The requirements of the proposed schemes are highlighted. The research objectives of the thesis are discussed as well.

Chapter Three develops the proposed clustering algorithm supported by a distributed architecture. The scheme to identify the content of interest is presented as well. All the design parameters are detailed in this chapter.

Chapter Four presents the performance of the designed clustering algorithm. Simulation setup and parameters of interest are discussed in this chapter as well. The performance with respect to aggregate throughput, energy consumption, throughput fairness, and area spectral efficiency are analyzed and discussed in detail.

Chapter Five investigates the effect of clustering on individual users. ML algorithms are proposed to develop a user segregation scheme that identifies the users that should be considered for clustering as opposed to putting all the users, interested in a particular content, in clusters. All the details of ML implementation are discussed in this chapter.

Chapter Six discusses and analyzes all the results associated with the user segregation scheme detailed in Chapter Five. The performance of the implemented ML algorithms, as well as other performance parameters, are discussed as well.

Chapter Seven concludes the thesis, highlights the contribution, and discusses future research directions.

CHAPTER 2

LITERATURE REVIEW

2.1 The Drive Towards Content-Centric Networks

There is an unprecedented demand for multicast applications which is one of the driving forces to move towards content-centric cellular networks [20]. It is expected that content-sharing, specifically videos, will constitute more than 70% of the future data traffic [21]. The expected distribution of data traffic (for the year 2021 – 2022) is represented in Figure. 2.1.

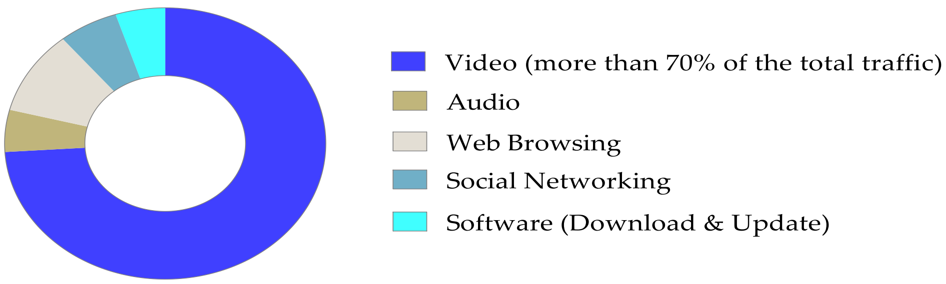

Figure. 2.1 The expected percentage of data traffic (by type) for the year 2021- 2022. Video (generation and sharing) dominates the total traffic generated by the cellular users [22].

Due to the surge in multimedia services [23, 24], existing centralized architectures and mechanisms may not be able to meet the content-sharing demands [2, 20]. Therefore, decentralized architectures and load mitigation techniques are required. It is worthwhile to offload some of the multimedia traffic to the D2D tier to reduce the load on the cellular network's infrastructure. In practice, popular content, such as Social Media platform content (Facebook, Twitter, etc.), YouTube videos, are requested much more frequently than others. As a result, the eNB often ends up serving different mobile users with the same contents using multiple duplicate transmissions. The initial transmission of popular content can be cached and users demanding the same contents, that are within the transmission distance, can receive the "cached" contents directly through D2D communication.

Ultra-dense networks have encouraged many researchers to study the feasibility of clustering and relaying techniques. Various research works have targeted D2D clustering which has found that relaying traffic to different cellular users by clustering them, can achieve lower signaling overhead and better spectral and energy efficiency [25, 26].

2.2 D2D Clustering Literature

An introductory study of D2D multicast is available in [18] that forms clusters with a predetermined number of D2D devices. It shows the gain in data rate achieved through cooperative multicast transmission for clustered D2D communication. Another study has been conducted in [27], with a point of view of reducing data distribution time. It provides an approach to better handle the transmission failures by letting CHs assist CMs to retransmit the failed transmission. Yaacoub et al., [28] utilize game theory for collaborative communication among D2D clusters. A comprehensive study has been presented in [29] to determine the effect of varying data transfer rates on energy consumption.

It is suggested that multicast performance can be improved by clustering [30, 31]. Moreover, it is emphasized that D2D multicast can yield better results as compared to non-cellular short-range technologies [32]. Another application supporting traffic safety introduced multi-hop architecture in [33], to maximize transmission distance and minimize transmission delay.

The users are actively involved in social interactions and wireless networks can gain benefits from such interactions [34]. Social ties have been exploited in [35] to evaluate the gains of cooperative communication in a D2D enabled cellular network. This work exploited social reciprocity and trust. Optimization of the D2D network is presented in [36] by introducing social ties among users. Clustered D2D communication has been studied in [37] where a joint optimization methodology was presented for precoding D2D and cellular transmissions. It is important to note that all the above-mentioned works consider distance among DUEs as the main metric for clustering. However, later in Chapter 4, it is shown that other parameters are important and should be considered while forming clusters.

D2D clustering and relaying have been popularly used for coverage improvements and traffic aggregation [38-42]. These works derived analytical expressions for throughput, power, and energy efficiencies. In [41], the trade-off between latency and transmit power is shown highlighting that transmit power can be reduced subject to increased latency. It has been shown in many works that the formation of appropriate clusters lead to better performance in underlay D2D networks. Many research works have taken into consideration different factors like location, social characteristics, and contents similarity while making clustering decisions. In [42], cluster formation was based on location. Another research work [43], jointly considers location and social characteristics for clustering. In [44] and [27], it is shown that the performance of a cluster is greatly affected by the cluster head selection. In these works, a cluster head is considered to be a device having content to share with the highest number of D2D links.

2.2.1 Inherent Benefits of D2D Clustering

While discussing clustering for D2D networks, it is very important to study the inherent benefits of introducing clustering into D2D-enabled networks. Different research works (listed in Table 2.1) have evaluated the throughput performance of the clustered network. Analytical and simulation-based results are presented to support their observations and findings. According to the results, the clustering algorithm for D2D pairs can produce higher throughput as compared to other works that do not consider clustering.

Carpio. et al. [45] suggest that the conventional method for increasing the throughput and capacity of the network includes physical-layer capacity improvement, increasing the area spectral efficiency by reducing the cell size but clustering promises to present a simpler and viable solution for the improvement in the capacity and throughput especially in case of high user density. Guo et al., [46] used the graph theory for cluster formation and resource assignment for the D2D network. The numerical results show that their algorithm can achieve better throughput as compared to conventional LTE networks. Authors in [47] show that cooperatively formed clusters for D2D users can produce the benefits of higher throughput and better energy consumption. Their results show that throughput is scalable with the increase in size and number of clusters. A greedy heuristic algorithm is proposed in [48] to maximize the number of satisfied users in a cluster. Results show that the throughput is much higher than the un-clustered networks and a significant increase in the number of satisfied users is seen as compared to other works considering the same performance parameters. A location-based clustering algorithm is utilized in [49] to propose a model for inter-cluster communication. They have used cooperative relays to improve the retransmission throughput as compared to a conventional network which does not consider clustering and cooperative relaying.

Some researchers have utilized game theory to jointly optimize power and resource allocation for overall improvement in the system throughput while others focus on better resource allocation strategies for throughput maximization via transmission rate improvement [50]. The major categorization of these models includes cooperative and non-cooperative game nature. All these models have proven to provide better resource management techniques with improved throughput, especially for a clustered network [51]. Stackelberg game is popularly used for resource allocation among Cellular User Equipment (CUEs) and DUEs [50, 52-54]. These works employ a buyer-seller scenario to maximize the utilities of the users and central controller. The authors in these works show that individual utilities of users and central controllers eventually reach an equilibrium where throughput, power, and interference levels cannot be further improved. The only problem is that throughput achieved via this technique is lower as compared to other methods targeting clustering applications. Moreover, though the game theory optimization techniques are effective, they increase the signaling load of the central controller.

2.3 The Requirement of Clustering Algorithm and Supporting Decentralized Network Architecture

Collaborative communication, content, and resource sharing can be realized by clustering the DUEs in proximity [55]. By allowing DUEs to form clusters collaboratively, network resources can be better utilized, interference can be managed, intra-cell and intragroup communication can be efficiently coordinated and social interactions can be improved as well [56].

D2D enables the merger of the attributes and capabilities of both distributed and centralized communication mechanisms. [57]. Moreover, D2D communication in a clustered environment can be easily integrated into a cellular network with the help of cognitive radio and cooperative communication.

It should be noted that D2D communication in clusters also helps in designing distributed mechanisms for network management. Distributed mechanisms are significant for today's cellular networks since the number of mobile users have increased rapidly over the past decade. Smartphones and other handheld devices have played a vital role in this exponential rise of mobile users (see Figure. 2.2 for relevant data).

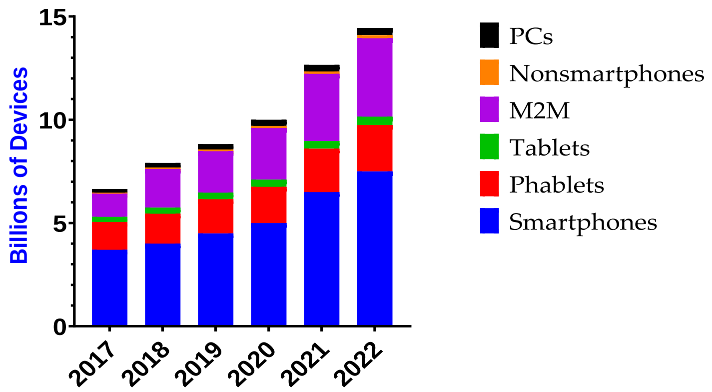

Figure. 2.2 Increasing trend of mobile users where smartphones have played a significant role [21].

With exponentially increasing network devices, it is becoming difficult to fulfil the QoS requirements of multimedia services [58, 59]. The growth of cellular devices has been addressed with the concept of dense networks having a large number of small cells. However, the limited capacity of the backhaul becomes a bottleneck in such a scenario [60]. It has been proposed that the significant growth in 5G networks and beyond can be accommodated by investing more in Content-Centric Networks (CCN). The rationale behind a CCN is to present a scalable and efficient mechanism for content delivery [61]. The techniques developed around CCN are expected to reduce the transmission delay by caching the data within the network. Data caching is performed closer to the group of mobile devices/content requesters by exploiting social ties (e.g., shared interest in similar content) [62, 63]. In CCN [64], contents are acquired by their name rather than the IP address of the entity hosting the content. This communication paradigm is expected to decrease network congestion while providing fast and secure distribution of the contents [65].

Device-level caching can be facilitated by D2D communication. D2D has been effectively used to disseminate data in various network scenarios. However, the network architectures presented in many of these works [62, 63, 66, 67] (and other works mentioned in Table 2.1) that consider D2D multicasting scenarios are centralized in nature and require massive message passing to make the whole scheme work. Such schemes do not meet the requirements of dense future networks such as 6G. Contrary to this, a decentralized architecture is proposed in this thesis (please see Chapter 3) that can effectively support the D2D multicasting scenario.

Apart from CCN, researchers have investigated other technologies that address the scalability of the network and the ability to cater to the growth in wireless traffic and services. The Network Virtualization (NV) concept has been widely used [60, 61, 63] in this context. Virtual Private Networks (VPN) is one such example. Network virtualization aims to slice the resources of cellular architecture into virtual resources to be shared among multiple users. It should be noted that by cellular resources we mean a licensed spectrum and infrastructure e.g., Core Network and Radio Access Network, etc. [63]. All the signalling and message passing that needs to take place to set up the virtual network is well researched and presented in the literature [60, 61, 63].

This thesis proposes a network architecture that combines the concepts of CCN and NV. This is different from the schemes found in literature, as we merge both the technologies. There are significant advantages to the proposed merger. One of the most important features is that not only the cellular resources but the actual contents can be shared. Duplicate transmissions exhaust the cellular resources. The content-sharing among the networks with the aid of virtualization can significantly reduce these redundant transmissions. Details of the proposed architecture are presented in the next chapter.

When a multicasting scenario is considered in the literature within the context of clustering, most works assume that users sharing a common interest have already been identified. Furthermore, if a system model is presented, how a typical cellular architecture can support such a model is not shown or glossed over. Most articles either discuss social tie/social interest modelling or clustering algorithm in detail but not both [68, 69]. For instance, in [68], a clustering algorithm has been proposed and social metric is also considered as an important factor. However, it does not provide any details around how the users having the same interest are identified. The architecture that supports their system model is not presented either. Research work presented in [69] discusses the social ties/social attributes in detail and effectively describes the mechanism behind modelling social metrics. It also considers clustering for a multicasting scenario but the clustering itself is assumed to have taken place by placing the users in a certain grid. Therefore, clustering and its effects on the performance parameters have not been discussed. It should be noted that that [68, 69] uses centralized mechanisms. Similar is the case with various other works mentioned in Table 2.1. It also suggests that most studies focus on throughput, energy consumption, and area spectral efficiency (ASE) but fairness has not been given due attention in the relevant literature which is elaborated in Figure. 2.3. On the other hand, research suggests that throughput fairness is an important parameter for evaluating a cellular network [70-73].

As shown in Table 2.1, a clustering algorithm that is supported by a distributed architecture, considers different performance parameters including fairness, and offers the flexibility to trade-off the performance for fairness is not seen in the recent and relevant literature. Moreover, since the content-sharing scenario is considered, a network architecture that conforms to the requirements of content identification is required as well. Therefore, the first part of this thesis is dedicated to accomplishing these tasks and addresses this gap by developing a clustering algorithm along with decentralized architecture and a mechanism to identify the content of interest.

Table 2.1 Summary of Clustering Related Articles and Their Performance Parameters

Research	Year	Distributed Architecture	Performance Parameters				User Segregation
			Throughput	Energy Consumption	ASE	Fairness	
Xu et al. [74]	2014	×	×	✓	✓	×	×
Militano et al. [75]	2015	×	×	✓	×	×	×
Asadi et al. [47]	2016	×	✓	✓	×	✓	×
Ashraf et al. [76]	2016	×	✓	×	×	×	×
Afshang et al. [77]	2016	×	×	×	✓	×	×
Afshang et al. [78]	2016	×	×	×	✓	×	×
Zhang et al. [79]	2017	×	×	✓	×	×	×

Doumiati et al. [80]	2017	×	×	✓	×	×	×
Yi et al. [81]	2017	×	×	×	✓	×	×
Xia et al. [82]	2017	×	×	✓	×	×	×
Niu et al. [83]	2017	×	×	✓	×	×	×
Ren et al. [84]	2017	×	✓	×	×	×	×
Kitagawa et al. [85]	2017	×	✓	×	×	×	×
Tulu et al. [86]	2017	×	✓	✓	×	×	×
Xu et al. [87]	2017	×	×	✓	✓	×	×
Huang et al. [88]	2017	×	×	✓	✓	×	×
Li et al. [89]	2017	×	×	✓	✓	×	×
Duan et al. [90]	2017	×	×	✓	✓	×	×
Yaacoub et al. [91]	2018	×	×	✓	×	×	×
Zhao et al. [92]	2018	×	×	✓	×	×	×
Sharafesddine et al. [93]	2018	×	✓	×	×	×	×
Yang et al. [94]	2018	×	×	✓	×	×	×
Huang et al. [95]	2018	×	✓	×	×	×	×
Rahman et al. [96]	2018	×	×	✓	×	×	×
Amer et al. [97]	2018	×	✓	✓	×	×	×
Pizzi et al. [98]	2019	×	✓	×	×	×	×
Aslam et al. [99]	2019	×	✓	✓	✓	×	×
Shi et al. [100]	2019	×	×	✓	×	×	×
Wu et al. [101]	2019	×	✓	×	×	×	×

Wang et al. [102]	2019	×	×	×	✓	×	×
Yin et al. [103]	2019	×	×	✓	✓	×	×
Zhou et al. [104]	2020	×	×	✓	✓	×	×
Zhang et al. [55]	2020	×	×	✓	×	×	×
Yin et al. [93]	2020	×	✓	×	×	×	×
Gong et al. [105]	2021	×	✓	×	×	×	×
Jian et al. [106]	2021	×	✓	✓	×	×	×
The Proposed Work	2021	✓	✓	✓	✓	✓	✓

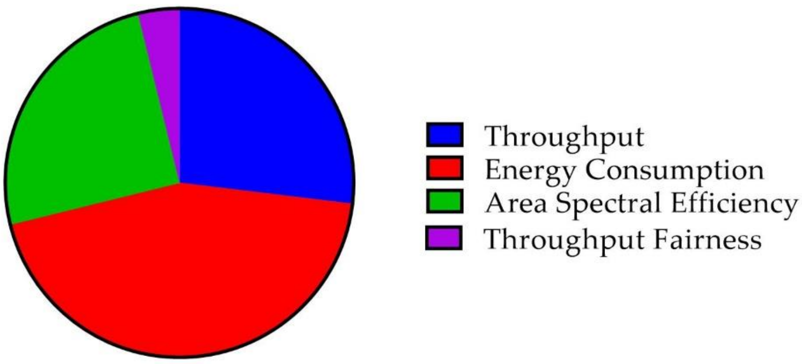

Figure. 2.3 Performance parameters considered by various research articles. It clearly shows that throughput fairness has not been widely considered by the clustering-related works. The proposed study considers Throughput Fairness while evaluating the performance of the clustered network.

2.4 The Requirement for Optimizing the Clustering Process

For content-sharing applications, it is important to design clustering algorithms and study content caching techniques. However, while these works improve the performance of the clustered network, optimization of the clustering scenario has not been well explored. A further investigation of the D2D clustering-related literature suggests that the focus of most research works is only on system-level performance. The effect of clustering on user-level performance needs to be investigated. Some users can be at a disadvantage after being placed in the clusters. Therefore, this thesis aims at optimizing the clustering process by investigating the effect of clustering on the performance of an individual user.

A study of recent clustering algorithms [74 - 106] shows that these works place all the users that are interested in sharing the same content in clusters. While this is a standard practice among clustering works, it does not reflect the effect of clustering on the performance of the individual users. More precisely 'all in cluster' approach does not consider the possibility of having two distinct groups of users: users better served in clusters, and users that are better served without being in clusters (i.e. connected to eNB). We aim to identify the users better served by not being in clusters communicating directly with the eNB. Therefore, a mixed-mode clustering approach is proposed (also called the 'user segregation' approach). It is shown that such an approach improves various performance parameters significantly.

As far as the author is aware, the concept of user segregation has not been explored before in the literature. All the research works mentioned in Table 2.1 do not consider the user segregation concept.

To perform the user segregation, ML has been utilized. Specifically, various classifiers have been explored and applied to segregate the users (details can be found in Chapter 5). The following sub-section describes the basics of ML, the rationale of using ML, present various applications of ML as applied to wireless networks and discuss related research works as well.

2.5 The Key Concepts of Machine Learning

Machine Learning aims at designing algorithms that train on the input data to predict an output. Conventional modelling often falls short of learning the complex relationships that exist among various parameters of wireless networks [107]. For example, it may be difficult to accurately model a communication scenario owing to the non-linearities brought into the system due to hardware impairments, propagation complexities, etc. [107, 108]. Therefore, ML algorithms are often utilized to address these challenges. A sequence of steps is followed to train the ML algorithm that are briefly described in the following text.

1. ***Data Collection and Data Preparation***

The first step is to gather the data. It is significantly important since the quantity and quality of the collected data impacts the performance of the trained algorithm. Once the data is collected, it needs to be prepared for training. To prepare the data, it needs to be cleaned (eliminate duplicates, address missing value issues, randomize the data, etc.) and most importantly split into training and testing sets.

2. ***Selection of Training Model***

Different ML algorithms are available, and it is important to choose a particular model that suits the data and the application scenario. The details on different ML algorithms can be found in the subsequent sub-section.

3. ***Train a Model***

At this stage, input data (usually set aside for training known as training data) is used to learn the relationship between input and output and make accurate predictions. It is an iterative process where each training epoch updates the algorithm.

4. ***Model Validation***

Once the model is trained, it is evaluated against unseen data. Difference performance metrics evaluate the trained model such as accuracy, loss, area under the curve, etc. Based on the results, various parameters (known as hyperparameters) are tuned to improve the performance of the trained algorithm.

5. ***Model Testing***

The last step utilizes the withheld data (known as test data) to test the model and make predictions for the problem under consideration. This step best approximates the practicality of the model to a real-world application.

2.5.1 Machine Learning Techniques

Various machine learning techniques exist in the literature that can be used to address different challenges of wireless networks. A brief description of these techniques is provided below.

- ***Supervised Learning***

A ML approach that aims to learn the relationship between input and output given that labelled data is utilized for learning. The output of the learning algorithm is compared with the intended output to determine the errors and this information is then utilized to improve the accuracy of the algorithm. This learning approach can be used either for classification (predicting a class label) or regression (predicting a numerical value). Support Vector Machines (SVM) and K- Nearest Neighbors are examples of supervised learning algorithms.

➢ ***Unsupervised Learning***

Unsupervised learning algorithms aim to learn the hidden functions/relations given that the data is not labelled. In this technique, the algorithm itself uncover the associations, similarities, and patterns. Among many examples, K-Means clustering algorithms and Principal Component Analysis (PCA) are the two popular learning techniques [109].

➢ ***Reinforced Learning***

The objective of Reinforced Learning (RL) is to optimize an objective function. An agent, a component of the system, is utilized for learning. The agents learn from their environment and perform the optimization.

➢ ***Neural Networks***

As the name suggests, Neural Networks are comprised of neurons, attached to their respective weights that pass through an activation function to provide an output. There can be several layers of this network, each layer can have a different number of neurons and different activation functions. To optimize the network different backpropagation techniques can be applied such as Adam etc. [110].

All of the above-mentioned learning techniques can be further divided into various algorithms. These are summarized in Figure. 2.4.

The problem discussed in this thesis falls under the category of classification and specifically, it is a binary classification problem (highlighted in Figure. 2.4). Binary classification refers to the problems that involve two classes. It assigns a data instance to one of the two classes. An example is classifying an email as "spam" or "not spam". In this thesis, binary classification is utilized to determine whether a node should be a part of the cluster or should be associated with the eNB. Various binary classifiers such as Decision Trees, Neural Networks and Supports Vector Machines are available and investigated in this thesis. Details can be found in Chapters 5 and 6.

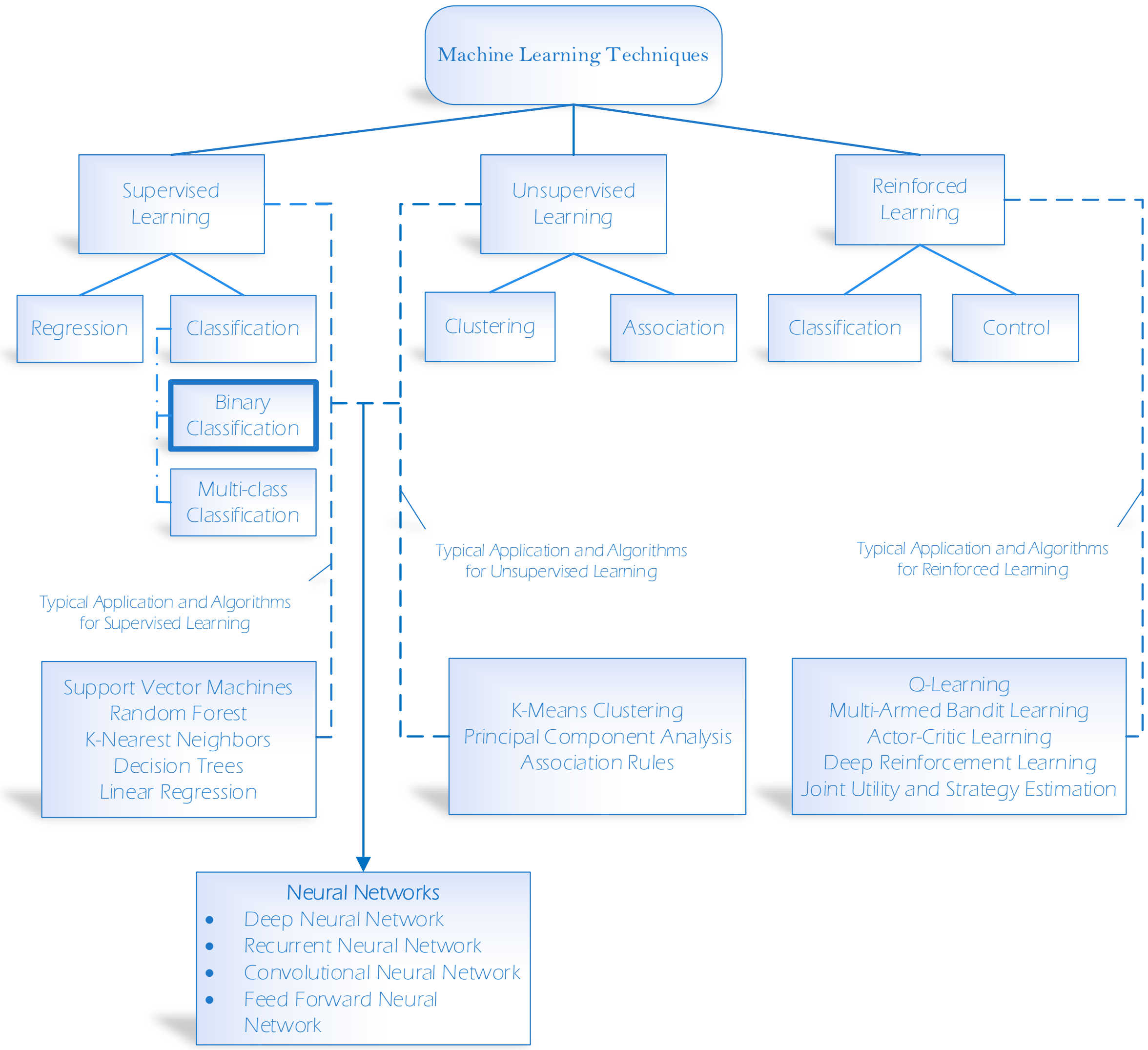

Figure. 2.4 Categorization of Various Machine Learning Algorithms. Neural Networks can be used for both Supervised and Unsupervised Learning. This study utilizes ML for Binary Classification.

2.6 Machine Learning for Wireless Networks

The last decade has seen numerous applications of ML for different fields of study. Similarly, ML has gained the attention of the wireless communications discipline as well. The reasons for utilizing ML for wireless network solutions are as follows:

- Conventional approaches present an undesirable burden on the central controllers owing to massive signaling/message passing between various entities of the cellular network.
- Simulation enables the researchers to create training sets that best approximate the real-world data.
- Explanations for various wireless solutions are not required rather a black box realization is sufficient.

ML has been utilized for providing solutions for wireless networks ranging from the physical layer to the application layer [111]. Some of these works focused on Wireless Sensors Networks (WSNs) [112, 113], Wireless Adhoc Networks (WANET) [114], Cognitive Radio (CR) [115], and IoT [116]. For instance, WSN employs machine learning techniques to adapt to dynamic environments. WSN utilizes machine learning to optimize energy consumption, scheduling, routing, and security, etc. [113]. Authors in [117] jointly consider cognitive radio and machine learning to investigate the complex spectrum requirements of the communication system. The authors propose a new model of anti-jamming, based on machine learning. On the other hand, [118] investigates the spectrum sharing problem. A multi-agent learning framework has been introduced to optimize the spectrum sharing process.

Considering the characteristics of future wireless networks, it is envisioned that ML will substantially improve network performance. This is possible by learning the real-time wireless environment.

Machine learning (ML) is applied to various problems in wireless networks [119]. To be able to run the network economically, the system needs to be self-aware and adaptive [120]. Conventional methods of network maintenance are not efficient any longer. However, machine learning allows the network to perform proactive and predictive maintenance. It has been reported in the literature that machine learning has found several applications in wireless networks that help the cellular operators to find unknown properties, identify anomalies in the network, identify correlation in the data generated by cellular networks that is impossible to be inspected otherwise, and based on these observations find novel ways to optimize the networks [121, 122].

The three important drivers of applying machine learning to cellular networks are cost & service, usage, and technology [121]. It is easy to optimize the network by applying machine learning algorithms trained offline which reduces the cost of network maintenance. On the other hand, cellular resources are not increasing proportionally to the traffic load, therefore it is important to manage the network traffic more efficiently. Recent research has proven that machine learning algorithms are efficient in regulating the network load [120-124]. Another important application of machine learning is that it helps balance the distributed and centralized functionality of the network [121, 122].

As we are moving towards the deployment of 5G, the focus has shifted towards developing the technologies necessary to realize the next generation cellular networks such as 6G. Besides the improvement in the current techniques, ML approaches have been recognized by many researchers as a potential tool to provide optimal solutions to the complexities of the 6G network [125]. It should be noted that ML algorithms not only deliver network optimization solutions, but they also profoundly change the architecture of the 6G network. Therefore, data storage and learning servers are now an essential part of the cellular network architecture [121, 122, 126-129].

Recent works have focused on ML applications of clustered cellular networks [130, 131]. 6G is envisioned to have various D2D use-cases and research suggests that many of these applications are better realized by applying clustering algorithms [3]. A few preliminary studies have shown that clustering is going to be an integral part of 6G [132]. A summary of various ML applications specific to cellular networks and clustering is presented in Figure. 2.5.

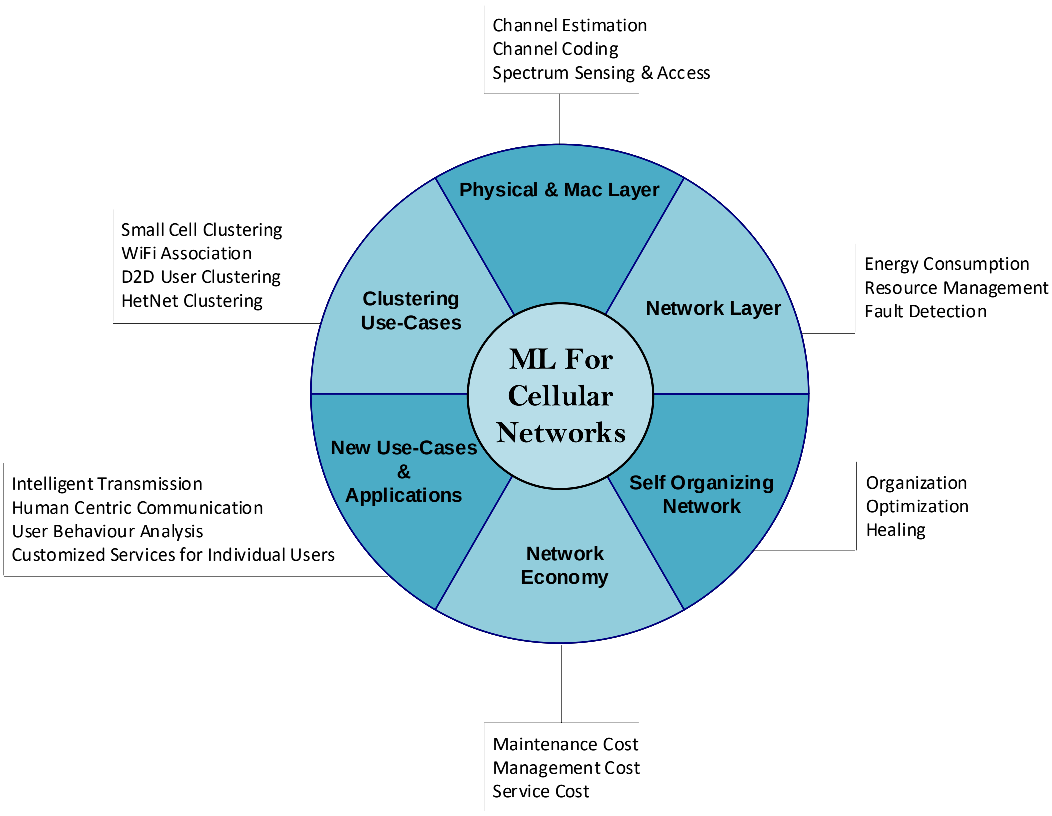

Figure. 2.5 Machine Learning Applications for Next Generation Cellular Networks.

While incorporating ML into wireless networks, two implementation mechanisms need to be considered namely, online and offline [122, 133]. Online learning considers the live network data whereas offline learning exploits a static dataset often referred to as batch learning. During offline learning, ML parameters are updated once the whole batch is consumed. On the other hand, in an online implementation, parameters are updated based on one training sequence.

Online ML algorithms are part of the networking algorithms or protocols whereas offline ML algorithms may be executed by a computing facility (remote or co-located) connected to network entities [122, 133]. Offline implementation is realized via offline learning servers that are now essentially part of the cellular network (more details can be found in Chapter 5). Offline ML is important due to the strict latency requirements of modern cellular networks.

Owing to the benefits of ML and offline learning, we utilize ML for the proposed work. It is synergistic to the other ML proposals as well. However, we are targeting a clustering application and proposing user segregation for performance improvement. Moreover, if we consider an online network, then solving this problem conventionally, without utilizing the ML, will present an impractical scenario, where the signalling required to set up an eNB assisted solution would be unrealistic. Therefore, offline training makes the scheme distributive and hence, realizable in a real-world scenario.

2.7 Research Objectives

While different research works have achieved promising results, based on the literature review, this thesis aims for the following research objectives:

RO. 1: Design clustering algorithm for content-sharing scenarios supported by a decentralized architecture.

1.1 Clustering based on parameters other than conventional 'node location'. Exploit the information acquired during the discovery phase and utilize it for cluster formation for a distributive mechanism.

1.2 Propose decentralized network architecture that can complement the proposed clustering algorithm supported by a content-identification technique.

1.3 Demonstrate the impact of social-interest on the throughput of the D2D clustered network.

1.4 Evaluate the clustering algorithm for various performance parameters such as throughput, energy consumption, area spectral efficiency, and throughput fairness.

1.5 Investigate the effect of forming a various number of clusters on performance parameters.

RO. 2: Optimize the clustering process by evaluating the effect of clustering on individual user's performance.

2.1 Utilize machine learning algorithms to perform user segregation. Determine the nodes that are better off without clusters.

2.2 Determine the effect of user segregation on various performance parameters.

2.3 Identify the data collection opportunities in the cellular networks to present a practical ML based solution.

2.4 Determine the machine learning algorithms that should be used for user segregation as several algorithms (or classifiers) exist in the literature.

2.5 Investigate the loading effects of user segregation.

2.6 Demonstrate that the user segregation improves the performance of the system irrespective of the clustering scheme applied i.e. improvement due to user segregation is independent of the clustering algorithm.

2.8 Research Methodology

Standard research methodology was utilized in this thesis that also conforms to the research work taking place in the field of cellular communications. The first phase of this research was dedicated to literature review. Literature review provided insights into the state-of-the-art schemes related to cellular architectures, clustering algorithms, optimization techniques, and performance parameters. The literature also steered the research towards MATLAB based simulation. In the second phase of the research, various classical and state-of-the-art clustering schemes were implemented for replication of the results presented in the literature. Lastly, the system model was developed, and the proposed clustering algorithm and user segregation schemes were thoroughly evaluated by exhaustive simulation. The proposed schemes were benchmarked against several algorithms and found to be superior.

CHAPTER 3

THE CLUSTERING ALGORITHM

The concept of clustering users in proximity sharing a common interest has been very popular for multicasting scenarios [20]. Typically, an intermediate node which is a CH fetches the content from the eNB and delivers it to several content requestors [51, 134, 135]. A similar multicasting scenario is considered in this chapter. The concept is illustrated in Figure. 3.1.

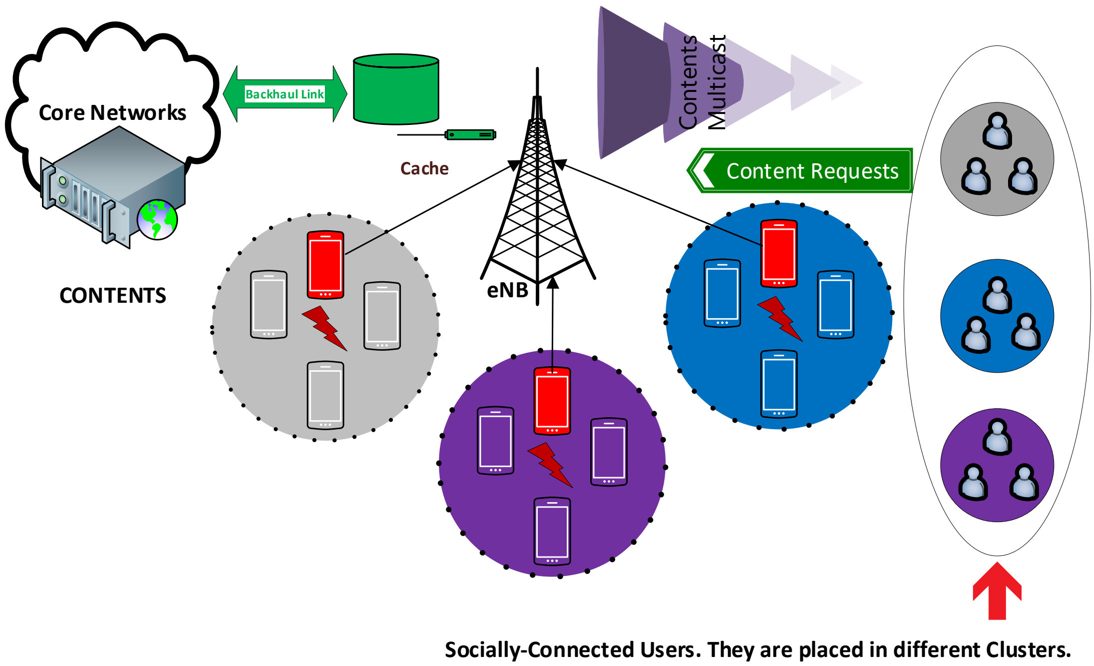

Figure. 3.1 A typical clustering scenario. A few users sharing a common interest are going to be served in clusters. Red devices represent CHs, which fetch data from the eNB and serve the cluster members.

To realize clustering that supports content-sharing via D2D, a suitable network architecture is necessary that not only conforms to the standards of future cellular networks but needs to be distributive as well. Moreover, it should be capable of handling a high user density. Therefore, we propose a decentralized architecture suitable for content-sharing. The proposed architecture is suited for 5G and beyond. As discussed, this architecture utilizes Content-Centric Networking and Network Virtualization, the key technologies required to realize a 5G architecture [136-141]. The proposal also employs the same frame structure as employed in the published article on 5G [142-145].

Recent literature suggests that the geographical distribution of mobile users plays a vital role in successful content caching [146]. Moreover, to ensure nearby availability of content, caching at a D2D device should consider social ties and request patterns. Therefore, it is important to identify users having common social characteristics.

In this study, the proposed decentralized architecture is supported by simple hash-based functions that have been previously used in multimedia broadcast networks for identifying users with a common interest. These users are then organized in clusters. The clustering takes place based on the multi-factor algorithm considering proximity, channel gain, and channel variance, detailed in Section 3.3. The clustering algorithm is optimized using fuzzy optimization which is useful in optimizing clustering as well as other parameters of a cellular network [147-150]. Once the clustering takes place, CHs are responsible for multicasting the required information to their cluster members. The summary of the proposed research is presented in Figure. 3.2.

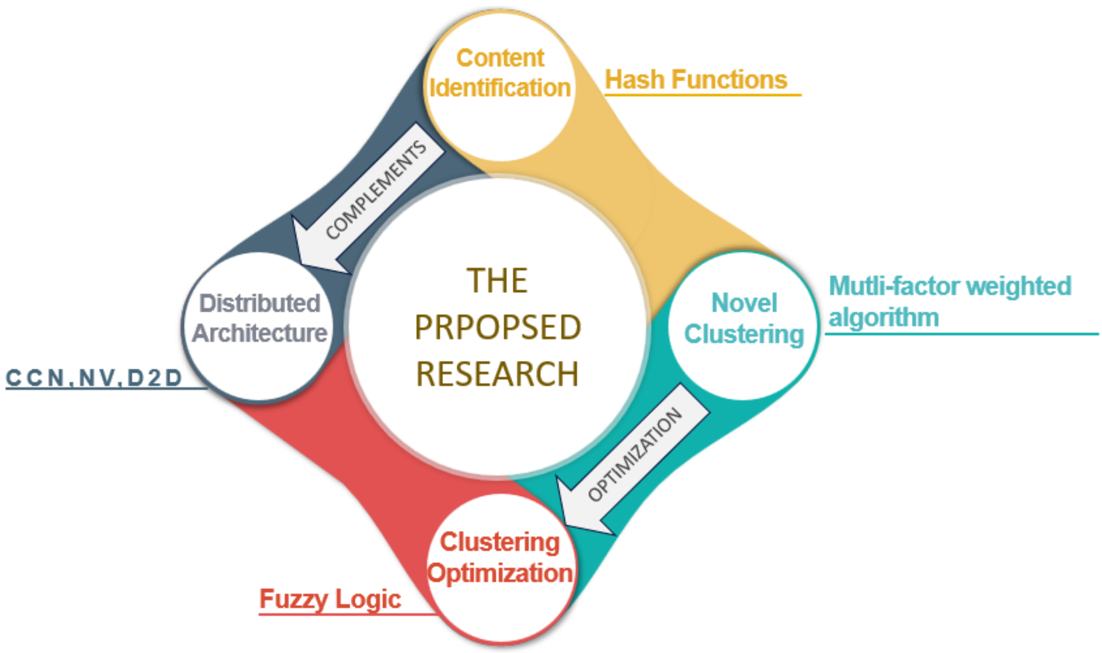

Figure. 3.2 Summary of the Proposed Mechanism. The four critical aspects are: distributed architecture implementation using Content-Centric Networking (CCN) and Network Virtualization (NV), identification of users with common interest using hash functions, cluster formation with the proposed multi-factor algorithm, and optimizing the clustering algorithm using fuzzy logic.

3.1 The Distributed Architecture

The concept of CCN is predicated on the requested content reaching the requester without needing to reach the content publisher/provider [63]. Therefore, caching the requested content at an intermediate node will enable content delivery with reduced energy consumptions and latency. Once the intermediate node has cached the content, it can be provided to several requesters. Architectures supporting CCN have been proposed in the literature [61, 63, 151, 152]. However, these works consider centralized mechanisms.

The architectures presented in [61, 63] involve eNB, and significant signaling is required to take place between eNB and the D2D nodes, before the content delivery. On the other hand, [151] does not provide any details on the architecture. The work presented in [152] does consider decentralized mechanisms, but it does not explicitly show any architecture that supports their mechanism. Our research considers a similar approach as presented in [61, 63] with necessary modifications to accommodate the clustering of users for content-sharing scenarios and making the scheme distributive. Figure. 3.3 shows the network model of the proposed content-centric architecture. It is different from a conventional wireless network such as Internet Protocol (IP).

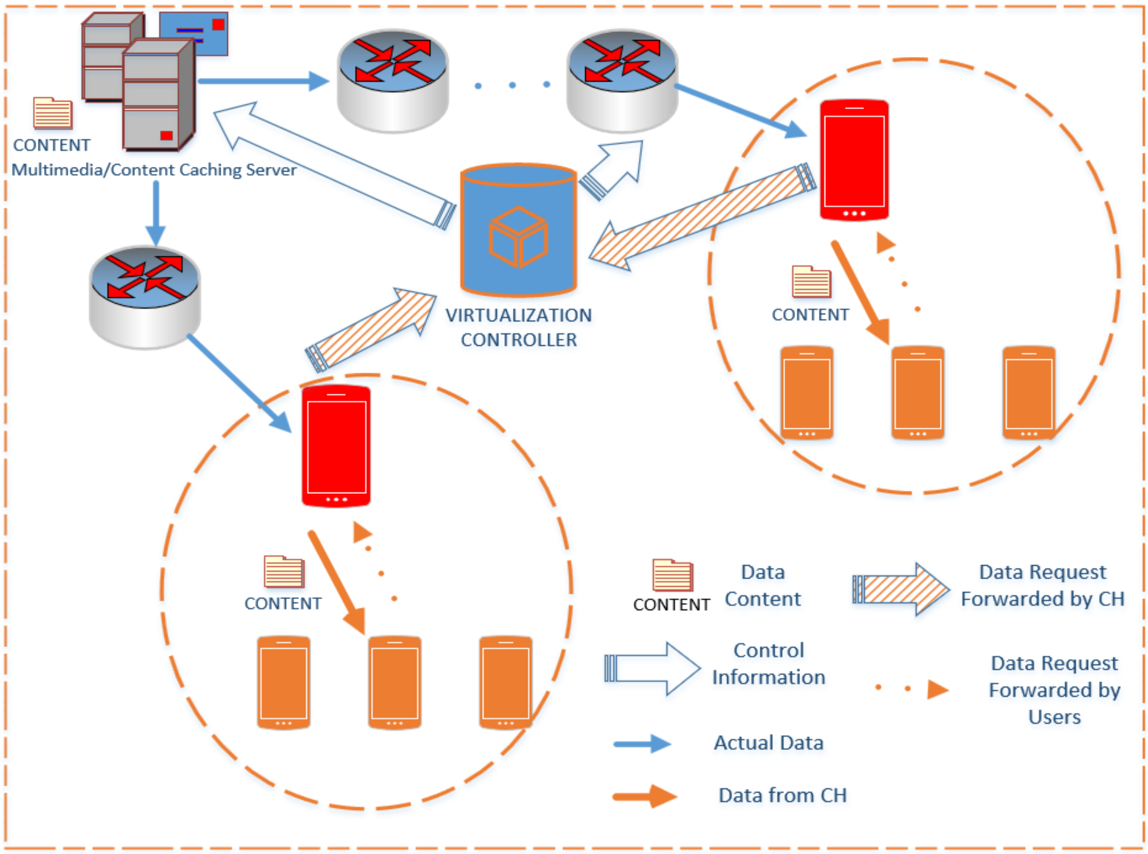

Figure. 3.3 The proposed distributed network architecture. Each red device represents a cluster head that forwards the requested content to the virtualization controller that connects with the multimedia servers to fetch the contents.

The basic difference lies in the establishment of the connection. IP based networks first establish the connection between the requestor and the provider before the content is delivered. In contrast, the content is requested without the establishment of the connection with the host/content provider in CCN. The proposed architecture utilizes CCN as well as network virtualization. The controller for virtualization, shown in Figure. 3.3, is responsible for providing the location of the content-holder as well as setting the virtual infrastructure components for the content delivery. With the help of virtualization, infrastructure and radio resources such as spectrum resource, RAN and core networks, etc., can be sliced into the virtual network resources shared by various cellular networks [61]. This provides operational efficiency, extra resources, and ease in delivering the contents for all the networks involved in virtualization. In the proposed merger of CCN and NV, virtualization controller (VC) is significant for efficient content delivery. The VC is responsible for managing and customizing the sliced network resources and most importantly providing programmability of the virtual resources [61]. VC decouples the data and control plane so that network operators can customize the sliced resources according to their requirements. Scheduling and forwarding of contents can be customized by VC as well. It is important to note that VC is not centralized in its operation since it controls several cellular networks participating in virtualization. Therefore, the architecture proposed in Figure. 3.3 is independent of eNB and depends on virtualization controller having distributed functionality.

Another important entity of the proposed architecture is the caching server. It is an integral part of the network which caches popular contents and reduces duplicate transmissions.

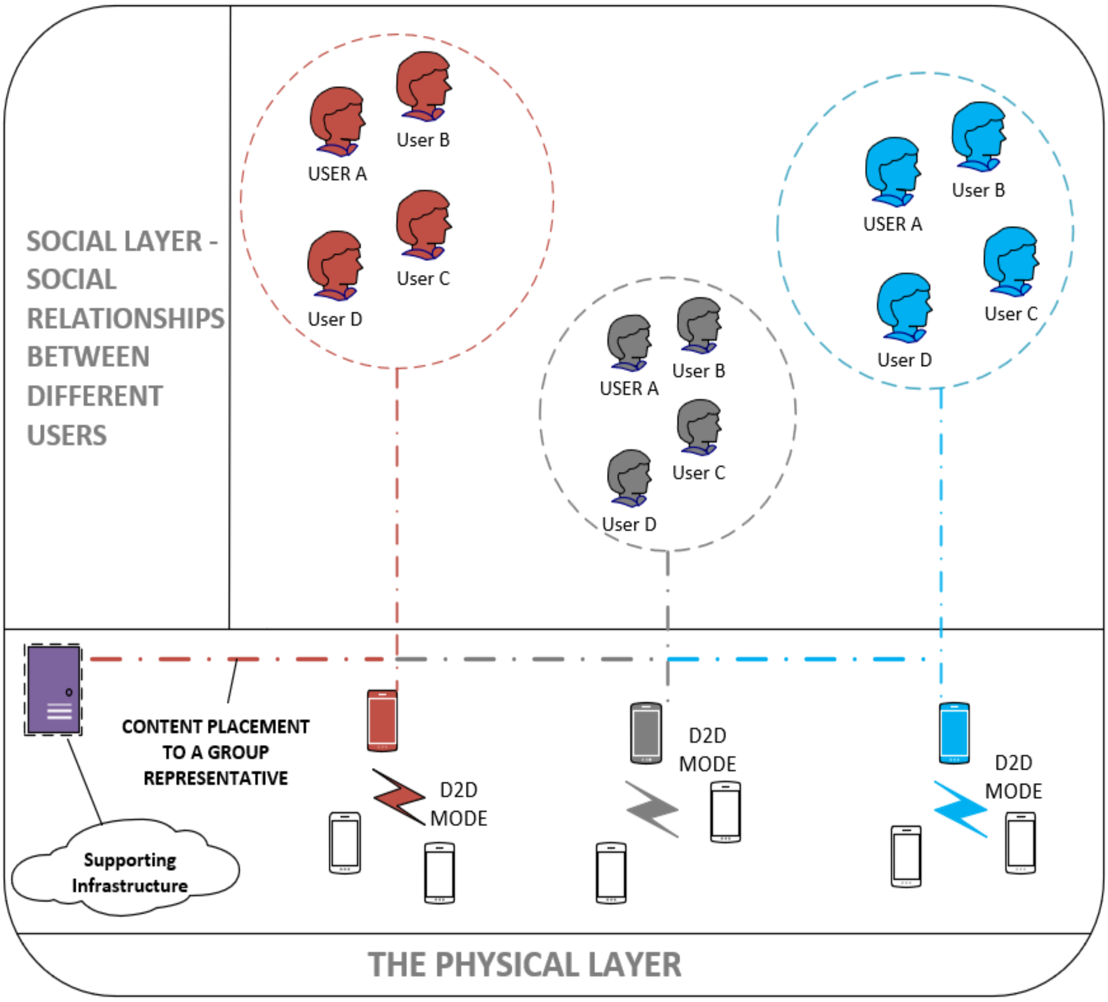

Figure. 3.4 The visualization of a layered network showing the interaction between social users and the corresponding physical layer.

The proposed architecture relies on CCN. Therefore, content can be delivered from any network location/device. Most elements in a CNN are assigned hash spaces where contents can be identified and stored. When a user requests a particular content, the request goes to the VC via a CH (please see Figure 3.3). The VC determines both the content location as well as the routing process. The content request is sent to the router, and if the requested content is cached in the corresponding router, the content is delivered to the user via CH, otherwise, the request is forwarded to the content/multimedia server.

Figure. 3.4 shows the layered network. The first layer, termed as a social layer, represents the social ties that exist among different groups of users. The physical layer represents the mobile devices that model the communication taking place in various clusters represented by a CH. It also shows the supporting infrastructure required to set up the communication and making content delivery possible.

3.2 Content Identification Using Hash Functions

One of the important aspects of the proposed architecture is the mechanism that addresses the content-identification. When several users request similar content, they need to be identified so that they can be part of the same group. We propose to utilize Hash Functions [153] for the content-identification.

Hash functions perform the mapping between the given data and hash of a specific length. The size/length of the output of a hash function does not depend on the length of the input. Hash can be regarded as a 'signature' for a given text [153-155]. The output binary sequence is termed as 'Hash' (Figure. 3.5). One of the major applications of hash functions lies in the field of multimedia broadcast networks, as a content identifier [153-155]. The hash function aids the network by providing the content identification to easily determine which content has been broadcasted, timing information, and to what station. Several hashing algorithms exist in the literature; we suggest using SHA-256 due to its reduced complexity and speed [156, 157].

Since this work is considering content-centric networks, it is important to name the contents rather than naming the network devices or hosts. For example, in a typical Named-Data Network, all the data/content is named and 'digitally signed' by the publisher of the content. The binary sequence generated by the hash functions for a particular 'text' or 'name' will always be the same.

Therefore, if we produce the hash of various contents at the content servers, then the hash value can be matched with the one generated by the content requestors. If the hash value matches, it means that the same content is being demanded. Moreover, based on the hash values, groups can be identified requesting the same content. It is clear that hashing will not only help in identifying the content but also the group of users sharing the same interest. Therefore, we believe, that hashing is a natural choice.

To further elaborate the concept, let us consider a scenario where a group of users is interested in a certain 'music video'. Based on the generated hash, users sharing the same interest are identified and the request is forwarded to the CH, the intermediate node. CH only uses the hash value to transmit it to the content-server that exists in the proposed architecture, given in Figure. 3.3. When such a match is found in the content server, the CH receives the content and broadcast it to the content requestors/users in a group. It should be noted that the content publishers/content servers do not need to share the details of the contents to the intermediate nodes or CH rather, only the information needs to be forwarded.

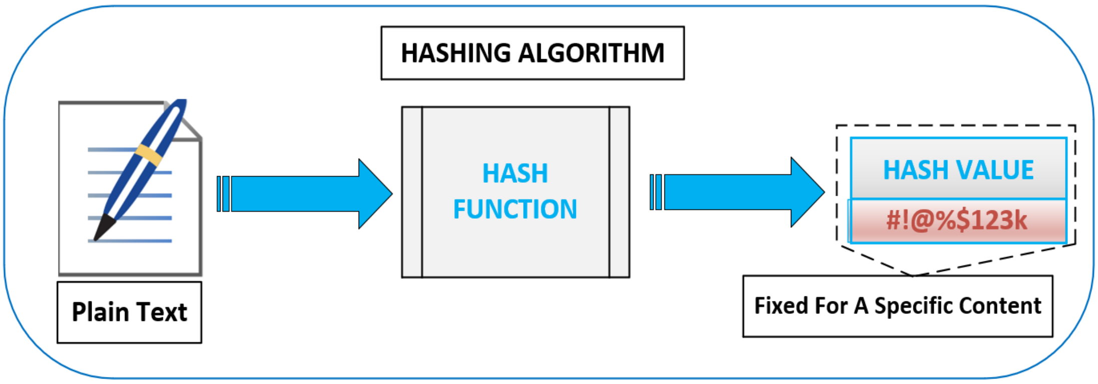

Figure. 3.5 The Hash Function: Plain Text to Hash Value.

3.3 Clustering Mechanism

Clustering commences once users demanding the same content have been identified. The user clustering process consists of three main steps: the selection of appropriate clustering metrics, identification of the devices suitable for being a CH, and finally, associating the cluster members with their respective CHs. The overall clustering process is shown in the flow chart of Figure. 3.6.

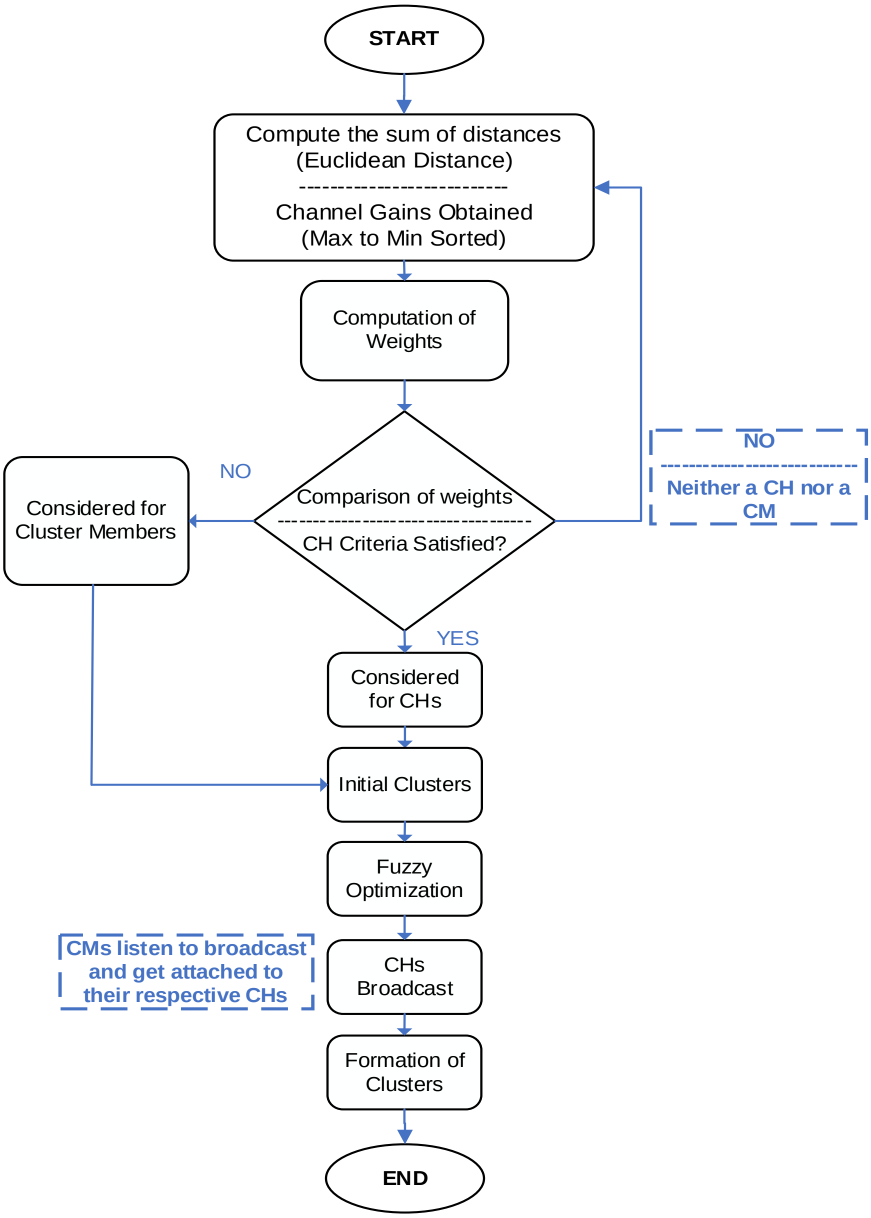

Figure. 3.6 Flow Chart of the Clustering Algorithm.

All the information required for forming clusters can be obtained via D2D discovery. This information is obtained and utilized to form clusters without the involvement of eNB or any other centralized controller. Therefore, the clustering algorithm can be characterized as decentralized.

It has been reported in the literature that distance and channel conditions can be obtained via the D2D discovery process [158, 159]. Each device/user runs the clustering algorithm, summarized in the flow chart of Figure. 3.6.

Before communication takes place, all the devices need to be discovered. Each device maintains a neighborhood table which is obtained by sending peer discovery signals [160]. This signal may contain the positioning information and device ID. Once the discovery signal is successfully received by other devices, a response signal is sent back with all the necessary information like channel conditions and distance to the user of interest. Therefore, all the devices after discovery possess the necessary information of clustering metrics. Each device calculates its weight and broadcast to its neighbors. Once the signals from multiple devices are received, the devices with the best weights broadcast cluster joining requests, and all the devices that receive this request, get associated to that CH that serves them the best, and thus clusters are formed. This process of cluster formation is solely dependent on users and independent of the eNB or any other central controller.

3.3.1 Weighted Clustering Approach

Cluster heads are selected on a per-frame basis. The duration of one frame is 10 ms following the relevant literature. All the users are considered CHs for the first frame; therefore, the clustering algorithm is implemented for the next frames. It is assumed that every node is capable of being a CH and has enough energy [94]. The position of the nodes remains the same during one frame.

However, for the next frame user distribution/placement of users change and, therefore, every simulation represents a different user distribution. This is following the standard literature relevant to multimedia multicasting scenarios [62, 94]. If the users are mobile, mobility will need to be accounted for with a clustering metric since CH needs to be in the cluster to serve its members. If the CH moves too fast, the stability of the cluster will be affected. Moreover, there will be frequent re-selection of CHs that will increase the computational complexity.

After the initialization, the algorithm gathers the information about the clustering metrics, and clusters are formed, details of which can be found in the subsequent sections. Before the clustering takes place, the distance among the devices and the channel conditions are obtained and conveyed to all the neighbors during the discovery phase as explained in the next subsection. Based on the information received, CHs announce its cluster members. It is assumed that multiple users can be detected simultaneously by the CH. All the members listen to the broadcast of the CHs and get attached to the one that serves them the best considering distance and channel conditions.

3.3.2 Device Discovery

Though the device discovery is out of the scope of this research, we utilized the information obtained through the device discovery. Therefore, device discovery is described within the context of the proposed algorithm.

Before the clustering takes place, it is necessary to discover the devices and create a neighbor list. It is assumed, as is common practice in the literature [48, 161-164], that the neighbor list is available with the nodes. For these tasks, we propose to use the Peer Discovery Resource (PDR). PDR represents a resource unit, used to transmit the discovery signal or beacon signal. Two of the standard PDR structures that are used in published literature are LTE-A and FlashlinQ. The PDR structures are shown in Figure. 3.7. Literature suggests that a considerable amount of information can be conveyed using either of these structures [47, 48, 163]. Moreover, different research works have utilized PDR to send clustering-related information [162, 163]. We propose to utilize the same concept and use the PDR to send the information regarding the predefined clustering metric detailed in the subsequent section. Therefore, the signaling load for the proposed clustering scheme will be accommodated by standard signaling taking place in a D2D network.

The users in proximity to one another receive the discovery signals. Devices decode this signal containing information such as device or user ID and its link characteristics (such as SINR, channel conditions) with the user. Based on these characteristics, a device decides which of the users whose signal it has received can be classified as neighbors. There are various advanced channel estimation algorithms and processes for 5G networks [164] that can be utilized for this purpose. Once the neighbor detection has taken place, every user possesses a list of its neighbors.

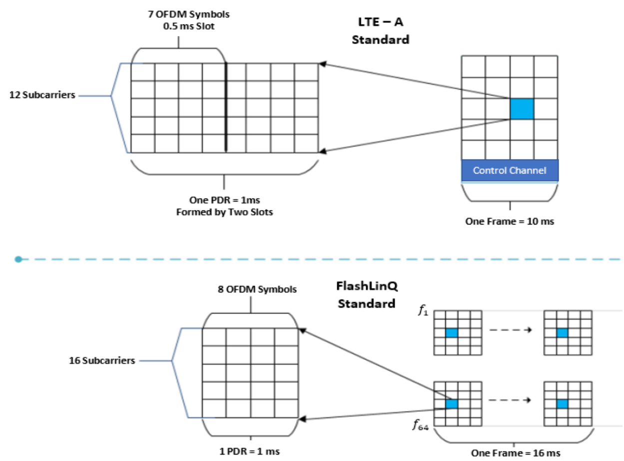

Figure. 3.7 The Two Standard PDR Structures.

3.3.3 Clustering Metrics

The selection of clustering metrics significantly impacts the system's performance. Therefore, the metrics selected for clustering the users are based on performance parameters. This work presents two different sets of clustering metrics that demonstrate the trade-off among various performance parameters. To target aggregate throughput, ASE and energy consumption, distance, and channel conditions among the users are considered. However, it was noticed that recent clustering algorithms have not considered fairness. The literature suggests that fairness is crucial for evaluating a cellular network [165, 166]. Therefore, we extended the proposed algorithm to suit the throughput fairness as well. To target fairness, we introduced another clustering metric—the variance among the channel conditions of the users. All these metrics are attached to their respective weight. The details of the clustering metrics are provided in the following text.

1. ***The Distance Among the Nodes***

Recent literature has identified the significance of the spatial distribution of users as it directly influences the caching efficiency [146]. Furthermore, there is a high probability of successful D2D transmission if the devices are in proximity [62]. Hence, we chose distance among the nodes as an important metric for forming appropriate clusters. It is also important since users that do not exist in proximity are not ideal candidates for being a part of the same cluster even with a strong social relationship.

2. ***Channel Conditions***

Since we are considering a multicasting scenario where a CH will be communicating with the rest of the cluster members, the cluster members need to have a good link with the CH. If we ignore these conditions, both inter-cluster and intra-cluster communication might be impaired. Therefore, we believe that the channel condition between the prospective CH and its cluster members is an important metric.

3. *Variance Among Channel Conditions*

Considering only absolute channel conditions might disregard many users having unfavorable channel conditions. Therefore, in that case, there would be a significant difference in the throughputs of the individual users. Owing to this reason, we introduced variance of channel conditions, and our assumption is validated by the results demonstrating throughput fairness, shown later in this section.

Moreover, since we are considering a multicasting scenario, we are looking for approximately similar channel conditions with each node. The reason being, in a multicasting scenario, if each device receives the transmission at significantly different rates, then the complexity of the system would increase and might become infeasible [18].

Let us take an example of a scenario where a video stream needs to be broadcasted to a group of users. Assume that there are a few users with higher rates as compared to the others. Since we are considering broadcasting, the maximum achievable rates are determined by the worst physical link in the group. Therefore, users having higher rates will face long delays waiting for the other users to catch up. This reason makes it even more significant to have variance in channel conditions as an important factor in addition to just the absolute channel conditions. Though, owing to the degradation in the throughput performance, variance in channel conditions cannot be selected as the sole criterion. Therefore, the clustering algorithm considers all three different metrics which are attached to their respective weights.

3.3.4 Cluster Head Selection

During the CH selection process, devices use PDR to broadcast beacons continuously. These beacons include predefined metrics (e.g., distance, channel conditions). Every device decoding the beacon stores the corresponding metric and its identifiers (as per the details given in Section 3.3.2). This information is vital for the devices to select a CH and delegating the control to it for further communication. If a certain device is not able to receive a beacon signal, it might be out of reach of another device, and it can self-select itself as CH. Once the metric information is received from the beacons, all the devices compare their metrics. The devices with the lowest metric values are identified as CHs. The remaining becomes the cluster members. It should be noted here that all the users need to fulfill the predefined criterion to be considered for clustering. It is based on the social relationship among the devices. The following steps summarize the proposed algorithm.

Step 1: Determine the neighbors of each node using D2D discovery. Parameters of interest are stored.

Step 2: Determine the nodes sharing the same interest/content using the hash function.

Step 3: Compute the sum of distances (Euclidean Distance) for all the nodes against all their neighbors.

$$D_{(a,b)} = \sqrt{({a_X}^2 - {b_X}^2) + ({a_Y}^2 - {b_Y}^2)} \tag{1}$$

where a and b represent any two neighboring devices.

Step 4: Weights of the nodes are accumulated as follows:

$$W_T = I.\left[w_1 * D_{(a,b)} + w_2 * (\frac{1}{h_{ab}})\right] \tag{2}$$

w_1, w_2 represents the weights given to distance, channel gains, respectively. h_{ab} represents the channel gain between the nodes a and b. The weights represented in Equation (2) are such that $\sum_{f=1}^{2} w_f = 1$. The node with the minimum W_T is chosen as the CH.

Since only nodes sharing a common interest should be considered for clustering, the total weight is being multiplied with a binary interest-factor denoted by "I", so that if $z = w_1 * D_{(a,b)} + w_2 * (\frac{1}{h_{ab}})$, then,

$$W_T = \begin{Bmatrix} z, & I \neq 0 \\ 0, & I = 0 \end{Bmatrix} \tag{3}$$

In Equation (3), I represent the interest factor. Since clusters are formed only for those users that are interested in sharing a specific content, I should be non-zero for a node to be considered for clustering.

Step 5: Compare the weights for each node and select the cluster head corresponding to the smallest W_T.

Step 6: For the remaining devices, repeat steps 3 and 4, until each node is either selected as a CH or a CM.

Step 7: Clustering optimization using Fuzzy.

It should be noted that only two clustering metrics (distance and channel conditions) have been considered in the above-mentioned cluster head selection process. These two metrics target throughput, ASE, and energy consumption. We can add the third metric "channel variance" to target throughput fairness. However, the algorithm remains the same, as represented in Figure. 3.6. The introduction of variance among channel conditions modifies Equation (2), to the following:

$$W_T = I.\left[w_1 * D_{(a,b)} + w_2 * (\frac{1}{h_{ab}}) + w_3 * var(h)\right] \quad (4)$$

w_3 represents the weight given to variance of the channel gains. The term "$Var(h)$" represents the variance among the channel conditions of the users. The weights represented in Equation (4) are such that $\sum_{f=1}^{3} w_f = 1$. The node with the minimum W_T is chosen as the CH.

3.3.5 Fuzzy Optimization of Clustering

The initial clusters formed based on the proposed algorithm need to be optimized. Therefore, a fuzzy optimization technique was applied. Fuzzy optimization partitions the users into C clusters based on the proposed criterion of clustering. Each input to this function is attached to an attribute such as the weights in our study (i.e., w_1, w_2, w_3 of Equation (4)). Fuzzy optimization aims at minimizing the objective functions given in Equation (6). In fuzzy optimization, the membership of each user is spread among all clusters. One of the advantages of this technique is that it handles the outliers effectively. Therefore, outliers do not influence the clustering decisions.

The fuzzy optimization initializes with the cluster heads obtained via the proposed algorithm. Hence, the fuzzy optimization algorithm converges faster as compared to conventional cases. After the initialization of CHs, the membership matrix (presented in Equation (5)) is calculated, and new clusters are formed. Finally, the absolute difference between two consecutive membership matrices is calculated to check the condition for convergence. This optimization is an iterative process where the steps mentioned above are repeated until the condition of convergence is achieved. The process stops when the clusters stabilize i.e. the clusters from the previous iteration are similar to those obtained in the current iteration, conforming to the given error threshold (∊). In this optimization, we have selected ∊ ≤ 0.01.

Fuzzy optimization is based on a membership matrix P where membership degree, p_{ik} ∈ {0,1}, of user i to cluster k, is defined as follows [167, 168];

$$\sum_{k=1}^{C} p_{ik} = 1\,;0 \leq\ p_{ik} \leq 1 \tag{5}$$

The objective function of Fuzzy Optimization is given by Equation (6) [167, 168]:

$$OF\ (P, V) = \sum_{i=1}^{N} \sum\nolimits_{k=1}^{C} p_{ik}\ \|X_i - V_k\|^2 \tag{6}$$

Equation (6) represents the objective function where C is the number of clusters and V is the set of cluster centers. N represents the number of samples (users in our case) and X_i is the i^{th} calculated sample where $\|.\|^2$represents the Euclidean norm, and p_{ik} denotes the membership of X_i to cluster k. Each element of the partition matrix is a measure of the extent to which a particular user belongs to a certain cluster. The complete optimization process is explained in the flow chart of Figure. 3.8.

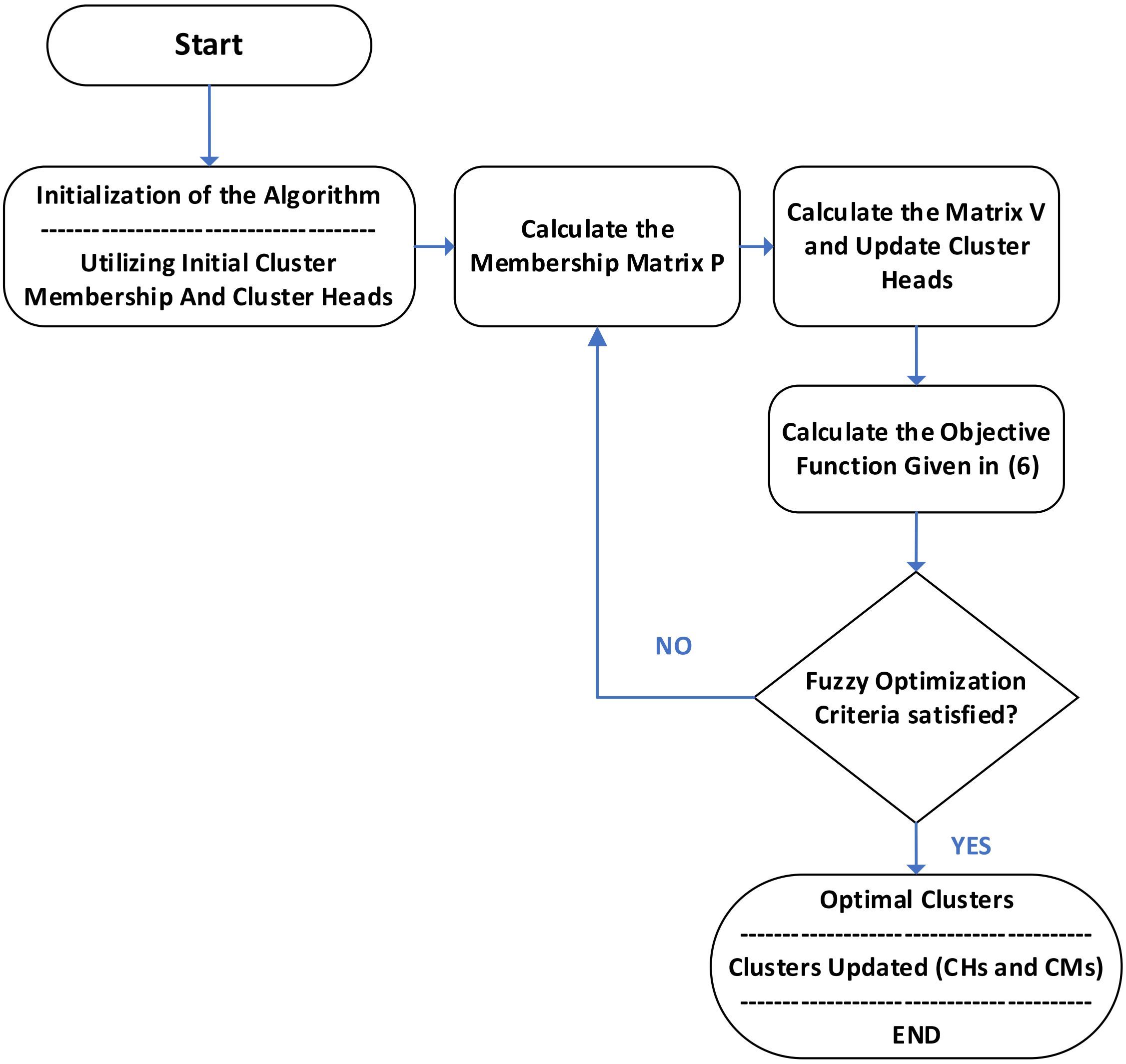

Figure. 3.8 The Optimization Flow Chart.

3.3.6 Communication

After the selection of CHs, they broadcast a message containing their IDs. They can use the same PDR used for neighbor detection to broadcast the results once the broadcast is received, and all the non-CH devices select those CHs to which they are closest and receive better channel conditions. The cluster members then associate themselves with a certain cluster, and the formation of the clusters is complete. The above-mentioned procedure is completely decentralized which is very important for dense networks. Once the clusters are formed, all the cluster members communicate via the CH. The operating phases of the proposed algorithm are shown in Figure. 3.9. The next frame follows the same activities.

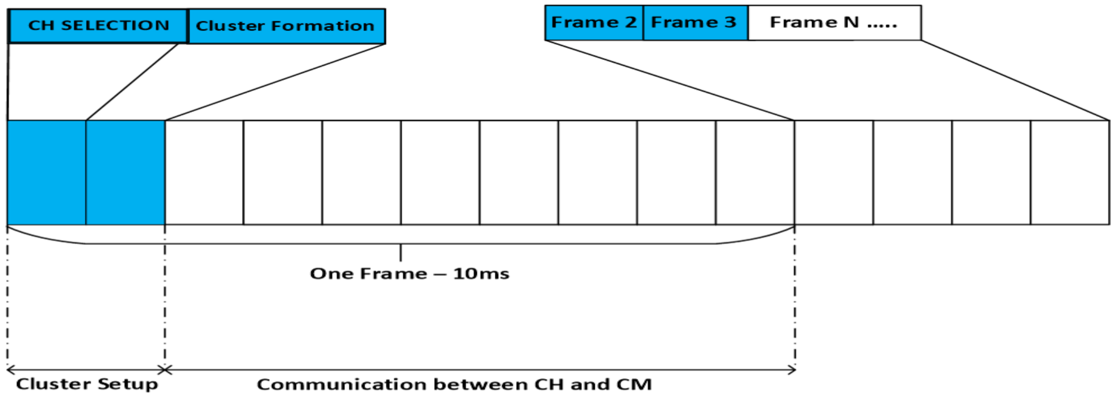

Figure. 3.9 Frame Structure for Clustering.

3.4 System Model

We consider a single cell where users are randomly distributed. In-band D2D communication using the underlaying concept is considered. In this case, D2D reuses cellular resources. These techniques are well researched [94, 169, 170]. The reason for considering the underlaying concept is that reutilizing the resources improve spectral efficiency. However, it creates interference and, therefore, was considered in the simulation scenario. Conventionally, eNB provides the requested content; however, it comes at the expense of increased energy consumption and usage [169, 170]. In contrast, the CH is responsible for delivering the contents to the requestors as depicted in Figure. 3.3. Once the data has been fetched by the CH, the requested content is distributed utilizing the D2D multicast communication.

3.5 Mathematical Models for Performance Parameters

3.5.1 Achievable Rates for Cluster Head and Cluster Members

There are total N users in the network which constitute the set $N = \{m_{1,}\, m_2, m_{3- - -,} m_N\}$. The CHs and cluster members are indexed as j and k, respectively. For clarity, all the other symbols are summarized in Table 3.1.

Table 3.1 List of Symbols

Symbol	Representation
$\boldsymbol{N}$	Set comprises of all the users
$\boldsymbol{k}$	Index of cluster member
$\boldsymbol{CH}$	Cluster Head
$\boldsymbol{R_{CH_j}}$	Achievable Rate of CH_j when receiving the contents from the Base Station (BS)
$\boldsymbol{SNR_{CH_j}}$	Signal-to-Noise Ratio of a CH_j
$\boldsymbol{N_o}$	Noise Spectral Density
$\boldsymbol{B}$	Bandwidth of the Transmission Channel
$\boldsymbol{h_{BS,CH_j}}$	Channel Gain between the BS and the CH_j
$\boldsymbol{P_B}$	Transmit Power of the BS
$\boldsymbol{R_{m_k}}$	Achievable Rate of cluster member m_k
$\boldsymbol{h_{m_k,CH_j}}$	Channel Gain between the cluster member m_k and CH_j
$\boldsymbol{P_{CH_j}}$	Transmit Power of the CH_j
$\boldsymbol{F_S}$	File Size (size of the demanded content)
$\boldsymbol{P_{chrx}}$	Power consumed by the CH to receive the contents from BS
$\boldsymbol{P_{mrx}}$	Power consumed by the cluster member to receive the content from cluster head (CH)

The achievable rate at the CH can be written as:

$$R_{CH_j} = B\, log_2\left(1 + SNR_{CH_j}\right) \tag{7}$$

where the SNR of the CH_j is given by:

$$SNR_{CH_j} = \frac{P_B \; h_{BS,CH_j}}{N_o \; B} \qquad (8)$$

Therefore, we may write Equation (7) as:

$$R_{CH_j} = B \log_2 \left(1 + \frac{P_B \; h_{BS,CH_j}}{N_o B} \right) \qquad (9)$$

Since we are considering a multicasting scenario, the achievable rate depends on the worst physical link. Otherwise, the successful reception of the content for all cluster members cannot be made certain. Therefore, the achievable rate at the cluster member m_k can be written as follow:

$$R_{m_k} = B \log_2 \left(1 + \frac{P_{CH_j} h_{m_k,CH_j}}{N_o \; B} \right) \qquad (10)$$

It should be noted that R_{m_k} is the minimum achievable rate to make sure that all the cluster members receive the content.

3.5.2 Energy Model

Downlink energy consumption is considered in this study. We utilized the energy consumption model presented in [28]. We assumed that the content demanded by the users is a file of size "F_S" $bits$. Suppose this file needs to be transmitted from CH_j to cluster member m_k with an achievable data rate of R_{m_k}. The time required to transmit this file is $\left(\frac{F_S}{R_{m_k}}\right)$ $seconds$. Therefore, energy consumption E_C in one of the clusters "C" can be written as:

$$E_c = \frac{F_S P_{chrx}}{R_{CH_j}} + \frac{F_S P_{CH_j}}{R_{m_k}} + \sum_{\substack{j \neq m \\ \forall \mathrm{m}}} \frac{F_S P_{mrx}}{R_{m_k}} \qquad (11)$$

Equation (11) represents the sum of three independent terms. Energy consumption of CH to receive data is represented by the first term, whereas the second term represents the energy consumed by the CH to transmit the data to their cluster members. The sum of the energy consumed by the cluster members to receive the demanded content is shown by the last term in Equation (11).

3.6 The Complexity of the Proposed Clustering Algorithm

The computational complexity is an important parameter to determine the applicability of the algorithm. Generally, an algorithm's complexity is defined by the worst-case running time, represented by the Big-O notation. We use the same notation to describe the complexity of the proposed clustering algorithm. It has been described in section 3.3.2 that the signaling required to implement the proposed clustering algorithm can be accommodated by the standard D2D signaling (discovery and communication). However, we describe the computational complexity of the proposed algorithm and compare the execution time of each of the implemented algorithms. The proposed clustering algorithm is summarized in the flow chart of Figure. 3.6. To determine the overall complexity of the algorithm, the complexity of each step of Figure. 3.6 is analyzed next.

1 – Computation of Sum of Distances (Euclidean Distance)

Complexity: $O(n)$

2 – Sorting of the Channel Gains

Sorting Algorithm: Heapsort

Complexity: $O(n \, log \, (n))$

3 – Comparison of Weights

Complexity: $O(n)$

4 – Initial Clusters Formation/ Communicating Information for Joining Clusters or CH selection

It is assumed that T_{Step} represent the time taken by the user to gather information about its neighbours. Hence, it also means that T_{Step} is the minimum time required before a user decides to join a cluster or announce itself as a CH. It is already mentioned in section 3.3.1 that multiple users can be detected simultaneously by the CH and all the members can get attached to the one CH that serves them the best (as per the defined clustering metrics). Therefore, in case, the user is a CH, a message needs to be broadcasted to its neighbours, which will take one T_{Step}. Same is the case if the user decides to join a cluster as a member. Since the cluster heads are selected on a per-frame basis, the complexity of the formation of the initial cluster is $O(1)$

5 – Optimization Using FCM

The objective function of FCM is given in Equation (6). It shows for the optimization, we need to communicate information such as initial cluster centers, membership matrix, and the number of clusters to be formed. Accordingly, the complexity of optimization is given as follows.

Complexity: $O(nn_C^2 n_f n_i)$

n = total number of nodes/users

n_C = number of clusters ($number\ of\ clusters\ are\ variable, details\ can\ be\ found\ in$ 4.4)

$n_f = number\ of\ features\ attached\ (two\ or\ three, please\ see\ Equations\ 2\ \&\ 4)$,

n_i = number of iterations

Overall Complexity:

$O(n) + O(nlogn) + O(n) + O(1) + O(nn_C^2 n_f n_i) = O(nn_C^2 n_f n_i)$

$\because \ O(nn_c^2 n_f n_i) > O(nlogn) > O(n) > O(1)$

3.6.1 The Execution Time

The execution time represents the time taken to form clusters by each of the implemented algorithms (benchmarked as well as the proposed clustering algorithm). The simulation setup and parameters utilize to implement these algorithms are summarized in Chapter 4 (Section 4.1). The experiments were performed on a 64-bit Intel 4600 GPU, Core i7-4790 CPU @ 3.60 - 4 GHz processor having a 12GB RAM. MATLAB was used for the implementation. The execution times for all the implemented algorithms are described in the following table.

Table 3.2 The Execution Times of Implemented Clustering Algorithms

Clustering Algorithm	Execution Time (sec)
K-Medoids	3.43755
FCM	6.12589
***Proposed in [86]**	13.08578
***Proposed in [171]**	9.11457
The Proposed Algorithm	11.13785

*These are the benchmarked algorithms considered in this study. The detail on these algorithms is given in Chapter 4 (Section 4.3).

3.7 Summary

Chapter three discusses the proposed distributed architecture and develops the clustering algorithm. The distributed architecture relies on the merger of the CCN and NV. The architecture is independent of the eNB and employs a VC. Moreover, hash functions are utilized to identify the contents. The proposed multi-factor clustering algorithm has been described in detailed as well. Three different clustering metrics are selected to facilitate a trade-off between throughput, energy consumption, area spectral efficiency, and throughput fairness. All the relevant details about collecting information for clustering metrics and optimization have been presented in this chapter as well. Finally, the complexity of the proposed and benchmarked algorithms is described and compared.

3.8 Related Publication

The work presented in this chapter has been published in the following research article:

S. Aslam, F. Alam, S. F. Hasan and M. A. Rashid, " A Novel Weighted Clustering Algorithm Supported by a Distributed Architecture for D2D Enabled Content-Centric Networks," *Sensors* 2020, *20*, 5509. https://doi.org/10.3390/s20195509.

URL: https://www.mdpi.com/1424-8220/20/19/5509

CHAPTER 4

PERFORMANCE EVALUATION OF THE CLUSTERING ALGORITHM

This chapter comprehensively evaluates the performance of the clustering algorithm proposed in the previous chapter.

4.1 Simulation Setup

The simulation environment was built on MATLAB. A single cell of 1 sq. km area was considered. For the conventional cellular communication scenario, the eNB was placed at the center of the cell. Moreover, it is important to mention that we explored the performance of a multimedia application (content). The packet size is 100 kB as suggested by relevant literature [172]. This simulation can easily be extended for any other multimedia application (e.g., online gaming, eHealth, etc.) by varying the file size and packet interarrival rates [172]. We selected the weights empirically, which can be adjusted according to the system requirements. The number of clusters formed to produce all the results were chosen using the Calinski–Harabasz criteria [173]. Various user densities have been considered to produce the results. The optimum number of clusters for various user densities are different, and hence, a specific number is not explicitly mentioned. All the simulation parameters of interest are detailed in Table 4.1. Parameters related to channel and energy consumption are adapted from the relevant literature [28],[88, 174, 175].

It should be noted that the results presented in Section 4.2 - 4.4 consider Equation (2) targeting throughput, energy consumption, and area spectral efficiency whereas Sections 4.5 & 4.6 evaluate the performance of the algorithm considering Equation (4), therefore targeting throughput fairness and demonstrating the trade-off between performance parameters.

4.2 Impact of Clustering and Social-Interest

The proposed study takes clustering and social interest into account, as both have a significant impact on the system. To demonstrate this impact, we consider three different scenarios. In the first scenario, conventional cellular communication takes place that does not involve D2D mode and clustering. The other two cases consider the proposed clustering algorithm, explained by the following text.

- ***Clustered D2D users with no interest factor***

In this case, we assume that users do not share a common interest i.e., all of them are not interested in a single file (content). Users demand files of various sizes varying from 10 to 100 kB in a random manner. Though this scenario does not consider the interest factor, we still cluster the users, as the literature suggests that even without the interest factor, clustering yields significant throughput gains [20, 62, 135, 145]. The clustering criteria for these nodes are the same as mentioned in Equation (2) except that the interest-factor "I" is not considered. The weights selected are as follows: $w_1 = 0.4, w_2 = 0.6$. These were empirically selected to maximize the throughput performance.

- ***Clustered D2D users with an interest factor***

The third scenario considers the interest factor i.e., all the users in a given cluster are interested in a single file of size 100 kbits. This emulates social gatherings such as a concert or a stadium, where there is a large gathering, interested in a similar video/content. This scenario was implemented using the proposed algorithm. The value of the two weights remains the same as discussed in the previous scenario.

Table 4.1 Simulation Parameters for Clustering Algorithm Implementation

Parameters	Value
Simulation Platform	MATLAB
Channel Model	Rayleigh Distributed
User Placement	Uniformly Distributed
Node Density	100 to 1000
Cluster Size	Variable
Path Loss for DUEs	2.5
Path Loss for CUEs	3.5
Transmission Power of BS	46dBm
Transmission Power of DUE	24dBm
Shadowing Standard Deviation	8dB
N_o	-174dBm/Hz
Number of Clusters	Variable
Transmit Power of CH	1.425 Joules/s
Power required to receive data from BS	1.8 Joules/s
Power required to receive data from CH	0.925 Joules/s
Content Considered	A file of size 100 kBits
Classical benchmarked Schemes	K-Medoids (KM), Genetic Algorithm (GA), Fuzzy C-Means (FCM)
State-of-the-art benchmarked Schemes	Proposed in [86]. (referred in this document as benchmarked I) Proposed in [171]. (referred in this document as benchmarked II)
Number of Simulation Runs	10,000

Figure. 4.1 shows the result of aggregate throughput versus the number of users. It clearly shows the impact of social awareness as the aggregate throughput was maximum when it was considered. On the other hand, aggregate throughput was considerably low when social awareness was

ignored. At the user density of one hundred, the difference between the two curves was approximately 19%. The throughput for a conventional cellular network with no clustering remained considerably low compared to the other two scenarios. This result, therefore, shows that clustering does play a vital role in enhancing the system's performance. Furthermore, it can be seen that both social-interest and physical parameters (e.g., spatial distribution and channel gains) should be considered while modelling a system as it may bring significant benefits for the users as well as the whole network.

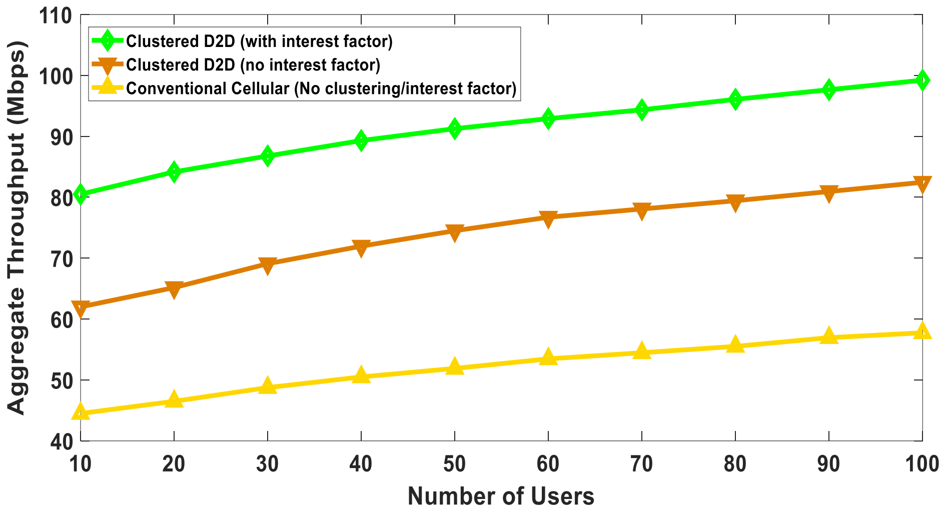

Figure. 4.1 The impact of clustering and social-interest on throughput.

4.3 Benchmarking against Existing Algorithms

We selected five algorithms to benchmark against. Three of these are classical algorithms that are widely found in the literature, namely K-Medoids, Fuzzy C-Means (FCM), and Genetic Algorithm (GA) based clustering. These three algorithms have not been investigated and benchmarked within the context of D2D clustering and content-sharing applications, though an initial investigation was performed in our previous work [99]. The remaining two are the state-of-the-art and recently proposed algorithms. "Benchmarked I" has been proposed by Tulu et al. [86]. This algorithm applies the concept of entropy of betweenness centrality (EBC) to select CHs for content-sharing. The entropy of betweenness is based on the social relationship between the nodes and the shortest paths that exist between the nodes. "Benchmarked II" is proposed by Kazez C.A et al. [171]. This algorithm takes the neighbors and distance among the users as inputs for the selection of CHs.

1. *Throughput Comparison*

The following result shows the comparison of the throughput performance. The proposed algorithm utilizes the social interest and physical parameters of the users to enhance the system's performance. This was discussed in the previous result, and it is further elaborated in Figure. 4.2, as it demonstrates that the proposed algorithm performs approximately 7% better than the next best algorithm (Benchmarked I) at one thousand nodes. Benchmarked algorithms I and II utilize the social interest, but they do not consider both distance and channel conditions among the users for clustering the users. Our result shows that consideration of both metrics does have a positive effect on the system's throughput. This is because many users that are in proximity to each other may not have better channel conditions due to various factors (e.g., shadowing).

2. *Energy Consumption of Users*

The result shown in Figure. 4.3 represents the energy consumption of the nodes in Joules with a varying number of users. It is evident from Figure. 4.2 that we achieved better throughput as compared to the rest of the algorithms. If the file size of 100 kbits is constant, then the energy consumptions will be significantly dependent on the transfer rate. Consequently, the proposed algorithm performed the best (demonstrated by least energy consumptions) at different user densities as compared to the other algorithms. At one thousand nodes, the proposed algorithm approximately consumed 6% less energy as compared to the second-best algorithm.

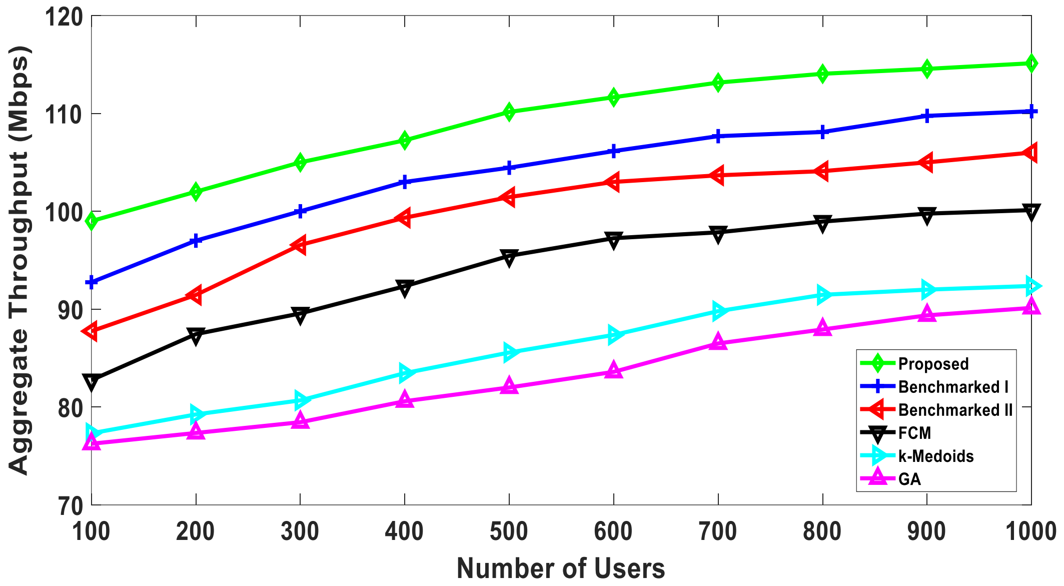

Figure. 4.2 Aggregate Throughput: comparison with the benchmarked ($w_1 = 0.4, w_2 = 0.6$).

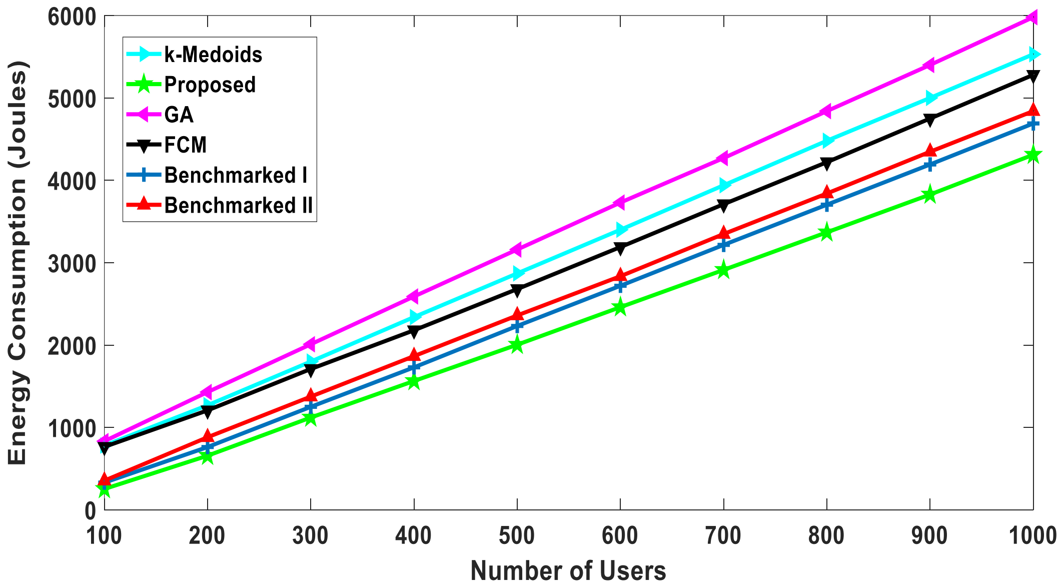

Figure. 4.3 Energy Consumption: comparison with the benchmarked ($w_1 = 0.4, w_2 = 0.6$).

The energy consumption of the proposed algorithm is further elaborated in Figure. 4.4. The Cumulative Distribution Function (CDF) of the energy consumption is presented for the proposed algorithm and the benchmarked clustering algorithms at a user density of one thousand. We can observe that even at the node level, energy consumption demonstrated by the proposed algorithm outperformed the benchmarked algorithms in all quartiles. Therefore, the overall lower energy consumption was not achieved by favoring a few nodes to a large extent while disregarding the others.

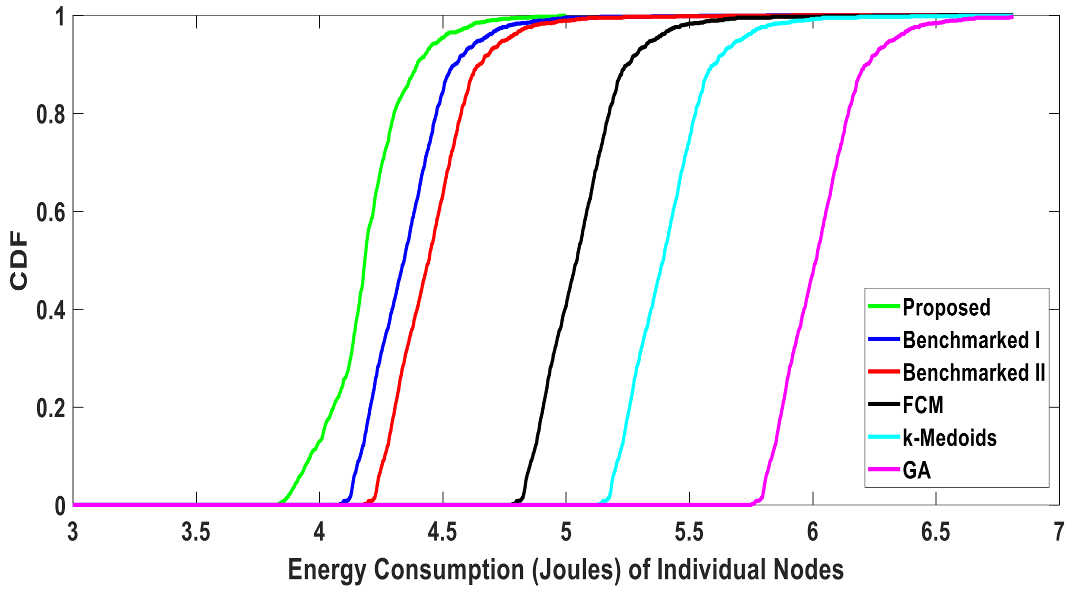

Figure. 4.4 Cumulative Distribution Function (CDF) of energy consumption ($w_1 = 0.4, w_2 = 0.6$).

3. *Area Spectral Efficiency*

Area Spectral Efficiency represents the sum of average achievable rates per unit bandwidth per unit area [176]. To the best of the authors' knowledge, ASE has not been evaluated for all the five benchmarked algorithms. It can be observed in Figure. 4.5 that the ASE of the proposed algorithm was better than all the benchmarked algorithms. ASE depends significantly on the average rates of the users if the area and per unit bandwidth remain constant. Therefore, the proposed algorithm has a higher ASE. It is also encouraging to observe that the performance improved for the proposed algorithm as the user density increased. This shows the scalability of the proposed algorithm. The proposed algorithm showed approximately 3% improvement in ASE at the node density of one thousand, as compared to the benchmarked scheme I that showed the second-best performance. The classical algorithms for generic clustering are not purpose-built for a D2D scenario and are far inferior to the proposed, “Benchmarked I” and “Benchmarked II” algorithms.

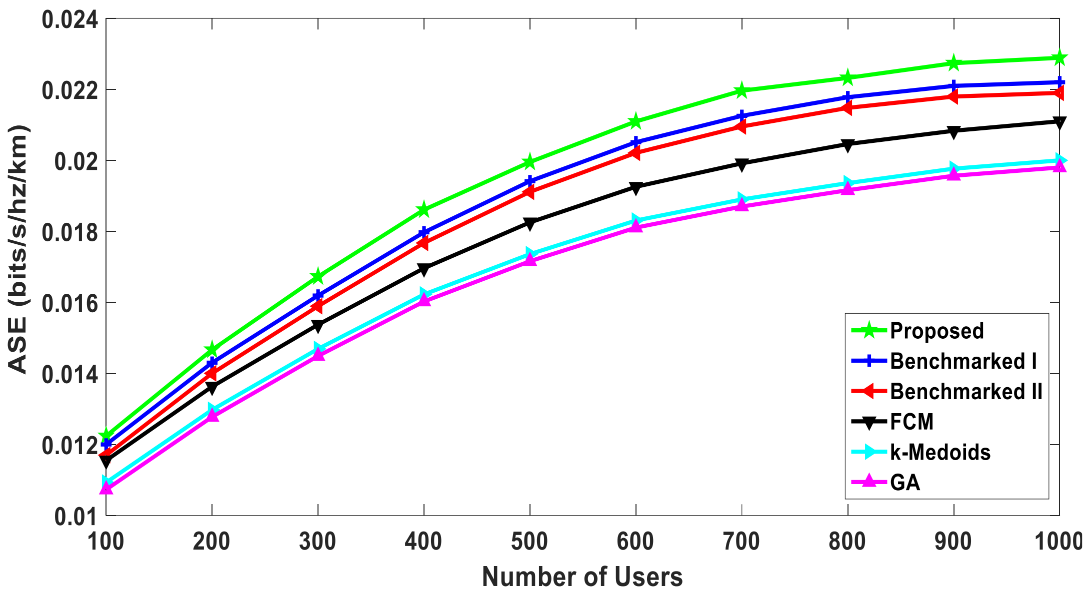

Figure. 4.5 ASE: comparison with the benchmarked ($w_1 = 0.4, w_2 = 0.6$).

4.4 The Optimal Number of Clusters

The three well-known criteria for selecting the number of clusters are: Calinski-Harabasz Criteria (Cal-Har Criteria), Silhouette Coefficient (SC), and Davies-Bouldin Index (DB-I) [177]. DB-I and SC have high computational complexity. Moreover, if the clusters are not well separated, DB-I and SC tend to form a large number of clusters creating a large overhead for cluster formation. Moreover, significant signalling will be required to manage the clusters. Therefore, in this thesis, Cal-Har criterion was selected. In this study, we demonstrated that the Cal-Har criterion can be effective in selecting the number of clusters since it impacts the performance parameters such as energy consumption and ASE. The results presented later, show a comparative study of performance parameters attained after forming clusters based on Cal-Har criterion against a range of randomly selected cluster numbers.

We investigated the effect of the number of clusters on energy consumption and ASE. It has not been reported in the literature considering D2D Content-Centric Networks. The selection of the number of clusters significantly affects the clustering performance. A trade-off always exists when it comes to selecting the number of clusters. Increasing the number of clusters up to a certain extent will bring benefits but at the expense of increased signaling and complexity. The clustering metrics that we selected for the proposed algorithm can vary significantly; thus, it is not easy to predetermine the cluster size. Therefore, the size of the cluster is variable. However, there should be a criterion that can help determine the number of clusters that can be formed based on a given scenario such as user distribution, values of the clustering metrics, etc. In this study, the Calinski–Harabasz (Cal–Har) criterion [173] was selected. It is also termed as the variance ratio criterion. Mathematically, it can be defined as:

$$Cluster\ Size_{(Cal-Har)} = \frac{V_B}{V_W} * \frac{(N-C)}{(C-1)} \tag{12}$$

In Equation (12), V_B represents the between-cluster variance. It can also be defined as the separation between clusters. On the other hand, V_W is the with-in cluster variance. It can be defined as the compactness of the cluster. The idea is to form non-overlapping clusters and therefore, the optimal number of clusters are obtained by maximizing the Cal-Har criterion. Since the input to this criterion was the clustering solution provided by the proposed algorithm, so weights used are similar to the ones defined in Equation (4).

The total number of users is denoted by N, whereas C is the number of clusters against which this criterion will be judged. Clustering metrics determine the variance between and within clusters. To find the optimal solution, Equation (12) needs to be maximized with respect to the number of clusters. As the ratio of the variances given in Equation (12) increases, user association with a certain cluster becomes more precise which leads to the optimal number of clusters.

For a user density of one thousand, the Cal–Har criterion result is depicted in Figure. 4.6. The criterion value was highest when the number of clusters was seven. Therefore, for the given user distribution and node density, the optimal number of clusters should be seven. We then investigated whether this is the optimal choice when considering energy consumption, and ASE. Results for both parameters for a various number of clusters are presented in Figures. 4.7 and 4.8, respectively. A trade-off can be seen for the choice of the number of clusters. As shown in Figure. 4.6, the result for energy consumption aligns with the Calinski–Harabasz criterion, as the lowest energy consumption occurred when the number of clusters was seven to ten. On the other hand, the optimal number of clusters for ASE appears to lie between 10 and 15.

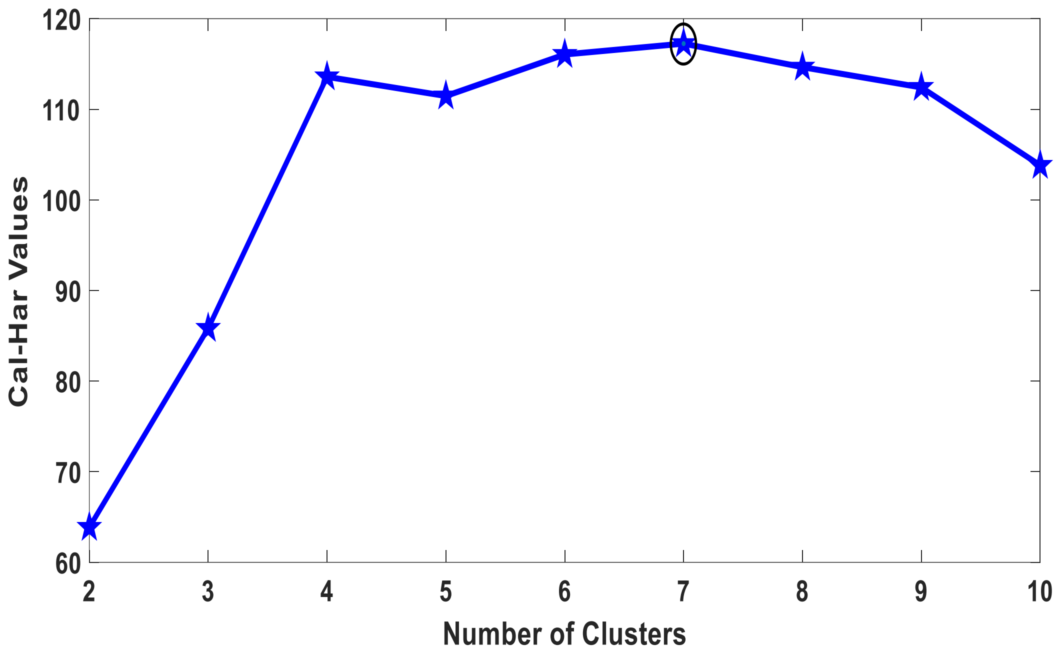

Figure. 4.6 Optimal number of clusters: Cal–Har criterion. To choose the value of the number of clusters, Cal-Har criterion needs to be maximized. For the given node density and distribution of users, the highest value occurs at 7, indicated by a circle.

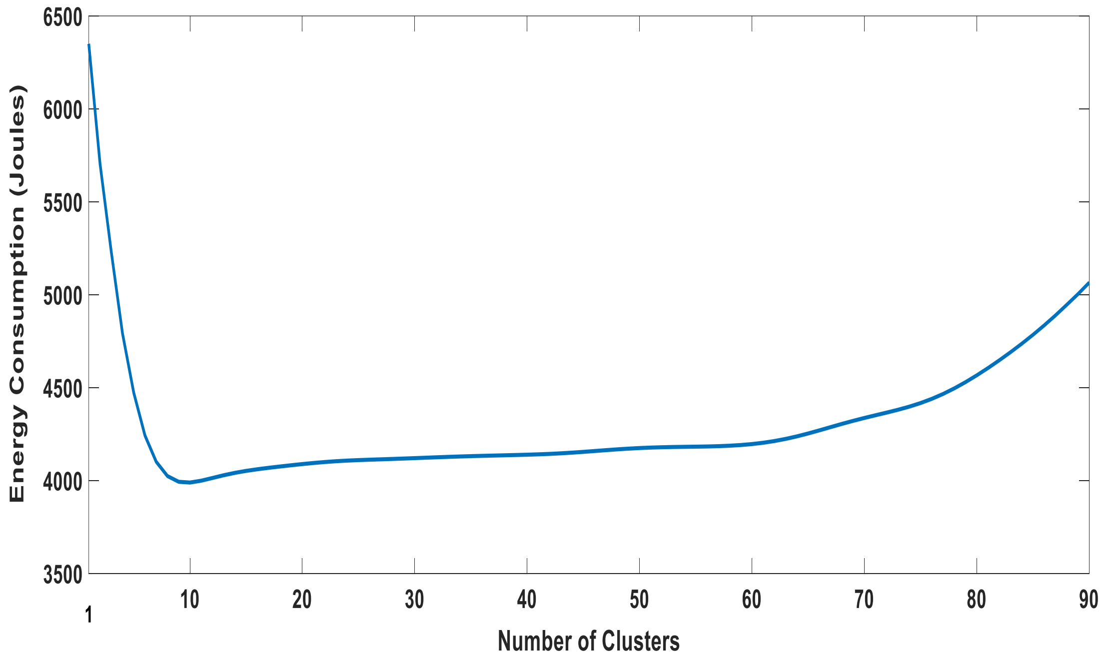

Figure. 4.7 Energy consumption and the number of clusters: a comparison.

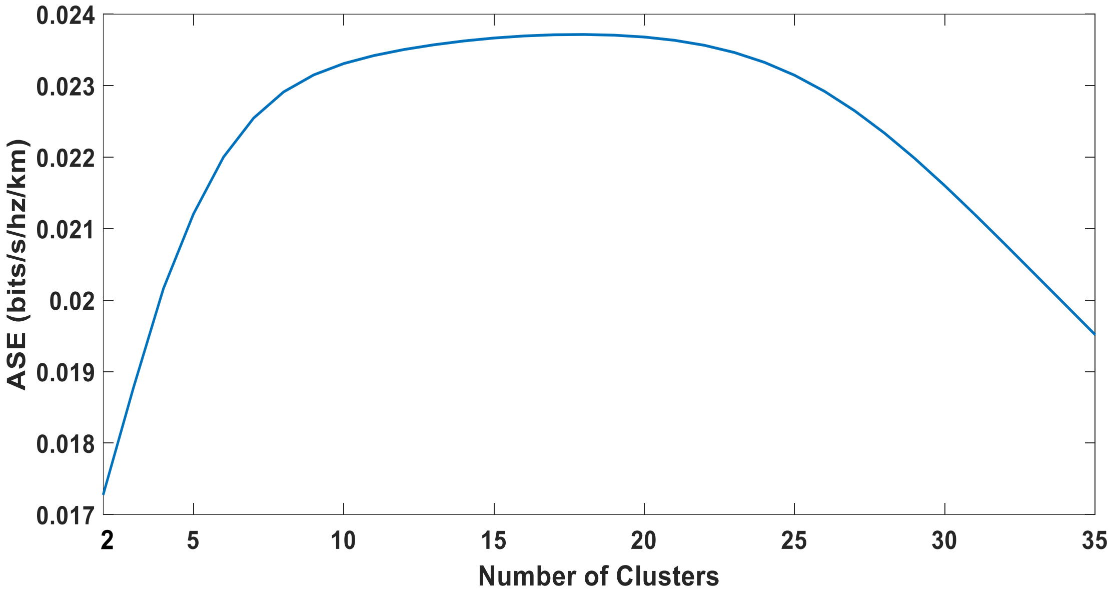

Figure. 4.8 ASE and the number of clusters: a comparison.

It is not straightforward to suggest a certain number of clusters based on the above-mentioned results. It is important to consider a few additional factors that might influence the selection. The signaling overhead and complexity of cluster maintenance increase with the increase in the number of clusters. Moreover, we can observe from the results presented in Figure. 4.8 that ASE did not differ significantly when forming seven to ten clusters as opposed to forming 10 – 15. Therefore, seven is likely to be a better choice, since it represents the lowest energy consumption while sacrificing minimal ASE gain for lower signaling overhead and complexity.

Various external factors influence the choice of the number of clusters as well. As discussed earlier, our study considers forming clusters only for those users that are interested in content-sharing. Therefore, in some situations, only a few users might be interested in content-sharing, and hence, making a certain number of clusters is not necessary. On the other hand, a scenario can build up where a large group of users is interested in content-sharing, but even in this case, the physical location of the users might influence the choice of the number of clusters. A large number of closely packed users at a concert or a sports event only need a few clusters. As opposed to this scenario, users sharing a common interest might be dispersed in a geographical area, requiring a higher number of clusters. More clusters can also be formed in a scenario where users might be in close vicinity, but they have different content of interest. In that case, users sharing the same interest are expected to be in one cluster, whereas the rest of the users should form a separate cluster. Moreover, some users may be better served by the BS, and it should not be mandatory for all the users to be considered for the clustering.

4.5 Throughput Fairness

Jain's fairness model [178] was used to evaluate the fairness performance of the proposed algorithm. The Jain's fairness index denoted by $J(x)$ is represented by Equation (13).

$$J(x) = \frac{(\sum_{i=1}^{M} x_i)^2}{M \sum_{i=1}^{M} {x_i}^2} \tag{13}$$

x_i represents the throughput of the i^{th} user, given that there are total M users. The algorithm was simulated with the weights as follows (for Equation (4)): $w_1 = w_2 = 0.1, w_3 = 0.8$. The weights selected for this result were empirically adjusted to enhance fairness. So, the maximum weight is given to the variance of channels. Though the selected weights did not yield the best energy consumption and ASE, the proposed algorithm did outperform all the benchmarked algorithms when it comes to fairness. This is depicted in Figure. 4.9. At the user density of one thousand, the proposed algorithm performed approximately 7% better than the benchmarked scheme I, which performed the best among the existing algorithms. This result demonstrates the flexibility of the proposed algorithm. By simply adjusting the weights, it is possible to achieve better fairness. It should be noted that the works reporting the benchmarked algorithms I and II do not discuss fairness. These algorithms also do not have any parameter to perform this trade-off.

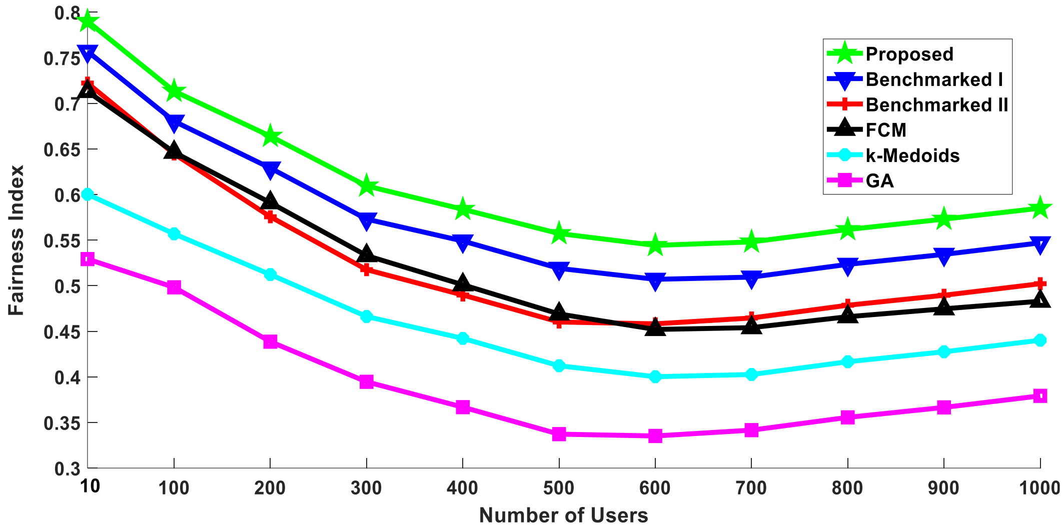

Figure. 4.9 Jain's Fairness Index: Comparison with the benchmarked ($w_1 = w_2 = 0.1, w_3 = 0.8$).

4.6 The Trade-Off Between Fairness and Other Performance Parameters

The energy consumption and ASE for the weights $w_1 = w_2 = 0.1$ and $w_3 = 0.8$ are presented in Figures. 4.10 and 4.11, respectively. It can be observed that the cost of improved fairness is a slight degradation in performance with respect to energy consumption and ASE. However, the performance of the proposed algorithm is satisfactory in the sense that it is better than the other four benchmarked algorithms and there is only a very small performance gap with the best scheme. The proposed algorithm is able to trade-off energy consumption and ASE with fairness, which is not possible in any of the benchmarked algorithms. Our algorithm provides this flexibility of weights adjustment to enhance the desired performance parameter. It can achieve the best performance for a given parameter while providing a satisfactory performance with respect to the rest.

To the best of our knowledge, the algorithms considered for benchmarking in this study have not been investigated for all the three performance parameters i.e., energy consumption, ASE, and fairness.

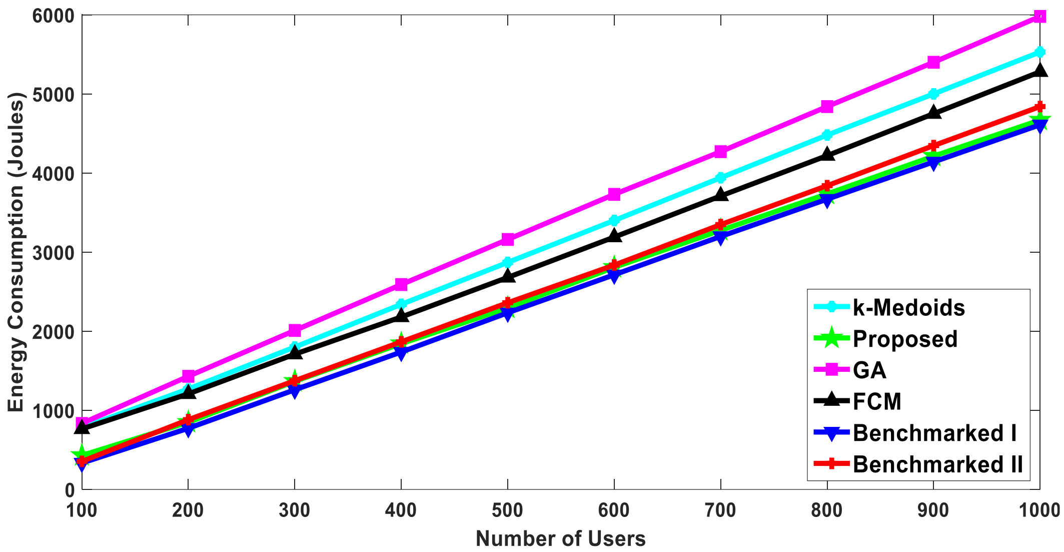

Figure. 4.10 Energy consumption: comparison with the benchmarked schemes ($w_1 = w_2 = 0.1, w_3 = 0.8$).

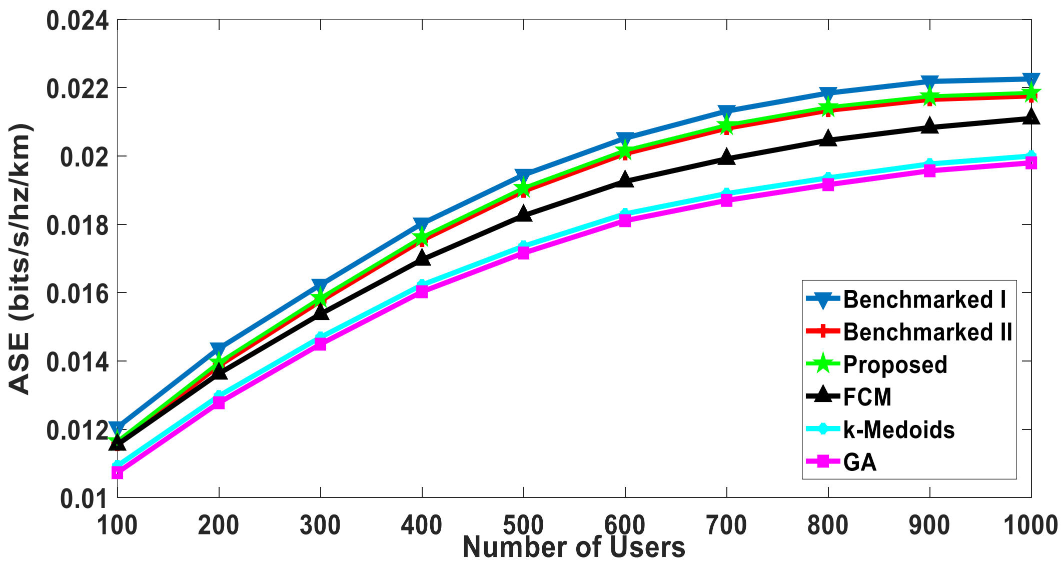

Figure. 4.11 ASE: comparison with the benchmarked schemes ($w_1 = w_2 = 0.1, w_3 = 0.8$).

4.7 Summary of Results

It is evident from the results that the proposed Multi-Factor clustering enhanced the system's performance. The performance of the proposed algorithm was thoroughly investigated against different popular clustering algorithms. The proposed algorithm shows a significant improvement in throughput performance, a 6% improvement in energy consumption while achieving 3% better ASE as compared to the best of the benchmarked algorithms. The improvement in energy consumption of the proposed scheme compared to all other schemes was 8%, 14.5%, 17%, and 24 % with respect to Benchmarked II, FCM, k-Medoids, and GA respectively. The improvement in ASE of the proposed scheme compared to all other schemes was 4% for Benchmarked II, 8% for FCM, 14% for k-Medoids, and 15% for GA.

Most of the research works focus on forming a fixed number of clusters for various user densities. As opposed to this approach, we use a criterion to form clusters appropriate for a given node density and geographical location.

The effect of the number of clusters on the energy consumption of users and ASE was also investigated. A trade-off exists between the two metrics in the selection of the number of clusters. The optimal energy consumption was achieved at a smaller number of clusters as compared to ASE. It is suggested that signalling overhead required to set up more clusters should be considered while selecting the number of clusters. Hence, the formation of a smaller number of clusters showing optimal energy consumptions at the cost of marginal degradation in ASE is acceptable.

A slight modification in the algorithm and weight adjustment improved throughput fairness up to 7%. This improvement came at the cost of a slight degradation in energy consumption and ASE.

4.8 Related Publication

The work presented in this chapter has been published in the following research article:

S. Aslam, F. Alam, S. F. Hasan and M. A. Rashid, " A Novel Weighted Clustering Algorithm Supported by a Distributed Architecture for D2D Enabled Content-Centric Networks," *Sensors* 2020, *20*, 5509. https://doi.org/10.3390/s20195509.

URL: https://www.mdpi.com/1424-8220/20/19/5509

CHAPTER 5

USER SEGREGATION: A MACHINE LEARNING APPROACH TO ENHANCE THE PERFORMANCE OF CLUSTERED NETWORK

The previous chapter demonstrated the significance of applying a clustering algorithm to the content-sharing network. Although the proposed algorithm produced significant performance gains, we believe that there is a need to optimize the clustering process as there always exist some users better off with the eNB rather than being in the clusters. Therefore, this chapter is dedicated to present the impact of user segregation on different performance parameters such as throughput, energy consumption, and fairness. We utilize machine learning algorithms namely, Deep Neural Network (DNN), Random Forest (RF), and Support Vector Machine (SVM), to segregate the users that are better off with the eNB and form clusters for the rest of the users. We explore the performance of all the algorithms when applied to the user's segregation problem (please see Section 6.3). Applying these methods, it has been demonstrated that the performance of the individual users as well as the whole network, has been improved significantly.

5.1 Introduction

The works presented in Chapter 2, show that the importance of clustering is not specific to content-centric broadcast networks and can be applied to various applications as shown in the literature [84-93, 103, 179-182]. Since all these clustering works are meant for performance enhancement of various parameters, therefore a fundamental question would be, is there any scope of improvement within the clustering process itself? To answer this question, let us consider a few users interested in the same content. Once users sharing a common interest are identified (several processes exist in the literature for content/interest identification such as shown in Chapter 3), they are either served directly by the eNB or by the Cluster Heads (CHs) in clusters. Clusters are formed only for the users that are interested in the same content. Different research works that consider clustering, place all the users/nodes in clusters whereas on the other hand if clustering is not performed then all the users are associated with eNB. However, we believe there is a better approach to clustering, so we suggest user segregation; some users communicate with CHs while being in clusters, whereas the remaining communicate directly with the eNB. Nevertheless, as a result of this segregation, only a certain percentage of users communicate directly with eNB and the majority of the users still stay in clusters and exploit D2D communication.

To further elaborate on the proposed concept, let us consider a social gathering, such as a football match. Users (socially aware nodes) are interested in the same videos of their favorite players. Conventionally, all these users sharing a common interest will be considered for clustering. However, even within a socially connected group, user segregation needs to take place to enhance the performance. Hence, in the proposed scheme, there are three types of users; CHs (responsible for fetching the content from eNB), cluster members (part of the cluster and receiving the requested content from CH via D2D), and segregated users (downloading the content directly from the eNB). This user segregation has not been reported in the literature. We will demonstrate that segregation has a significant impact on the performance of both individual users as well as the network.

It should be noted that the concept of user segregation is not specific to any particular application of the D2D clustered network. It can be applied to any clustering scenario and/or application. However, we have evaluated the proposed scheme for a content-sharing D2D enabled network. It is also understood that D2D communication is meant to offload the central controller and socially connected users are meant to be in the same group. However, we focus on optimizing the clustering scenario by introducing the segregation concept. eNBs often have the spare capacity [183-185] and we are proposing to exploit this capacity leading to better performance and more efficient utilization of resources. Various research works focus on exploiting the spare capacity to improve the system's performance [186-189], but not for clustering. Therefore, this study proposes trading off significant performance improvement with an increase in load on the central controller. Later in the Results chapter, we show the performance improvement at various loading factors.

For the proposed scheme to work, the fundamental task would be to categorize the users into two groups; one group contains all the users better served in clusters while the other group consists of users better off with the eNB. To perform the segregation, this study opts for the ML approach. ML is an ideal tool for the proposed problem since we take advantage of offline training, without involving the eNB, making the training, and segregation process distributive. The machine learning algorithms employed are: Support Vector Machine (SVM) [190, 191], Random Forest (RF)[192], and Deep Neural Network (DNN)[193]. Literature suggests that SVM and DNN have been widely used for classification problems [109]. Moreover, the problem discussed in this thesis is a binary classification problem which has found even more applications for SVM and DNN [113]. Similarly, RF has found many use-cases for classification and regression problems targeting wireless networks [109, 194]. The concept of the proposed scheme; user segregation, utilization of ML algorithms is shown in Figure. 5.1.

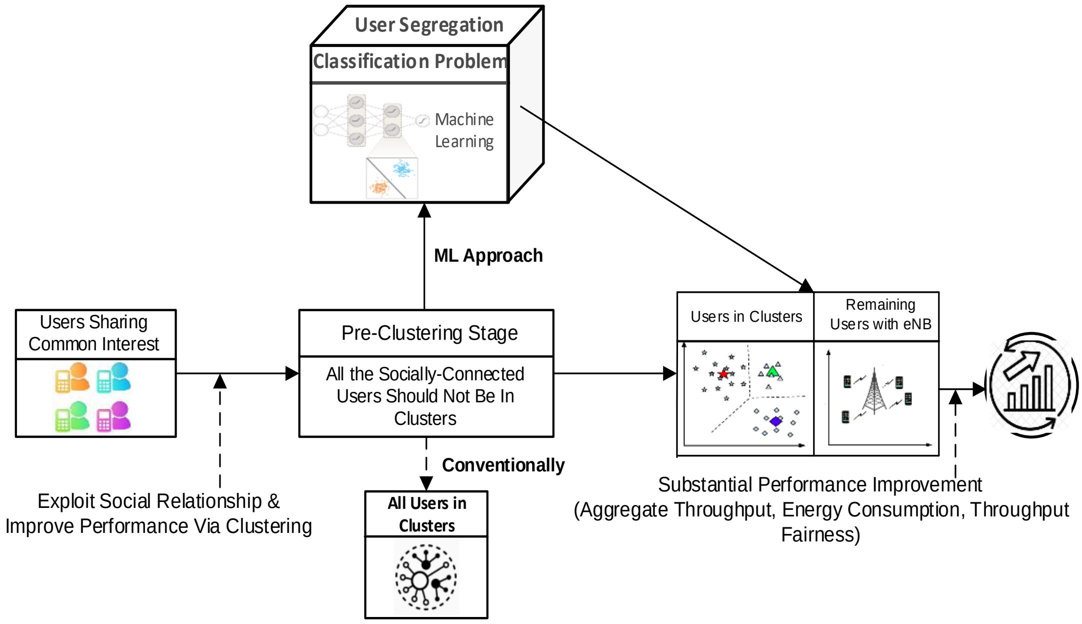

Figure. 5.1 Proposed User Segregation before clustering. Segregation is carried out using Machine Learning and as a result, few users are shown communicating in clusters whereas the rest communicate directly with the eNB.

5.2 Data Collection for Machine Learning

The significance of applying ML to various cellular network scenarios has already been presented in section 2.6. To realize all the applications of the machine learning paradigm, it is important to first address the data collection required for the learning to take place. Due to extensive interest in machine learning schemes, data storage is now an essential part of the cellular network architecture therefore data is stored by COs to make it readily available for analysis [121, 122, 126-129]. Specifically, the collection of data can be accomplished by UEs, Core Network (CN), and the Radio Access Network (RAN)[195]. Figure. 5.2 presents the details of different data collection sources within the cellular network [195]. Once the data from different sources come in, it needs to be processed so that machine learning algorithms can be designed to improve the performance of the network.

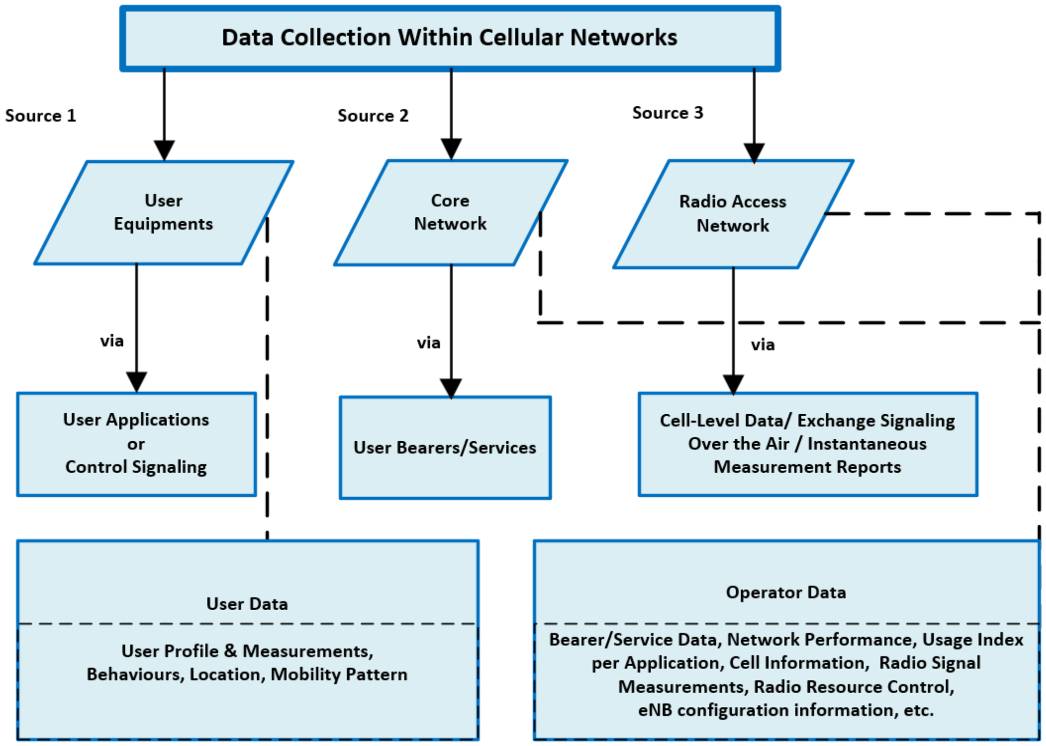

Figure. 5.2 Data Collection Opportunities to Construct Training Corpus in Cellular Networks.

5.3 The System Model

We assume that the eNB is placed at the center of the cell. It should be noted that the placement of the eNB does not affect machine learning since users are randomly distributed and therefore, the distance of users to the eNB and/or distance of users among themselves is random as well. Moreover, the geographical location of users for which the clustering is taking place does not specifically represent a macro cell or small cell. The learning does not consider a particular shape of the cell. The concept can be validated for any scenario since the learning was not subjected to any of these factors (geographical location of eNB and/or type of cell). Therefore, the scheme can be adapted for next generation networks such as 6G, where eNB is likely to exist to support the users even though the concept of cell is not relevant.

A typical network model is presented in Figure. 5.3. It is assumed that users are interested in certain content. These users are going to be served in clusters via a CH (UEs shown with red tops). However, there are certain nodes better served with the eNB and are communicating directly in Figure. 5.3 (UEs shown with blue tops). Moreover, the machine learning aspect is shown as well, complementing the network model.

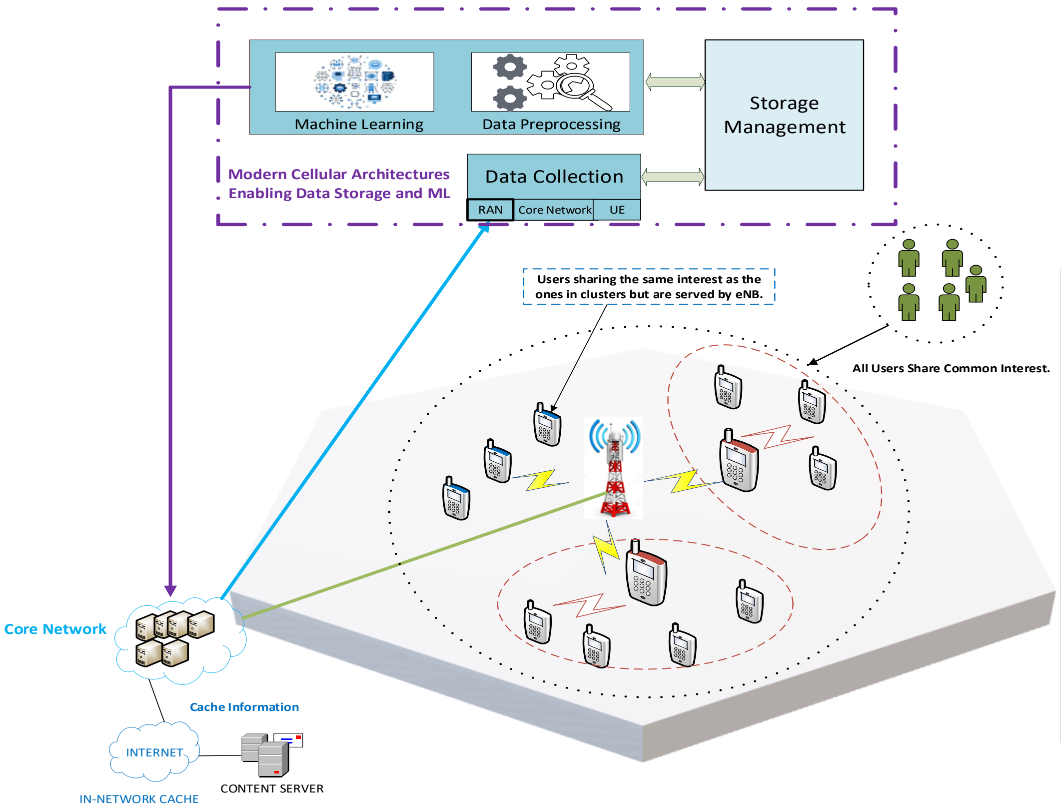

Figure. 5.3 The network model supported by Machine Learning. A separate learning server, linked with core networks, has been considered an important part of modern cellular networks.

The users are represented by the set U, which comprises of both CUE and DUE, such that, $U = 2U_D + U_C$, where $2U_D$ represents U_D D2D pairs and U_C are cellular users. We may write the capacity of the system represented by both types of users as:

$$C_U = W\left(\sum_{i=1}^{U_C} \log_2\left(1 + SNR_{U_{C_i}}\right) + \sum_{j=1}^{U_D} \log_2\left(1 + SNR_{U_{D_j}}\right)\right) \tag{14}$$

where i & j denote indices of CUEs and DUEs respectively and W represents the bandwidth of the cellular network.

SNR_{U_C} and SNR_{2U_D} represents SNR of the cellular and D2D users respectively, which can be written as

$$SNR_{U_{C_i}} = \frac{P_{eNB} h_{eNB,U_{C_i}}}{N_o B} \tag{15}$$

Where P_{eNB} is the transmit power of eNB.

The channel between eNB and ith CUE is represented by $h_{eNB,U_{C_i}}$.

$$SNR_{U_{D_j}} = \frac{P_m^T h_{U_{D_j},m}}{N_o B} \tag{16}$$

P_m^T is the power of the D2D transmitter indexed m. The channel between the mth DUE transmitter and D2D receiver is denoted by $h_{U_{D_j},m}$.

The channel gain h is modelled by a Rayleigh fading channel. It is important to note that we encapsulate both distance-dependent attenuation (i.e. path loss) and shadowing effect into the channel model. Shadowing is defined by a Gaussian random variable with a zero mean and standard deviation, σ (σ = 8dB was considered in our simulation study presented in the next chapter).

As can be observed from Equation (14 -16), the sum capacity of a network depends on the accumulative SNRs of both CUEs and DUEs. Moreover, if we assume that the transmission power (of both eNB and DUE) and channel bandwidth are constant, then the SNR will significantly depend on the channel between the transmitter and the receiver, which is valid for both types of users. Nevertheless, it is worth mentioning that even with the varying transmission power, SNR will be influenced by the channel conditions. Therefore, the physical link between the transmitter and the receiver plays a vital role in determining achievable data rates. This is important in selecting the input features for training the classifier.

The goal is to increase the capacity represented by Equation (14), utilizing the concept of user segregation, executed by ML algorithms. From a broader perspective, it is a binary classification problem where a user needs to be added to either of the two groups i.e. with eNB or with clusters. To perform this segregation, ML classifiers are utilized. The target for ML is based on maximizing the throughput of a particular node. It is mentioned in section 3.5 that since we are considering a multicasting application, the maximum achievable rate of a cluster member depends on the worst physical link, to ensure that all the members receive the required data. The achievable rate of the cluster member is already defined by Equation (10).

A user is selected for either of the two groups based on the achievable data rate being higher in that group (i.e. either with the eNB or the CH) as opposed to being part of the other group. This fact gives the user opportunity to not only reduce its energy consumptions but also increase the throughput fairness of the system. It should be mentioned here that the ML algorithm is trained offline before it is tested on a mobile network (or, as in our case tested with the simulation environment). For the classification, we propose to utilize SVM, RF, and DNN. These are benchmarked against four widely used classification techniques and are found to be superior.

Here we provide details about the data labelling process. ML training data was generated via simulations. We used the complex gaussian channel model that follows Rayleigh fading. All the simulation parameters are the same as used in Chapter 4 (listed in Table 4.1). It is observed in the previous chapter that the weights attached to distance and channel conditions were not the same, therefore, it was necessary to have a mechanism that learns the relationship of each node with respect to these parameters and classify the users into the two groups (i.e., with eNB or in clusters). Hence, we utilized ML algorithms. Uniformly random distributed users interested in content-sharing are clustered using the proposed clustering algorithm of Chapter 3. The throughput performance of the users and the system was evaluated. Then, the same set of users, sharing common interest, were communicated in a conventional cellular manner and their performance was noted as well. After these simulations, we have the data that clearly shows whether the node should be in a cluster or communicate directly to the eNB so it was labelled accordingly. An example of a learning set is provided in appendix A.

5.4 The Proposed Machine Learning Approach

The data generated for the training is obtained via the simulations, following the practice reported in the ML applications of wireless networks [196-198]. It is explained in [124], that the comparison of different ML algorithms can be made if we have common datasets. However, while this is true for computer vision, voice, and image processing, the wireless communication domain is unfortunately not having common datasets because it inherently deals with the data that can be accurately generated by simulations. However, as discussed in Section 5.2, data is available to the COs, that could be potentially used to set up common benchmarks so the effectiveness of the simulation-based training can be evaluated. The overall learning algorithm is presented in Figure. 5.4. The algorithm is trained completely offline. However, data required (distance and channel conditions) for an online implementation, are the typical data required for forming clusters and can be obtained via the D2D discovery process which takes place before the communication starts. The details of such a relevant process can be found in the clustering literature, e.g. see [47, 182] that shows network latency is not impacted significantly.

The selection of input features and data labelling took place keeping in view the target of throughput maximization. The details of the explored ML algorithms including the tuning of hyperparameters have been provided in subsequent subsections and Chapter 6.

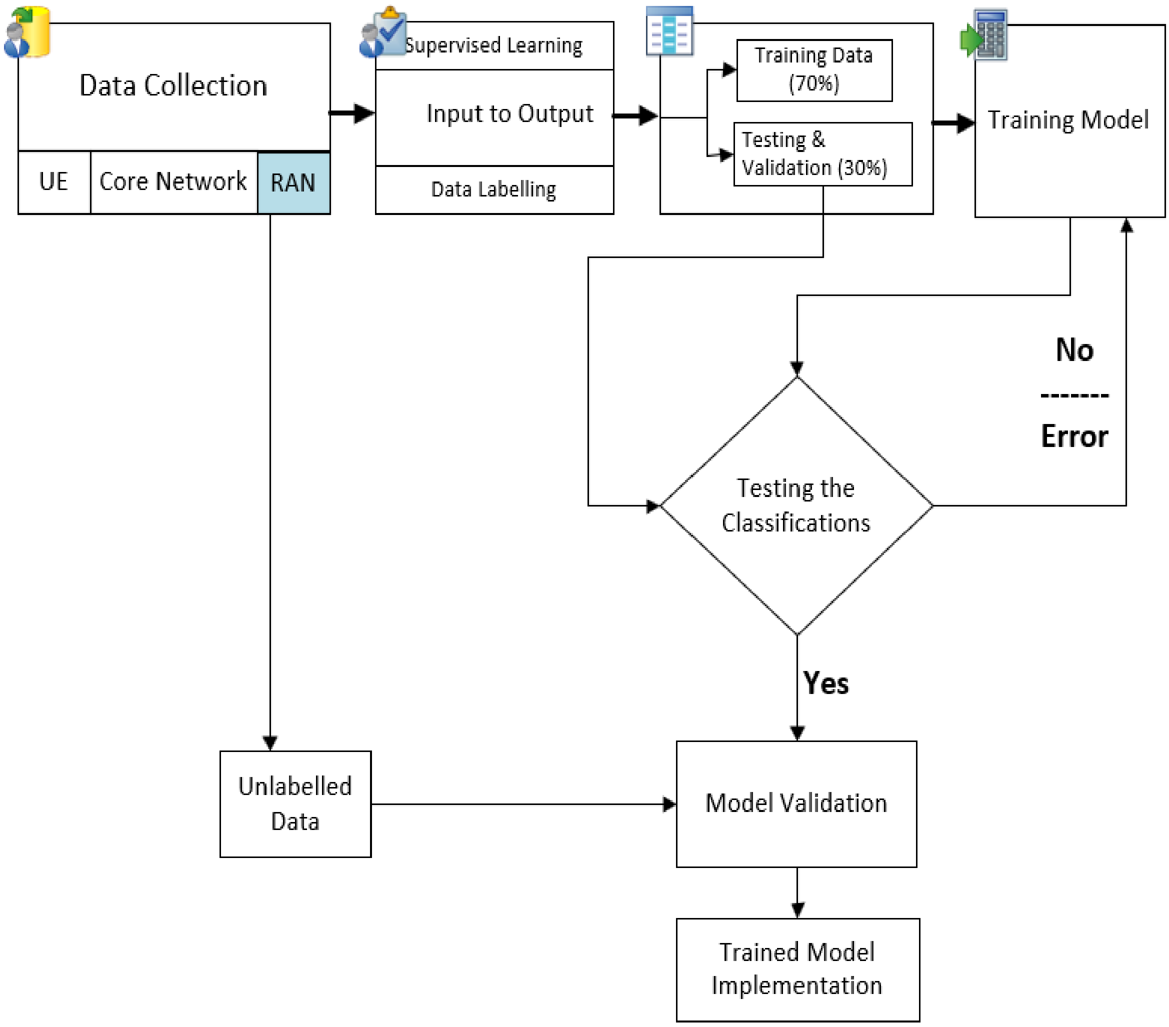

Figure. 5.4 The Learning Algorithm flow chart. It should be noted that all the tasks are performed offline except the trained model implementation.

5.4.1 The DNN Architecture

The proposed architecture is shown in Figure. 5.5. Utilizing this architecture, user segregation is performed for the proposed scheme. The output of the DNN gives a probability value, which classifies a node that should be either serviced by the eNB or the CH. The DNN consist of three layers. The input layer is termed as L_{IN}, whereas the hidden layers are termed as L_{H1}, L_{H2} respectively. The selected features and dependent vector/label are presented in Table 5.1.

Table 5.1 Input Features and True Labels

Feature (Mean Value)	True Labels	Labels Representation
1. Distance of a user from eNB	0	Users with eNB
2. Distance of a user from other users 3. Pathloss of a user with eNB 4. Pathloss of a user with other users	1	Users in Clusters

The data generated for the training purpose is obtained via the simulations. It is in conformity with different research works presented in the field of wireless networks. Seventy percent of the total generated data is randomly selected data for training purposes and the rest thirty percent is used for validation/test.

Before the actual values of the weights can be found, the weights are randomly initialized from values between 0 and 1. Once the weights are initialized, we move to the first hidden layer where a dot product of the initialized weights and input vector is performed. It can be represented as the following equation:

$$X_{L_{H1}} = V_{L_{IN}} \, . \, W_{L_{IN}} + b_{L_{IN}} \tag{17}$$

$V_{L_{IN}}$ represents the input vector (shown in Figure. 5.5), $b_{L_{IN}}$ is the bias and weights are represented by $W_{L_{IN}}$. Once the input passes through the first layer, it becomes a neuron to be processed by the other layers.

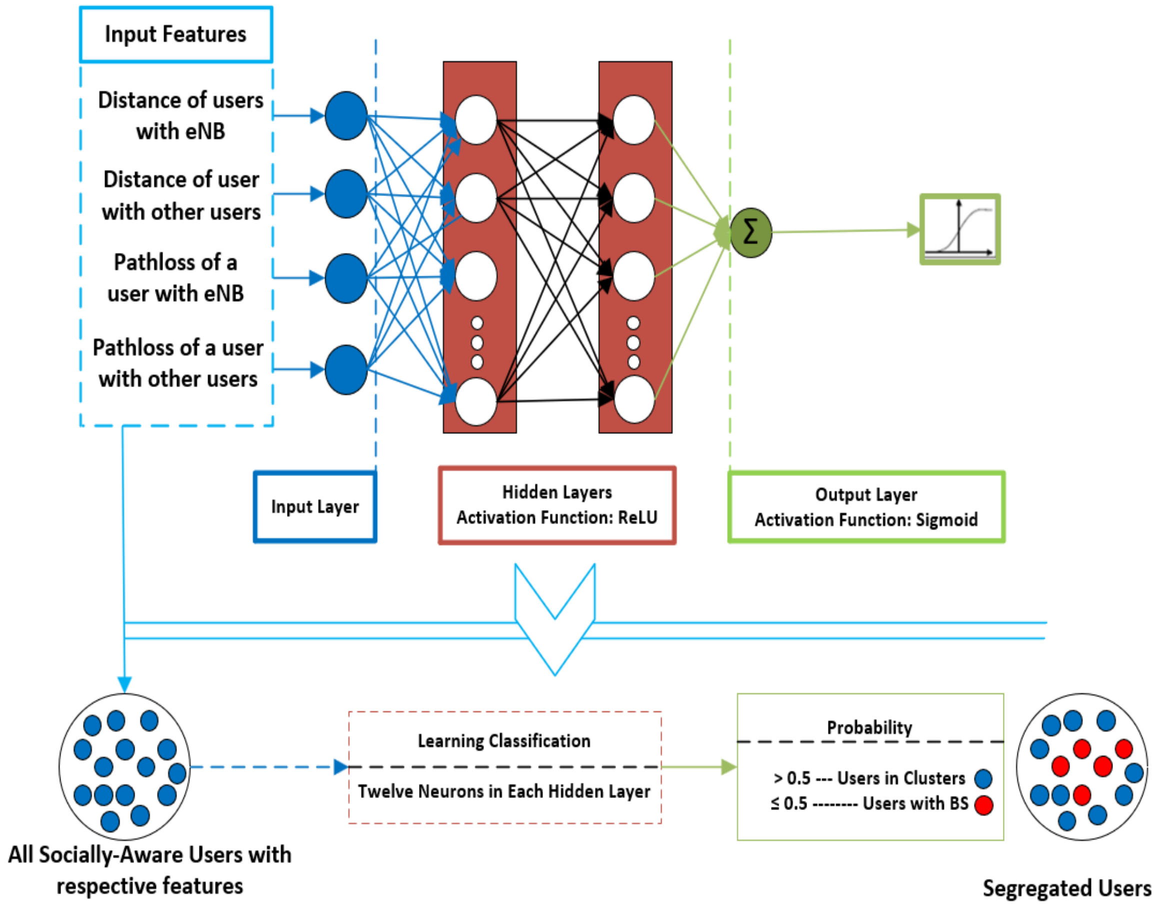

Figure. 5.5 The DNN architecture for proposed mixed-mode clustering. It consists of two hidden layers having twelve neurons each and a single neuron output layer. The number of hidden layers and the neurons they contain is obtained by the Random Search scheme.

Now we use $X_{L_{H1}}$ to pass through our first activation function i.e. Rectified Linear Unit (ReLU) [199], which creates the first hidden layer and its output becomes the input of the second hidden layer.

The output layer of the proposed DNN architecture is composed of one neuron with the output given by Equation (18).

$$Out_{L3} = sig(Out_{L2} \, . \, W_{L_{H2}} + b_{L_{H2}}) \tag{18}$$

$(Out_{L2}.\ W_{L_{H2}})$ represents the dot product between the input vector $V_{L_{H2}}$(i.e. the data vector from the second hidden layer) and the corresponding weights $W_{L_{H2}}$. The bias of the second hidden layer is represented by $b_{L_{H2}}$.

Out_{L3} is the output from the last layer. Sig is the sigmoid function given by Equation (19).

$$sig\ (x) = \frac{1}{1 - e^{-x}} \tag{19}$$

It computes the probabilities for the two classes. It can be mathematically represented as the following:

$$p_{out} = \begin{cases} p_{cluster} & if\ Out_{L3} > 0.5; \\ p_{eNB} & otherwise \end{cases} \tag{20}$$

In Equation (20), $p_{cluster}$ represent the nodes that should be in clusters whereas the nodes that should be served by the eNB are represented by p_{eNB}. Ideally, all the nodes with clusters should have a probability of 1, and others should have a probability of zero.

The output is then evaluated for error. It is calculated using binary cross-entropy. A loss function is defined to determine the misclassification between the target and the predicted one. The loss function is evaluated after every training iteration. It is given by:

$$LF(p_{cluster}, p_{eNB}) = -\frac{1}{N}\sum_{i=1}^{N}\Big([True\ Label_{cluster}]log\big(p_{out_i}\big) + [[True\ Label_{eNB}]]log\big(1 - p_{out_i}\big)\Big) \tag{21}$$

The loss function is averaged over all the training samples N at the end of each iteration. Once, the first iteration is complete, we backward propagate our gradient descent to update our weight parameters. Once the error is known, we want to minimize it as much as possible. The whole process is represented in the Figure. 5.6.

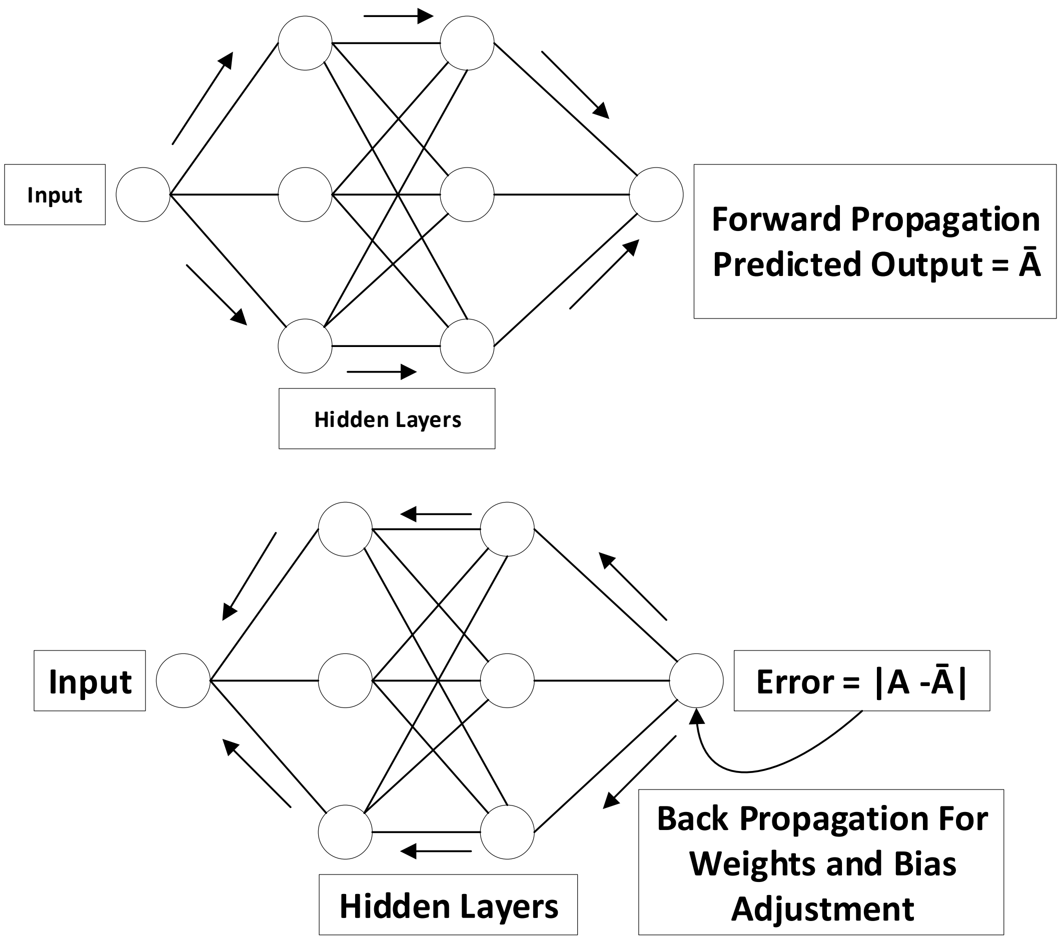

Figure. 5.6 The Forward and Back Propagation for Learning the Algorithm.

5.4.2 Support Vector Machine

The objective of the SVM is to find an optimal hyperplane that distinctly classifies the input vectors. An SVM represents an N-dimensional hyperplane, that optimally segregates the input data into two categories [128]. To classify the data points, there can be several options of hyperplanes, however, the aim is to find a plane that can maximize the distance between the classes [128]. This maximization reinforces the confidence with which we can classify the future values. This concept is illustrated in Figure. 5.7.

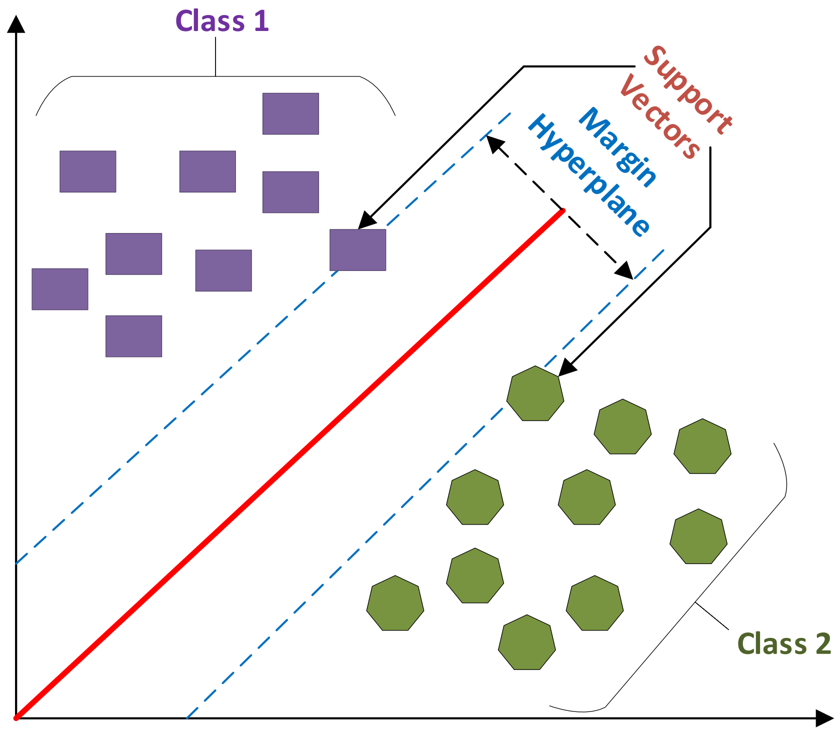

Figure. 5.7 The SVM Concept: A Hyperplane to Optimally Segregate the Data.

In this study, the hyper plan segregates the users (e.g. whether a particular user should be in the cluster as opposed to being with eNB). The performance of the SVM is dependent on the Kernel functions. We empirically selected Sigmoid Kernel after comparing it with other functions such as Polynomial, Gaussian, and Radial Basis Function, etc. The Sigmoid Kernel provided the highest accuracy as compared to other functions (please see Table 5.2).

Table 5.2 Comparison of Accuracy for Different SVM Kernel Functions

Kernel Function Explored	Accuracy (%)
Linear	82.33
Gaussian	88
Polynomial	93.33
Radial Basis Function	94.5
Sigmoid	96

The overall schematic of SVM applied to the proposed study is shown in Figure. 5.8.

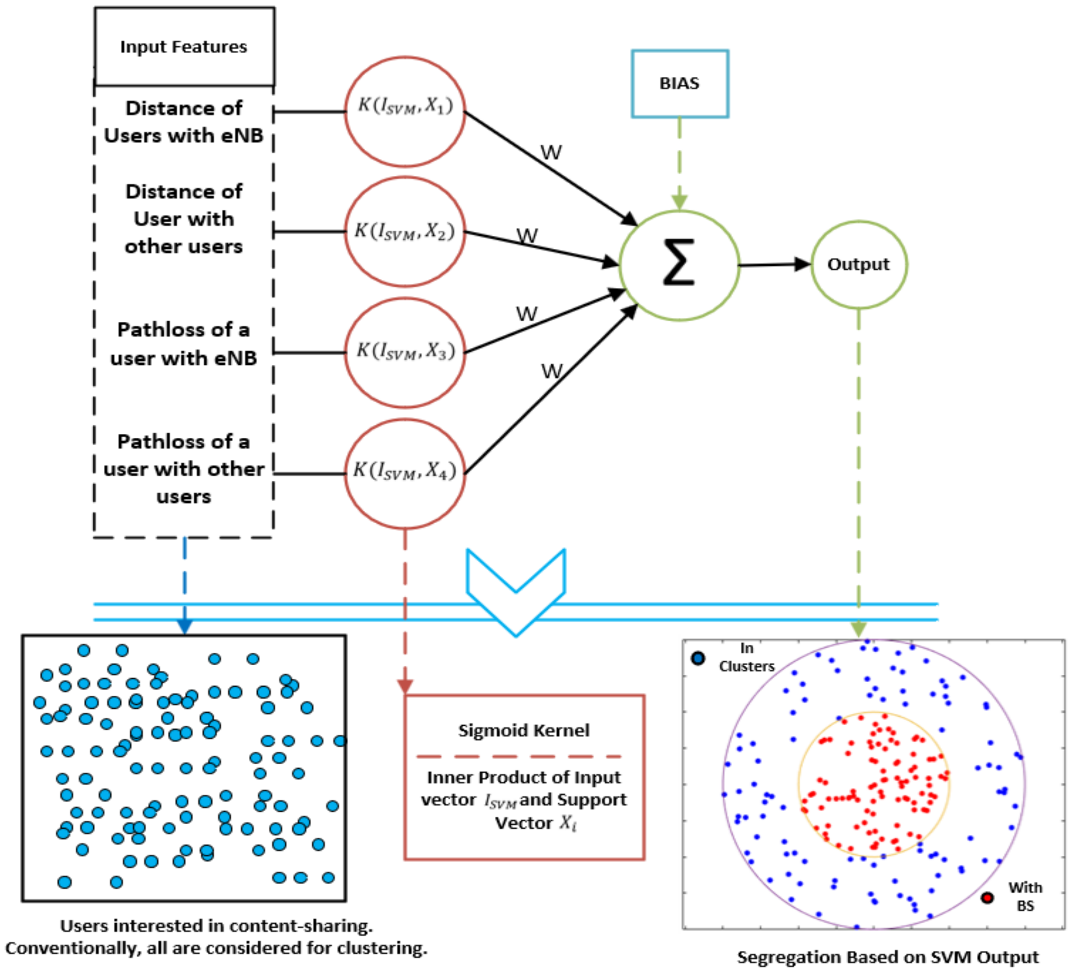

Figure. 5.8 SVM architecture implemented with Sigmoid Kernel, for the proposed mixed-mode clustering.

Each input instance of an SVM, denoted by I_{SVM}, represents a pair (a_i, b_i), where $a_i \in \mathbb{R}^n$ is the data instance (shown in Figure. 5.8), and b_i represents the binary class label (given in Equation (23)). A class can be characterized as either positive or negative. Therefore, in this setting, 'nodes with clusters' belong to the positive class while 'nodes with eNB' belong to the negative class. Given this information, we may write the hyperplane as;

$$w.I_{SVM} + C = 0 \tag{22}$$

Where the classifier can be defined as;

$$f_{(I_{SVM})} = \begin{cases} +1, if\ w.I_{SVM} + C \ \geq 0 \\ -1, if\ w.I_{SVM} + C \ < 0 \end{cases} \tag{23}$$

In the above-given equation, w represents attached weights and C is a constant. +1 and -1 are binary class labels (b_i).

5.4.3 Random Forest

Random Forest belongs to the class of ensemble algorithms (e.g. bagging and boosting) that utilizes the combination of trees to increase the accuracy and make stable predictions. RF can be used for both classification and regression problems. In this study, we use RF for the proposed binary classification problem [200]. The tree ensemble created by RF is trained on bootstrapped training data. The majority vote of the trees decides the classifier output [201]. The classification decision is dependent on the attribute/feature selection approach. In this work, Gini Index (GI) has been utilized, which evaluates the impurity of a feature with respect to the class. GI can be written as given in Equation (24) [202];

$$\sum \sum_{a \neq b} \big((f(NS_a, TS)/|TS|\big) \big((f(NS_b, TS)/|TS|\big) \tag{24}$$

Where NS_a is the node selection class 'a' (node belonging to cluster in this case), TS is the training set. The probability of the selected node is defined by $(f(NS_a, TS)/|TS|)$.

Once the RF is trained on $'N'$ trees, each new dataset is passed down the trees, and the forest chooses a class based on the majority vote of trees. As described earlier, we utilize RF to classify the nodes that should be with eNB as opposed to being in the clusters. The concept of RF as applied to the user segregation problem is illustrated in Figure. 5.9.

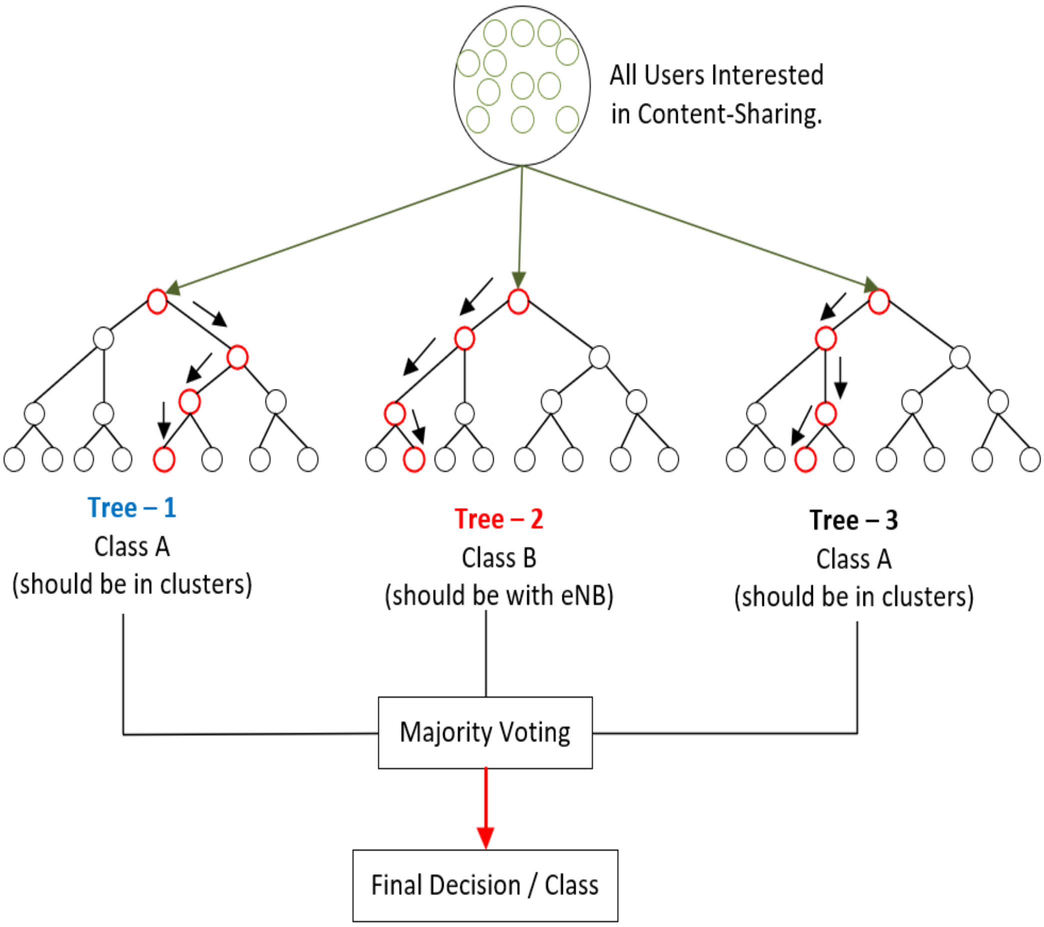

Figure. 5.9 The Random Forest concept as applied to the User Segregation problem.

5.5 Related Publication

The work presented in this chapter has been published in the following research article:

S. Aslam, F. Alam, S. F. Hasan and M. A. Rashid, "A Machine Learning Approach to Enhance the Performance of D2D-Enabled Clustered Networks," in *IEEE Access*, vol. 9, pp. 16114-16132, 2021, doi: 10.1109/ACCESS.2021.3053045.

URL: https://ieeexplore.ieee.org/document/9328769

CHAPTER 6

PERFORMANCE EVALUATION OF THE USER SEGREGATION SCHEME

The user segregation scheme, proposed in the previous chapter, is evaluated in this chapter.

We generated 340,000 learning sets (with input features mentioned in Table 5.1), 70% were used to train the model (i.e., training set) while the remainder 30% is equally partitioned for testing & validation. The dataset used for training can be accessed at GitHub (https://github.com/Saad7861004/Machine-Learning-For-Wireless-Cellular-Networks.git).

A standard technique used in the literature to define training and testing sets for validating the learning model is cross-validation [203]. Therefore, k-fold cross-validation was applied in this study with k=10, a commonly used value [204]. Other common values of k such as k = 3, 5 were explored showing similar trends.

The ML related results such as accuracy, loss, and Receiver Operating Characteristic (ROC) curves are obtained from the validation set whereas the test data is used to evaluate the system's performance. It should be noted that we are testing the proposed scheme for a multimedia application in a multicast scenario. Users are interested in the content of size 100kB similar to the previous section. Various user densities have been considered to demonstrate the performance and effect of increasing density on different parameters. Moreover, these users are interested in sharing a common interest. Hence, it does not represent the total users of a cell. As discussed earlier in Chapter 5, as a result of classification, all the users are not in clusters. Rather, based on the classification results, some users will be in clusters whereas the remainders are going to be communicating directly with the eNB. Users in clusters will be communicating in D2D mode since the CH fetches the requested content and deliver it to its cluster members whereas the rest will download the content directly from the eNB. All the simulation parameters are similar to the previous section, summarized in Table 4.1.

6.1 Hyperparameters Optimization

The objective is to find the optimal solution for each of the employed ML algorithms. To achieve this goal, we explored various values of the hyperparameters, mentioned in Table 6.1, for each of the classification techniques. Literature suggested that we can exploit the following search strategies to find the optimal solution. These strategies include; Random Search, Grid Search, Heuristic, and Exhaustive Search [205].

An extensive study on hyperparameter optimization [206] suggests that Random Search is a significantly better technique for various types of data sets as compared to Search Space, Grid Search, etc. It should be noted that many other related articles targeting wireless network applications are using the Random Search method for hyperparameters optimization as well [207]. We, therefore, applied Random Search to optimize the hyperparameters for all the ML algorithms. We train the learning model using the training data and use Random Search on k-fold cross-validation to tune the hyperparameters. All the values of hyperparameters are detailed in Table 6.1.

6.2 Complexity of the Trained Algorithms

It is mentioned earlier that the proposed study takes advantage of offline training. Only the trained classifier/algorithm will be implemented in a live network. Therefore, the proposed algorithm training does not affect network latency significantly. The total execution time for training the algorithms is detailed in Table 6.2.

The experiments were performed on a 64-bit Intel 4600 GPU, Core i7-4790 CPU @ 3.60 - 4 GHz processor. The system had 12 GB of RAM. MATLAB was used for the training purpose. SVM, RF, and DNN are standard classifiers. Therefore, the complexity of these algorithms in terms of hardware requirements and Big-O notations can be found in the literature [208-210] and are quoted in Table 6.2.

6.3 Comparison of Classification Techniques

It has been reported in the literature that SVM, RF, and DNN are useful for classification purposes especially for applications targeting wireless networks [211]. However, other classification techniques (e.g. Decision Trees, k-Nearest Neighbor, etc.) are available and investigated as well. The accuracy of each technique is detailed in Table 6.3. SVM, RF, and DNN give the best accuracy among all the classification techniques. In this context, accuracy is defined as the percentage of correctly classified users. As discussed in Chapter 5, the target was to select a user for one of the two groups based on the throughput maximization, and hence the output of three algorithms was compared with true labels to determine the accuracy of the classification. The results given in Table 6.3 show that we achieved better accuracy for SVM as compared to RF and DNN. Moreover, it is suggested in the literature that SVM and RF have a lower execution time as compared to DNN (supported by our findings shown in Table 6.2), therefore it presents a lower computational burden on the learning servers as well [210, 212].

Table 6.1 Hyperparameters Explored for Various Machine Learning Algorithms

Algorithm	Hyperparameters Explored		Optimized Solution/Hyperparameters
DNN	η (Learning Rate for Stochastic Gradient Descent)	$10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}$,1	Hidden Layer: 2 Number of Neurons:12, Gradient Descent: 10^{-3}
	Epochs	150	
	Batch Size	256	
	Hidden Layers	1,2,3,4,5	
	Activation Function	ReLU, Sigmoid	
	Weight and Bias Initiation	Random	
	Number of Neurons/Nodes in First & Second Hidden Layer	First Hidden Layer: 5 to 40 (step of 2) Second Hidden Layer: 0 to 40 (step of 2)	
SVM	Kernel	Linear, Gaussian, Polynomial, RBF, Sigmoid.	Kernel: Sigmoid C = 1 $\Upsilon = 10^{-1}$
	C	0.03, 0.1, 0.2, 0.3, 0.4. 1,2,3,4,5,10,100.	
	ϒ	$10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}$	
Random Forest	Number of Trees/Estimators	10, 50, 100, 200, 300	Number of Tress/Estimators: 200 Criterion: Gini Max_Depth: 5
	Max Depth	2, 5, 10, 20	
	Function/Criterion to evaluate each split	Gini, Entropy	
	Bootstrap	True, False	
	Min, samples leaf	[1,20]	
	Min. samples split	[2,20]	
Decision Trees	Learning Rates	0.0001, 0.001, 0.01, 0.1, 0.2, 0.3	Learning Rate = 0.01, Decision Tress = 300, Splitter = Random.
	Number of Decision Trees/ Estimators	100, 200, 300, 400, 500	
	Splitter	Best, Random	

Table 6.2 Comparison of Complexity for Different Machine Learning Algorithms

Algorithm	Multipliers	Adders	Complexity in General (Worst Case Running Time)	Total Execution Time (sec)
DNN	$\sum_{k=2}^{K} N_{k-1} N_k$	$\sum_{k=2}^{K} (N_{k-1} N_k) N$	O(L^4)	3401.5
SVM	$N_{SV} * M$	$2N_{SV} * M$	O($N_{SV} D$)	2307.1
RF	0 $(since\ it\ is\ a\ classifica$	$N_{trees} - 1$	O($M N_{trees}$)	1874

$$N_{SV} = number\ of\ support\ vectors,$$

$$M = total\ number\ of\ features,$$

$$L = total\ number\ of\ layers\ of\ DNN,$$

$$N_k = number\ of\ neurons\ in\ the\ kth\ layer,$$

$$D = data\ points, N_{trees} = number\ of\ trees.$$

Table 6.3 Comparison of Accuracy for Different Classification Techniques

Classification Techniques Accuracy (%)							
Decision Trees	Discriminant Analysis	k-Nearest Neighbor (k = 5)	Ensemble Classifier		DNN	RF	SVM
Medium Tree	Linear Quadratic	Weighted KNN	Boosted Trees	Bagged Trees			
73.3	76.3	61.4	74.3	73.6	85.8	93.33	96

6.4 Comparison of Accuracy and Loss of Trained Algorithms

It is important to note that the number of epochs cannot be pre-selected for a particular machine learning implementation. The number of epochs depend on the type of application and data. Overfitting is one of the critical issues of training the ML. To avoid overfitting and to increase the generalization of the trained model, the model should be trained for number of epochs that achieve this. Since the problem is a binary classification problem, the loss is calculated using binary cross-entropy. The number of epochs was selected to be 150 based on the satisfactory error rate and flattening of the curves. As the accuracy increases the misclassifications are reduced which show up as loss decreasing in Figure. 6.1 and Figure 6.2.

Figure. 6.1 presents the results of the three algorithms for testing data against different epochs. It can be observed that the accuracy of all three algorithms keeps on increasing with the increasing number of epochs. On the other hand, it can be seen in Figure. 6.2 that the loss values keep on decreasing with the increase in the number of epochs suggesting no overfitting. It is further elaborated in the next sub-section.

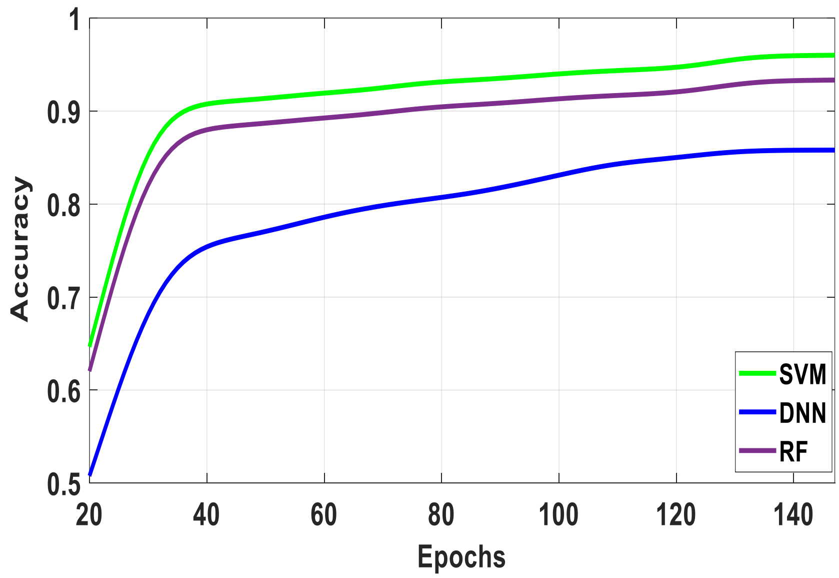

Figure. 6.1 Accuracy of the Trained Algorithms V/S Epochs.

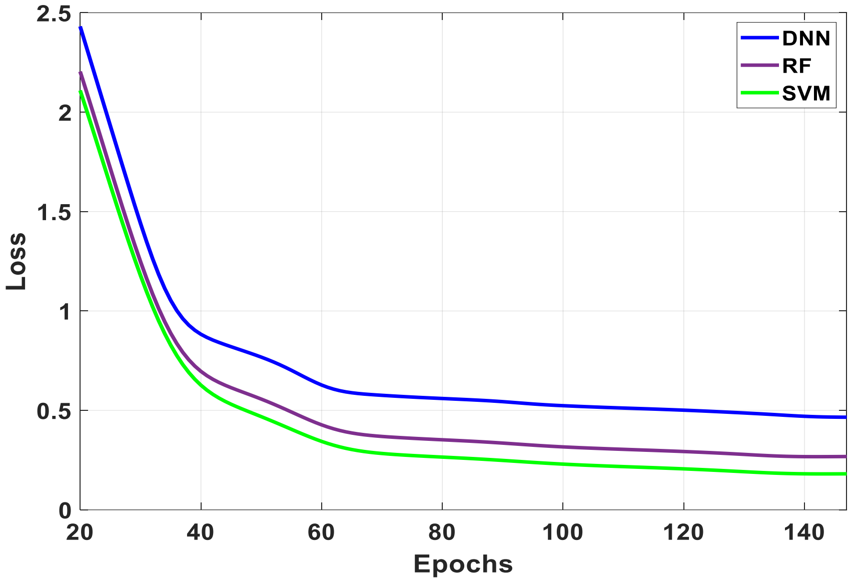

Figure. 6.2 Loss of the Trained Algorithms V/S Epochs.

6.5 ROC Curves

ROC curve as shown in Figure. 6.3 demonstrates that the model is not overfitting as the curves are not all into the left corner. Since the curves are closer to the true positive axis, therefore, it shows the false positive rate is well within limits. Moreover, it can be seen from Figure. 6.3 that the area under the SVM curve is greater than that of RF and DNN which is another indication of SVM being more accurate subject to the user segregation problem.

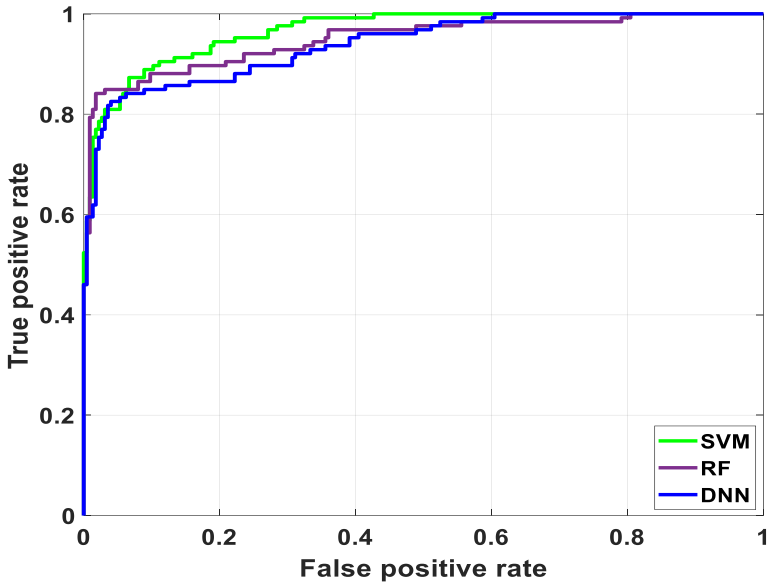

Figure. 6.3 The ROC Curves for the Trained Algorithm.

6.6 Analysis of the Performance Parameters

In this section, we explore the impact of the proposed mixed-mode scheme on the performance of the system. All the results shown in this section are based on the SVM classification since it is shown in the previous section that it produces the best classification results.

To demonstrate the performance, we have applied the user segregation scheme and segregated the users for the proposed clustering algorithm (presented in Chapter 3) and three other popular clustering algorithms. The clustering algorithm proposed in Chapter 3 is termed; "Multi-Factor"[182]. The rest of the three algorithms represent state-of-the-art and classical clustering techniques. The classical algorithm, K-Medoids and Density-Based Spatial Clustering of Applications With Noise (DBSCAN) [213] can be widely found in the literature whereas the work presented in [86], is termed as "EBC", described earlier in section 4.3. We deliberately selected these algorithms since they consider the social interest and it has been shown in the literature that it improves the performance of the network. Therefore, the proposed segregation concept is compared with the best performing algorithms (e.g. the proposed 'Multi-Factor' clustering algorithm is performing the best as shown in Chapter 4). The performance of the proposed scheme is benchmarked against that of the corresponding standard clustering scheme that keeps all the users within the clusters. Our results highlight that irrespective of the clustering algorithm, the proposed scheme significantly enhances the system's performance.

It should be noted that since we are considering different densities of socially aware nodes, so the number of clusters formed are once again following the Cal-Har criteria, described in section 4.4.

1. ***Throughput***

The throughput performance is demonstrated in Figure. 6.4. The solid line represents 'all in clusters' scenario whereas the dotted line in each case demonstrates the impact of the proposed scheme. It should be noted that dotted lines represent the same clustering approach as solid lines, the only difference is that some users (based on classification) are serviced by eNB. At the user density of one thousand, the user segregation scheme improves the throughput of the proposed Multi-Factor algorithm by 30%. In the case of classical schemes, the percentage increase in the throughput is approximately 42% and 34% for K-Medoids and DBSCAN respectively whereas for the EBC scheme the increase is approximately 30%. It is a significant improvement in all cases. The proposed scheme was able to achieve the best improvement for K-Medoids since it only considers distance for clustering users and distance only may not be the best metric for clustering since users in proximity may not have the best channel conditions due to various factors such as shadowing. However, in the proposed user segregation scheme, shadowing was considered for training the algorithms.

The throughput result is further elaborated in Figure. 6.5. It represents the Cumulative Distribution Function (CDF) of throughput for the Multi-Factor scheme. The two curves represent CDFs of the same algorithm but for 'all in clusters' against the proposed 'user segregation'. We are presenting the result for the Multi-Factor scheme only since it produces the best throughput as compared to the other three cases. It can be seen that the proposed scheme is performing better. A clear difference in the performance of the users can be seen at the 90th percentile. Similar trends were observed for all clustering schemes and other user densities.

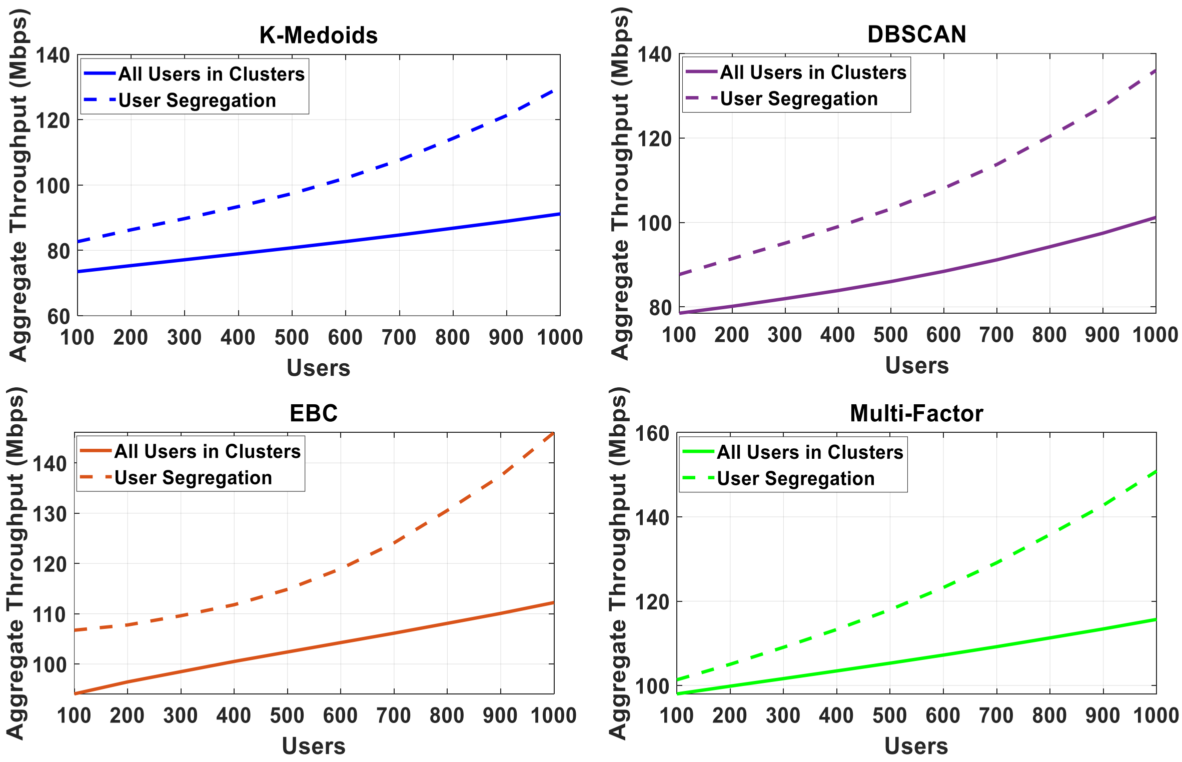

Figure. 6.4 Throughput Performance: A Comparison.

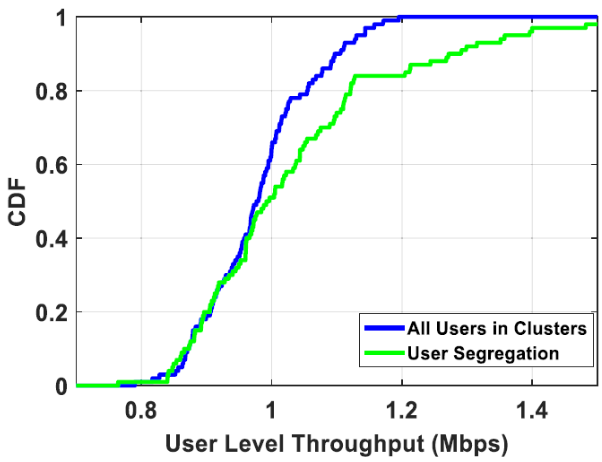

Figure. 6.5 Throughput CDF (100 Users). Similar trends were observed at other user densities with performance gap more pronounced.

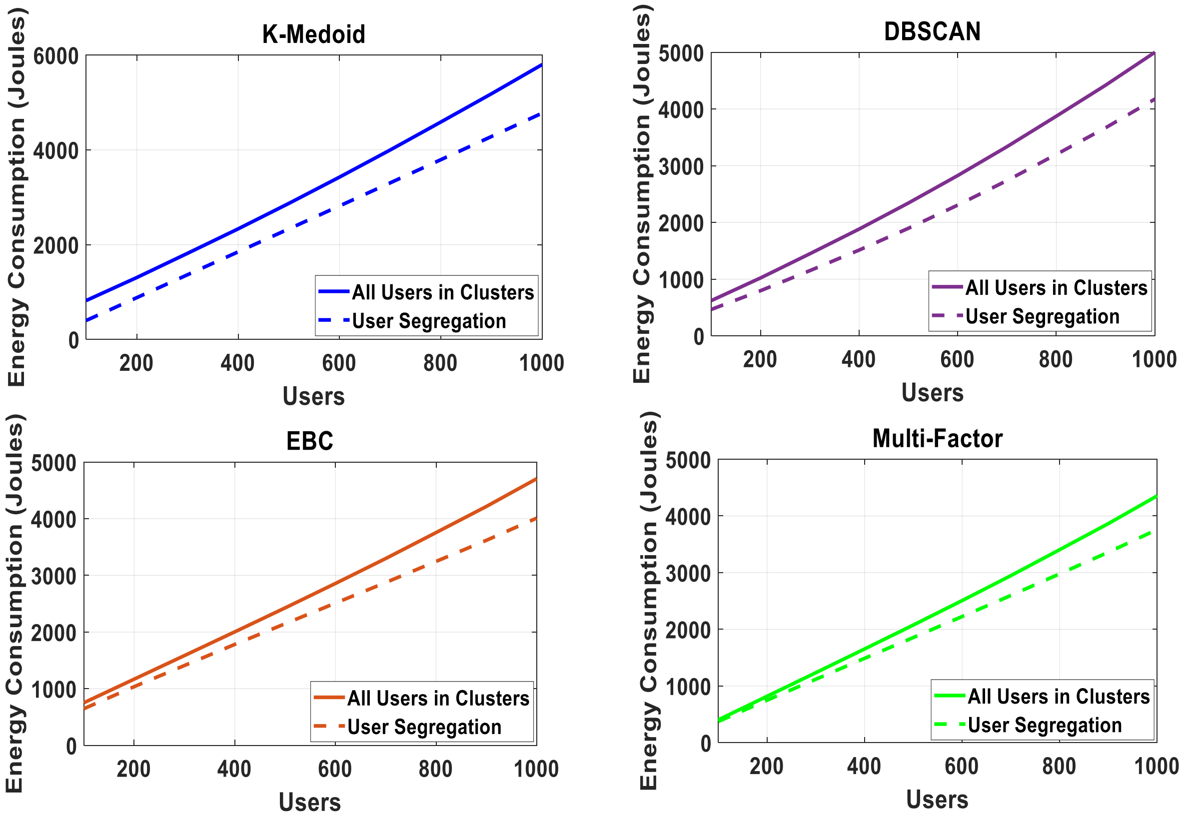

Figure. 6.6 Energy Consumption: A Comparison.

2. *Energy Consumption*

The result for the energy consumption of the users is demonstrated by Figure. 6.6. Downlink energy consumption is considered in this study. We have utilized the same model that was presented in Section 3.5.2 and used for evaluating the energy consumption of the proposed clustering algorithm in Section 4.3.

Since the result shown in Figure. 6.4 demonstrated that user segregation improves the aggregate throughput, therefore energy consumption will be reduced as well. The reduction in energy consumptions at a user density of one thousand is approximately 13.66% for the proposed Multi-Factor algorithm, whereas, for other schemes, the reduction is 17.67% for K-Medoids, 16.5% for DBSCAN, and 14.77% for EBC. The CDF of energy consumption of the proposed scheme against that of the Multi-Factor algorithm is shown in Figure. 6.7. The performance of the proposed scheme is better in most of the quartiles.

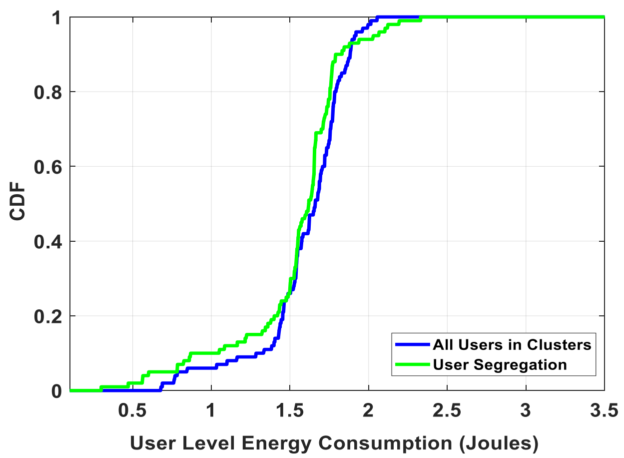

Figure. 6.7 Energy Consumption CDF (100 Users). Similar trends were observed at other user densities with performance gap more pronounced.

3. *Throughput Fairness*

Similar to Section 4.5, Jain's Fairness model (given in Equation (13)) was used to evaluate fairness performance. The result for throughput fairness is shown in Figure. 6.8. At the user density of one thousand, the proposed Multi-Factor algorithm shows an improvement of 20%. The improvement in fairness for other schemes is around 26% for both K-Medoids and DBSCAN, and 13.5% for EBC. Since we have already shown earlier that as compared to the 'all in clusters' scenario, user segregation improves achievable rates for a considerable percentage of users (precisely 20-30% users on average), as a result, distribution of throughput among the users is significantly improved therefore the system is fairer to the users. The result presented in Figure. 6.9 shows the CDF achieved with user segregation compared to standard clustering. It is clear that after the 10th percentile, fairness for the user segregation scenario is better in all percentiles. Therefore, better fairness is not achieved by only favoring a few users while neglecting a large number of users.

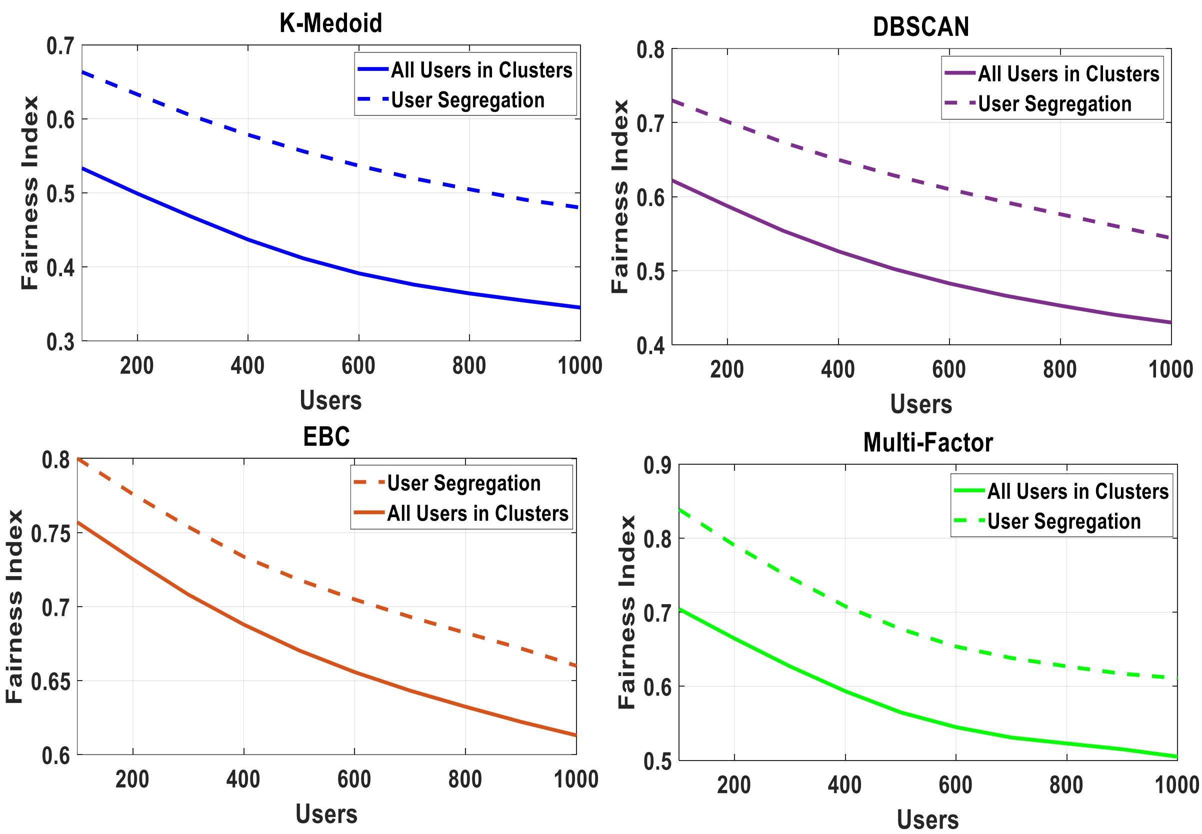

Figure. 6.8 Throughput Fairness: A Comparison.

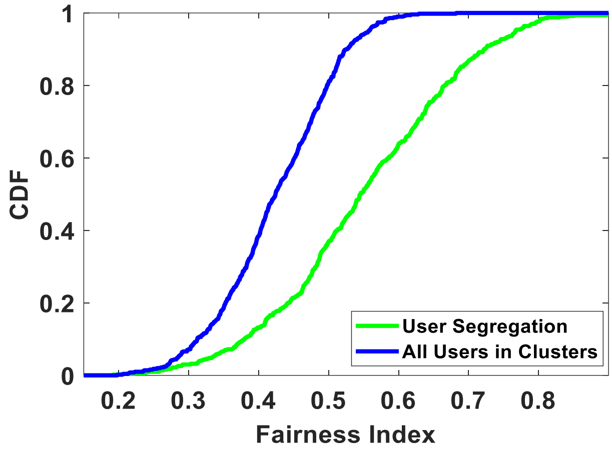

Figure. 6.9 Throughput Fairness CDF (100 Users). Similar trends were observed at other user densities with performance gap more pronounced.

6.7 Trade-Off Between Performance Parameters and eNB Loading

Our simulations suggest that the performance parameters are getting significantly improved at the expense of eNB loading. Hence, it becomes necessary to select an appropriate loading factor depending on the spare capacity of the eNB. To present this trade-off, we randomly selected various percentages of users from the total number of users for which the performance was improved as a result of applying user segregation. We considered one hundred different random combinations, calculated the performance parameters for each combination, and then averaged the results. Figure. 6.10 shows two performance bounds; an upper bound of performance improvement (topmost curve, user segregation with 100% loading) meaning all users who have been identified to be better off with the eNB are being serviced by the eNB, and the lower bound (not segregated, all users in clusters). However, we may select a certain percentage of users according to the spare capacity of the eNB. The three middle curves represent three different loading factors, 10%, 50%, and 80% (bottom to top: low loading, average loading, and high loading) of the total users for which the performance was improved.

Greater the loading factor, greater is the performance improvement. Therefore, it presents an opportunity for the cellular network to select a particular loading factor and trade it off with an improvement in the performance.

It should be noted that the result shown in Figure. 6.10 considers randomly selected users so it might disadvantage some users that are having the best performance. Therefore, to investigate this, we selected the same percentage of best users (i.e. the users that have the best improvement of all the users) and compared the performance with the random selection. Table 6.4 shows performance improvement is not significant, as we change the selection criteria from random to best users. Moreover, this percentage increase in performance is reported at a user density of one thousand. Therefore, this improvement will be even less at other user densities considered in this study. Similar marginal benefits (shown in Table 6.4) were observed for the other performance parameters i.e. energy consumption and throughput fairness. Therefore, we can conclude that a binary classifier is adequate, and training a multiclass classifier is not warranted.

Table 6.4 Random to Best Selection: Performance Improvement for Different Parameters

Clustering Scheme	**Aggregate Throughput (% Increase)**			**Energy Consumption (% Decrease)**			**Throughput Fairness (% Increase)**		
	Low Loading	Average Loading	High Loading	Low Loading	Average Loading	High Loading	Low Loading	Average Loading	High Loading
K-Medoid	2.74	3.03	3.59	1.11	1.66	2.18	1.88	2.11	2.67
DBSCAN	1.75	3.13	3.46	1.29	1.78	2.28	1.76	2.07	2.49
EBC	1.81	3.31	3.22	0.96	1.21	1.77	1.01	1.67	2.11
Multi-Factor (proposed)	0.83	2.79	2.86	0.67	0.98	1.27	1.56	1.97	2.32

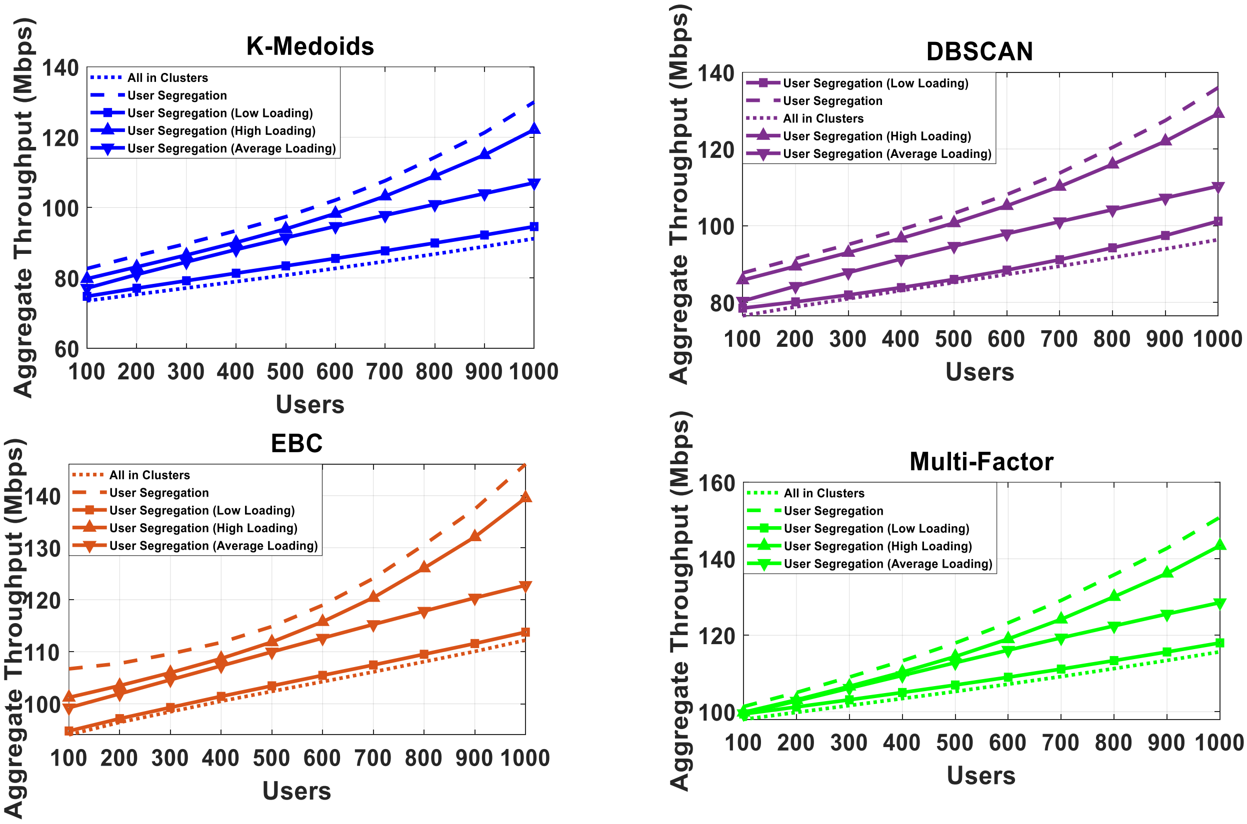

Figure. 6.10 Improvement in Aggregate Throughput for Different Loading Factors.

6.8 Summary of Results

The results demonstrated the impact of the proposed user segregation scheme on various performance parameters such as throughput, energy consumption, and fairness. Specifically, at a node density of one thousand, throughput gets improved by 30% for the proposed Multi-Factor clustering algorithm. The throughput of other clustering schemes improved as well, 42% for K-Medoids, 34% for DBSCAN, and approximately 30% for EBC. Energy consumption of the Multi-Factor clustering algorithm was reduced by 13.66% whereas the reduction for other algorithms was: 17.67% for K-Medoids, 16.5% for DBSCAN, and 14.77% for EBC. Throughput fairness showed improvement by 20% for the proposed Multi-Factor clustering algorithm, approximately 26% for both K-Medoids and DBSCAN, and 13.5% for EBC. All these results demonstrate that the user segregation scheme improves the performance irrespective of the clustering algorithm.

As a result of segregation, some users communicate directly to the eNB, therefore we presented a trade-off in performance improvement for various loading factors. The margin of improvement can be selected based on eNB's loading capability and spare capacity. This study also demonstrated that as compared to DNN and RF, SVM performs better with relatively smaller training samples subject to classification scenarios.

6.9 Related Publication

The work presented in this chapter has been published in the following research article:

S. Aslam, F. Alam, S. F. Hasan and M. A. Rashid, "A Machine Learning Approach to Enhance the Performance of D2D-Enabled Clustered Networks," in *IEEE Access*, vol. 9, pp. 16114-16132, 2021, doi: 10.1109/ACCESS.2021.3053045.

URL: https://ieeexplore.ieee.org/document/9328769

CHAPTER 7

CONCLUSION AND FUTURE RESEARCH DIRECTIONS

The exponential increase in the demand for multimedia traffic poses a significant challenge for current cellular networks and is one of the main drivers for the next generation of cellular networks. There are many solutions proposed for next generation cellular networks that either try to increase the efficiency of the available resources or aim at providing new radio resources or infrastructures. D2D communication is a good example of such proposed solutions, by which a user communicates directly to its receiver bypassing eNB. There are different ways to integrate D2D communications in a network. In this thesis, clustering was employed to integrate D2D with the cellular network. We designed as well as optimized the clustering algorithm that showed significant performance gains for the D2D network.

The thesis presented a content-sharing framework for D2D communication in a multicasting scenario. Content-Centric Networking and Network Virtualization were utilized to propose a distributive architecture. A novel weighted clustering algorithm was incorporated into the proposed architecture. Various performance parameters (such as Throughput, Energy Consumption, ASE, and Throughput Fairness) have been considered to evaluate the performance of the proposed algorithm for a content-sharing scenario. To the best of the author's knowledge, all these performance parameters have not been targeted by any single research work. This study showed the significance of considering spatial distribution and social ties on different parameters and established that both are vital for enhancing the performance of the Content-Centric Network.

To further enhance the performance of the clustered network, a mixed-mode clustering scheme was proposed based on user segregation. The concept relies on the fact that all users should not be part of a cluster as there are always some users that are better served by the eNB. We applied ML algorithms to perform this classification and compared the accuracy of different classification techniques. SVM, RF, and DNN were found to be the most promising classifiers. The results shown for accuracy, loss, and ROC demonstrate the effectiveness of the trained algorithm for the proposed scheme. The trained model was tested on a D2D-enabled content-sharing multicasting scenario. As per the classification outcome, a portion of the users were directly fetching the required content from the eNB.

7.1 Contribution

The contributions of this thesis are in two major parts: (i) decentralized network architecture & clustering algorithm, (ii) optimization of the clustering process. Specifically:

1) Development of a Novel Clustering Algorithm

A novel multi-factor weighted clustering has been proposed. The performance of the proposed algorithm is shown to be superior compared to the five benchmarked algorithms. The designed clustering algorithm consists of various clustering metrics attached to their respective weights. These weights can be adjusted to suit the system's requirements. This flexibility in trading off the performance with respect to various parameters is not available for existing algorithms. The benchmarked algorithms are tested for throughput fairness which has not been reported in the literature on clustering. Moreover, different from the existing works, the impact of the number of clusters on energy consumption and area spectral efficiency is also demonstrated.

2) *Distributed Architecture*

A distributed architecture is proposed that is effectively supported by hash functions to identify the socially connected users. It also supports the designed clustering algorithm.

3) *Development of User Segregation Scheme using ML classifiers*

A user segregation scheme targeting D2D clustering has not been reported in the literature. Our work clearly shows that substantial improvement in terms of throughput, energy consumption, and fairness can be achieved as a result of applying user segregation. It should be noted that we applied this concept to the proposed clustering scheme as well as three other algorithms, and it improves the performance of every algorithm.

A binary classification model has been designed and trained to identify the users that should be in clusters while the rest communicate directly to eNB. This model is trained completely offline and therefore does not increase the workload of the central controller. Moreover, owing to the offline training, explicit network measurements of the live network are not required and hence network latency is not substantially affected. Moreover, our results also show that binary classification is adequate, and training a multiclass classifier is not warranted.

Multiple machine learning algorithms namely, Support Vector Machines (SVM), Deep Neural Network (DNN), and Random Forest (RF) are investigated to ascertain their suitability as classifiers for user segregation.

4) Identification of Data Collection Opportunities in a Cellular Network

We have explored and identified various data collection opportunities in a cellular network for constructing the machine learning training corpus. These opportunities are outlined with respect to the user segregation problem as well.

5) Trade-off Between eNB loading and Performance Improvement

This work also demonstrates the trade-off between eNB loading and performance improvement. It provides an opportunity for the cellular network to select an improvement factor based on the serving capacity of the eNB.

7.2 Future Research Directions

Based on the assumptions, results, and analysis presented in this thesis, we provide several directions for future research as listed below:

- We empirically selected weights of the proposed Multi-factor clustering algorithm. Future research should explore developing algorithms to select optimum weights. ML based regressors can be used for such weight selection.
- The performance of the proposed clustering algorithm needs to be evaluated for multiple multimedia applications.
- Additionally, the comparative study of the signaling overhead required for implementing the proposed algorithm needs to be explored as well.
- Moreover, while this study provided an approach to finding the number of clusters, one of the future directions could be a comprehensive study of optimal selection of the number of clusters. This optimal selection should consider the geographical distribution of users, target performance parameters, etc.

- We trained the ML algorithms on simulated data. It would be interesting to compare the results obtained from ML algorithms trained on live network data. Common data sets/training corpus from live networks need to be available to perform benchmarking and evaluate the effectiveness of the trained algorithms.
- Moreover, ML can predict multimedia traffic demand, especially for social events. Therefore, having information on the anticipated traffic load would make the user segregation process even more optimized. The resource allocation would be more effective as well.
- Finally, going further, the proposed user segregation method should be explored for other applications as well. For instance, segregating a group of users that act as relays for coverage extension or in case of infrastructure failure, segregating users that can provide emergency services for public safety.

REFERENCES

[1] J. T. Penttinen, "3GPP Mobile Communications: LTE/SAE and LTE-A," 2013.

[2] Y. Benchaabene, N. Boujnah, and F. Zarai, "5G cellular: Survey on some challenging techniques," in *2017 18th International Conference on Parallel and Distributed Computing, Applications and Technologies (PDCAT)*, 2017: IEEE, pp. 348-353.

[3] S. Zhang, J. Liu, H. Guo, M. Qi, and N. Kato, "Envisioning device-to-device communications in 6G," *IEEE Network,* vol. 34, no. 3, pp. 86-91, 2020.

[4] Y. He, J. Ren, G. Yu, and Y. Cai, "D2D communications meet mobile edge computing for enhanced computation capacity in cellular networks," *IEEE Transactions on Wireless Communications,* vol. 18, no. 3, pp. 1750-1763, 2019.

[5] K. David and H. Berndt, "6G vision and requirements: Is there any need for beyond 5G?," *IEEE Vehicular Technology Magazine,* vol. 13, no. 3, pp. 72-80, 2018.

[6] Z. Zhang *et al.*, "6G wireless networks: Vision, requirements, architecture, and key technologies," *IEEE Vehicular Technology Magazine,* vol. 14, no. 3, pp. 28-41, 2019.

[7] A. Asadi, Q. Wang, and V. Mancuso, "A survey on device-to-device communication in cellular networks," *IEEE Communications Surveys & Tutorials,* vol. 16, no. 4, pp. 1801-1819, 2014.

[8] Y.-D. Lin and Y.-C. Hsu, "Multihop cellular: A new architecture for wireless communications," in *Proceedings IEEE INFOCOM 2000. Conference on Computer Communications. Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies (Cat. No. 00CH37064)*, 2000, vol. 3: IEEE, pp. 1273-1282.

[9] X. Wu *et al.*, "FlashLinQ: A synchronous distributed scheduler for peer-to-peer ad hoc networks," *IEEE/ACM Transactions on networking,* vol. 21, no. 4, pp. 1215-1228, 2013.

[10] G. Fodor, S. Sorrentino, and S. Sultana, "Network assisted device-to-device communications: use cases, design approaches, and performance aspects," in *Smart Device to Smart Device Communication*: Springer, 2014, pp. 135-163.

[11] A. Aijaz, H. Aghvami, and M. Amani, "A survey on mobile data offloading: technical and business perspectives," *IEEE Wireless Communications,* vol. 20, no. 2, pp. 104-112, 2013.

[12] G. Fodor *et al.*, "An overview of device-to-device communications technology components in METIS," *Ieee Access,* vol. 4, pp. 3288-3299, 2016.

[13] O. Bello and S. Zeadally, "Intelligent device-to-device communication in the internet of things," *IEEE Systems Journal,* vol. 10, no. 3, pp. 1172-1182, 2014.

[14] X. Lin, J. G. Andrews, A. Ghosh, and R. Ratasuk, "An overview of 3GPP device-to-device proximity services," *IEEE Communications Magazine,* vol. 52, no. 4, pp. 40-48, 2014.

[15] S. Y. Lien and H. L. Tsai, "D2D in Vehicular Networks," *Wiley 5G Ref: The Essential 5G Reference Online,* pp. 1-18, 2019.

[16] X. Cheng, L. Yang, and X. Shen, "D2D for intelligent transportation systems: A feasibility study," *IEEE Transactions on Intelligent Transportation Systems,* vol. 16, no. 4, pp. 1784-1793, 2015.

[17] S.-Y. Lien, C.-C. Chien, F.-M. Tseng, and T.-C. Ho, "3GPP device-to-device communications for beyond 4G cellular networks," *IEEE Communications Magazine,* vol. 54, no. 3, pp. 29-35, 2016.

[18] J. Seppälä, T. Koskela, T. Chen, and S. Hakola, "Network controlled device-to-device (D2D) and cluster multicast concept for LTE and LTE-A networks," in *2011 IEEE Wireless Communications and Networking Conference*, 2011: IEEE, pp. 986-991.

[19] A. S. Rostami, M. Badkoobe, F. Mohanna, A. A. R. Hosseinabadi, and A. K. Sangaiah, "Survey on clustering in heterogeneous and homogeneous wireless sensor networks," *The Journal of Supercomputing,* vol. 74, no. 1, pp. 277-323, 2018.

[20] M. F. Khan, K.-L. A. Yau, R. M. Noor, and M. A. Imran, "Survey and taxonomy of clustering algorithms in 5G," *Journal of Network and Computer Applications,* vol. 154, p. 102539, 2020.

[21] G. Forecast, "Cisco visual networking index: global mobile data traffic forecast update, 2017–2022," *Update,* vol. 2017, p. 2022, 2019.

[22] A. Ericsson, "Ericsson mobility report June 2018," *Recuperado de: https://www. ericsson. com/assets/local/mobility-report/documents/2018/ericsson-mobility-report-june-2018. pdf (Accedido el 09/09/2018),* pp. 3-13, 2018.

[23] C. V. N. Index, "Global mobile data traffic forecast update, 2015–2020," *Cisco white paper,* p. 9, 2016.

[24] A. Ericsson, "Ericsson mobility report: On the pulse of the networked society," *Ericsson, Sweden, Tech. Rep. EAB-14,* vol. 61078, 2015.

[25] Y. Gu, L. X. Cai, M. Pan, L. Song, and Z. Han, "Exploiting the stable fixture matching game for content sharing in D2D-based LTE-V2X communications," in *2016 IEEE Global Communications Conference (GLOBECOM)*, 2016: IEEE, pp. 1-6.

[26] S. Chen *et al.*, "Vehicle-to-everything (V2X) services supported by LTE-based systems and 5G," *IEEE Communications Standards Magazine,* vol. 1, no. 2, pp. 70-76, 2017.

[27] B. Peng, T. Peng, Z. Liu, Y. Yang, and C. Hu, "Cluster-based multicast transmission for device-to-device (D2D) communication," in *2013 IEEE 78th Vehicular Technology Conference (VTC Fall)*, 2013: IEEE, pp. 1-5.

[28] E. Yaacoub and O. Kubbar, "Energy-efficient device-to-device communications in LTE public safety networks," in *2012 IEEE globecom workshops*, 2012: IEEE, pp. 391-395.

[29] B. Narottama, A. Fahmi, and B. Syihabuddin, "Impact of number of devices and data rate variation in clustering method on device-to-device communication," in *2015 IEEE Asia Pacific Conference on Wireless and Mobile (APWiMob)*, 2015: IEEE, pp. 233-238.

[30] B. Zhou, H. Hu, S.-Q. Huang, and H.-H. Chen, "Intracluster device-to-device relay algorithm with optimal resource utilization," *IEEE transactions on vehicular technology,* vol. 62, no. 5, pp. 2315-2326, 2013.
[31] L. Militano, M. Condoluci, G. Araniti, A. Molinaro, and A. Iera, "When D2D communication improves group oriented services in beyond 4G networks," *Wireless Networks,* vol. 21, no. 4, pp. 1363-1377, 2015.
[32] L. Militano, M. Condoluci, G. Araniti, A. Molinaro, A. Iera, and F. H. Fitzek, "Wi-Fi cooperation or D2D-based multicast content distribution in LTE-A: A comparative analysis," in *2014 IEEE International Conference on Communications Workshops (ICC)*, 2014: IEEE, pp. 296-301.
[33] D. Li and H. Zhao, "An analytical approach to relay access delay in multihop relay chain of vanets," *International Journal of Distributed Sensor Networks,* vol. 11, no. 3, p. 836106, 2015.
[34] M. Peng, Q. Hu, X. Xie, Z. Zhao, and H. V. Poor, "Network coded multihop wireless communication networks: Channel estimation and training design," *IEEE Journal on Selected Areas in Communications,* vol. 33, no. 2, pp. 281-294, 2014.
[35] X. Chen, B. Proulx, X. Gong, and J. Zhang, "Exploiting social ties for cooperative D2D communications: A mobile social networking case," *IEEE/ACM Transactions on Networking,* vol. 23, no. 5, pp. 1471-1484, 2014.
[36] Y. Zhang, E. Pan, L. Song, W. Saad, Z. Dawy, and Z. Han, "Social network enhanced device-to-device communication underlaying cellular networks," in *2013 IEEE/CIC International Conference on Communications in China-Workshops (CIC/ICCC)*, 2013: IEEE, pp. 182-186.
[37] H. Tang, Z. Ding, S. B. Yoo, and J. Hämäläinen, "Outage constrained joint precoding for D2D underlay cellular networks," in *2013 IEEE Global Communications Conference (GLOBECOM)*, 2013: IEEE, pp. 3540-3545.
[38] A. Asadi and V. Mancuso, "Energy efficient opportunistic uplink packet forwarding in hybrid wireless networks," in *Proceedings of the fourth international conference on Future energy systems*, 2013, pp. 261-262.
[39] C. Choi, S. Park, and D.-h. Cho, "User-cooperation scheme based on clustering for energy efficiency in cellular networks with D2D communication," in *2014 IEEE 25th Annual International Symposium on Personal, Indoor, and Mobile Radio Communication (PIMRC)*, 2014: IEEE, pp. 1365-1369.
[40] A. Asadi and V. Mancuso, "On the compound impact of opportunistic scheduling and D2D communications in cellular networks," in *Proceedings of the 16th ACM international conference on Modeling, analysis & simulation of wireless and mobile systems*, 2013, pp. 279-288.
[41] G. Rigazzi, N. K. Pratas, P. Popovski, and R. Fantacci, "Aggregation and trunking of M2M traffic via D2D connections," in *2015 IEEE international conference on communications (ICC)*, 2015: IEEE, pp. 2973-2978.
[42] A. Asadi and V. Mancuso, "WiFi Direct and LTE D2D in action," in *2013 IFIP Wireless Days (WD)*, 2013: IEEE, pp. 1-8.
[43] Z. Wu, L. Wang, G. Araniti, and Z. Han, "Exploiting social-interest interactions on user clustering and content dissemination in device-to-device

communications," in *2015 IEEE/CIC International Conference on Communications in China (ICCC)*, 2015: IEEE, pp. 1-6.
[44] J. Liu, L. Fu, J. Zhang, X. Wang, and J. Xu, "Modeling multicast group in wireless social networks: A combination of geographic and non-geographic perspective," *IEEE Transactions on Wireless Communications,* vol. 16, no. 6, pp. 4023-4037, 2017.
[45] L. F. Del Carpio, A. A. Dowhuszko, O. Tirkkonen, and G. Wu, "Simple clustering methods for multi-hop cooperative device-to-device communication," in *2015 IEEE 81st Vehicular Technology Conference (VTC Spring)*, 2015: IEEE, pp. 1-6.
[46] J. Guo, L. Moalic, J.-N. Martin, and A. Caminada, "Cluster resource assignment algorithm for Device-to-Device networks based on graph coloring," in *2017 13th International Wireless Communications and Mobile Computing Conference (IWCMC)*, 2017: IEEE, pp. 1700-1705.
[47] A. Asadi and V. Mancuso, "Network-assisted outband D2D-clustering in 5G cellular networks: Theory and practice," *IEEE Transactions on Mobile Computing,* vol. 16, no. 8, pp. 2246-2259, 2016.
[48] V. Mancuso, A. Asadi, and P. Jacko, "Tie-breaking can maximize fairness without sacrificing throughput in D2D-assisted networks," in *2016 IEEE 17th International Symposium on A World of Wireless, Mobile and Multimedia Networks (WoWMoM)*, 2016: IEEE, pp. 1-9.
[49] C. Saha and H. S. Dhillon, "D2D underlaid cellular networks with user clusters: Load balancing and downlink rate analysis," in *2017 IEEE Wireless Communications and Networking Conference (WCNC)*, 2017: IEEE, pp. 1-6.
[50] Z. Zhou, M. Dong, K. Ota, B. Gu, and T. Sato, "Stackelberg-game based distributed energy-aware resource allocation in device-to-device communications," in *2014 IEEE International Conference on Communication Systems*, 2014: IEEE, pp. 11-15.
[51] L. Feng *et al.*, "Resource allocation for 5G D2D multicast content sharing in social-aware cellular networks," *IEEE Communications Magazine,* vol. 56, no. 3, pp. 112-118, 2018.
[52] L. Song, D. Niyato, Z. Han, and E. Hossain, "Game-theoretic resource allocation methods for device-to-device communication," *IEEE Wireless Communications,* vol. 21, no. 3, pp. 136-144, 2014.
[53] C. Xia, S. Xu, and K. S. Kwak, "Resource allocation for device-to-device communication in LTE-A network: A stackelberg game approach," in *2014 IEEE 80th Vehicular Technology Conference (VTC2014-Fall)*, 2014: IEEE, pp. 1-5.
[54] Z. Zhou, M. Dong, K. Ota, R. Shi, Z. Liu, and T. Sato, "Game-theoretic approach to energy-efficient resource allocation in device-to-device underlay communications," *IET Communications,* vol. 9, no. 3, pp. 375-385, 2015.
[55] G. Zhang, K. Yang, and H.-H. Chen, "Socially aware cluster formation and radio resource allocation in D2D networks," *IEEE Wireless Communications,* vol. 23, no. 4, pp. 68-73, 2016.

[56] G. I. Tsiropoulos, A. Yadav, M. Zeng, and O. A. Dobre, "Cooperation in 5G HetNets: Advanced spectrum access and D2D assisted communications," *IEEE Wireless Communications,* vol. 24, no. 5, pp. 110-117, 2017.

[57] J. Liu, N. Kato, J. Ma, and N. Kadowaki, "Device-to-device communication in LTE-advanced networks: A survey," *IEEE Communications Surveys & Tutorials,* vol. 17, no. 4, pp. 1923-1940, 2014.

[58] E. Semenciuc, A. Pastrav, T. Palade, and E. Puschita, "Performance evaluation of a Cloud-based QoS support mechanism," in *2016 International Conference on Communications (COMM)*, 2016: IEEE, pp. 243-246.

[59] E. Puschita, T. Palade, A. Moldovan, R. Colda, and I. Vermesan, "An innovative qos paradigm based on cognitive in-network management of resources for a future unified network architecture: I-name qos model," in *2010 Second International Conference on Advances in Future Internet*, 2010: IEEE, pp. 37-43.

[60] Y. Wu *et al.*, "Challenges of mobile social device caching," *IEEE Access,* vol. 4, pp. 8938-8947, 2016.

[61] C. Liang, F. R. Yu, and X. Zhang, "Information-centric network function virtualization over 5G mobile wireless networks," *IEEE network,* vol. 29, no. 3, pp. 68-74, 2015.

[62] P. Lin, K. S. Khan, Q. Song, and A. Jamalipour, "Caching in heterogeneous ultradense 5G networks: A comprehensive cooperation approach," *IEEE Vehicular Technology Magazine,* vol. 14, no. 2, pp. 22-32, 2019.

[63] X. Zhang and Q. Zhu, "Information-centric virtualization for software-defined statistical QoS provisioning over 5G multimedia big data wireless networks," *IEEE Journal on Selected Areas in Communications,* vol. 37, no. 8, pp. 1721-1738, 2019.

[64] V. Jacobson, D. K. Smetters, J. D. Thornton, M. F. Plass, N. H. Briggs, and R. L. Braynard, "Networking named content," in *Proceedings of the 5th international conference on Emerging networking experiments and technologies*, 2009, pp. 1-12.

[65] D.-h. Kim, J.-h. Kim, Y.-s. Kim, H.-s. Yoon, and I. Yeom, "Mobility support in content centric networks," in *Proceedings of the second edition of the ICN workshop on Information-centric networking*, 2012, pp. 13-18.

[66] Z. Zhou, C. Gao, C. Xu, Y. Zhang, S. Mumtaz, and J. Rodriguez, "Social big-data-based content dissemination in internet of vehicles," *IEEE Transactions on Industrial Informatics,* vol. 14, no. 2, pp. 768-777, 2017.

[67] Z. Zhou, H. Yu, C. Xu, Y. Zhang, S. Mumtaz, and J. Rodriguez, "Dependable content distribution in D2D-based cooperative vehicular networks: A big data-integrated coalition game approach," *IEEE Transactions on Intelligent Transportation Systems,* vol. 19, no. 3, pp. 953-964, 2018.

[68] Y. Zhu, X. Qin, and P. Zhang, "An efficient multicast clustering scheme for D2D assisted offloading in cellular networks," in *2018 IEEE/CIC International Conference on Communications in China (ICCC)*, 2018: IEEE, pp. 480-484.

[69] D. Wu, J. Yan, H. Wang, D. Wu, and R. Wang, "Social attribute aware incentive mechanism for device-to-device video distribution," *IEEE Transactions on Multimedia,* vol. 19, no. 8, pp. 1908-1920, 2017.

[70] K. Montuno and Y. Zhacfi, "Fairness of resource allocation in cellular networks: A survey," *Resource allocation in next generation wireless networks,* vol. 5, p. 249, 2006.
[71] H. Jin, B. C. Jung, and V. C. Leung, "Fundamental limits of CDF-based scheduling: Throughput, fairness, and feedback overhead," *IEEE/ACM Transactions on Networking,* vol. 23, no. 3, pp. 894-907, 2014.
[72] C. Wang and Y.-C. Huang, "Delay-scheduler coupled throughput-fairness resource allocation algorithm in the long-term evolution wireless networks," *IET Communications,* vol. 8, no. 17, pp. 3105-3112, 2014.
[73] X. Sun and L. Dai, "Towards fair and efficient spectrum sharing between lte and wifi in unlicensed bands: fairness-constrained throughput maximization," *IEEE Transactions on Wireless Communications,* vol. 19, no. 4, pp. 2713-2727, 2020.
[74] L. Xu, G. M. O'Hare, and R. Collier, "A balanced energy-efficient multihop clustering scheme for wireless sensor networks," in *2014 7th IFIP Wireless and Mobile Networking Conference (WMNC)*, 2014: IEEE, pp. 1-8.
[75] L. Militano, M. Condoluci, G. Araniti, A. Molinaro, A. Iera, and G.-M. Muntean, "Single frequency-based device-to-device-enhanced video delivery for evolved multimedia broadcast and multicast services," *IEEE Transactions on Broadcasting,* vol. 61, no. 2, pp. 263-278, 2015.
[76] M. I. Ashraf, M. Bennis, W. Saad, M. Katz, and C.-S. Hong, "Dynamic clustering and user association in wireless small-cell networks with social considerations," *IEEE Transactions on Vehicular Technology,* vol. 66, no. 7, pp. 6553-6568, 2016.
[77] M. Afshang, H. S. Dhillon, and P. H. J. Chong, "Modeling and performance analysis of clustered device-to-device networks," *IEEE Transactions on Wireless Communications,* vol. 15, no. 7, pp. 4957-4972, 2016.
[78] M. Afshang, H. S. Dhillon, and P. H. J. Chong, "Fundamentals of cluster-centric content placement in cache-enabled device-to-device networks," *IEEE Transactions on Communications,* vol. 64, no. 6, pp. 2511-2526, 2016.
[79] Z. Zhang, L. Wang, and J. Zhang, "Energy efficiency of D2D multi-user cooperation," *Sensors,* vol. 17, no. 4, p. 697, 2017.
[80] S. Doumiati, H. Artail, and K. Kabalan, "A framework for clustering LTE devices for implementing group D2D communication and multicast capability," in *2017 8th International Conference on Information and Communication Systems (ICICS)*, 2017: IEEE, pp. 216-221.
[81] W. Yi, Y. Liu, and A. Nallanathan, "Modeling and analysis of D2D millimeter-wave networks with poisson cluster processes," *IEEE Transactions on Communications,* vol. 65, no. 12, pp. 5574-5588, 2017.
[82] Z. Xia, J. Yan, and Y. Liu, "Cooperative content delivery in multicast multihop device-to-device networks," *IEEE Access,* vol. 5, pp. 6314-6324, 2017.
[83] Y. Niu, Y. Liu, Y. Li, X. Chen, Z. Zhong, and Z. Han, "Device-to-device communications enabled energy efficient multicast scheduling in mmWave small cells," *IEEE Transactions on Communications,* vol. 66, no. 3, pp. 1093-1109, 2017.

[84] H. Ren, F. Jiang, and H. Wang, "Resource allocation based on clustering algorithm for hybrid device-to-device networks," in *2017 9th International Conference on Wireless Communications and Signal Processing (WCSP)*, 2017: IEEE, pp. 1-6.

[85] K. Kitagawa, H. Homma, Y. Suegara, and Y. Kishi, "A user selection algorithm for D2D multicast communication underlaying cellular systems," in *2017 IEEE Wireless Communications and Networking Conference (WCNC)*, 2017: IEEE, pp. 1-6.

[86] M. M. Tulu, R. Hou, C. Li, and M. D. Amentie, "Cluster head selection method for content-centric mobile social network in 5G," *IET Communications,* vol. 12, no. 4, pp. 402-408, 2017.

[87] L. Xu, G. M. O'Hare, and R. Collier, "A smart and balanced energy-efficient multihop clustering algorithm (smart-beem) for mimo iot systems in future networks," *Sensors,* vol. 17, no. 7, p. 1574, 2017.

[88] J. Huang, P. Zhou, K. Luo, Z. Yang, and G. He, "Two-stage resource allocation scheme for three-tier ultra-dense network," *China Communications,* vol. 14, no. 10, pp. 118-129, 2017.

[89] Y.-y. Li, Z.-g. Ma, W.-b. Yan, and L.-y. Liu, "Weighted graph based clustering and local mobility management for dense small cell network with x2 interface," *Wireless Personal Communications,* vol. 95, no. 4, pp. 3771-3783, 2017.

[90] X. Duan, Y. Liu, and X. Wang, "SDN enabled 5G-VANET: Adaptive vehicle clustering and beamformed transmission for aggregated traffic," *IEEE Communications Magazine,* vol. 55, no. 7, pp. 120-127, 2017.

[91] E. Yaacoub, "Cooperative energy efficient D2D clustering in LTE-A with enhanced QoS," *Telecommunication Systems,* vol. 67, no. 3, pp. 401-414, 2018.

[92] Y. Zhao and W. Song, "Energy-aware incentivized data dissemination via wireless D2D communications with weighted social communities," *IEEE Transactions on Green Communications and Networking,* vol. 2, no. 4, pp. 945-957, 2018.

[93] S. Sharafeddine and O. Farhat, "A proactive scalable approach for reliable cluster formation in wireless networks with D2D offloading," *Ad Hoc Networks,* vol. 77, pp. 42-53, 2018.

[94] L. Yang, D. Wu, S. Xu, G. Zhang, and Y. Cai, "Social-energy-aware user clustering for content sharing based on D2D multicast communications," *IEEE Access,* vol. 6, pp. 36092-36104, 2018.

[95] H. Huang, B. Liu, M. Hu, Y. Tao, and W. Xiang, "User-information-aware D2D multicast file distribution mechanism," *Sensors,* vol. 18, no. 10, p. 3389, 2018.

[96] M. A. Rahman, Y. Lee, and I. Koo, "Energy-efficient power allocation and relay selection schemes for relay-assisted d2d communications in 5g wireless networks," *Sensors,* vol. 18, no. 9, p. 2865, 2018.

[97] R. Amer, M. M. Butt, M. Bennis, and N. Marchetti, "Inter-cluster cooperation for wireless D2D caching networks," *IEEE Transactions on Wireless Communications,* vol. 17, no. 9, pp. 6108-6121, 2018.

[98] S. Pizzi, F. Rinaldi, A. Molinaro, A. Iera, and G. Araniti, "Energy-Efficient Multicast Service Delivery Exploiting Single Frequency Device-To-Device Communications in 5G New Radio Systems," *Sensors,* vol. 18, no. 7, p. 2205, 2018.

[99] S. Aslam, F. Alam, S. F. Hasan, and M. Rashid, "Performance Analysis of Clustering Algorithms for Content-Sharing Based D2D Enabled 5G Networks," in *2019 29th International Telecommunication Networks and Applications Conference (ITNAC)*, 2019: IEEE, pp. 1-7.

[100] R. Shi, B. Wang, F. Shi, D. Piao, and J. Hu, "A Design of D2D-Clustering Algorithm for Group D2D Communication," in *International Conference on Communications and Networking in China*, 2019: Springer, pp. 356-367.

[101] Y. Wu, D. Wu, L. Yang, X. Shi, L. Ao, and Q. Fu, "Matching-coalition based cluster formation for D2D multicast content sharing," *IEEE Access,* vol. 7, pp. 73913-73928, 2019.

[102] Y. Wang and Q. Zhu, "Performance analysis of clustered device-to-device networks using matern cluster process," *Wireless Networks,* vol. 25, no. 8, pp. 4849-4858, 2019.

[103] L. Yin, J. Gui, and Z. Zeng, "Improving energy efficiency of multimedia content dissemination by adaptive clustering and D2D multicast," *Mobile Information Systems,* vol. 2019, 2019.

[104] L. Zhou, Y. Wu, and H. Yu, "A Two-Layer, Energy-Efficient Approach for Joint Power Control and Uplink–Downlink Channel Allocation in D2D Communication," *Sensors,* vol. 20, no. 11, p. 3285, 2020.

[105] W. Gong, L. Pang, J. Wang, M. Xia, and Y. Zhang, "A social-aware K means clustering algorithm for D2D multicast communication under SDN architecture," *AEU-International Journal of Electronics and Communications,* vol. 132, p. 153610, 2021.

[106] F. Jiang, L. Zhang, C. Sun, and Z. Yuan, "Clustering and resource allocation strategy for D2D multicast networks with machine learning approaches," *China Communications,* vol. 18, no. 1, pp. 196-211, 2021.

[107] L. Dai, R. Jiao, F. Adachi, H. V. Poor, and L. Hanzo, "Deep learning for wireless communications: An emerging interdisciplinary paradigm," *IEEE Wireless Communications,* vol. 27, no. 4, pp. 133-139, 2020.

[108] T. Erpek, T. J. O'Shea, Y. E. Sagduyu, Y. Shi, and T. C. Clancy, "Deep learning for wireless communications," in *Development and Analysis of Deep Learning Architectures*: Springer, 2020, pp. 223-266.

[109] Y. Sun, M. Peng, Y. Zhou, Y. Huang, and S. Mao, "Application of machine learning in wireless networks: Key techniques and open issues," *IEEE Communications Surveys & Tutorials,* vol. 21, no. 4, pp. 3072-3108, 2019.

[110] S. Ruder, "An overview of gradient descent optimization algorithms," *arXiv preprint arXiv:1609.04747*, 2016.

[111] J. Wang, C. Jiang, H. Zhang, Y. Ren, K.-C. Chen, and L. Hanzo, "Thirty years of machine learning: The road to Pareto-optimal wireless networks," *IEEE Communications Surveys & Tutorials,* 2020.

[112] R. V. Kulkarni, A. Förster, and G. K. Venayagamoorthy, "Computational intelligence in wireless sensor networks: A survey," *IEEE communications surveys & tutorials,* vol. 13, no. 1, pp. 68-96, 2010.
[113] M. A. Alsheikh, S. Lin, D. Niyato, and H.-P. Tan, "Machine learning in wireless sensor networks: Algorithms, strategies, and applications," *IEEE Communications Surveys & Tutorials,* vol. 16, no. 4, pp. 1996-2018, 2014.
[114] A. Forster, "Machine learning techniques applied to wireless ad-hoc networks: Guide and survey," in *2007 3rd international conference on intelligent sensors, sensor networks and information*, 2007: IEEE, pp. 365-370.
[115] L. Gavrilovska, V. Atanasovski, I. Macaluso, and L. A. DaSilva, "Learning and reasoning in cognitive radio networks," *IEEE Communications Surveys & Tutorials,* vol. 15, no. 4, pp. 1761-1777, 2013.
[116] L. Cui, S. Yang, F. Chen, Z. Ming, N. Lu, and J. Qin, "A survey on application of machine learning for Internet of Things," *International Journal of Machine Learning and Cybernetics,* vol. 9, no. 8, pp. 1399-1417, 2018.
[117] X. Wang *et al.*, "Dynamic spectrum anti-jamming communications: Challenges and opportunities," *IEEE Communications Magazine,* vol. 58, no. 2, pp. 79-85, 2020.
[118] X. Wang *et al.*, "Machine Learning Empowered Spectrum Sharing in Intelligent Unmanned Swarm Communication Systems: Challenges, Requirements and Solutions," *IEEE Access,* vol. 8, pp. 89839-89849, 2020.
[119] A. Klautau, P. Batista, N. González-Prelcic, Y. Wang, and R. W. Heath, "5G MIMO data for machine learning: Application to beam-selection using deep learning," in *2018 Information Theory and Applications Workshop (ITA)*, 2018: IEEE, pp. 1-9.
[120] S. Han, I. Chih-Lin, G. Li, S. Wang, and Q. Sun, "Big data enabled mobile network design for 5G and beyond," *IEEE Communications Magazine,* vol. 55, no. 9, pp. 150-157, 2017.
[121] M. G. Kibria, K. Nguyen, G. P. Villardi, O. Zhao, K. Ishizu, and F. Kojima, "Big data analytics, machine learning, and artificial intelligence in next-generation wireless networks," *IEEE access,* vol. 6, pp. 32328-32338, 2018.
[122] M. Chen, U. Challita, W. Saad, C. Yin, and M. Debbah, "Artificial neural networks-based machine learning for wireless networks: A tutorial," *IEEE Communications Surveys & Tutorials,* vol. 21, no. 4, pp. 3039-3071, 2019.
[123] X. Cheng, L. Fang, X. Hong, and L. Yang, "Exploiting mobile big data: Sources, features, and applications," *IEEE Network,* vol. 31, no. 1, pp. 72-79, 2017.
[124] X. Cheng, L. Fang, L. Yang, and S. Cui, "Mobile big data: The fuel for data-driven wireless," *IEEE Internet of things Journal,* vol. 4, no. 5, pp. 1489-1516, 2017.
[125] R. Shafin, L. Liu, V. Chandrasekhar, H. Chen, J. Reed, and J. C. Zhang, "Artificial intelligence-enabled cellular networks: A critical path to beyond-5G and 6G," *IEEE Wireless Communications,* vol. 27, no. 2, pp. 212-217, 2020.
[126] S. Y. Han, N. B. Abu-Ghazaleh, and D. Lee, "Efficient and consistent path loss model for mobile network simulation," *IEEE/ACM Transactions on Networking,* vol. 24, no. 3, pp. 1774-1786, 2015.

[127] J. Liu, R. Deng, S. Zhou, and Z. Niu, "Seeing the unobservable: Channel learning for wireless communication networks," in *2015 IEEE Global Communications Conference (GLOBECOM)*, 2015: IEEE, pp. 1-6.

[128] S. Hu, Y.-D. Yao, and Z. Yang, "MAC protocol identification using support vector machines for cognitive radio networks," *IEEE Wireless Communications,* vol. 21, no. 1, pp. 52-60, 2014.

[129] R. D. Timoteo, D. C. Cunha, and G. D. Cavalcanti, "A proposal for path loss prediction in urban environments using support vector regression," in *Proc. Advanced Int. Conf. Telecommun*, 2014, pp. 1-5.

[130] C. Jiang, H. Zhang, Y. Ren, Z. Han, K.-C. Chen, and L. Hanzo, "Machine learning paradigms for next-generation wireless networks," *IEEE Wireless Communications,* vol. 24, no. 2, pp. 98-105, 2016.

[131] M. Polese, R. Jana, V. Kounev, K. Zhang, S. Deb, and M. Zorzi, "Machine learning at the edge: A data-driven architecture with applications to 5G cellular networks," *IEEE Transactions on Mobile Computing,* 2020.

[132] Z. Na, Y. Liu, J. Shi, C. Liu, and Z. Gao, "UAV-supported clustered NOMA for 6G-enabled Internet of Things: Trajectory planning and resource allocation," *IEEE Internet of Things Journal,* 2020.

[133] R. Boutaba *et al.*, "A comprehensive survey on machine learning for networking: evolution, applications and research opportunities," *Journal of Internet Services and Applications,* vol. 9, no. 1, pp. 1-99, 2018.

[134] Y. Cao, T. Jiang, X. Chen, and J. Zhang, "Social-aware video multicast based on device-to-device communications," *IEEE Transactions on Mobile Computing,* vol. 15, no. 6, pp. 1528-1539, 2015.

[135] Z. Chang, Y. Hu, Y. Chen, and B. Zeng, "Cluster-oriented device-to-device multimedia communications: joint power, bandwidth, and link selection optimization," *IEEE Transactions on Vehicular Technology,* vol. 67, no. 2, pp. 1570-1581, 2017.

[136] I. F. Akyildiz, S. Nie, S.-C. Lin, and M. Chandrasekaran, "5G roadmap: 10 key enabling technologies," *Computer Networks,* vol. 106, pp. 17-48, 2016.

[137] A. Morgado, K. M. S. Huq, S. Mumtaz, and J. Rodriguez, "A survey of 5G technologies: regulatory, standardization and industrial perspectives," *Digital Communications and Networks,* vol. 4, no. 2, pp. 87-97, 2018.

[138] D. Bega, M. Gramaglia, C. J. Bernardos Cano, A. Banchs, and X. Costa-Perez, "Toward the network of the future: From enabling technologies to 5G concepts," *Transactions on Emerging Telecommunications Technologies,* vol. 28, no. 8, p. e3205, 2017.

[139] Z. Feng, C. Qiu, Z. Feng, Z. Wei, W. Li, and P. Zhang, "An effective approach to 5G: Wireless network virtualization," *IEEE Communications Magazine,* vol. 53, no. 12, pp. 53-59, 2015.

[140] E. Hossain and M. Hasan, "5G cellular: key enabling technologies and research challenges," *IEEE Instrumentation & Measurement Magazine,* vol. 18, no. 3, pp. 11-21, 2015.

[141] A. A. Ateya, A. Muthanna, M. Makolkina, and A. Koucheryavy, "Study of 5G services standardization: specifications and requirements," in *2018 10th International Congress on Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT)*, 2018: IEEE, pp. 1-6.
[142] G. O. Pérez, J. A. Hernández, and D. Larrabeiti, "Fronthaul network modeling and dimensioning meeting ultra-low latency requirements for 5G," *IEEE/OSA Journal of Optical Communications and Networking,* vol. 10, no. 6, pp. 573-581, 2018.
[143] S.-Y. Lien, S.-L. Shieh, Y. Huang, B. Su, Y.-L. Hsu, and H.-Y. Wei, "5G new radio: Waveform, frame structure, multiple access, and initial access," *IEEE communications magazine,* vol. 55, no. 6, pp. 64-71, 2017.
[144] T. S. Rappaport, S. Sun, and M. Shafi, "Investigation and comparison of 3GPP and NYUSIM channel models for 5G wireless communications," in *2017 IEEE 86th Vehicular Technology Conference (VTC-Fall)*, 2017: IEEE, pp. 1-5.
[145] I. O. Sanusi, K. M. Nasr, and K. Moessner, "Channel assignment and power control for D2D-enabled cellular networks," in *2019 International Conference on Computing, Electronics & Communications Engineering (iCCECE)*, 2019: IEEE, pp. 225-228.
[146] P. Yang, N. Zhang, S. Zhang, L. Yu, J. Zhang, and X. S. Shen, "Content popularity prediction towards location-aware mobile edge caching," *IEEE Transactions on Multimedia,* vol. 21, no. 4, pp. 915-929, 2018.
[147] P. Padmavathi, V. Eswaramurthy, and J. Revathi, "Fuzzy Social Spider Optimization Algorithm for Fuzzy Clustering Analysis," in *2018 International Conference on Current Trends towards Converging Technologies (ICCTCT)*, 2018: IEEE, pp. 1-6.
[148] A. A. Khan, M. Abolhasan, W. Ni, J. Lipman, and A. Jamalipour, "A hybrid-fuzzy logic guided genetic algorithm (H-FLGA) approach for resource optimization in 5G VANETs," *IEEE Transactions on Vehicular Technology,* vol. 68, no. 7, pp. 6964-6974, 2019.
[149] E. Cardoso, K. Silva, and R. Francês, "Intelligent handover procedure for heterogeneous LTE networks using fuzzy logic," in *2017 13th International Wireless Communications and Mobile Computing Conference (IWCMC)*, 2017: IEEE, pp. 2163-2168.
[150] A. Alhammadi, M. Roslee, M. Y. Alias, I. Shayea, S. Alriah, and A. B. Abas, "Advanced handover self-optimization approach for 4G/5G HetNets using weighted fuzzy logic control," in *2019 15th International Conference on Telecommunications (ConTEL)*, 2019: IEEE, pp. 1-6.
[151] D. Kim, J.-h. Kim, C. Moon, J. Choi, and I. Yeom, "Efficient content delivery in mobile ad-hoc networks using CCN," *Ad Hoc Networks,* vol. 36, pp. 81-99, 2016.
[152] Q. Wang, D. Chen, N. Zhang, Z. Qin, and Z. Qin, "LACS: A lightweight label-based access control scheme in IoT-based 5G caching context," *IEEE Access,* vol. 5, pp. 4018-4027, 2017.
[153] M. Fritter, N. Ould-Khessal, S. Fazackerley, and R. Lawrence, "Experimental evaluation of hash function performance on embedded devices," in *2018 IEEE*

Canadian Conference on Electrical & Computer Engineering (CCECE), 2018: IEEE, pp. 1-5.
[154] A. L. Varna and M. Wu, "Modeling and analysis of content identification," in *2009 IEEE International Conference on Multimedia and Expo*, 2009: IEEE, pp. 1528-1531.
[155] J. Malhotra and J. Bakal, "A survey and comparative study of data deduplication techniques," in *2015 International Conference on Pervasive Computing (ICPC)*, 2015: IEEE, pp. 1-5.
[156] M. Padhi and R. Chaudhari, "An optimized pipelined architecture of SHA-256 hash function," in *2017 7th International Symposium on Embedded Computing and System Design (ISED)*, 2017: IEEE, pp. 1-4.
[157] X. Qiuyun, H. Ligang, L. Qiming, G. Shuqin, and W. Jinhui, "The Verification of SHA-256 IP using a semi-automatic UVM platform," in *2017 13th IEEE International Conference on Electronic Measurement & Instruments (ICEMI)*, 2017: IEEE, pp. 111-115.
[158] O. Hayat, R. Ngah, and Y. Zahedi, "Device discovery for D2D communication in in-band cellular networks using sphere decoder like (SDL) algorithm," *EURASIP Journal on Wireless Communications and Networking,* vol. 2018, no. 1, pp. 1-7, 2018.
[159] H. Tang, Z. Ding, and B. C. Levy, "Enabling D2D communications through neighbor discovery in LTE cellular networks," *IEEE Transactions on Signal Processing,* vol. 62, no. 19, pp. 5157-5170, 2014.
[160] S. Jaffry, "Increasing the capacity of 5G networks using mobile-cells: a thesis presented in partial fulfilment of the requirements for the degree of Doctor of Philosophy in Engineering at Massey University, New Zealand," Massey University, 2019.
[161] S. Jaffry, S. K. Zaidi, S. T. Shah, S. F. Hasan, and X. Gui, "D2D neighborhood discovery by a mobile device," in *ICC 2019-2019 IEEE International Conference on Communications (ICC)*, 2019: IEEE, pp. 1-6.
[162] Y. Wang, Y. Li, W. Wang, and M. Song, "A locality-based mobile caching policy for D2D-based content sharing network," in *2016 IEEE Global Communications Conference (GLOBECOM)*, 2016: IEEE, pp. 1-6.
[163] X. Lin, J. G. Andrews, and A. Ghosh, "Spectrum sharing for device-to-device communication in cellular networks," *IEEE Transactions on Wireless Communications,* vol. 13, no. 12, pp. 6727-6740, 2014.
[164] M. Kuerbis, N. M. Balasubramanya, L. Lampe, and A. Lampe, "On the use of channel models and channel estimation techniques for massive MIMO systems," in *2016 24th European Signal Processing Conference (EUSIPCO)*, 2016: IEEE, pp. 823-827.
[165] H. T. Cheng and W. Zhuang, "An optimization framework for balancing throughput and fairness in wireless networks with QoS support," *IEEE Transactions on Wireless Communications,* vol. 7, no. 2, pp. 584-593, 2008.

[166] S. Huaizhou, R. V. Prasad, E. Onur, and I. Niemegeers, "Fairness in wireless networks: Issues, measures and challenges," *IEEE Communications Surveys & Tutorials,* vol. 16, no. 1, pp. 5-24, 2013.
[167] T. A. Runkler and C. Katz, "Fuzzy clustering by particle swarm optimization," in *2006 IEEE international conference on fuzzy systems*, 2006: IEEE, pp. 601-608.
[168] J. Zhang and Z. Ma, "Hybrid fuzzy clustering method based on FCM and enhanced logarithmical PSO (ELPSO)," *Computational intelligence and neuroscience,* vol. 2020, 2020.
[169] A. Antonopoulos and C. Verikoukis, "Multi-player game theoretic MAC strategies for energy efficient data dissemination," *IEEE Transactions on Wireless Communications,* vol. 13, no. 2, pp. 592-603, 2013.
[170] A. Antonopoulos, E. Kartsakli, and C. Verikoukis, "Game theoretic D2D content dissemination in 4G cellular networks," *IEEE Communications Magazine,* vol. 52, no. 6, pp. 125-132, 2014.
[171] A. C. Kazez and T. Girici, "Clustering-based device-to-device cache placement," *Ad Hoc Networks,* vol. 84, pp. 170-177, 2019.
[172] A. Ometov *et al.*, "Toward trusted, social-aware D2D connectivity: bridging across the technology and sociality realms," *IEEE Wireless Communications,* vol. 23, no. 4, pp. 103-111, 2016.
[173] S. Łukasik, P. A. Kowalski, M. Charytanowicz, and P. Kulczycki, "Clustering using flower pollination algorithm and Calinski-Harabasz index," in *2016 IEEE Congress on Evolutionary Computation (CEC)*, 2016: IEEE, pp. 2724-2728.
[174] r. G. P. Project, "3GPP TR 25.814 3GPP TSG RAN Physical Layer Aspects For Evolved UTRA, v7. 1.0," 2006.
[175] K. Mahmud, M. Inoue, H. Murakami, M. Hasegawa, and H. Morikawa, "Energy consumption measurement of wireless interfaces in multi-service user terminals for heterogeneous wireless networks," *IEICE transactions on communications,* vol. 88, no. 3, pp. 1097-1110, 2005.
[176] H. Jung and I.-H. Lee, "Coverage probability and area spectral efficiency of clustered linear unmanned vehicle sensor networks," *Sensors,* vol. 17, no. 11, p. 2550, 2017.
[177] Y. Liu, Z. Li, H. Xiong, X. Gao, and J. Wu, "Understanding of internal clustering validation measures," in *2010 IEEE international conference on data mining*, 2010: IEEE, pp. 911-916.
[178] R. K. Jain, D.-M. W. Chiu, and W. R. Hawe, "A quantitative measure of fairness and discrimination," *Eastern Research Laboratory, Digital Equipment Corporation, Hudson, MA,* 1984.
[179] Z. Khan and P. Fan, "A multi-hop moving zone (MMZ) clustering scheme based on cellular-V2X," *China Communications,* vol. 15, no. 7, pp. 55-66, 2018.
[180] W. Qi, Q. Song, X. Wang, L. Guo, and Z. Ning, "SDN-enabled social-aware clustering in 5G-VANET systems," *IEEE Access,* vol. 6, pp. 28213-28224, 2018.
[181] M. Yin, W. Li, L. Feng, P. Yu, and X. Qiu, "Emergency Communications Based on Throughput-Aware D2D Multicasting in 5G Public Safety Networks," *Sensors,* vol. 20, no. 7, p. 1901, 2020.

[182] S. Aslam, F. Alam, S. F. Hasan, and M. Rashid, "A Novel Weighted Clustering Algorithm Supported by a Distributed Architecture for D2D Enabled Content-Centric Networks," *Sensors,* vol. 20, no. 19, p. 5509, 2020.

[183] N. U. Hasan, W. Ejaz, N. Ejaz, H. S. Kim, A. Anpalagan, and M. Jo, "Network selection and channel allocation for spectrum sharing in 5G heterogeneous networks," *IEEE access,* vol. 4, pp. 980-992, 2016.

[184] I. Filippini, A. E. C. Redondi, and A. Capone, "Beyond Cellular Green Generation: Potential and Challenges of the Network Separation," *Mobile Information Systems,* vol. 2017, 2017.

[185] A. Tanveer, Z. Khan, A. N. Malik, and I. M. Qureshi, "Flexible Queuing Model for Number of Active Users in Cognitive Radio Network Environment," *Wireless Communications and Mobile Computing,* vol. 2018, 2018.

[186] C. Xu *et al.*, "Efficiency resource allocation for device-to-device underlay communication systems: A reverse iterative combinatorial auction based approach," *IEEE Journal on Selected Areas in Communications,* vol. 31, no. 9, pp. 348-358, 2013.

[187] Y. Li, D. Jin, J. Yuan, and Z. Han, "Coalitional games for resource allocation in the device-to-device uplink underlaying cellular networks," *IEEE Transactions on wireless communications,* vol. 13, no. 7, pp. 3965-3977, 2014.

[188] Y. Zhao, Y. Li, X. Chen, and N. Ge, "Joint optimization of resource allocation and relay selection for network coding aided device-to-device communications," *IEEE Communications Letters,* vol. 19, no. 5, pp. 807-810, 2015.

[189] D. Wang, B. Song, D. Chen, and X. Du, "Intelligent cognitive radio in 5G: AI-based hierarchical cognitive cellular networks," *IEEE Wireless Communications,* vol. 26, no. 3, pp. 54-61, 2019.

[190] H. Kaneko and K. Funatsu, "Application of online support vector regression for soft sensors," *AIChE Journal,* vol. 60, no. 2, pp. 600-612, 2014.

[191] M. A. Hearst, S. T. Dumais, E. Osuna, J. Platt, and B. Scholkopf, "Support vector machines," *IEEE Intelligent Systems and their applications,* vol. 13, no. 4, pp. 18-28, 1998.

[192] L. Breiman, "Random forests," *Machine learning,* vol. 45, no. 1, pp. 5-32, 2001.

[193] X. Yuan, B. Huang, Y. Wang, C. Yang, and W. Gui, "Deep learning-based feature representation and its application for soft sensor modeling with variable-wise weighted SAE," *IEEE Transactions on Industrial Informatics,* vol. 14, no. 7, pp. 3235-3243, 2018.

[194] Y. Wang, K. Wu, and L. M. Ni, "Wifall: Device-free fall detection by wireless networks," *IEEE Transactions on Mobile Computing,* vol. 16, no. 2, pp. 581-594, 2016.

[195] K. Zheng, Z. Yang, K. Zhang, P. Chatzimisios, K. Yang, and W. Xiang, "Big data-driven optimization for mobile networks toward 5G," *IEEE network,* vol. 30, no. 1, pp. 44-51, 2016.

[196] M. Najla, Z. Becvar, P. Mach, and D. Gesbert, "Predicting Device-to-Device Channels from Cellular Channel Measurements: A Learning Approach," *arXiv preprint arXiv:1911.07191,* 2019.

[197] T. O'Shea and J. Hoydis, "An introduction to deep learning for the physical layer," *IEEE Transactions on Cognitive Communications and Networking,* vol. 3, no. 4, pp. 563-575, 2017.
[198] T. Wang, C.-K. Wen, H. Wang, F. Gao, T. Jiang, and S. Jin, "Deep learning for wireless physical layer: Opportunities and challenges," *China Communications,* vol. 14, no. 11, pp. 92-111, 2017.
[199] S. Qian, H. Liu, C. Liu, S. Wu, and H. San Wong, "Adaptive activation functions in convolutional neural networks," *Neurocomputing,* vol. 272, pp. 204-212, 2018.
[200] A. Liaw and M. Wiener, "Classification and regression by randomForest," *R news,* vol. 2, no. 3, pp. 18-22, 2002.
[201] P. O. Gislason, J. A. Benediktsson, and J. R. Sveinsson, "Random forest classification of multisource remote sensing and geographic data," in *IGARSS 2004. 2004 IEEE International Geoscience and Remote Sensing Symposium*, 2004, vol. 2: IEEE, pp. 1049-1052.
[202] M. Pal, "Random forest classifier for remote sensing classification," *International journal of remote sensing,* vol. 26, no. 1, pp. 217-222, 2005.
[203] K. K. Dobbin and R. M. Simon, "Optimally splitting cases for training and testing high dimensional classifiers," *BMC medical genomics,* vol. 4, no. 1, p. 31, 2011.
[204] G. James, D. Witten, T. Hastie, and R. Tibshirani, *An introduction to statistical learning*. Springer, 2013.
[205] D. Stathakis, "How many hidden layers and nodes?," *International Journal of Remote Sensing,* vol. 30, no. 8, pp. 2133-2147, 2009.
[206] J. Bergstra and Y. Bengio, "Random search for hyper-parameter optimization," *The Journal of Machine Learning Research,* vol. 13, no. 1, pp. 281-305, 2012.
[207] H. Yang, X. Xie, and M. Kadoch, "Machine Learning Techniques and A Case Study for Intelligent Wireless Networks," *IEEE Network,* vol. 34, no. 3, pp. 208-215, 2020.
[208] M. J. Kearns, *The computational complexity of machine learning*. MIT press, 1990.
[209] S. Sra, S. Nowozin, and S. J. Wright, *Optimization for machine learning*. Mit Press, 2012.
[210] Y. Zhao *et al.*, "Low-complexity and joint modulation format identification and OSNR estimation using random forest for flexible coherent receivers," *Optics Communications,* vol. 457, p. 124698, 2020.
[211] N. Ahad, J. Qadir, and N. Ahsan, "Neural networks in wireless networks: Techniques, applications and guidelines," *Journal of network and computer applications,* vol. 68, pp. 1-27, 2016.
[212] K. Li, Y. Wu, Y. Nan, P. Li, and Y. Li, "Hierarchical multi-class classification in multimodal spacecraft data using DNN and weighted support vector machine," *Neurocomputing,* vol. 259, pp. 55-65, 2017.
[213] M. Du, S. Ding, Y. Xue, and Z. Shi, "A novel density peaks clustering with sensitivity of local density and density-adaptive metric," *Knowledge and Information Systems,* vol. 59, no. 2, pp. 285-309, 2019.

An Example of a Trainig Dataset

S.#	Coordinates		Features								Throughput (kbps)		Labels
			Average Path Loss (dB)				Average Channel Gains (ABS)						
	XX	YY	BS	Cluster 1	Cluster 2	Cluster 3	BS	Cluster 1	Cluster 2	Cluster 3	BS	Cluster	
1	0.882515017	0.52656658	110.572	0	0	105.0235	3.89E-06	0	0	9.32E-06	5.48E+00	**9.20E+00**	Cluster
2	0.388909111	0.50115413	89.05234	0	0	111.3595	4.63E-05	0	0	4.50E-06	**13.214**	6.10E+00	BS
3	0.299822438	0.81032638	109.9189	91.94101	0	0	4.19E-06	3.26E-05	0	0	5.89E+00	**15.74**	Cluster
4	0.752914143	0.8048382	111.1363	0	0	111.3999	3.64E-06	0	0	4.47E-06	5.68E+00	**6.08E+00**	Cluster
5	0.265169501	0.11782083	113.2971	0	104.1214	0	2.84E-06	0	1.30E-05	0	3.55E+00	**13.49**	Cluster
6	0.525548325	0.61835023	90.54664	0	0	106.1517	3.90E-05	0	0	8.19E-06	**16.81**	8.88E+00	BS
7	0.229234718	0.77272752	110.6115	79.93622	0	0	3.87E-06	1.30E-04	0	0	5.51E+00	**21.04**	Cluster
8	0.056385551	0.28483845	114.9395	0	98.53031	0	2.35E-06	0	2.47E-05	0.00E+00	3.16E+00	**16.44**	Cluster
9	0.827781052	0.26912024	111.3471	0	0	109.6269	3.55E-06	0	0	5.49E-06	4.14E+00	**5.21E+00**	Cluster
10	0.115814114	0.46611625	110.6732	0	107.1901	0	3.84E-06	0	9.11E-06	0.00E+00	5.46E+00	**9.11E+00**	Cluster