Caching Strategies in NDN Based Wireless Ad Hoc Network: A Survey

Ahmed Khalid[1], Rana Asif Rehman[1] and Byung-Seo Kim[2,*]

[1]Department of Computer Science, FAST School of Computing, National University of Computer and Emerging Sciences, Lahore, 54000, Pakistan

[2]Department of Software and Communications Engineering, Hongik University, Sejong City, 30016, Republic of Korea

*Corresponding Author: Byung-Seo Kim. Email: jsnbs@hongik.ac.kr

ABSTRACT

Wireless Ad Hoc Networks consist of devices that are wirelessly connected. Mobile Ad Hoc Networks (MANETs), Internet of Things (IoT), and Vehicular Ad Hoc Networks (VANETs) are the main domains of wireless ad hoc network. Internet is used in wireless ad hoc network. Internet is based on Transmission Control Protocol (TCP)/Internet Protocol (IP) network where clients and servers interact with each other with the help of IP in a pre-defined environment. Internet fetches data from a fixed location. Data redundancy, mobility, and location dependency are the main issues of the IP network paradigm. All these factors result in poor performance of wireless ad hoc networks. The main disadvantage of IP is that, it does not provide in-network caching. Therefore, there is a need to move towards a new network that overcomes these limitations. Named Data Network (NDN) is a network that overcomes these limitations. NDN is a project of Information-centric Network (ICN). NDN provides in-network caching which helps in fast response to user queries. Implementing NDN in wireless ad hoc network provides many benefits such as caching, mobility, scalability, security, and privacy. By considering the certainty, in this survey paper, we present a comprehensive survey on Caching Strategies in NDN-based Wireless Ad Hoc Network. Various caching mechanism-based results are also described. In the last, we also shed light on the challenges and future directions of this promising field to provide a clear understanding of what caching-related problems exist in NDN-based wireless ad hoc networks.

KEYWORDS

Content centric network; Internet of Things; mobile ad hoc network; named data network; vehicular ad hoc network

1 Introduction

The Internet is based on a TCP/IP stack. When data are uploaded to the Internet, they are divided into small packets [1]. The Internet was invented by the Advanced Research Project Agency Network in the 1960s [2]. Packets on the Internet travel from source to destination using TCP and IP. IP contains information on how to send the data from the client to the server, whereas TCP ensures that the user receives the complete data, no data packets are lost during transmission, and the data are reliable. TCP is a protocol for communication that addresses how packets of data are exchanged between different Internet layers. It provides some key features such as reliable data delivery, packet switching,

connection-oriented communication, best-effort delivery, and IP addressing. TCP establishes a reliable communication connection between multiple devices before transmission begins. It also imposes a flow control mechanism that helps with the data transmission rate. TCP allows data to flow in both directions. The Internet has some issues that need to be addressed as soon as possible. Some of the important issues of traditional TCP/IP are:

- The Internet is host-centric, and users demand content rather than location.
- For locating content distribution servers (CDNs), the Internet uses the domain name system (DNS).
- Security issue (because the location of data is known).
- Mobility issue (because for a new location, we have to require new IP).
- TCP/IP model is difficult to manage, set up, and control.
- TCP/IP model does not represent other protocol stacks other than TCP/IP [3].
- Chances of failure when traffic grows rapidly with time.
- Inefficient process of packet delivery.

To overcome the issue of the inefficient processing of packet delivery, researchers used the CDN mechanism [4]. In this mechanism, a new server is created that has all the data of the original server. If the content request is the same, the query is transferred to the new duplicate server. This saves time and energy because the content query is redirected to the duplicate server instead of the original server, as the original server is far from the user. However, the content is changed dynamically if the user request is difficult to support or mobility issues occur. CDN is based on IP, so when the client is far away from the duplicate server, there is still an issue that the client request will not be served on time by the duplicate server. Therefore, there is a need for a new network that can overcome these issues.

With the evolution of the Internet, new requirements and demands have emerged for the network, such as mobility, distribution of data in a scalable form, and security. These requirements have encouraged scientific researchers and stakeholders to study and propose alternative architectures for the network to overcome these issues. Thus, a new network paradigm was introduced called information-centric networking (ICN). This is a next-level approach for the future of the Internet. ICN retrieves content in two ways: content delivery and content discovery. ICN is a new paradigm that works on the content-centric approach. ICN content is distinguished by its unique identifiers and names. Content names are hierarchical and contain multiple attributes, such as location, publisher, and content type information. A request in ICN is forwarded based on the content name. ICN then caches the content closer to the user within the network. ICN enables content distribution, flexibility, and scalability. ICN has four main design components:

1. Data objects based on the name: photos, documents, songs, and webpages.
2. Security and naming: unique names to distinguish data objects.
3. Routing and forwarding: Data routing and forwarding paths and decisions.
4. Caching: In-network caching.

Major ICN projects in the EU include scalable adaptive Internet solutions, content mediator architecture for content-aware networks, content-aware searching retrieval and streaming, publish-subscribe Internet technology, ALICANTE, and CONVERGENCE. Major U.S. projects include content-centric networks (CCN) and NDN. CCN and NDN are the most useful projects. In 2009, Vasilakos et al. [5] elaborated the idea of the CCN paradigm. CCN is now implemented in the NDN project. Both of these projects were proposed by the Palo Alto Research Center. The main goal of NDN

is to replace the traditional IP-based network with a name-based content model. NDN is necessary for the following reasons:

- NDN provides an effective and simple model of communication.
- Instead of IP, the content name is used.
- NDN focuses on "what" instead of "where."
- The name of the content is more important than the location of the content.
- It is a decentralized approach (less chance of failure).
- Consumers initiate the communication.
- Less scalability issue (because naming is hierarchical, as names are human-readable).
- Provides in-network caching.
- Not only end-users, but intermediate nodes are also aware of user requests.
- Support mobility.
- Better security (because the location is not known).
- It is proposed by the USA researchers hence it is important from a business perspective.

A comparison of IP and NDN is illustrated in Table 1, where we can easily see why NDN is better than a traditional IP network. NDN can use the major services of IP, such as DNS. It is also capable of using different IP protocols and can easily use NDN with minor modifications. There are two packets in NDN: the INTEREST packet and the DATA packet. The INTEREST packet is a packet that is used for content requests, whereas the DATA packet is the actual packet of data. These packets hold the names that distinguish multiple pieces of content. The main node transmits the packet in the network until the requested content is found or a producer of that content is not found [6].

Table 1: Comparison between IP *vs*. NDN

IP	NDN
Host-centric network	Information-centric network
Complex and less effective	Simple and effective
Focus on IP	Focus on content name
Focus on "Where"	Focus on "What"
Centralized approach	Decentralized approach
Not provide in-network caching	Provide in-network caching
Does not support mobility	Support mobility
Security issue	Better security
More scalability issues	Fewer scalability issues

A wireless ad hoc network is a broad network consisting of different nodes and devices, each of them connected to the others using various wireless connections. Traditional networks use a sink node or router with nearby devices. Some of the key benefits of wireless ad hoc networks are listed below:

- They provide mobility.
- They are easy to set up and expand.
- They provide better global coverage.
- They provide greater flexibility.
- They are cheaper as compared to wired networks because it does not require cables.

As wireless users increase, traffic on wireless networks also increases. Demand for quick data retrieval also increases accordingly. To fulfill user demand, we have to shift from traditional wireless networks toward a new paradigm in which users' queries and demands are fulfilled in a fast and efficient manner. The content-centric approach should be considered to satisfy users' demands. A user request is served by the nodes that are available near the user. To support this, NDN nodes support caching to store the content. The concept of caching content on NDN nodes is different from other networks. In a wireless ad hoc network, the mobility of nodes is also considered because if the user or node moves to another region, then data should be formulated against the changing scenario. Therefore, a wireless ad hoc network should be integrated into NDN, which would result in many benefits. Due to limited storage space and cost, the caching of content on wireless ad hoc network nodes is still a problem in NDN-based wireless ad hoc networks [7]. The main issue with caching is determining with which criteria data should be cached on wireless nodes. NDN nodes are integrated with various cache policies that determine which content is to be replaced or deleted and on what basis.

In this paper, we comprehensively discuss different caching strategies in NDN-based wireless ad hoc networks. Though many different works have been carried out on caching in NDN, those studies only focused on limited caching strategies and not on NDN-based wireless ad hoc networks.

1.1 Our Contribution

In this survey paper, we present a survey of caching strategies in NDN-based wireless ad hoc networks. We highlight different aspects (how NDN works and its applications, caching in NDN, wireless ad hoc networks, and caching in NDN-based wireless ad hoc networks) that are necessary to better understand this topic. This is the first survey paper to comprehensively cover the up-to-date caching schemes of NDN-based wireless ad-hoc networks, including MANETs, the IoT, and VANETs, in a single paper. This paper contributes the following:

1. As compared with other survey papers, we present a discussion of the need for caching strategies in NDN and the classification of caching in NDN-based wireless ad hoc networks in detail.
2. Different caching strategies used in NDN-based MANETs, IoT, and VANETs are explained.
3. We identify the challenges and future directions of this promising field.

1.2 Survey Paper Structure

Section 1 discusses the current Internet network paradigm and the need for a new network paradigm as well as the benefits of wireless ad hoc networks and why we should integrate these with NDN. Section 2 describes the NDN architecture and its applications. Section 3 elaborates on caching in NDN and the need for caching policies. Section 4 describes caching in NDN-based wireless ad hoc networks. Section 5 explains caching in NDN-based MANETs and different caching strategies (cooperative, popularity, location, priority, hybrid, and multifarious caching). Caching strategies in NDN-based IoT (cooperative, freshness, popularity, hybrid, and multifarious caching) are explained in Section 6. Section 7 explains caching strategies in NDN-based VANETs (cooperative, popularity, probability, proactive, optimization, hybrid, and multifarious caching). In Section 8, we list some challenges and future directions regarding caching in NDN-based wireless ad hoc networks. Finally, the survey paper is concluded in Section 9.

2 NDN Working and Its Applications

In this section, we discuss the background of our proposed work, the motivation behind our research work, and the problem statement that drove us to contribute here.

2.1 NDN Architecture and Working

NDN architecture is like an hourglass structure. A comparison between IP and NDN architectures is illustrated in Fig. 1. IP and NDN have similar structures with a minor difference. NDN's first layer is the application layer, which includes different applications such as browsers. The second layer in NDN is the transport layer, which includes different protocols. After the second layer, the content name replaces the IP in NDN because NDN focuses on the content name rather than location. The NDN structure has three main tables:

1. Content store (CS) table: To save bandwidth, enhance sharing possibilities, and reduce retrieval time of the content, NDN routers cache a copy of DATA packets and store this in the CS table until new content is placed. Matching in the CS table can be done through the matching of exact characters.
2. Pending interest table (PIT): PIT handles the INTEREST. PIT is used to keep a record of interfaces. It maintains every incoming packet entry until its lifetime expires or a corresponding DATA packet arrives. Data are delivered to requested users based on interfaces. A PIT entry is deleted when data reaches the user or the lifetime of the INTEREST packet is expired. Matching in PIT can be done through the matching of exact characters.
3. Forwarding information base (FIB) table: The role of FIB is to maintain information on the next nodes to which a node in the INTEREST packet should be directed [8].
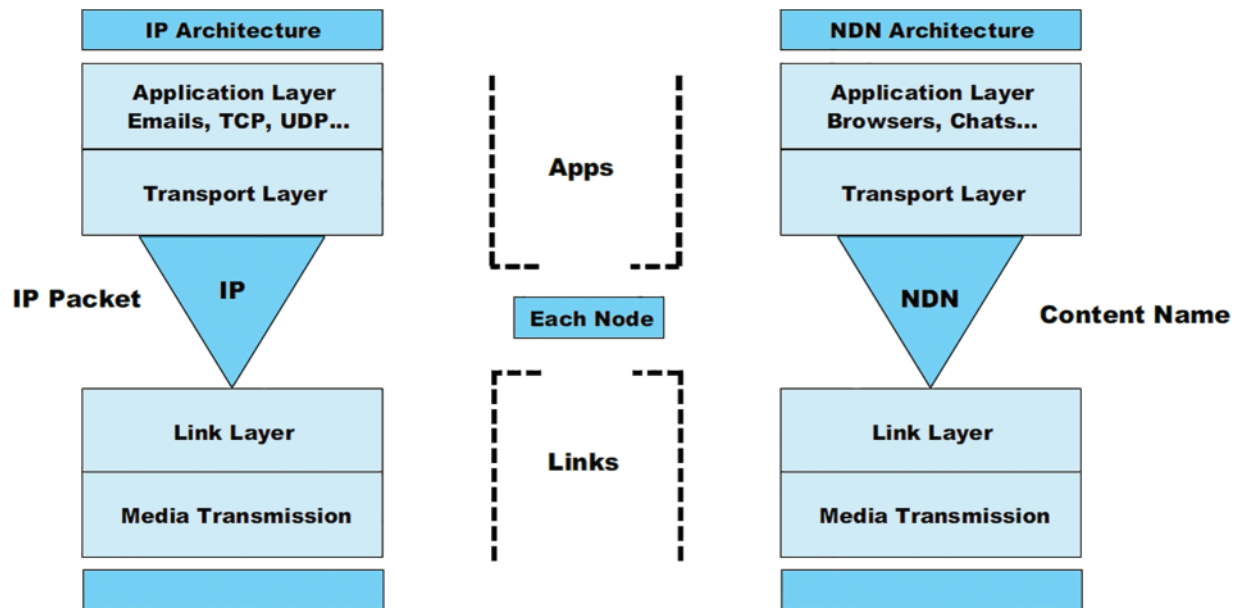

Figure 1: Comparison of glass model of IP *vs*. NDN architecture

The data forwarding process occurs with the help of both packets. When a user wants data, an INTEREST packet is sent to the NDN main device. This device searches the user's query in the CS table to check if an exact match is found. If a perfect match exists, the DATA packet is signed and

sent to the user immediately. If the user query is not available in the first table, then the NDN device checks the PIT to find the exact match. If the PIT has the exact match for the user query, then the interface is added to the demanded packet in the PIT list. This interface is added because the user will get their DATA packet when the required DATA packet is available. If a PIT entry is not found for an incoming packet, then the NDN router passes the user's INTEREST packet to the FIB, which further tries to match the user's query by using the longest matching via the prefix. The INTEREST packet is forwarded to the other nodes when the FIB entry containing the matching query is found, and a brand new entry in PIT is created for this interface. If the INTEREST packet does not exist in FIB, then this INTEREST packet is deleted or flooded to all the outgoing interfaces. This is decided by the NDN's router policies. When a content packet moves to the router, all the entries in PIT are properly searched for an exact match for the user's query. The DATA packet is forwarded to all interfaces that are listed in the interfaces table to determine if the specific entry is present in the PIT. Moreover, content is saved in the CS table, and the corresponding entry in PIT is deleted. Data are stored in the CS table because NDN has an in-network caching ability; thus, the NDN router stores data so that future user requests for the same data are tackled quickly [6].

2.2 NDN Applications

In this section, we discuss different application services in which NDN is used widely. Multiple applications are used in our daily lives and cover a huge amount of human workload. Therefore, it is necessary to implement NDN in these applications. NDN applications, such as the wireless sensor network (WSN), ad-hoc network, vehicular network, educational network, and entertainment network, are illustrated in Fig. 2.
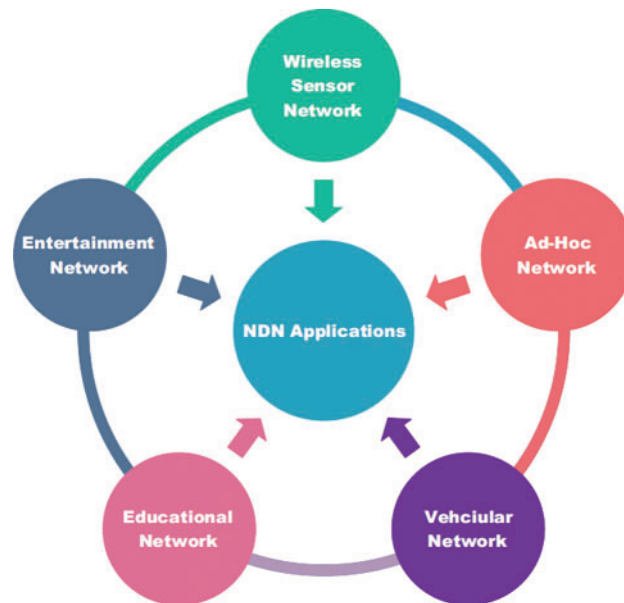

Figure 2: NDN applications

1. WSN: WSN is widely used in different domains. It is made up of multiple sensor nodes. These nodes are integrated in a wide area. The attributes of NDN-based WSN are:
   - Task nature: Tells us the task name.

- Task duration: Includes the time duration of the task and can also include values in real-time.
- Location: Includes the geographic area identification and can also include different logical or region-based names.
- Data duplication: Discards duplicate data. If multiple sensors fulfill the same interest request, the sink node must identify and discard the duplicate data [9].

2. Ad-hoc network: This is a decentralized wireless network in which multiple wireless devices establish a connection with other devices. Ad-hoc networks provide multiple benefits in different domains. They are widely used for connecting army equipment and personnel at headquarters and on the battlefield. Many state-of-the-art army communications systems solely depend on traditional IP networks. Ad-hoc networks face multiple challenges such as mobility, high latency, limited connectivity, and limited bandwidth. In many army communication systems, different nodes shift their region as they move from one region to another, which results in mobility issues in a wide range. Therefore, it is necessary to shift from a traditional IP network mechanism towards a better network paradigm. NDN is a network that overcomes these issues with ad-hoc networks. NDN's convincing characteristics and topologies make it a strong candidate. The navy's network consists of mobile nodes (ships) that use a mechanism to transmit crucial content to other nodes. NDN-based scenarios used in army special vehicles also contain moveable nodes and work as the network backbone. If NDN is implemented with proper guidelines, it will work more efficiently than existing ad-hoc networks [10].

3. Vehicular network: This is a network in which vehicles are treated as nodes that can move anywhere at any time. Furthermore, with the development of a transportation system based on intelligence, simple Internet vehicles' and autonomous vehicles' data are sent to the provider instead of just the consumer [11]. Internet vehicles and different autonomous vehicles are based on IP networks. These vehicles analyze and collect data such as road safety, navigation data, the status of road traffic, and weather conditions. These vehicles share and communicate their data to lessen road traffic jams and provide information on speeders, road closures or openings, and road conditions. Vehicles are fully loaded with different units and sensors that are used for different purposes. Due to their diverse mechanisms, designing a scalable and flexible scheme for data dissemination is a difficult challenge in the vehicular network domain [12]. Combining NDN with vehicular networks is a solution that may reduce the issues faced by traditional vehicular networks. NDN is relatively fast and more efficient than a traditional network. Two models of communication are used in vehicular networks: ICN and an address-centric network. ICN is a routing architecture of NDN. Named data are used for communication establishment. Joint online optimization of the data sampling rate is a process of collecting data from sensors in real-time by adjusting sampling rates to achieve objective functions. It is a sophisticated approach that provides a balance between system performance, quality, and resources. In contrast, the preprocessing mode for edge-cloud collaboration-enabled industrial IoT is a way of leveraging between cloud and edge computing with the help of data preprocessing to enhance the industrial IoT system. It helps improve the use of raw data before they are stored or used. Another optimization approach is the vacation queue-based approach, which enables better energy consumption and latency using a queue-based approach. It dynamically allocates the applications in cloud computing devices by considering efficient energy resources. In vacation queue mode, the server shows the status as inactive when it is not participating in processing tasks. During this timeframe, the server saves its energy by minimizing its resources. In contrast, the address-centric model uses the nodes' addresses, such as their IDs, to generate communication paths. Cluster-based routing in named data is an algorithm that lessens the

cons of the vehicular network. Some of the main aspects of this algorithm are highlighted below:

- It uses the NDN mechanism to reduce the error rate.
- It uses the NDN mechanism to provide greater bandwidth and efficient usage of the network.
- Provides greater delivery ratio of data by using ICN setup.
- It is a self-configuring and lightweight technique.

By implementing a vehicular network with cluster-based routing in the named data, the results are highly improved [13]. This clearly shows that if different vehicular techniques are combined with NDN, we will get huge benefits.

4. Educational network: The educational network is one of the primary aspects of today's world. Traditional IP networks are used to connect students and teachers. Sometimes, IP networks fail to meet teachers' and students' demands. The reasons for IP network failure are too much traffic, less throughput, and more delays. Therefore, it is essential to use a network that can overcome these issues. By implementing NDN in educational networks, all these issues may be overcome. Therefore, a new program named inherent Indonesian topology was started by the Indonesian government. This program connects 33 Indonesian cities' educational institutes using satellite or fiber optics networks. It performs different experiments by considering factors such as throughput, delay, and packet loss. Throughput is how many bits are transmitted successfully in a given time frame. Delay is defined as the time spent sending or receiving information. Packet loss is the number of all the sent packets that do not safely reach their destination. In their scenario, they stored 100 packets of INTEREST with a payload size of 1024. They also used 1000 CS. Two producer nodes and four consumer nodes were used. The best route policy was also used. Network Simulator 3 (NS3) was used to carry out the simulations. All files were run on ndnSIM. They ran simulations approximately every 20 s. The routing information protocol was used in the IP scenario. The simulation results of NDN and IP are listed below:

- The throughput of NDN was better than IP. Because the NDN CS table caches the information, there is no need to go to the destination to get information. In contrast, the throughput of IP was inferior because all information was gained by the nodes from the original location.
- Two types of delay were measured by ndnSIM: last delay and full delay. The last delay is the total time a packet needs to arrive from one place to another. In this duration, the path length is not considered. Thus, this is a smaller delay. In contrast, full delay is measured as the total time a packet needs to travel from one place to another. During this duration, the path length is calculated. Thus, this is a larger delay. The delay was higher for the IP network because the nodes had to collect data from the destination, which may be far away. NDN had a low delay because all nodes were cached, and any node could fulfill the user's query.
- For both the NDN and IP networks, no packet loss was detected. This was because the scenario that they used did not have enough consumers [14].

Videoconferencing in real-time is another important feature used in educational institutions. Especially in online classes, almost all lectures are attended via videoconferencing. Implementing NDN in videoconferencing in real time is quite an appreciated approach. Videoconferencing can benefit from NDN's forwarding scheme, caching scheme, data signature, and aggregation. The main goals of implementing NDN in videoconferencing in real time are listed below [15]:

- Audio/video in low latency.
- Caching ability by all nodes.
- Verification of data.
- Network assumption.
- Fast data retrieval.

5. Entertainment networks: Multimedia is the basic source of entertainment, and the prime form of multimedia is video. As the number of users increases, streaming of different videos on computers, mobile phones, tablets, and laptops over traditional IP networks is limited to network capacity, bandwidth, and throughput. To overcome these issues, a new network paradigm should be considered. NDN is particularly well-suited to the speedy delivery of video for clients. Different experiments were performed. These experiments included streaming many videos of up to 18 Mbps and resolutions up to 4 K. Multiple clients and channels were considered. Different building blocks for comparing NDN and IP were used. Some comparisons are listed below:

- In-network recovery of loss was handled by the NDN with the help of caches.
- Better bandwidth because the demanded data could already be in the cache.
- NDN had better load balancing as NDN nodes store data [16].

3 Caching in NDN

In this section, we focus on the classification of caching policies. Nodes store the objects, data, and information temporarily to support peer-to-peer and content-centric approaches. An advantage of caching is that it speeds up the data transmission process. Caching decreases the extra cost of data transmission and saves time on data recovery. It also helps lessen the expensive cost of downstream and up-streaming of user requests and information. In NDN caching, when a user request is sent by the client, the node that is closest to the client will try to find a DATA packet against that INTEREST. If a DATA packet is found, then it forwards the DATA packet to the client, which results in client satisfaction promptly. Caching provides several benefits:

- Reduces traffic load on nodes.
- Improves response time as almost all nodes cache the content.
- Better bandwidth utilization.
- Reduces data waste using multiple cache policies.

3.1 Need for Caching Strategies in NDN

In CCN, data content is the most precious component of communication. As we discussed earlier, NDN offers several advantages over traditional IP networks, such as mobility, security, and simplicity. Furthermore, it enhances the diverse model of communication by transmitting heterogeneous data. Two terms need to be considered: homogeneous caching and heterogeneous caching. In homogeneous caching, routers cache the same content packets that are passed from them. NDN and CCN architectures also support homogeneous caching [17]. For example, as seen in Fig. 3, two routers (R1 and R2) cache the same content (D1, D2, and D3). In heterogeneous caching, some routers cache the DATA packet along the path. For example, as seen in Fig. 4, routers R1 and R2 cache the content differently. D1 and D2 are cached by router R1, whereas D3 is cached by router R2. Each router has a different cache size in heterogeneous caching [18]. Therefore, NDN is the future of the Internet, and caching architecture based on NDN is beneficial to gain notable performance.
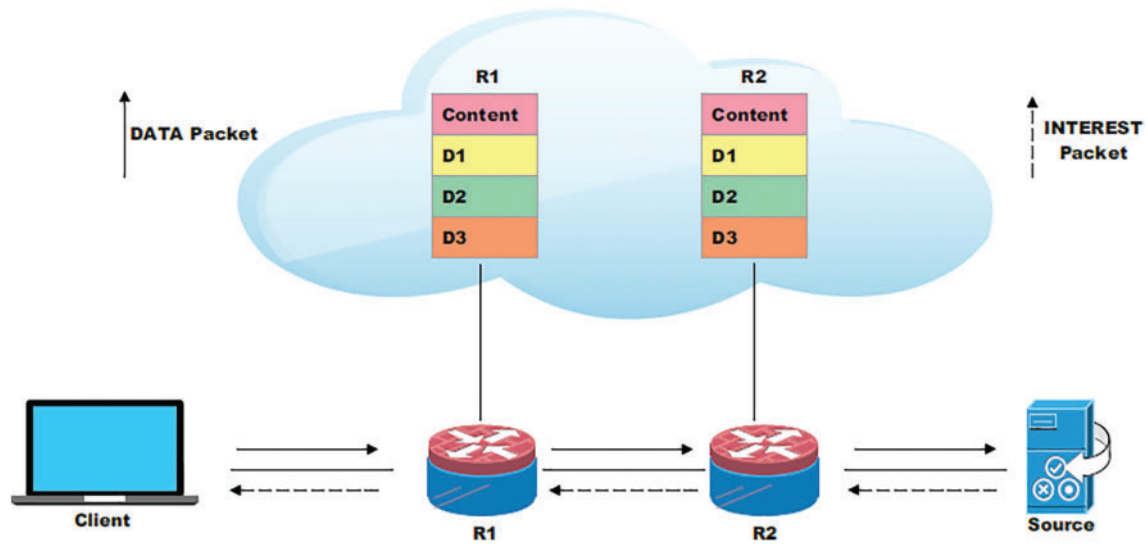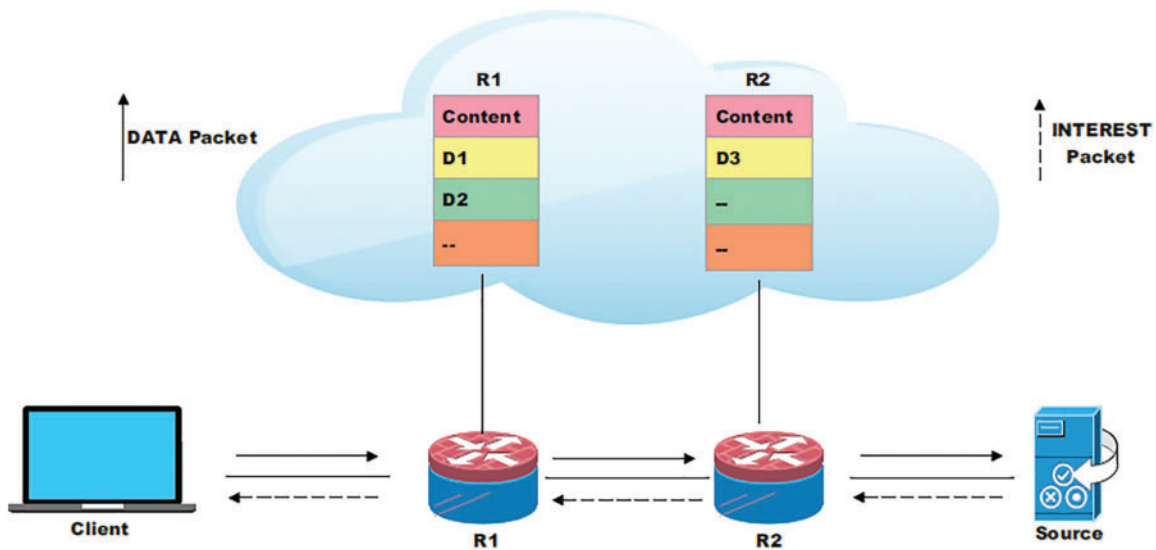
Figure 3: Homogeneous caching

Figure 4: Heterogeneous caching

Three main caching aspects are used in NDN scenarios:

1. In-network caching: This is a strategy in which data are generally stored in an area nearer to the user who requested those data. For example, a browser stores its site resources on a web cache that is closer and more reliable for the user. By doing this, the web query is efficiently satisfied. Caching in network has two different forms:

   - Non-coordinated caching: In this form, data and different objects are selected without any precautions. Data duplication and redundancy are not considered. By doing this, neighboring nodes may contain identical data. Load on the original data distributor is high, as data need to be extracted from the original data distributor. The hop count of the router is also maximized.

- Coordinated caching: In this form, the data and different objects are carefully selected to avoid data duplication in the nodes. This enhances the latency. By doing this, neighboring nodes do not contain identical data, and the load on the original data distributor is reduced. The hop count of the router is also minimized. A few research questions still require researchers' attention [17]: When should we use a coordinated caching scheme? What is the best way to use a coordinated caching scheme? Which metrics should be used to differentiate between recently used data to avoid duplication?

2. On-path caching: This is a strategy in which a place decides the type of caching. In this, when a user requests content, the data content are cached along a path. On-path caching is easily applicable in NDN. The mechanism of on-path caching is illustrated in Fig. 5, which shows that routers that lie in the path between the client and source can cache the data. This is the most important module because its approach is flexible towards content caching. CCN, NDN, and DONA also support on-path caching [17].

3. Off-path caching: In off-path caching, when a user requests a DATA packet, the data content may or may not be cached along the path. In NDN, a centralized topology exists. This means that the server handler that is placed in the center is the only server that decides where to cache the data. A central server handler decides where a DATA packet should be cached [18]. The off-path caching mechanism is illustrated in Fig. 6, which shows that the server handler decides where to cache data. It is necessary to have caching strategies. However, by implementing more efficient performance regarding caching, results may be more favorable. Efficient performance of caching demands more efficient strategies that are capable of searching for the appropriate nodes or routers for data storage.
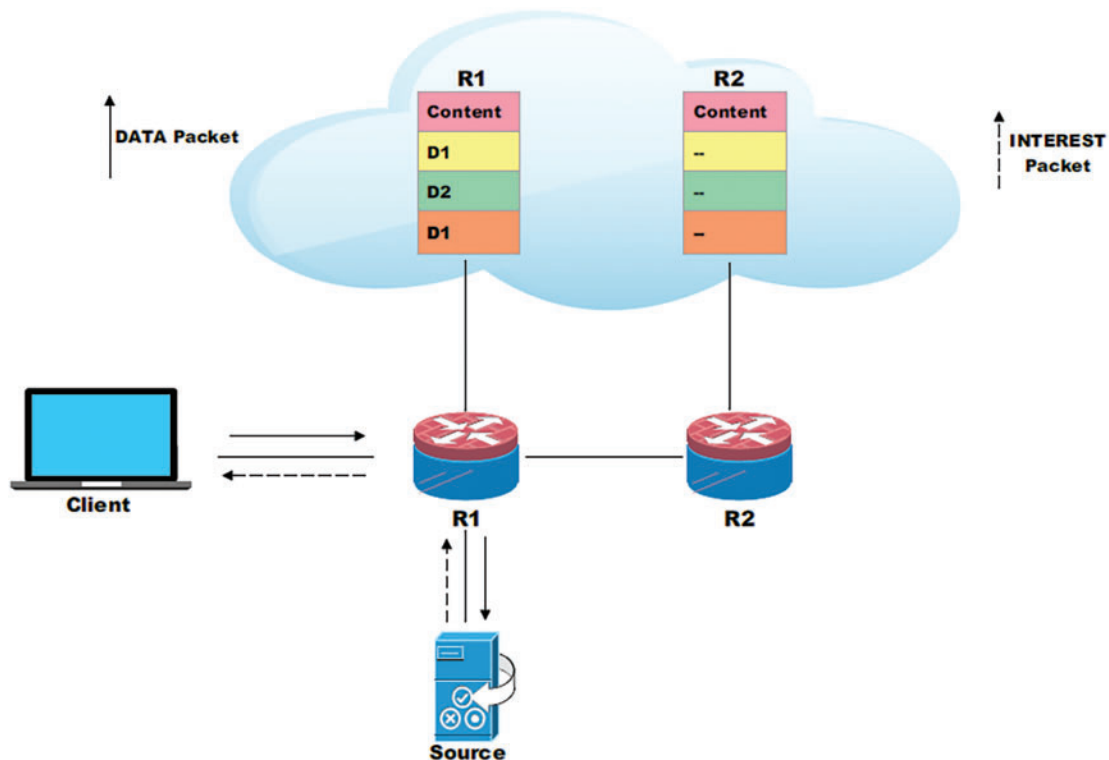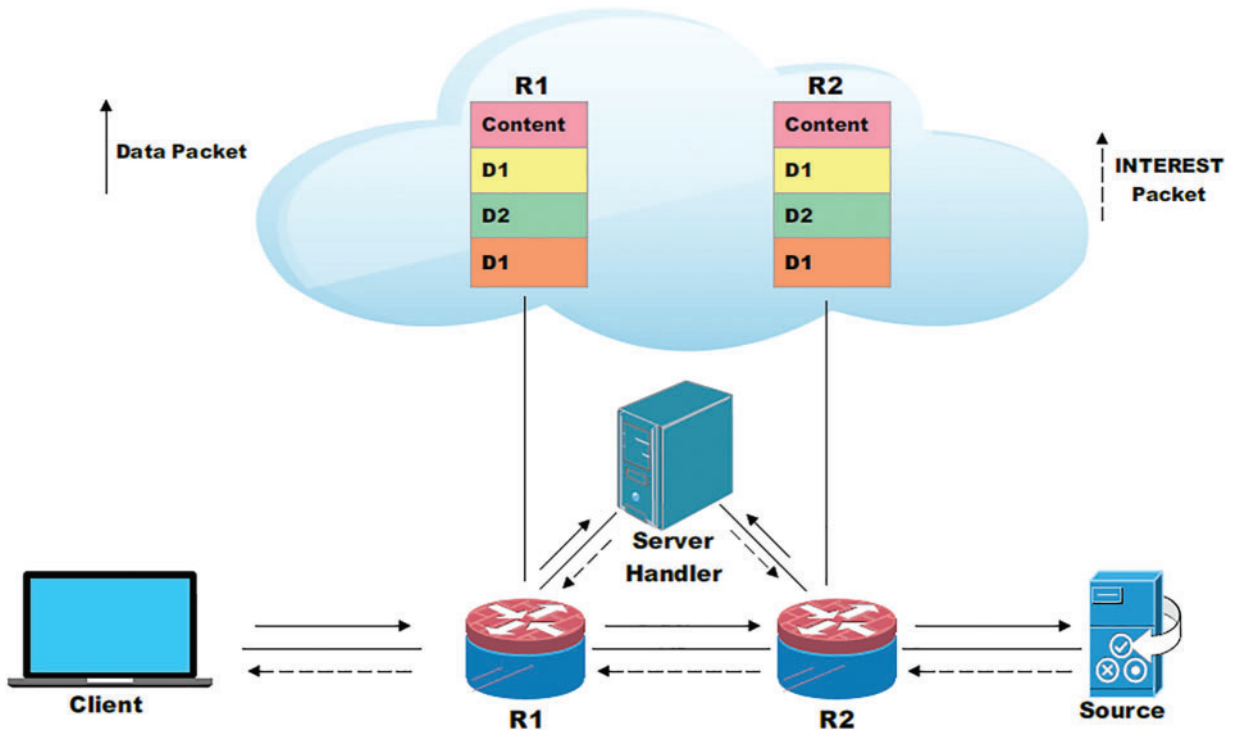

Figure 5: On-path caching

Figure 6: Off-path caching

3.2 Caching Policies

Decision policies for caching decide where DATA packets should be cached, such as whether they are cached on interposed routers or not. For better understanding, there are two main aspects: placement of the cache (at which nodes or routers the cache is placed) and replacement of the cache (which cached data are replaced with new cached data).

1. Placement of cache: This is a caching technique in which the contents of the data are stored in the cache. This decides where to store the data content [6]. There are many policies regarding content storage. A DATA packet that is available in the network will be cached at downstream neighboring nodes. Thus, the packet is cached along the data delivery path. A DATA packet that is available in the network will be cached among all the nodes or routers. If a new request for content approaches, intermediate nodes will try to satisfy the request because they have a copy of that particular content. The utilization of bandwidth is increased in the on-path caching mechanism. Many ICN architectures support different caching techniques.

2. Replacement of cache: Replacement of the cache is used to decide which data should be replaced with the newly cached data. Basically, it is the main aspect of cache policies [6]. Many replacement policies for cache storage exist. Sometimes, the content that is accessed the least recently is dropped by the nodes or routers. In others, content that is used less is the first to be dropped. Drop timing may depend on the router.

4 Caching in NDN-Based Wireless Ad Hoc Networks

A wireless network consists of multiple moveable devices that are connected via some kind of signal. It is a huge group of nodes that exchange information with each other. These nodes are connected to each other without any fixed infrastructure or topology. They use radio signals for communication between other wireless devices. This is a kind of dynamic network. In this network, nodes do not rely on a predefined infrastructure; because of this specification, they are called wireless ad hoc networks [19]. A radio network consists of multi-hop devices for communication. Each single node is capable of directly communicating with the other nodes. Different wireless technologies (such as Wi-Fi and Bluetooth) use an ad hoc wireless network. These networks are very helpful in emergency scenarios. In contrast, NDN supports caching in which data are exchanged between different devices in an asynchronous manner. Devices at both ends of the wireless ad hoc network may or may not be connected simultaneously. Maintaining a path between two end devices is also not necessary.

Mobility is the main challenge in a wireless ad hoc network because it is difficult for a device to perform its services continuously after switching from one region to another. As NDN is decentralized, it provides a flexible and efficient caching mechanism in the network. If a central server station fails in a simple wireless ad hoc network, then coverage issues may occur. Using an NDN-based caching mechanism in a wireless ad hoc network increases the response time of the network compared with a traditional IP-based wireless network. An NDN-based caching mechanism with a wireless ad hoc network can be used in offices, on roads (i.e., syncing side unit signs from the road via moving from one place to another), by educational institutions (i.e., arranging online classes in less time), by the military (i.e., establishing communication setup between army headquarters in less time), and in industry (i.e., dealing with foreign clients via video meetings by establishing quick connections). It also provides support to devices and nodes to help a node deal with mobility issues because there is no need for different network configuration parameters (such as an IP address) in NDN. Nodes can easily take advantage of the CCN approach as their location is not fixed in NDN, and data can be fetched from anywhere at any time. Implementing an NDN caching mechanism can improve various wireless ad hoc networks, such as MANETs, IoT, and VANETs. Avoidance of packet collision and aggregation of INTEREST packets (i.e., if the same INTEREST packet is received by the node, then it denies the INTEREST transmission) can increase the performance of wireless ad hoc networks based on NDN. Moreover, multiple strategies in different wireless ad hoc networks using an NDN scheme resulted in better and improved versions of those wireless ad hoc networks [8]. Some of the advantages of caching using wireless ad hoc networks are listed below:

- Enhances the data transmission in a sped-up manner.
- Decreases the extra cost of data transmission and saves time on data recovery.
- Reduces the traffic load.
- Improves node response time.
- Improves bandwidth utilization.
- Decreases unwanted content by using different node cache policies.
- Improves data availability.
- Lessens query delay time.
- Widely used in homes, educational institutions, industry, and private and public sectors.

There are many types of NDN-based wireless ad hoc networks; we discuss caching mechanisms for the following:

1. Caching in NDN-based MANETs
2. Caching in NDN-based IoT
3. Caching in NDN-based VANETs

5 Caching in NDN-Based MANETs

MANETs are basically networks that consist of mobile devices or nodes that are capable of moving from one place to another. Advancements in MANETs have made them an ideal candidate as the next main source of communication. Nodes in MANETs create a network that is decentralized. These nodes automatically connect over a link (i.e., wireless). Nodes present in the same range (e.g., radio signals) interact directly with each other. There is no need for a dedicated router or node because all routers and nodes are capable of self-forwarding the packet. Each device or node can work as a router and can transfer the packet without any difficulty. This network is very helpful in several environments (e.g., earthquake rescues, battles, and disaster recovery) [20].

Caching in NDN-based MANETs is a process in which different mobile nodes cache the data for future use. If a user demands specific data from the network, the mobile nodes contact each other and check their cache storage. If the cache storage contains the data, then they are forwarded to the user. Through caching, overall time is saved. If the data are not found in the mobile nodes, then a data request is sent to the data originator or server. The data originator or server sends the data to the demanding node, and the mobile nodes cache the data for future user requests. NDN-based MANETs works in different challenging and active environments. They are limited in terms of connectivity, battery, and storage. They also face many problems such as retransmission of packets, packet collision, data duplication, and packet flooding. In NDN-based MANETs, INTEREST packets are forwarded via different methods [21]. Different types of caching in NDN-based MANETs may be used depending on the scenario. Caching is the core aspect of NDN in MANETs, and it provides multiple advantages:

- Avoids caching of similar data content on adjacent nodes.
- Lessens the duplicate data in the network.
- Enhances the cache hit ratio.
- Quick response time.
- Decreases data retrieval latency.
- Lessens duplicate data requests in the server.
- Covers mobility issues in some cases.

The working mechanisms of MANETs are depicted in Fig. 7, which shows different wireless devices communicating with each other and messages being forwarded from source to destination using different wireless nodes. In MANETs, caching is also a challenge because the environment is different and resources are limited. Caching in MANETs depends upon the placement and distribution of data. Many aspects limit the performance of caching mechanisms in MANETs. These aspects are:

- Node mobility.
- Limited resources.
- Limited energy.
- Limited bandwidth.
- Limited memory.

Strategies for data caching in NDN-based MANETs are necessary to deal with energy, memory, bandwidth, and resource constraint issues. Caching strategies in NDN-based MANETs decide how to efficiently select the best caching scheme. The design of NDN-based caching in MANETs' schemes depends upon the context and complete network scenario. A detailed classification of different caching strategies in NDN-based MANETs is illustrated in Fig. 8.
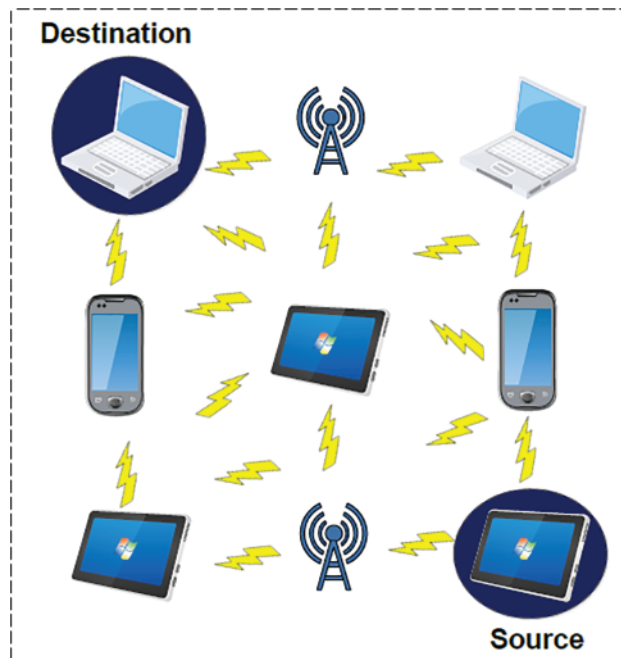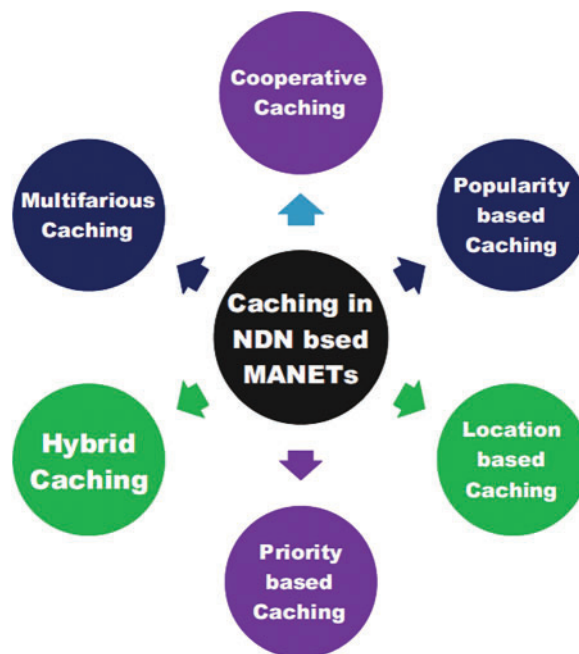
Figure 7: Working of MANETs

Figure 8: Caching strategies in NDN-based MANETs

We divide the different caching strategies for NDN-based MANETs into different branches:

1. Cooperative caching
2. Popularity-based caching

3. Location-based caching
4. Priority-based caching
5. Hybrid caching
6. Multifarious caching

5.1 Cooperative Caching

In wireless ad hoc networks, the cooperative caching technique is used the most often. Due to several challenges of node size and network nature, there was a need for an efficient caching scheme in wireless ad hoc networks, which is fulfilled by cooperative caching. Cooperative caching is a type of in-network caching. In-network caching enables more than one node to work together to enhance the caching process. By doing in-network caching, many challenges and limitations can be resolved. Cooperative caching allows sharing and mutual coordination between multiple moving nodes. This helps improve network performance.

Cooperative caching consists of four components. The first is data locating, which focuses on locating the data in the cache. When data are required by the node, it first determines whether the required data are available in the node locally. If data are not available in the node, then this component evaluates how to access the data. The second is cache management, which caches all the data regardless of where they come from. It may use certain parameters (e.g., bandwidth, distance between the nodes, and latency) to decide what to cache. The third is replacement of the cache, which is when the decision to eliminate data content is decided independently by the node using its native information. The fourth is cache regularity, which ensures whether cached content is valid or not. Due to congestion in the network, a user request may not be very important if it takes a long time to respond to the user query. Therefore, some kind of attribute (e.g., TTL) should be added in the caching process that tells the system about data validity [22].

In [23], a caching strategy in a CCN-based MANETs (CSCM) technique based on cooperative caching was proposed by the authors. This technique selects the suitable node for caching by considering the factor of mobility. It also lessens the mobility issue of nodes. Their model consists of different mobile nodes. These nodes work as the consumer, intermediate, or producer. Nodes can freely move in the region. The selection of appropriate nodes is an important task. For better overall performance, each and every node calculates its ability to cache the incoming content. For this, each node calculates its selection score (SC). A threshold value is also defined. If the SC is less than the threshold value, then it transfers the content to other nodes. If the SC is greater than or equal to the threshold value, then it caches the content. Thus, only a few nodes are selected for the cache instead of every node. This helps with energy efficiency. In this way, content duplication is also reduced. For this, SC is used to calculate the best suitable node for caching the content. The SC for every single node and for all incoming data was calculated. The authors tackled different metrics for different limitations of MANETs, such as available cache memory and middling nodes. These metrics were also calculated and combined with SC. The equation for these metrics is presented in [23].

Middling of the node was also performed. This means that a node that has the highest middling ranking among the other nodes enhances the cache accessibility and has more successful cache hits. Node middling is used to find the most important node in the network [24]. In CSCM, node middling is measured using the eigenvector centrality formula [25]. This formula indicates if a node is connected to other crucial nodes that have greater values of eigenvector centrality, in which case, it is a middling node. If a node that has a low eigenvector centrality value is connected with multiple other nodes, then the node's eigenvector centrality becomes higher, and this node is labelled a middling node. The

middling value score of every node can be calculated according to its connection with other nodes. These middling values are broadcast to the adjacent nodes. If nodes that are fewer middling nodes are now middling nodes after knowing that their neighbor node is a middling node. This helps with tackling topology or mobility changes.

Checking available cache memory is important for caching new content. There are two types of mobility: mobility of the node within the area (content availability never occurs) and mobility of the node across the area (content availability may occur). Therefore, mobility across the area is considered here. If a node guesses that it is about to move from one area to another, it selects the most suitable node to transfer all of its content to so that the new node works as the cache node in that area. In CSCM, each middling node checks its status. If its status is about to move, then it broadcasts a message to its adjacent nodes. The broadcasted message includes a list of available data. After that, every node calculates its SC. If its SC is less than the threshold or status of the node that is about to move, then it cannot send INTEREST packets. If the SC is greater than the value of the threshold, and the status of the node is not about to move, then the node forwards its INTEREST packet. After receiving the INTEREST packet, the node sends the content to the new middling node [23]. In [23], the performance of CSCM was evaluated using NS3. Fifty nodes were installed in an area. These nodes moved from one place to another randomly. One node was labeled as a producer node, while the remaining nodes were labeled as client or consumer nodes. The results showed that the network traffic was reduced, the hit rate increased, and content retrieval time increased.

In [20], the group caching (GC) technique based on cooperative caching was described by the authors. MANETs can exchange information with other nodes at any time. Therefore, a cooperative caching scheme was used in this technique. In this technique, each mobile node created a group with its adjacent nodes that were one hop away. The caching status of all nodes was recorded in the group. By using this technique, the caching space was utilized efficiently, average latency was reduced, and data duplication was decreased. In this model, graph (M) shows different vertices representing nodes. Edge (E) represents the connectivity between them. Edge (E) is present between nodes only if another node is in range. This edge may be bi-directional. A group is formed by the nodes that are connected to their one hop adjacent nodes. A unique group ID was assigned to each node. The node that was connected to the highest number of other nodes was known as the leader node. A "SYNC" message was broadcast to group members to check the connectivity of nodes. In this way, each node could identify its neighbor nodes. Each node forwarded its caching status to its group member nodes. Thus, when new content became available, the leader node selected the most suitable node from the group for caching. Each node from the group contained two records for maintaining the caching status of the group. These two tables are the collective table and individual table. The collective table stores the group members' node caching status. It contains the columns cached content ID, content source ID, timestamp, and cached content. In contrast, the individual table stores the caching status. It contains the columns cached content ID, content source ID, timestamp, and ID of the group member. When a mobile node gets a caching status notification from the other nodes in the group, then it modifies the collective table.

Leader nodes know which content is stored in which node in the group with the help of the collective table. When an INTEREST packet is received by the leader node, then it searches both records and finds the requested content. A message is broadcast to every node in the group to check the available cache space of each node. The message contains information about different fields (e.g., group member ID, cached content ID, available cache space, and timestamp). The leader node updates the collective table after receiving the messages from the other nodes. When new content arrives, the leader node checks its cache storage space. If the storage space is greater than the new content, then

it stores the data; otherwise, it checks the collective table and forwards the data to the node in the group that has greater or equal storage compared to the newly arrived content. If no sufficient space is available in all nodes in the group, then the leader node checks whether this content is already cached in other nodes in the group. If yes, then it discards the data. If no, then the leader node selects the node that has the oldest cached data timestamp, and the content is placed in that node. When clients demand data from the source, the leader node first checks its individual table and then the collective table. If found, then it forwards the content to the client. If not found in both tables, then it forwards the request to another leader via constructing a routing path to the data originator. The next leader repeats the same process and checks both tables first. This process continues until the data are found. If the data originator receives a request for data, it sends the data, and cache placement and replacement is done by the intermediate nodes. In [20], GC performance was evaluated using NS2 for 95 nodes. The radio range was about 99 meters, and 999 content packets were distributed among all the leader nodes. These nodes moved from one place to another randomly. The results show that the average hop count was reduced, the cache hit rate increased, and the average latency decreased. Another study presented a cooperative scheme combined with popularity for improved cache hit ratio [26]. Improved caching was also found by using a valuable node for suitable caching in MANETs [27]. Furthermore, cooperative caching schemes in MANETs were described by the authors in [22,28].

5.2 Popularity-Based Caching

Every NDN-based mobile router or node keeps track of the total number of times specific content is requested by the client. It calculates the total number of data chunks by seeing the demand for that particular datum [6]. For example, if a user demands a piece of content, the mobile nodes first check their caches to determine if the demanded content is available or not. They also count the total number of requests for that specific content. If the content is not available, they request that content from the data generator and cache the content for future use. If the count for a specific piece of content increases beyond the threshold value, then this content is labeled as popular content. Popular content has the highest hit ratio.

Energy, cache space, and mobility are issues in MANETs. Therefore, time-dependent caching (TDPC) techniques were proposed by the authors in [21] as a solution to overcome these issues. TDPC is popularity-based caching. It caches the data along the forwarding path. It is a time-dependent scheme that chooses the content on the basis of its popularity. It also measures the storage capacity and cache distribution of MANETs' nodes. In this technique, not all content is cached on all nodes along the routing path, but the content is cached on a few nodes. It uses two factors for caching the content: the gap between consumer and time-based popularity. The gap between nodes is measured by checking the hop counts. The total number of nodes that are passed by the INTEREST packet helps calculate the hop count. A new column is added to the INTEREST packet that measures the hop count. When a user demands the content, then the INTEREST packet is forwarded to the adjacent nodes. First, the CS table is checked for the required content; if it is found in the CS table, then the content is immediately sent to the user. If the required content is not found, then the PIT is checked. If the content entry is not found in the PIT, then it adds the entry in PIT, increases the hop count in the INTEREST packet, and forwards the packet further. Every time the INTEREST packet is moved forward, the hop count is increased by one. When the content is found, it goes backward, and the PIT is checked for the interface. If the interface is not found in the PIT, then it discards the data. If the interface is found, then it transfers the content to the user. The device with the smallest hop count is considered the nearest node. In the future, if the same node requests the content, then it gets the data from this node. When an INTEREST packet is received by the data originator, the total hop count is

fetched and added to the content. Popularity based on time is calculated here. The amount of content is divided by the current time to calculate the popularity of the content. If the popularity is greater than the old limit, then the content is stored in the CS table. In [21], the performance of TDPC was evaluated using NS3 for 55 MANETs nodes. These nodes randomly moved from one place to another at speeds of 9 to 24 m/s. Some nodes were labeled as producer, consumer, and intermediate nodes. The results showed that the content retrieval time was better, the cache hit rate increased, and unnecessary replacements decreased.

In [29], a scheme called less space still faster (LF) was proposed by the authors to enhance the caching of NDN-based MANETs. LF places the content close to the requesting node. LF provides a faster response using less space. Four factors are responsible for LF's fast pace. Caching only occurs if the free space is greater than the space required for caching. Distance is measured by using the total number of hops that a packet passed by. A new field is added in the INTEREST packet that counts the hops passed. The hop field also checks when a DATA packet is forwarded to the destination. The node with the minimum hop count is selected as the caching node. Content popularity is represented by frequency. The frequency of each packet increases by one when a particular piece of content is demanded. Content with the highest number of requests is considered popular content. These contents are cached as urgent. A new field is added to the INTEREST packet that determines the data redundancy. In order to detect data duplication, a new field provides values that indicate whether the data are already cached or not. The performance of the LF was evaluated using NS3 [29]. Sixty MANETs nodes were used. The distance between these nodes was at least 240 m. The results showed that the response time was better, and the cache hit rate increased.

5.3 Location-Based Caching

In location-aware caching, all NDN nodes in MANETs cache the data according to the mobile nodes' locations. Performing caching according to location parameters results in different improvements. If the location is known by the NDN-based mobile nodes, then they may cache the data in the nearest node, which helps improve the performance of the overall network. The performance of the network depends heavily on the location of the caching mobile nodes.

In [30], a novel caching scheme, location-based caching in CCN MANETs, was described by the authors. It is based on the location of the mobile nodes. Each mobile node uses a location system (i.e., GPS) to know the exact location of itself. The location of every node is shared while sending and receiving INTEREST and DATA packets. When a DATA packet is needed by the client node, an INTEREST packet is forwarded to the whole network. The INTEREST packet also contains location coordinates. When this packet is received, the total distance is calculated using the Euclidean formula. If the distance between the client node and the intermediate node is less, then it forwards the packet further. If the distance is greater than, it discards the packet. Every single node in the network performs distance-based caching. This avoids a slow response time as content is cached on the nodes that are nearest to the client node. In order to lessen content duplication and collision, every mobile node continuously senses the whole network for a limited amount of time. If a similar DATA or INTEREST packet is received by the mobile nodes during this period, then this transmission is terminated. The authors evaluated the performance using ndnSIM. Mobile nodes were found to travel up to 700 m. The nodes cached 990 packets. The results showed that the content retrieval time was improved, and packet retransmission decreased.

The authors of [31] proposed a scheme called location-based on-demand multipath caching and forwarding (LOMCF). This scheme was also based on location caching. In NDN, if a node demands

content, then a standard NDN communication procedure is followed. In LOMCF, the location of both the sender and receiver is considered during the forwarding mechanism. Here, distance-based caching is performed by every single node. When a DATA packet is needed by the client node in the network, an INTEREST packet is forwarded. Location coordinates are encapsulated in the INTEREST packet. The distance from other nodes is calculated using the Euclidean formula. If the distance is greater than, it discards the packet, and if the distance between the client node and intermediate node is less, then it forwards the packet to the other nodes. This avoids slow response times as the content is cached by the nodes that are nearest to the client node. By calculating the total distance between the client and intermediate node, unnecessary packet flooding between the networks is reduced. The authors evaluated the performance of LOMCF using NS3 for 990 DATA packets. The results showed that packet retransmission decreased, and content retrieval time improved.

5.4 Priority-Based Caching

Priority-based caching means that the content is cached on the basis of priority. The priority assigned to the different data content indicates which data content will be swapped first. When different nodes try to exchange data content, they swap their high-priority data content first. Content with a high priority will be found in NDN more frequently compared to content with a low priority. Content with a low priority may suffer from poor latency. Priority can be of any type depending on the developer [6].

Caching strategies in NDN-based MANETs focus less on content precedence. This affects the crucial data usability. In [32], a priority-based content caching scheme was proposed by the authors. Different data are prioritized by considering different demands of the data. Content priority is the key factor in this scheme. The authors added a new field, named field of priority, to the NDN packet. A greater priority value is added in the field for identifying priority. Frequency of content was also introduced to calculate how many times a particular piece of content was demanded in the CS table. This is used to distinguish between the popular and unpopular content. Frequency of content is measured by multiplying the content size by content priority. The authors evaluated the performance using ndnSIM for 45 mobile nodes. The results showed that content availability improved, and important content was better cached.

5.5 Hybrid Caching

Hybrid caching is when multiple schemes are combined with each other to obtain better results. In [26], a novel cooperative scheme combining a popularity mechanism for caching in CCN MANETs was proposed by the authors. They named this scheme CCMANET. They developed a random topology and implemented the cache scheme on different levels. They also elaborated a way of calculating the popularity of content in the network. They evaluated this scheme using NS3. Both the cache hit ratio and server load improved. In [27], cooperative caching combining popularity and probability were implemented via groups in NDN-based MANETs. Their model selects the most crucial adjacent node for cache placement. It excludes the content that has the lowest probability. They evaluated this scheme using ndnSim and found that the cache hit probability improved.

5.6 Multifarious Caching

Here, we summarize the different caching schemes that do not adopt any targeted schemes. This involves all the caching policies that depend on different parameters. Many more caching techniques exist in NDN-based MANETs. In [33], a technique for caching in MANETs was described. This

technique works on a normal caching scheme. A new parameter (interval of data) was added to the DATA packet. This parameter manages the interval of data along the forwarding path. Performance was evaluated using the QualNet network simulator. The results showed that overhead decreased, cache hit rate increased, and average path length decreased. In [34], a technique for cache space efficient caching in MANETs was described. This technique works on the graph scenario, which represents the mobile nodes. If graph nodes are connected, this means that mobile nodes are connected. Performance was evaluated using ndnSIM, and the results showed better cache space utilization and increased cache hit rate.

6 Caching in NDN-Based IoT

The IoT is usually considered to be a type of WSN. The IoT has received much attention due to its recent advancements. It enables different hardware objects to think, learn, hear, and perform different crucial tasks. The IoT uses devices, actuators, and sensors that interact with each other and perform multiple operations. It converts these devices into smart devices by giving them these abilities. Radio frequency is necessary for IoT communication. However, many technologies have been introduced that are similar to radio frequency. The IoT can connect almost any device with others. There are many state-of-the-art techniques for operating the IoT. The first technique is to use fully automated methods of retaining and organizing content. The second technique is to use a new method that is more productive to get information. The third technique is to use a secured method to combine IoT data regardless of where they originated from. The IoT is used in different applications [35].

The IoT consists of a data-centric network. IoT devices heavily depend on resources, and sometimes these resources are limited. The nature of the IoT network and inefficient addressing make the IoT a perfect candidate for implementing NDN. The IoT naming system lists multiple sensing tasks. The NDN naming scheme is similar to the IoT naming scheme. For example, the aggregator node depicts what type of data is necessary. Applying NDN in WSN has several advantages:

- Provides caching as sensor nodes have storage capability.
- Provides easy retrieval and searching of content via hierarchical naming.
- Provides flexibility in deployment by implementing NDN on top of a traditional IP network.
- Provides better scalability as several nodes easily retrieve data using the NDN approach.
- Provides easy development of applications [9].

The performance of IoT nodes is affected by many parameters, such as the limited storage space of IoT nodes, limited energy resources, and limited bandwidth. Strategies for data caching in NDN-based IoT are necessary to deal with these aspects. The design and selection of NDN-based IoT caching schemes depends on the context and complete network scenario. A detailed classification of caching strategies in NDN-based IoT is illustrated in Fig. 9. We divide different caching strategies for NDN-based MANETs into the following branches:

1. Cooperative caching
2. Freshness-based caching
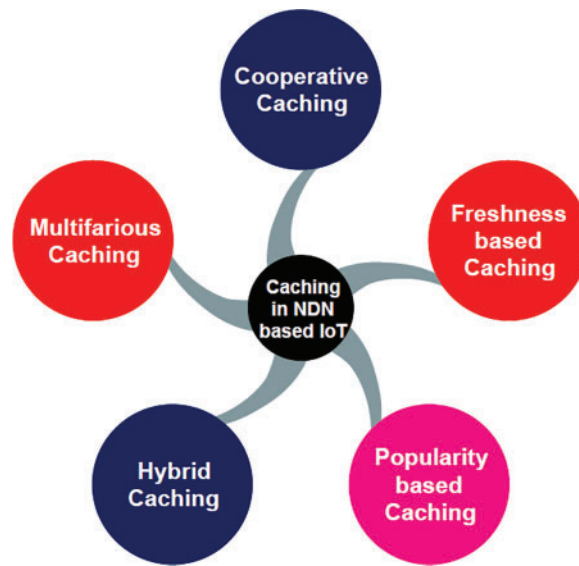3. Popularity-based caching
4. Hybrid caching
5. Multifarious caching

Figure 9: Caching strategies in NDN-based IoT

6.1 Cooperative Caching

In the IoT, the cooperative caching technique is widely used. Cooperative caching allows sharing and mutual coordination between multiple moving nodes. Cooperative caching works on the mechanism of in-network caching. NDN also supports in-network caching because more than one node can work together to enhance the caching process. Cooperative caching helps improve network performance by resolving several restrictions, such as slow response time to user queries, the limited storage space of IoT nodes, limited energy resources, and limited bandwidth. It is based on four important parameters.

In [36], a disaster management system (DMS) scheme based on cooperative caching was proposed by the authors. In this technique, the producer node sends a fire alert notification to the adjacent consumer. A fire scenario is considered in school. In this scheme, the nodes collaborate with each other and manage the situation efficiently. PIT is restricted because, in a disaster, it is useless to maintain PIT entries.

The reason for this is that the main focus is to send the fire alert notification to the adjacent nodes. This helps decrease the delay. If the value of a fire sensor crosses the threshold limit, then an alert notification is broadcast to the network using a push-based technique. Fire sensors are implemented in schools. When the sensor values are less than the threshold level, then the sensor indicates that the environment is safe. When sensor values are greater than the threshold level, then the sensor indicates that the environment is unsafe. In an unsafe environment, all nodes send alert information to each node. In this way, all nodes are notified, and all instructors and children know about the fire incident. The authors evaluated the performance of DMS using NS2 for 49 nodes placed in a school. The simulation results indicated that energy utilization and delay were reduced.

In [37], the authors proposed a form of cooperative caching called information-centric wireless sensor networking (ICWSN) for an energy-efficient WSN. As NDN sensor nodes have a limited battery and may have to wait for a long time for a request, this results in an inefficient use of energy. Therefore, the authors suggested an energy-efficient cooperative caching scheme that improves fast

data fetching from cache nodes and better energy usage. NDN-based WSN sensors were deployed that frequently sense the network and forward the required data. Normally, three working modes are considered in an IoT scenario: normal, mild sleep, and full sleep. In normal mode, the nodes perform their operations normally. In mild-sleep mode, the nodes are in power-saving mode and use less energy, whereas in full-sleep mode, the nodes are sleeping and are activated by an alarm in the case of any processing. When an INTEREST packet enters the network, NDN sensors check the PIT to detect whether this packet was requested earlier or not. If it was not requested earlier, then the sensors activate the policy of cooperation and calculate the other parameters, such as eigenvector centrality [25], to determine the node with a large enough neighborhood to cache the content. The authors evaluated the performance of ICWSN using the NetworkX library for 95 nodes. The simulation results showed that delays were reduced, and energy consumption was achieved.

Data freshness is used for caching schemes that cache the content on the basis of content lifetime [38]. In [39], cooperative caching was proposed for NDN-based IoT. An ICN architecture scheme was applied in a low-power IoT system for better energy consumption. As we know, NDN offers in-network caching, which is beneficial because if IoT devices are in sleep mode, then the content still remains available. CoCa is an IoT protocol. In this scheme, CoCa was implemented with NDN to enhance effective caching in the IoT environment. The authors evaluated the performance using the remote interactive optimization testbed. The results showed that energy consumption was reduced, and the availability of recently used content was almost 91%.

In [40], an NDN-based IoT technique known as SPICE-IT was implemented for controlling the pandemic situation. This scheme works on a pull-based mechanism. Individuals were monitored for body temperature and mask adherence. The authors evaluated the performance using NS3, and the results showed that the cache overflow was reduced, and network congestion was improved. A different popularity model can also be used to collaborate with the cooperative mechanism [41].

6.2 Freshness-Based Caching

Freshness-based caching means that the freshest or latest content that was required in the network is determined. This concept is vital in the field of IoT, as IoT nodes have limited storage space. Caching fresh content helps with eliminating the old content that is rarely needed in the network. If the freshness of the content is not considered, then there is a chance of slow response times for the user and network overload. Freshness caching is a way of caching only the fresh content that circulates in the NDN-based IoT network. It also helps speed up the caching process because old content is evicted and space is made for fresh content in the nodes. Here, DATA packets should be cached by considering some value that differentiates the fresh content from the old content.

In [42], a freshness approach to an IoT-based scenario that works on the freshness caching mechanism was proposed by the authors. In an NDN-based IoT network, new information is consumed and produced by the consumer and data generator, respectively. Users mostly demand state-of-the-art information. Previously, a sequence number was assigned to the content, but this process was not very useful because when a new node entered the system, it did not know which sequence number was required. To address the old sequence number process issue, a new parameter was added named the latest parameter. It determines if a particular packet will remain in the network and for how long. If the latest parameter is not added, then aggregate nodes always forward the old packet. Two values are assigned to the latest parameter: low and high. Content that is new to the network is labeled as high, and content that is old in the network is labeled as low. Because different users demand different types of content, the user specifies the needed content. If the desired content is not available

in the CS due to a lack of freshness, it is considered a cache miss, and this query is transferred to the data originator. A new field (that determines the content freshness) is added to the INTEREST packet and a check (timestamp indicating that the content is currently cached) is added in the CS table. Old cached content is replaced by new content. The authors evaluated the performance using NS3. Nine producer nodes and four consumer nodes were placed. The simulation results showed that the cache hit and overall caching performance increased.

In [43], a caching scheme named least fresh first (LFF) was proposed by the authors. In this scheme, cached contents that are considered invalid based on the sensors' time forecasting are evicted first. It is used to enhance the freshness of data in the network. The least fresh content is selected if the cached content has not been modified in the cache since its last request. The main idea of this scenario is to predict the sensor event time to determine a rough estimate of the remaining life. Prediction of freshness depends on the behavior of the traffic in the IoT network. If data are flowing non-stop, then it is called continuous traffic. If data are flowing in a fixed time interval, then it is called periodic. If traffic is periodic, then after a certain amount of time, cached content is no longer fresh because new content is flowing into the network. Fresh content is calculated from this observation. The authors evaluated the performance of LFF using ccnSim. Many nodes were placed in the evaluation scenario. The simulation results found that the cache freshness percentage and overall retrieval time increased.

Content freshness can also be calculated by eliminating the content cache time from the total time and then dividing it by the total time [44]. In [45], an energy-efficient content caching scheme was proposed that works on the mechanism of content freshness. This scheme considers the energy issue via caching the content. They used a mechanism in which real-time user request rates increased, and the energy and content freshness was efficiently maintained. The authors evaluated the performance using ndnSim in which they used nodes with a memory of about 50 to 100 MB. The results showed that energy consumption was reasonably reduced, and response time increased.

In [46], caching considering content freshness and energy was described by the authors. In their scheme, a new field was added that manages all cached content entries, maintains fresh data, and removes the old data. They evaluated the performance using ndnSim. Performance was measured using different parameters such as the quantity of nodes, mobility, and simulation time. The results showed that data freshness increased, and caching efficiency improved. More than one distinct caching strategy can also be combined (i.e., popularity and freshness) for efficient caching [47].

6.3 Popularity-Based Caching

Content popularity directly affects caching efficiency. If some nodes frequently demand the same data, then this results in less delay time, reduced load on the network, and faster access time. NDN-based WSN nodes calculate the total amount of times a piece of content is requested by the consumer. When the client wants a piece of content, IoT nodes first check their cache to see whether that particular content is available in the CS table. The total number of requests for specific content is also recorded by the NDN-based IoT nodes. If content is not available in the CS, then they request that content from the data producer and cache the content for future use. If the count for specific content increases above the threshold value, then this content is labeled as popular content.

Due to advancements in the IoT, NDN is considered the only future network paradigm that efficiently maintains all the services of the IoT. Therefore, in [48], a scheme called periodic caching (PC) was proposed by the authors for an NDN-based smart city. Smart city is one of the key fields in the IoT. It is the ideal scenario for almost all fields. PC is a flexible strategy for efficient caching in NDN-based IoT networks. Each node in the smart city contains a separate table in which interest for

each request is calculated. This table includes multiple fields such as data name, threshold, frequency count, and recent accessed time. The most used content is calculated with the help of this table. The threshold algorithm calculates the maximum value for the threshold. This indicates which content is used the most. For each specific piece of content, if the interest touches the threshold value, then this content is marked as the most-used content. The most-used content is cached at the edge routers. If the cache space in the edge routers is full, then some data packets are transferred to the central node. All the new INTEREST packets pass through these central nodes. Furthermore, if requested data are found at these central nodes, then the request is fulfilled there, and the INTEREST packet is not sent to the edge routers. LRU policy is used for content eviction. Desired content is placed near the user in the edge router, which results in less data retrieval time and a shorter path length. However, the cache hit ratio increases because now all the INTEREST packets pass through the central node. The authors evaluated the PC performance using SocialCCN simulator. A user load of 4000 was put on the PC, and the tree topology was implemented. Different cache sizes of 90 and 999 were used. The results showed that the cache hit ratio increased, and data retrieval time decreased. Popularity and freshness have also been combined for better caching in NDN-based IoT devices [49].

In [50], a popularity-based caching scheme was proposed by the authors. This collaborative strategy caches the content by considering the node degree and its distance. The authors evaluated the performance using CCNx Contiki and ran this simulation nine times on different parameters. The results showed that the cache hit ratio and energy consumption were improved. Caching for heterogeneous content in the IoT can also be possible by adopting one or more caching schemes together in a single framework [51].

6.4 Hybrid Caching

In [38], a cooperative scheme that integrates a data freshness mechanism for caching in NDN-based IoT was proposed by the authors. This scheme uses data freshness and cooperative mechanisms. They named this scheme lifetime cooperative caching. The scheme uses the request rate and lifetime of the IoT content. They also proposed a threshold level for caching that automatically adjusts different data rates under different conditions. They evaluated this scheme by developing a simulator in C++. The results showed that data retrieval delay and energy consumption were improved.

In [41], a cooperative caching scheme that combines content popularity in NDN-based IoT was proposed. This scheme works on machine-to-machine NDN-based IoT networks. The goal was to improve the ICN-IoT domain. The lifetime of WSN data was considered for efficient retrieval and nodes' battery consumption. Performance was evaluated using the TOSSIM simulator, and the results showed that the cache hit ratio increased and energy consumption and retrieval delay were improved.

In [44], a freshness and probability-based caching scheme named probability-based IoT in-network caching was proposed by the authors. Their model caches the content on the basis of content freshness and then calculates its probability using different parameters. Content is cached by eliminating the less probable content. The authors evaluated the performance for 399 routers using MATLAB. The results showed that cost reduction and network load were improved.

In [47], edge caching by considering freshness and popularity in NDN-based IoT was proposed by the authors. This scheme works under different scenarios depending upon the requirement. Two different caching strategies were combined: autonomous and coordinated. Performance was evaluated using the GEANT topology. The results showed that cache hit ratio increased.

In [49], a scheme named caching fresh and popular content (CFPC) was proposed by the authors. In this scheme, the authors proposed an NDN-based IoT that caches the data by calculating the

content popularity and lifetime in the network. The main aim was to cache the most popular data by considering the maximum lifetime of the cached content. The decision about content popularity was based on the NDN-based IoT routers' content forwarding. After a fixed interval of time, the router calculates the total number of received requests and each distinct content request. Popularity is measured by dividing each distinct content request by the total number of requests received. Then, some exponential weightage is added for more accurate calculation. After a fixed amount of time, these calculations are reset to zero, and a new popularity is defined. If the content is popular, then it is cached; otherwise, it is dropped. The authors evaluated the performance of CFPC using ndnSim simulator for 10 NDN routers with a binary tree topology. The results showed that the cache hit ratio increased, and data retrieval time decreased.

In [51], an edge caching-based heterogeneous scheme in the IoT was proposed by the authors. This scheme uses artificial edge caching for heterogeneous IoT applications that mostly considers the content's popularity and probability. This scheme uses edge clustering and edge caching processes that count and cache the popular content. The probability of specific content is predicted via collaborative filtering. Content with the highest probability is stored at the edge nodes. The authors evaluated the performance using the Icarus simulator. The results showed that the cache hit ratio increased and average hop count and content retrieval time decreased.

In [52], a probabilistic caching strategy for the IoT (pCASTING) was proposed by the authors. It is a probability-based caching technique that uses the data freshness concept for caching content. Probability-based caching in NDN-based IoT means routers cache the content by calculating the probability that a particular request for content will be generated. It also includes the probability of whether the demanded content will be available in the cache store or not. Here, different node features, the lifetime of the content, storage space, and energy are considered. With the help of these parameters, each IoT node's probability is measured. This is a distributed scheme. Many IoT nodes that resource constrained are connected with each other via multi-hop WSN (e.g., sensors in streets or buildings). The data generator generates state-of-the-art content. The remaining nodes work as content forwarders that cache the content. Each forwarder node focuses on the attributes of the content and device and compare these with the other nodes. By considering these parameters, the caching probability is measured via the utility function and is assigned a value of 0 or 1. In this way, probability is measured. The authors evaluated the performance of pCASTING using the ndnSim simulator for 59 NDN nodes. The results showed that energy consumption and data retrieval time improved.

In [53], the efficient popularity-aware probabilistic caching scheme was proposed by the authors. This scheme caches the content that is requested more often, shortlisting and forwarding it in the probability process. Here, each node independently maintains the tables in which they track the most requested content. The probability of the most requested contents is measured, and then they are cached in NDN-based IoT nodes. The authors evaluated the performance using the Icarus simulator and found that network resources and content retrieval time improved.

6.5 Multifarious Caching

In [54], a technique called poolcache for caching in the IoT was described. This technique evaluates the node's storage before caching and combines it with other nodes' storage to act as one main storage. It manages the main content storage that is shared between all the neighboring nodes. It also uses the hash functions. The authors evaluated the performance using the transit-stub topology. The results showed that overall performance was improved.

In [55], a smart health mechanism using NDN-IoT was elaborated. The old system used a traditional IP network, which resulted in mobility, security, and privacy issues. Therefore, there was a need to implement NDN with the IoT in the smart healthcare sector to overcome these problems because NDN provides built-in support for mobility. The main goal of this technique was to process, collect, analyze, and forward patient health data to hospitals with the help of NDN naming. As shown in Fig. 10, a sensor layer was used to collect patient data. Various sensors, such as temperature, glucometer, and position sensors, are placed on the patient's body. The NDN gateway layer collects and forwards patient data to the next layer using the NDN mechanism. The data storage and analyzing layer stores, analyzes, and visualizes the patient data. All patient queries and data are maintained here. In the main handler layer, NDN uses different reliable algorithms for discovering the hospital. The main task of the handler is to obtain the network topology, content location, and content service on demand. Other caching aspects [56] and other details [57] of NDN in the IoT have been elaborated.
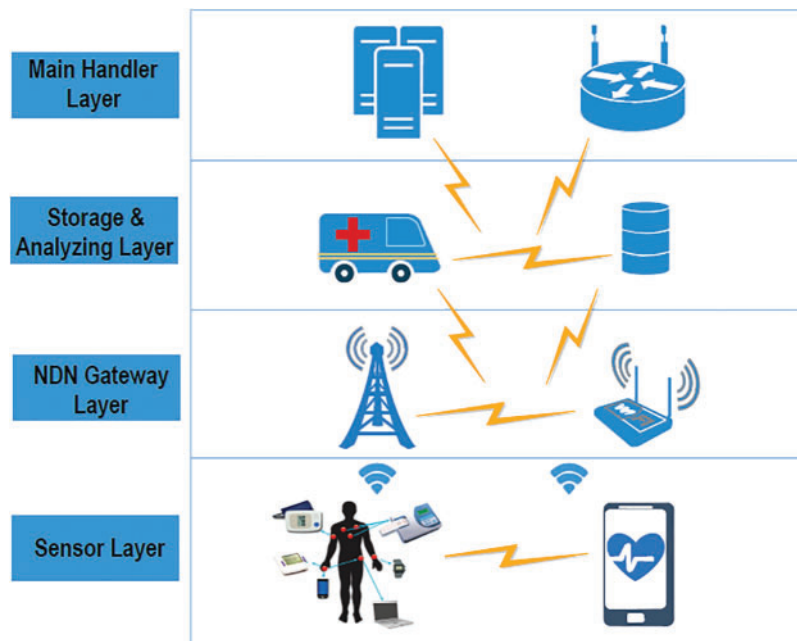

Figure 10: Working mechanism of smart-health using NDN-IoT

7 Caching in NDN-Based VANETs

VANETs is a group of vehicles that can communicate with each other without having any central controller and base station. MANETs changed the lifestyle of humans by allowing users to exchange information anywhere at any time. This led the researchers to think about a new communication paradigm. The new paradigm of communication is among different vehicles named as VANETs. This helps in driver assistance, sensing road conditions, calculating traffic load on the road, and safety. VANETs are responsible for passing message packets from one moving vehicle to another. A vehicle can directly exchange information with another vehicle this can be named as one vehicle to another vehicle while a vehicle can exchange information with electric units that are placed at the side of roads named as vehicle to infrastructure communication. Some of the important applications of VANETs are listed below:

1. Global Positioning System can be used in VANET to see your current location.
2. VANETs can be helpful in road safety so that they capture road signs or electric units.
3. VANETs provides Comfort. For example, inform the user about electronic payment of tolls.
4. VANETs provides Predictable Patterns of Mobility.

All these above applications are important in our daily life [58]. By using traditional IP networks in VANETs, many limitations arise such as huge network weak scalability, inefficient addressing in huge mobility areas, and poor efficiency. Implementing NDN in VANETs reflects many benefits. Fig. 11 shows the working mechanism of NDN based VANETs. In Fig. 11, V1 sends an INTEREST packet towards the V2 which then forwards that INTEREST towards V3, V4, and V6, respectively. V6 requests that INTEREST packet to roadside electric unit. The electric unit sends that DATA packet towards the V6, which then forwardsthe V4, V3, V2, and V1, respectively. After getting the particular DATA packet, this packet is cached at any one vehicle that forwards the specific INTEREST packet. If any of these vehicle demands that DATA packet, then this request can be fulfilled by these vehicles. Working Mechanism of NDN based VANETs is listed below:

- Packet Structure: Almost same as it is in NDN. If VANETs in NDN network wants to fetch data then it sends an INTEREST packet towards the data generator. DATA packet and INTEREST packet remains the same. Just minor changes are added such as new record (Vehicle Status) is added in the INTEREST packet of VANETs. Content popularity record is added into the DATA packet, to enhance cache hit ratio. In the case of mobility, INTEREST packet and DATA packet are modified and velocity and location of consumer is added in the INTEREST packet. By doing this, if node is moving from one region to another then suitable forwarding decision is done by the routers.
- Tables: Three tables used in NDN based VANETs environment. Same as NDN CS table, PIT, and FIB.
- Nodes functionalities: Three functions are done by nodes. Data forwarder, Data consumer, and Data Producer. The data forwarder sends the both INTEREST packet and DATA packet. Data Producer may be the Side Electric Unit (SEU) or Simple Side Unit (SSU). DATA packet is generated by these units. They also tell other nodes that what they produced or cached. Data Consumer sends the INTEREST packet and requests for particular data.
- Working of Nodes for INTEREST packet: After receiving the INTEREST packet, SEU or SSU at the receiver side first search in its own CS table. If found here, then SEU or SSU sends the DATA packet towards the node that is nearest to the SEU or SSU. If not found in CS table, then it checks the PIT. Here, the same procedure of NDN is followed.
- Working of Nodes for DATA packet: SEU or SSU check the entry for received content in their PIT. If entry is found, then forwards the DATA packet towards the requested node. If entry is not found then DATA packet is discarded [59].

Different caching strategies in NDN based VANETs are described in Fig. 12. We divide different caching strategies for NDN based VANETs into many branches. They are:

1. Cooperative Caching
2. Popularity based Caching
3. Probability based Caching
4. Proactive based Caching
5. Optimization based Caching
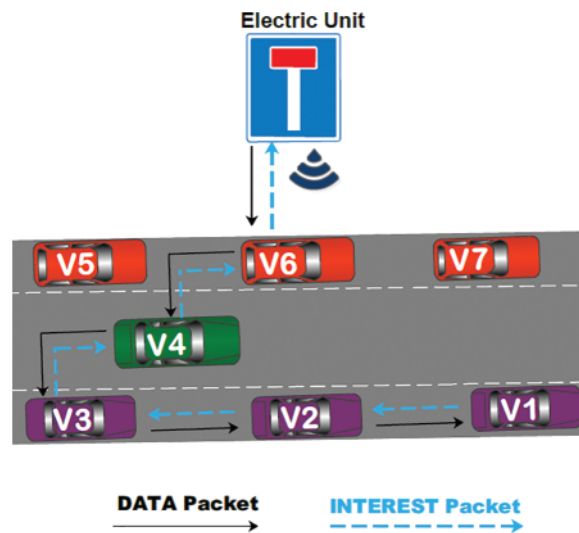6. Hybrid Caching
7. Multifarious Caching
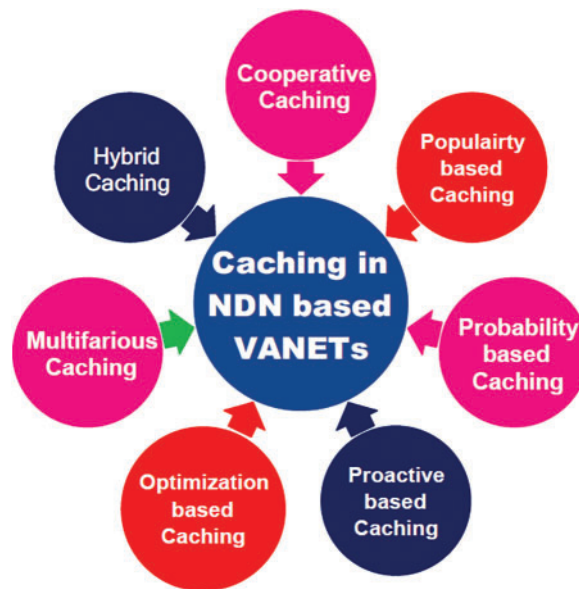
Figure 11: VANETs communication scenario

Figure 12: Caching strategies in NDN based VANETs

7.1 Cooperative Caching

In NDN based VANETs Cooperative Caching technique is a prominent caching technique. In VANETs, higher utilization of resources is achieved using Cooperative caching. Better content delivery is also achieved using this scheme. Cooperative caching demands different recent information to be swapped between different resource constraint wireless nodes. Cooperative caching turns individual node to work in a form of one big global cache. Many resource and other restrictions can be vanished by implementing cooperative caching. It restricts different VANETs nodes to cache the similar data in

a specific area. This helps in enhance the efficient resource utilization, improved cache hit ratio, and better content retrieval time [59].

In [60], a cooperative scheme that uses popular content for caching named Collaborative Caching Based on Socialized Relations (CCBSR) for caching in NDN based VANETs are proposed by the authors. This technique selects the suitable node for caching. Caching point is decided in order to overcome the mobility issue of nodes. Highway is considered where vehicles are moving. The driver wants to download some multimedia from smartphone. Multimedia contents are considered as a series of content. A unique name is assigned to every packet of content. An INTEREST packet is sent to the nearby vehicles for the multimedia content. If data is not available in the vehicle then it sends the same INTEREST packet towards the next vehicle and this process continues until corresponding DATA packet is not found. When the corresponding DATA packet is found, then it sends backwards the nodes along the path. At the forwarding nodes data can be cached. Request for data is done using content name rather than IP. Appropriate node for Caching is selected via the concept of graph theory. Minimum vertex problem is used in VANETs to cover the complete NDN based VANETs network. By doing this, minimum nodes are required to cache the content. So that, data is accessed using one hop distance. When the new vehicle enters the region it is connected to the topology, and it becomes one of the nodes of the minimum vertex topology. Performance is evaluated using NS3 simulator. More than 99 vehicles are placed and they maintain a speed of 5 kilometers. Simulation results in [60], show that content retrieval time improved. In NDN based VANETs in order to match social parameters cooperative scheme uses factors of popularity-based caching scheme [61]. Cooperative schemes sometimes used cluster-based caching mechanism [62] in NDN based VANETs. Inter and Intra cluster caching is done via cooperative caching [63]. Different issues in VANETs such as VANETs velocity, on-road traffic, and data access patterns of vehicles can also be analyzed via caching [64]. In [65], authors proposed a cooperative caching strategy named as layered cooperative cache management for vehicular content-centric network (LCC-VCCN) that choose adjacent nodes that lies in the broadcast range. Layer level caching is done here. Maximum demanded content layer is pushed towards the user. All this is done be cooperative caching. Performance is evaluated using NS3 simulator. Results show that content retrieval time and video streaming are improved. Partial matching that estimates the VANETs mobility in different regions by calculating its old mobility history is also a technique [66]. In [67], authors proposed a caching strategy based on cooperative caching. Deep reinforcement and asynchronous learning are used that is based on vehicular edge computing. Content is predicted and location for cooperative caching is obtained. First asynchronous federated learning is used that finds a global model. Dueling deep Q-network (DQN) algorithm helps in minimizing the content delay. Their proposed model captures multiple features and learns from vehicles local data. Model training time is reduced in every single round. Model accuracy is improved with the help of vehicle characteristics such as vehicle velocity, position, and mobility. Performance is evaluated using Python 3.8. Results show that proposed scheme outperforms other baseline caching schemes.

7.2 Popularity Based Caching

Here, content is cached by calculating the number of times a particular content is accessed by the user. Content that has more demand will be cached by more nodes. It evaluates the caching process. It is basically a signal for the most demanded content. It is widely used in VANETs because content is demanded so nodes in VANETs stores the most popular content in their storage to reduce the content retrieval time and to improve cache hit ratio [59]. A popularity based caching scheme is proposed by the authors named Node Value and Content Popularity (NVCP) in [68]. It is an improved technique for caching the contents on the basis of popularity in NDN based VANETs. NVCP first finds whether the

cache is local or global. Because it affects the performance. A routing protocol is used for evaluating the shortest path. It is done in term of graphs where nodes represent the VANETs and their link defines connectivity. If a node is connecting with many nodes, then there are more chances of cache hit. If a router that has content is placed between different routers with smallest path between them is considered useful. Because this router then controls most of the packets. Nodes centrality can also be found using Eigenvector formula. Popularity is calculated by dividing the count for requesting specific content by maximum count. Content with greater value is considered as popular. Performance is evaluated using NS3 simulator. More than 45 nodes and more than 145 links are used. Results show in [68], transmission latency, cache hit ratio, and hop count improved. Content popularity and freshness may be used in a single caching VANETs mechanism [69].

In [70], authors proposed a cache management technique that uses service quality as a parameter. They divide the traffic into distinct classes. Cache storage is also divided into further classes. Each class has different requirements. A density level of popularity is added to the contents. Content that has the lowest level of popularity will be replaced first. On the other hand, contents with the highest popularity level will be at the top of the selection. Performance is evaluated using Python Language. Results show that the quality of network improved.

In [71], a caching scheme named as Popularity-based Content Caching (PopCC) are proposed by the authors. The hidden Markov Model is used to estimate the content popularity. Here, popularity is based on the interest that are received such as content request ratio, priority, and frequency. Performance is evaluated using ONE simulator. Results show that overall success ratio, and access delay improved.

In [72], authors propose a caching mechanism that uses heterogeneous information network (HIN) and that caches the popular content in the caching storage. They made the network heterogeneous. By doing this, load from network is reduced and user can cache the popular content in real time. Results show that quality and user experience of overall network is improved. Caching storage problem for caching Road Side Unit (RSU) for VANETs can be also solved by using the popularity based caching mechanism [73]. Cache duplication in VANETs can reduced by using data popularity with hop numbers [74]. By using bloom filter mechanism for cache in VANETs, overall performance increase [75]. Cache insertion policy can increase the better resource utilization, and better network performance [76].

In [77], a caching strategy named Distributed Caching Strategy (DCS) is proposed by authors. This scheme caches the content on the basis of popularity. Here, content popularity is measured by calculating the total incoming request for specific content. It is then compared with the threshold level for detection of most popular content. Popular content cached in the form of chunks. It helps lessen the congestion and storage. Popular content chuck is cached at router that has the best closeness centrality. It will then transfer towards the edge router. Data dissemination is reduced by implementing distributed strategy for caching that works on content popularity. Performance is evaluated using SocialCCNsim simulator. Results shows that memory, cache hit ratio, and eviction ratio improved.

7.3 Probability Based Caching

In probability based caching, VANETs caches the data which has probability defined earlier when they received a content. Data is stored in already defined probability [59]. Here nodes cache the content by calculating their probability. Probability is a kind of prediction of a content request. Probability is calculated by using different parameters. The probability of any content can be calculated using different formulas and its desired values may be different for different schemes. If the probability value

is equal to our desired value then probability of content is achieved and this content may be cached in the CS now.

In [78], a scheme named Distributed Probabilistic Caching (DPC) is proposed by authors. It is a strategy in which individual node take their caching decision independently. By considering the total entries of interest caching decision takes place. DPC consists of three main factors. First is user demand for particular content. Second is node importance, and third is VANET movement. Probability of content is measured via probability of user (i.e., capture interest is analyzed), VANET importance (i.e., find vehicles centrality that defines its importance), VANET mobility (i.e., mobility is considered for caching decision), and cache calculation (i.e., probability of cache is measured by using VANET importance, and mobility). Performance is measured using NS3 and SUMO simulator. Many different files are provided by the producer. Results in [78] show that greater cache hit ratio, and low data duplication is achieved.

In [79], Extended CCN Vehicle Plus (ECV+) scheme proposed by authors. ECV+ is a caching strategy that caches the content via channel usage probability in NDN based VANETs. ECV is a kind of preliminary model for EVC+. This strategy works on the probabilistic caching. The main work of ECV is to reduce the cache miss. In ECV, forwarding and replying of a packet is categorize into two different parts. Replying of packet has greater priority. Two slots are assigned to each vehicle. One for INTEREST and the second for DATA packet. DATA packet slots are assigned first because this slot has to reply as fast as possible. If cache probability is small then there is a chance of cache miss. Therefore, it is necessary for the ECV+ to adjust probability of cache by the usage rate of channel. Performance is evaluated using Scenargie simulator. Results in [79] show that the number of hops are reduced, and content access rate improved.

7.4 Proactive Based Caching

In proactive based caching, decision about cache is taken before request of the content. Caching is based on driving trajectory, and historical request. Proxy based caching is a caching technique in which proactive caching is used. Here, data is placed on the recommended nodes. Data is placed even before the request of that data [59]. Proactive caching helps lessen the latency. Proactive caching is also used VANETs that help in lessening many different problems. The probability of content is measured for caching the content via proactive caching where nodes can anywhere place the data [80]. Chunks probability can also be predicted via proactive caching [81].

In [82], proactive based caching named as PeRCeIVE are proposed by the authors. It proactively caches the content within ICN based VANETs. It also facilitates caching based on the RSU. Performance is evaluated using NS3 simulator. The result shows that INTEREST, and one-hop ratio improved. Limited capacity and huge deploying cost of RSU, only limited number of VANETs access the cached content which results in low cache hit ratio. To overcome this issue, a proactive caching strategy that integrate probability should exist [83].

7.5 Optimization Based Caching

In optimization based caching in NDN based VANETS, selection of content is presented as an optimization problem. Here, at different scenarios for different application mostly distinct objective function is implemented for particular application. For example, optimization for transmission efficiency that may be used for entertainment and optimization of latency for messages in emergency situations.

In [84], a caching scheme is proposed by the authors. This scheme caches the data in different scenarios such as Vehicle-to-Infrastructure. Here, along the road multiple access points are installed that give data to moving VANETs. They tackle the content placement problem as a linear integer. Different optimization targets are considered. Such as the minimum size of the cache, and maximum content retrievable time, if link and storage capacity are limited. Performance is evaluated using ndnSIM simulator. Results show that the network is improved. Model for network availability, and 2-D Markov process can also be used for different optimization and probability based VANETs caching [85].

7.6 Hybrid Caching

In [61], a technique named as Cooperative Caching scheme based on Social Attributes and Mobility Prediction (CCSAMP) is proposed by authors. It combines content popularity with its mechanism. This technique observes the VANETs contact with each other. If some similarity between drivers social or other interest exists then these vehicles contact each other. If a caching node shares social attributes with the requester it means that the requester wants more this type of content. Here, we have to measure social resemblance, centrality, and predict the next route. Social resemblance is measured by comparing requesting node and forwarding node. Centrality is measured by seeing the connectivity of vehicle with other nodes. The next route is predicted via old pattern of mobility. Each node adds its social features in the INTEREST packet, and social feature from the provider node is also added by the provider node for the corresponding node in the DATA packet. The route record of every vehicle is uploaded to the nearest SSU or SEU. All the records are synchronized by the SEU. SEU determines the pressure zone where most vehicles intersect. Social attributes are sent by vehicles to their adjacent vehicles. These vehicles then update their social records. When INTEREST packet has arrived, vehicle search its CS table for cache hit. Same NDN forwarding process is used here. When a DATA packet is received social similarity is calculated via social attribute of source node. Node centrality is measured via old social records. Path prediction is measured by Hidden Markov Model. It determines the pressure zone. Greater the social similarity, greater the caching. SEU counts the vehicles that are in range. It then broadcasts vehicles pressure zone list with other SEU and vehicles. If storage is limited then content with greater popularity is cached. If greater number of social attributes are shared by the forwarding node with the requester than there is a great chance of more similar caching between these needs. A node which has higher number of links with other vehicles are labeled as central vehicles. Future path prediction is used for maximum caching. The performance of CCSAMP is evaluated using ONE simulator. Simulation results in [61] show that cache hit ratio increases, and access delay is decreased.

In [62], a scheme that uses both cooperative and popularity parameters is used in cluster-based caching mechanisms for NDN based VANETs. The scheme is known as Cluster-based Selective Cooperative Caching (CSCoC). This scheme caches the popular data for the VANETs. These VANETs made a proper stable communication mechanism. Mobility pattern of most vehicles are observed and by using method of cluster division different vehicles are selected for cluster head that works as caching nodes. Cooperative caching mechanism is used here to lessen the mobility issue that enable a local cluster member to extract data from main head. Performance is measured via NS3 simulator. Results show that cache hit ratio, hop count, and access delay is improved.

In [63], another cooperative and popularity scheme that uses cluster-based mechanism for VANETs are elaborated by the authors named as cluster-based cooperative caching approach with mobility prediction (COMP). They use the idea of develop a communication mechanism between different vehicles that almost have to move with each other. This lessens the mobility issue. Because

VANETs nodes with the same mobility pattern have reliable connection. Inter and Intra cluster caching is done via cooperative caching. Performance is measured via SUMO, and ndnSIM. Results show that network performance, and data delivery is improved. In [64], cooperative and popularity based edge caching scheme in VANETs are proposed by the authors. They develop a model that caches the popular content and decides when and where to get the copy of data when an on-going vehicle demands it. Performance is evaluated using MATLAB. Results show that overall performance is improved.

In [67], probabilistic and cooperative based caching named as ICN-based Cooperative Caching solution (ICoC) in NDN based VANETs are described by authors. They use a traffic model. Traffic is bi-directional. VANETs change lanes by using probability. Caching is used to improve experience quality. Performance is evaluated using NS3 simulator. Results show that delay is improved. In [69], a caching scheme called Multi-Metric Cache Replacement (MMCR) is proposed by authors. In this scheme, authors explore the NDN based VANETs and note the storage issue in VANTEs nodes. This scheme uses popularity and freshness factors for content caching. They realized that data lifetime, and CS size are the main issue that effects on the overall performance. They focus on popular content that needs to be replaced in the CS when the replacement occurs. This scheme focuses on three parameters for decision of which content is placed in the CS. The first parameter is freshness. Freshness is measured by how many times a particular content resides in the cache. The second parameter is popularity. It is measured by how many times content is demanded. The third parameter is the distance between the node at the start when it caches the content and its latest location. When content is cached in the node, it is updated after some interval. CS status is updated after fixed time. It results in updating of CS information such as distance between cache node at start and now, total cache hits, and content freshness level. To select the appropriate content for replacement, cache hit frequency, and freshness for particular content are calculated. After this distance of cache node is measured. Popularity is the main factor. If content is not fresh but it is popular than it will not be replaced. Two methods were further proposed by the authors for appropriate content selection. In this first method, they added a lifetime to a content. In the second method, they use distance, and freshness parameter. Performance is evaluated using NS3 and SUMO simulator. More than 100 vehicles are used. Results in [69] show that cache hit ratio improved.

In [73], authors proposed a caching scheme in NDN based VANETs. It uses a policy for cache storage that is based on popularity. Whereas, VANETs nodes caches the data packet by using probabilistic caching. Using the decision tree data is cached. This tree is then compared with probabilistic scheme. Performance is evaluated using SUMO simulator. Results show that QoS improved using this caching strategy. In [74], a popularity and probability based scheme proposed by authors. Their policy is called Community Similarity and Population based Cache Policy (CSPC). Probability based scheme is used here for evaluating and designing the different vehicles' similarity report. To lessen cache redundancy, hop number based vehicle is selected that has popular content cached. Performance is evaluated using OMNeT++. Results show that cache hit ratio, time delay, and minimum cache hit distance improved. In [75], probability and popularity based caching schemes are proposed by authors. Caching scheme based on Bloom filter is used for better cache management. This mechanism uses the cache functionalities for caching in VANETs to enable data transmission in cooperative manner. Frequency is used for content popularity calculation. This scheme is like of data structure type of probabilistic caching. Performance is evaluated using ndnSim. Results show that content retrieval time improved.

In [76], caching scheme named as Utility Gradient (UG) cache are proposed by authors. UG-Cache uses popularity and probability factors for caching the content in the network. Cache placement decision are based on request frequency from the sender side and the total distance between them.

Popularity is measured and popular content becomes more available. Cached item probability varies from maximum to minimum. Performance is evaluated using SUMO, OMNet++, and Veins. Results show that cache hit ratio, and on-demand caching improved. In [80], a caching strategy are proposed by the authors in [80]. This approach consists of proactive caching and combines probability for caching the content in VANETs nodes. Using proactive caching, caching can take place anywhere in the whole network. The authors use the Markov predictor that helps in forecasting the VANETs RSU. Entropy metric is selected for balancing the uncertainty that may occur in mobility prediction. In last, node that prefetches the data efficiently is selected. Performance is evaluated using ndnSIM. Results shows that user that have lower latency demand will fit in proactive caching.

In [81], another an optimized proactive based caching technique named as Proactive based Caching-based Mobility Prediction (PCMP) are proposed by authors. In PCMP, it predicts the chunks probability. In proactive way, from one hop it will be fetched. Perfect placement position also considered. Performance is evaluated using NS3 simulator. Cache utilization, network delay, and hop reduction improved. In [83], authors proposed a caching strategy for RSUs that combines proactive and probability. Here, content is distributed according to the VANETs movements. Performance is evaluated using NS3 simulator. Results show that delay, content availability and cache hit ratio improve. Authors in [85], proposed a caching scheme that uses the concept of optimization and probability in VANETs. They use a model for network availability. 2-D Markov process is used for VANETs interaction. Online caching mechanism is designed that enhance the overall network energy efficiency. Caching decisions treated as an optimization model. Performance is evaluated using MATLAB. Results show that system gain, cache hit ratio, and utilization is improved.

7.7 Multifarious Caching

Caching based on in-network using TCP scheme in VANETs is proposed in [86]. It tries to overcome low content delivery in VANETs. In [87], a Blockchain mechanism is implemented in NDN based VANETs for securing the caching process. It results in more trust between user vehicle and cache stores. In [88], different caching techniques are compared and a new one is proposed by authors. Their caching scheme caches the content in selected node. It reduces content duplication. An algorithm for caching in parking area for VAENTs are proposed by the authors in [89]. To lessen the pollution attacks on the cache, a mechanism is elaborated in [90]. Here, each INTEREST packet is completely monitored to identify the valid packet. Another in-network caching scheme that uses traffic characteristics for caching in VANETS are described in [91]. For efficient caching is VANETs using position based clusters, a mechanism is proposed in [92]. Caching is done via leader cluster head based on its position. Transient data dissemination and caching effects in NDN based VANETs is considered in [93]. To solve the complex issues in VANETs a hierarchal cluster-based caching scheme is recognized in [94]. They minimize packet loss, and delay issues that are caused by mobility. Here, data packets are broadcast by the routers that are get by consumers. To lessen the network delay a new caching scheme is described in [95]. Continuously changing environments and variable network are a challenge in VANETs. There proposed scheme uses vehicle attributes, motion, and characteristics. These factors help in calculating caching probability. Content caching diversity could be increased with the help of multiple content parameters [96].

8 Challenges and Future Directions

Although a huge amount of research is considered in NDN based Wireless Ad Hoc Networks, but still there are many things that need to be improved. Due to diverse nature of MANETs, IoT, and

VANETs many issues are still untouched. This section describes the challenges and future direction in this field. Some of the caching key challenges in this domain are listed below:

- Minimum Cache Decision Time: As wireless ad hoc networks move fast from one place to another which demands minimum cache decision time. Overall caching decision time in these networks is not minimal.
- Minimum Cache Capacity: As we know, almost all wireless ad hoc networks have limited storage capacity. This limitation results in difficulty in cache placement and replacement policies.
- Performance in Dense Network: Let consider the scenario of traffic jam. Most vehicles may be in range of other vehicles in traffic jam. Here in dense network, performance of cache nodes are limited because so many devices connect or disconnect with the nodes [59].
- Low Utilization Rate: In the case of hot region, where many nodes are in the same hot region, the performance of these nodes is not efficient because too much connection may result in poor connectivity, data collision, and poor utilization rate.
- Topology Changes: Nodes are connected with other nodes and adapting different network topologies. When node is moving from one region to another than topology of that node changes. These nodes have to adapt to new topologies continuously. Joining of new topology may arises the mobility issue.
- Security: Multiple security aspects such as various attacks (i.e., DDoS attacks [97], Content Poisoning Attacks, and replay attacks) may result in poor performance of VANETs [98]. Due to mobility, exchange of security messages suffers delay. For time-critical applications this is not a good approach. Decentralized mechanism for authenticity is missing in VANETs [99].
- Broadcast Storm: NDN based vehicles that work as mobile nodes broadcast the packets in their range [100]. This broadcast mechanism may increase congestion, packet loss, and network load.

These caching key challenges needs to be addressed for fast, better and efficient caching in NDN based wireless ad hoc network. Using cooperative and other caching techniques in NDN based MANETs, IoT, and VANETs enhanced the overall performance, but there are still many methods to enhanced caching in NDN based wireless ad hoc network. Cloud computing based on NDN refers to services and resources on cloud that solely depends on NDN principles for data retrieval and transmission. Cloud computing enhances the overall functionality of wireless networks and improves efficiency, and scalability by providing resources in centralized manner [101]. It also provides several services remotely. Collaborative caching in NDN based cloud computing networks enables coordination and cooperation among cloud nodes. By doing this overall network efficiency, cache hit rates, and data redundancy could also be improved. State-of-the-art applications of cloud computing such as virtual reality, multimedia streaming, and augmented reality should be considered for caching strategies. Context-aware caching schemes should also be explored that enables device capabilities, user preferences and enhance content delivery, and cache management. Here below some important future directions are listed:

- Overcome Congestion in Hot Region: In hot region to overcome network congestion, INTEREST requests should be filtered and important INTEREST packets should be transferred in the network to overcome network traffic.
- Use Other Domains: Wireless Ad hoc Networks are flexible and easily implemented with other domains. By implementing Artificial Intelligence (AI) in wireless ad hoc network we can get certain benefits. AI has ability to learn without doing so much work on it. Implementing AI in caching will enhance the network and it automatically adapts the new network topology. We

can also think of Cloud or Fog based architectures. This will result in caching at cloud nodes and better service to customers [59].

- Improvements in Factors: Nodes caching capacity should be increased. If not then content efficient caching policies should be adapted that cache only useful content. In IoT nodes should be in sleep mode if not working currently. This will help in better energy consumption.
- Caching Quality: Caching should be efficient and in timely manner. To achieve quality of service in caching we should implement quality aware protocols on NDN routers.

In our experience, implementing caching in NDN-based MANETs, Wireless, Ad-Hoc networks, and IoT have multiple opportunities, and challenges. Scalability and dynamic network conditions emerge as a primary concern that limits the growth of these domains. Addressing these challenges requires better edge computing resources that adopts better caching strategies that automatically adopt user behavior, and real-time changes. Lightweight caching schemes are also needed due to the factor of resource constraints. Vulnerabilities must be resolved with the help of encryption, and privacy-preserving techniques. Tailoring caching strategies are the requirements of different IoT applications and are crucial towards advancing research and standardizing caching mechanisms in NDN-based networks. Researchers who works on this domain should follow the above considerations in order to keep their NDN environment better and efficient.

9 Conclusion

Wireless Ad Hoc Networks are commonly used in today's life. Communication is slow in traditional IP-based networks because IP does not provide in-network caching. Therefore, there is a need for new network paradigm is necessary. NDN is a suitable futuristic network paradigm that when implemented in a wireless ad hoc network results in better security, caching, mobility, and network performance. In this survey paper, we did a detailed survey of Caching Strategies in NDN-based Wireless Ad Hoc Networks. We discussed the NDN working, caching in NDN-based wireless ad hoc networks, caching strategies in NDN-based wireless ad hoc networks that are categorized into three domains (MANETs, IoT, and VANETs), challenges, and future research directions. Comparison between IP and NDN is also elaborated which helps in the better overview of NDN. For a better understanding of the caching process, different caching strategies are explained in detail. Results are also described which tell us which strategy improves the cache hit ratio, efficiency, latency, and network performance. In the last, we have provided some challenges and future directions for researchers.

This survey paper concludes on the fact that NDN-based Wireless Ad Hoc Networks such as MANETs, IoT, and VANETs are still in their inception. It is visualized that implementing NDN brings new levels of user experience and better service quality. This survey paper attempts to explore the NDN-based MANETs, IoT, and VANETs in terms of their caching capability in detail. I hope that my exploration of caching on NDN-based wireless ad hoc network here may motivate stakeholders and researchers to elevate efforts in this encouraging field to enhance better performance in terms of caching.

Author Contributions: The authors confirm contribution to the paper as follows: study conception and design: Ahmed Khalid, Rana Asif Rehman; data collection: Ahmed Khalid; analysis and interpretation of results: Ahmed Khalid, Rana Asif Rehman; draft manuscript preparation and review: Ahmed Khalid, Rana Asif Rehman, Byung-Seo Kim. All authors reviewed the results and approved the final version of the manuscript.

Availability of Data and Materials: This article does not involve data availability and this section is not applicable.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1]   "Chapter 4—Networking: Vision and packet switching 1959–1968 circuit switching," Accessed: Sept. 24, 2022. [Online]. Available: https://historyofcomputercommunications.info/

[2]   "When was the Internet invented? What to know about the creators of it and more," Accessed: Sept. 24, 2022. [Online]. Available: https://www.usatoday.com

[3]   "Advantages and disadvantages of TCP/IP," Accessed: Sept. 24, 2022. [Online]. Available: https://www.rfwireless-world.com/Terminology/Advantages-and-Disadvantages-of-TCP-IP.html

[4]   L. V. Yovita and N. R. Syambas, "Caching on named data network: A survey and future research," *Int. J. Electr. Comput. Eng.*, vol. 8, no. 6, pp. 4456, 2018. doi: 10.11591/ijece.v8i6.pp4456-4466.

[5]   A. V. Vasilakos, Z. Li, G. Simon, and W. You, "Information centric network: Research challenges and opportunities," *J. Netw. Comput. Appl.*, vol. 52, no. 7, pp. 1–10, 2015. doi: 10.1016/j.jnca.2015.02.001.

[6]   D. Saxena, V. Raychoudhury, N. Suri, C. Becker, and J. Cao, "Named data networking: A survey," *Comput. Sci. Rev.*, vol. 19, no. 7, pp. 15–55, 2016. doi: 10.1016/j.cosrev.2016.01.001.

[7]   "Wireless technology disadvantages of wireless networking," Accessed: Apr. 5, 2024. [Online]. Available: https://www.nibusinessinfo.co.uk/content/pros-and-cons-wireless-networking

[8]   A. Tariq, R. A. Rehman, and B. S. Kim, "Forwarding strategies in NDN-based wireless networks: A survey," *IEEE Commun. Surv. Tutorials*, vol. 22, no. 1, pp. 68–95, 2019. doi: 10.1109/COMST.2019.2935795.

[9]   M. Amadeo, C. Campolo, A. Molinaro, and N. Mitton, "Named data networking: A natural design for data collection in wireless sensor networks," presented at the IFIP Wireless Days, Valencia, Spain, Mar. 25–27, 2013, pp. 1–6.

[10]   B. Etefia and L. Zhang, "Named data networking for military communication systems," presented at the IEEE Aerospace Conf., Big Sky, Montana, USA, Mar. 3–10, 2012, pp. 1–7.

[11]   H. Al-Omaisi, E. A. Sundararajan, and N. F. Abdullah, "Towards VANET-NDN: A framework for an efficient data dissemination design scheme," presented at the 2019 Int. Conf. Electr. Eng. Inform. (ICEEI), Bangkok, Thailand, 2019, pp. 412–417.

[12]   E. Kalogeiton, T. Kolonko, and T. Braun, "A topology-oblivious routing protocol for NDN-VANETs," *Ann. Telecommun.*, vol. 73, no. 9, pp. 577–587, 2018. doi: 10.1007/s12243-018-0661-4.

[13]   S. P. Ardakani, C. F. Kwong, P. Kar, Q. Liu, and L. Li, "CNN: A cluster based named data routing for vehicular networks," *IEEE Access*, vol. 9, pp. 159036–159047, 2021. doi: 10.1109/ACCESS.2021.3131198.

[14]   N. R. Syambas, H. Tatimma, A. Mustafa, and F. Pratama, "Performance comparison of named data and IP-based network-case study on the indonesia higher education network," *J. Commun.*, vol. 13, no. 10, pp. 612–617, 2018. doi: 10.12720/jcm.13.10.612-617.

[15]   P. Gusev and J. Burke, "NDN-RTC: Real-time videoconferencing over named data networking," presented at the 2nd ACM Conf. Inf. Centr. Netw., San Francisco, CA, USA, 2015, pp. 117–126.

[16]   J. Samain *et al.*, "Dynamic adaptive video streaming: Towards a systematic comparison of ICN and TCP/IP," *IEEE Trans. Multimedia*, vol. 19, no. 10, pp. 2166–2181, 2017. doi: 10.1109/TMM.2017.2733340.

[17] I. Abdullahi, S. Arif, and S. Hassan, "Survey on caching approaches in information centric networking," *J. Netw. Comput. Appl.*, vol. 56, no. 7, pp. 48–59, 2015. doi: 10.1016/j.jnca.2015.06.011.

[18] M. Zhang, H. Luo, and H. Zhang, "A survey of caching mechanisms in information-centric networking," *IEEE Commun. Surv. Tutorials*, vol. 17, no. 3, pp. 1473–1499, 2015. doi: 10.1109/COMST.2015.2420097.

[19] R. Di Pietro, S. Guarino, N. V. Verde, and J. Domingo-Ferrer, "Security in wireless ad-hoc networks–a survey," *Comput. Commun.*, vol. 51, no. 3, pp. 1–20, 2014. doi: 10.1016/j.comcom.2014.06.003.

[20] Y. W. Ting and Y. K. Chang, "A novel cooperative caching scheme for wireless ad hoc networks: Group caching," presented at the 2007 Int. Conf. Netw. Architect. Storage (NAS 2007), Beijing, China, IEEE, 2007, pp. 62–68.

[21] E. Elahi, "Time dependent popularity caching scheme for NDN based manets," *Lahore Garrison Univ. Res. J. Comput. Sci. Info. Technol.*, vol. 5, no. 1, pp. 41–46, 2021. doi: 10.54692/lgurjcsit.2021.0501174.

[22] P. T. Joy and K. P. Jacob, "Cooperative caching techniques for mobile ad hoc networks," presented at the 2012 Int. Conf. Data Sci. Eng. (ICDSE), Cochin, India, IEEE, 2012, pp. 175–180.

[23] S. Naz *et al.*, "A dynamic caching strategy for CCN-based manets," *Comput. Netw.*, vol. 142, no. 11, pp. 93–107, 2018. doi: 10.1016/j.comnet.2018.05.027.

[24] "Centrality," Accessed: Oct. 26, 2022. [Online]. Available: https://en.wikipedia.org/wiki/Centrality

[25] "Eigenvector centrality," Accessed: Oct. 26, 2022. [Online]. Available: https://en.wikipedia.org/wiki/Eigenvectorcentrality

[26] Y. Liu, D. Zhu, and W. Ma, "A novel cooperative caching scheme for content centric mobile ad hoc networks," presented at the 2016 IEEE Symp. Comput. Commun. (ISCC), Messina, Italy, IEEE, 2016, pp. 824–829.

[27] S. Soleimani, X. Tao, and Y. Li, "Cooperative group caching strategy in content-centric wireless ad hoc networks," presented at the 2018 IEEE/CIC Int. Conf. Commun. China (ICCC), Shanghai, China, IEEE, 2018, pp. 793–797.

[28] S. Glass, I. Mahgoub, and M. Rathod, "Leveraging manet-based cooperative cache discovery techniques in vanets: A survey and analysis," *IEEE Commun. Surv. Tutorials*, vol. 19, no. 4, pp. 2640–2661, 2017. doi: 10.1109/COMST.2017.2707926.

[29] L. Zhang, J. Zhao, and Z. Shi, "LF: A caching strategy for named data mobile Ad Hoc networks," presented at the 4th Int. Conf. Comput. Eng. Netw., Springer, 2015, pp. 279–290.

[30] R. A. Rehman and B. S. Kim, "Location-aware forwarding and caching in CCN-based mobile ad hoc networks," *IEICE Trans. Inf. Syst.*, vol. 99, no. 5, pp. 1388–1391, 2016. doi: 10.1587/transinf.2015EDL8244.

[31] A. Rana and B. Kim, "LOMCF: Forwarding and caching in named data networking based manets," *IEEE Trans. Veh. Technol.*, vol. 66, no. 10, pp. 9350–9364, 2017. doi: 10.1109/TVT.2017.2700335.

[32] Q. Zheng, Y. Bao, Z. Gao, L. Yuan, J. Yang and X. Tan, "Research on content priority-based caching strategy in NDManet," presented at the 2020 3rd Int. Conf. Hot Info. Centric Netw. (HotICN), Shanghai, China, IEEE, 2020, pp. 59–64.

[33] Y. Zeng and X. Hong, "A caching strategy in mobile ad hoc named data network," presented at the 2011 6th Int. ICST Conf. Commun. Netw. China (CHINACOM), Harbin, China, IEEE, 2011, pp. 805–809.

[34] T. Zhang, X. Xu, L. Zhou, X. Jiang, and J. Loo, "Cache space efficient caching scheme for content-centric mobile ad hoc networks," *IEEE Syst. J.*, vol. 13, no. 1, pp. 530–541, 2018. doi: 10.1109/JSYST.2018.2851394.

[35] S. H. Shah and I. Yaqoob, "A survey: Internet of things (IoT) technologies, applications and challenges," presented at the 2016 IEEE Smart Energy Grid Eng. (SEGE), Oshawa, ON, Canada, 2016, pp. 381–385.

[36] Z. Ali, M. A. Shah, A. Almogren, I. Ud Din, C. Maple and H. A. Khattak, "Named data networking for efficient IoT-based disaster management in a smart campus," *Sustainability*, vol. 12, no. 8, pp. 3088, 2020. doi: 10.3390/su12083088.

[37] Y. Yang and T. Song, "Energy-efficient cooperative caching for information-centric wireless sensor networking," *IEEE Internet Things J.*, vol. 9, no. 2, pp. 846–857, 2021. doi: 10.1109/JIOT.2021.3088847.

[38] Z. Zhang, C. H. Lung, I. Lambadaris, and M. St-Hilaire, "IoT data lifetime-based cooperative caching scheme for ICN-IoT networks," presented at the 2018 IEEE Int. Conf. Commun. (ICC), Kansas City, MO, USA, 2018, pp. 1–7.

[39] O. Hahm, E. Baccelli, T. C. Schmidt, M. Wählisch, C. Adjih and L. Massoulié, "Low-power internet of things with NDN & cooperative caching," presented at the 4th ACM Conf. Info. Centric Netw., Berlin, Germany, 2017, pp. 98–108.

[40] M. T. R. Khan, M. M. Saad, M. A. Tariq, J. Akram, and D. Kim, "SPICE-IT: Smart COVID-19 pandemic controlled eradication over NDN-IoT," *Inf. Fusion*, vol. 74, no. 8, pp. 50–64, 2021. doi: 10.1016/j.inffus.2021.03.005.

[41] S. Anamalamudi, M. S. Alkatheiri, E. Al Solami, and A. R. Sangi, "Cooperative caching scheme for machine-to-machine information-centric IoT networks," *IEEE Can. J. Electr. Comput. Eng.*, vol. 44, no. 2, pp. 228–237, 2021. doi: 10.1109/ICJECE.2020.3046844.

[42] J. Quevedo, D. Corujo, and R. Aguiar, "Consumer-driven information freshness approach for content-centric networking," presented at the 2014 IEEE Conf. Comput. Commun. Workshops (INFOCOM WKSHPS), Toronto, ON, Canada, 2014, pp. 482–487.

[43] M. Meddeb, A. Dhraief, A. Belghith, T. Monteil, K. Drira and H. Mathkour, "Least fresh first cache replacement policy for NDN-based IoT networks," *Pervasive Mob. Comput.*, vol. 52, no. 2, pp. 60–70, 2019. doi: 10.1016/j.pmcj.2018.12.002.

[44] S. Vural, P. Navaratnam, N. Wang, C. Wang, L. Dong and R. Tafazolli, "In-network caching of internet-of-things data," presented at the 2014 IEEE Int. Conf. Commun. (ICC), Sydney, NSW, Australia, 2014, pp. 3185–3190.

[45] H. Shrisha and U. Boregowda, "An energy efficient and scalable endpoint linked green content caching for named data network-based internet of things," *Results Eng.*, vol. 13, no. 2, pp. 100345, 2022. doi: 10.1016/j.rineng.2022.100345.

[46] R. Shrimali *et al.*, "Proposed caching scheme for optimizing trade-off between freshness and energy consumption in name data networking based IoT," *Adv. Internet Things*, vol. 7, no. 2, pp. 11–24, 2017. doi: 10.4236/ait.2017.72002.

[47] M. Amadeo, C. Campolo, G. Ruggeri, and A. Molinaro, "Beyond edge caching: Freshness and popularity aware iot data caching via ndn at internet-scale," *IEEE Trans. Green Commun. Netw.*, vol. 6, no. 1, pp. 352–364, 2021. doi: 10.1109/TGCN.2021.3124452.

[48] M. A. Naeem, R. Ali, B. S. Kim, S. A. Nor, and S. Hassan, "A periodic caching strategy solution for the smart city in information centric internet of things," *Sustainability*, vol. 10, no. 7, pp. 2576, 2018. doi: 10.3390/su10072576.

[49] M. Amadeo, G. Ruggeri, C. Campolo, A. Molinaro, and G. Mangiullo, "Caching popular and fresh IoT contents at the edge via named data networking," presented at the IEEE INFOCOM 2020-IEEE Conf. Comput. Commun. Workshops (INFOCOM WKSHPS), Toronto, ON, Canada, 2020, pp. 610–615.

[50] G. Jaber and R. Kacimi, "A collaborative caching strategy for contentcentric enabled wireless sensor networks," *Comput. Commun.*, vol. 159, no. 2, pp. 60–70, 2020. doi: 10.1016/j.comcom.2020.05.018.

[51] D. Gupta, S. Rani, S. H. Ahmed, S. Verma, M. F. Ijaz and J. Shafi, "Edge caching based on collaborative filtering for heterogeneous ICN-IoT applications," *Sensors*, vol. 21, no. 16, pp. 5491, 2021. doi: 10.3390/s21165491.

[52] M. A. Hail, M. Amadeo, A. Molinaro, and S. Fischer, "Caching in named data networking for the wireless internet of things," presented at the 2015 Int. Conf. Recent Adv. Internet of Things (RIoT), Singapore, 2015, pp. 1–6.

[53] M. A. Naeem, T. N. Nguyen, R. Ali, K. Cengiz, Y. Meng and T. Khurshaid, "Hybrid cache management in iot-based named data networking," *IEEE Internet Things J.*, vol. 9, no. 10, pp. 7140–7150, 2021.

[54] B. Alahmri, S. Al-Ahmadi, and A. Belghith, "Efficient pooling and collaborative cache management for ndn/iot networks," *IEEE Access*, vol. 9, pp. 43228–43240, 2021. doi: 10.1109/ACCESS.2021.3066133.

[55] D. Saxena, V. Raychoudhury, and N. SriMahathi, "SmartHealth-NDNoT: Named data network of things for healthcare services," in *Mobile-Health@ MobiHoc*, 2015, pp. 45–50.

[56] A. Aboodi, T. C. Wan, and G. C. Sodhy, "Survey on the incorporation of NDN/CCN in IoT," *IEEE Access*, vol. 7, pp. 71827–71858, 2019. doi: 10.1109/ACCESS.2019.2919534.

[57] S. Rani, R. Maheswar, G. Kanagachidambaresan, and P. Jayarajan, *Integration of WSN and IoT for Smart Cities*. Springer, 2020.

[58] S. U. Rehman, M. A. Khan, T. A. Zia, and L. Zheng, "Vehicular adhoc networks (VANETs)–an overview and challenges," *J. Wirel. Netw. Commun.*, vol. 3, no. 3, pp. 29–38, 2013.

[59] C. Chen, C. Wang, T. Qiu, M. Atiquzzaman, and D. O. Wu, "Caching in vehicular named data networking: Architecture, schemes and future directions," *IEEE Commun. Surv. Tutorials*, vol. 22, no. 4, pp. 2378–2407, 2020. doi: 10.1109/COMST.2020.3005361.

[60] L. C. Liu, D. Xie, S. Wang, and Z. Zhang, "CCN-based cooperative caching in VANET," presented at the 2015 Int. Conf. Connected Veh. Expo (ICCVE), Shenzhen, China, 2015, pp. 198–203.

[61] L. Yao, Y. Wang, X. Wang, and W. Guowei, "Cooperative caching in vehicular content centric network based on social attributes and mobility," *IEEE Trans. Mob. Comput.*, vol. 20, no. 2, pp. 391–402, 2019. doi: 10.1109/TMC.2019.2944829.

[62] W. Huang, T. Song, Y. Yang, and Y. Zhang, "Cluster-based selective cooperative caching strategy in vehicular named data networking," presented at the 2018 1st IEEE Int. Conf. Hot Inf. Centric Netw. (HotICN), Shenzhen, China, 2018, pp. 7–12.

[63] W. Huang, T. Song, Y. Yang, and Y. Zhang, "Cluster-based cooperative caching with mobility prediction in vehicular named data networking," *IEEE Access*, vol. 7, pp. 23442–23458, 2019. doi: 10.1109/ACCESS.2019.2897747.

[64] Z. Su, Y. Hui, Q. Xu, T. Yang, J. Liu and Y. Jia, "An edge caching scheme to distribute content in vehicular networks," *IEEE Trans. Veh. Technol.*, vol. 67, no. 6, pp. 5346–5356, 2018. doi: 10.1109/TVT.2018.2824345.

[65] Y. Wei, C. Xu, M. Wang, and J. Guan, "Cache management for adaptive scalable video streaming in vehicular content-centric network," presented at the 2016 Int. Conf. Netw. Netw. Appl. (NaNA), Hakodate, Japan, 2016, pp. 410–414.

[66] L. Yao, A. Chen, J. Deng, J. Wang, and G. Wu, "A cooperative caching scheme based on mobility prediction in vehicular content centric networks," *IEEE Trans. Veh. Technol.*, vol. 67, no. 6, pp. 5435–5444, 2017. doi: 10.1109/TVT.2017.2784562.

[67] Q. Wu, Y. Zhao, Q. Fan, P. Fan, J. Wang and C. Zhang, "Mobility-aware cooperative caching in vehicular edge computing based on asynchronous federated and deep reinforcement learning," *IEEE J. Sel. Top. Signal Process.*, vol. 17, no. 1, pp. 66–81, 2023. doi: 10.1109/JSTSP.2022.3221271.

[68] J. Gu, Y. Ji, W. Duan, and G. Zhang, "Node value and content popularity based caching strategy for massive vanets," *Wirel. Commun. Mob. Comput.*, vol. 2021, no. 2, pp. 1–10, 2021. doi: 10.1155/2021/5585896.

[69] S. Ostrovskaya *et al.*, "Towards multi-metric cache replacement policies in vehicular named data networks," presented at the 2018 IEEE 29th Ann. Int. Symp. Person. Indoor Mob. Radio Commun. (PIMRC), Bologna, Italy, 2018, pp. 1–7.

[70] H. Khelifi, S. Luo, B. Nour, and H. Moungla, "A QoS-aware cache replacement policy for vehicular named data networks," presented at the 2019 IEEE Global Commun. Conf. (GLOBECOM), Waikoloa, HI, USA, 2019, pp. 1–6.

[71] L. Yao, Y. Wang, Q. Xia, and R. Xu, "Popularity prediction caching using hidden markov model for vehicular content centric networks," presented at the 2019 20th IEEE Int. Conf. Mob. Data Manag. (MDM), Hong Kong, China, 2019, pp. 533–538.

[72] Y. Zhang, R. Wang, M. S. Hossain, M. F. Alhamid, and M. Guizani, "Heterogeneous information network-based content caching in the internet of vehicles," *IEEE Trans. Veh. Technol.*, vol. 68, no. 10, pp. 10216–10226, 2019. doi: 10.1109/TVT.2019.2936792.

[73] Z. Wei, J. Pan, K. Wang, L. Shi, Z. Lyu and L. Feng, "Data forwarding and caching strategy for RSU-aided V-NDN," presented at the Int. Conf. Wirel. Algorith. Sys. Appl., Hong Kong, China, 2019, pp. 605–612.

[74] W. Zhao, Y. Qin, D. Gao, C. H. Foh, and H. C. Chao, "An efficient cache strategy in information centric networking vehicle-to-vehicle scenario," *IEEE Access*, vol. 5, pp. 12657–12667, 2017. doi: 10.1109/AC-CESS.2017.2714191.

[75] A. Dua, M. Shishodia, N. Kumar, G. S. Aujla, and N. Kumar, "Bloom filter-based efficient caching scheme for content distribution in vehicular networks," presented at the 2019 IEEE Int. Conf. Commun. Workshops (ICC Workshops), Shanghai, China, 2019, pp. 1–6.

[76] F. de Moraes Modesto and A. Boukerche, "Utility-gradient implicit cache coordination policy for information-centric ad-hoc vehicular networks," presented at the 2017 IEEE 42nd Conf. Local Comput. Netw. (LCN), Singapore, 2017, pp. 10–17.

[77] Y. Meng, M. A. Naeem, R. Ali, Y. B. Zikria, and S. W. Kim, "DCS: Distributed caching strategy at the edge of vehicular sensor networks in information-centric networking," *Sensors*, vol. 19, no. 20, pp. 4407, 2019. doi: 10.3390/s19204407.

[78] G. Deng, L. Wang, F. Li, and R. Li, "Distributed probabilistic caching strategy in VANETs through named data networking," presented at the 2016 IEEE Conf. Comput. Commun. Workshops (INFOCOM WKSHPS), San Francisco, CA, USA, 2016, pp. 314–319.

[79] T. Nakazawa, S. Tang, and S. Obana, "CCN-based inter-vehicle communication for efficient collection of road and traffic information," *Electronics*, vol. 9, no. 1, pp. 112, 2020. doi: 10.3390/electronics9010112.

[80] N. Abani, T. Braun, and M. Gerla, "Proactive caching with mobility prediction under uncertainty in information-centric networks," presented at the 4th ACM Conf. Info. Centric Netw., Berlin, Germany, 2017, pp. 88–97.

[81] H. Khelifi, S. Luo, B. Nour, A. Sellami, H. Moungla and F. Naït-Abdesselam, "An optimized proactive caching scheme based on mobility prediction for vehicular networks," presented at the 2018 IEEE Global Commun. Conf. (GLOBECOM), Abu Dhabi, United Arab Emirates, 2018, pp. 1–6.

[82] D. Grewe, M. Wagner, and H. Frey, "PeRCeIVE: Proactive caching in ICN-based VANETs," presented at the 2016 IEEE Veh. Netw. Conf. (VNC), Columbus, OH, USA, 2016, pp. 1–8.

[83] S. Park, S. Oh, Y. Nam, J. Bang, and E. Lee, "Mobility-aware distributed proactive caching in content-centric vehicular networks," presented at the 2019 12th IFIP Wirel. Mob. Netw. Conf. (WMNC), Tokyo, Japan, 2019, pp. 175–180.

[84] G. Mauri, M. Gerla, F. Bruno, M. Cesana, and G. Verticale, "Optimal content prefetching in ndn vehicle-to-infrastructure scenario," *IEEE Trans. Veh. Technol.*, vol. 66, no. 3, pp. 2513–2525, 2016. doi: 10.1109/TVT.2016.2580586.

[85] Y. Zhang, C. Li, T. H. Luan, Y. Fu, W. Shi and L. Zhu, "A mobility aware vehicular caching scheme in content centric networks: Model and optimization," *IEEE Trans. Veh. Technol.*, vol. 68, no. 4, pp. 3100–3112, 2019. doi: 10.1109/TVT.2019.2899923.

[86] H. Tian, Y. Otsuka, M. Mohri, Y. Shiraishi, and M. Morii, "Leveraging in-network caching in vehicular network for content distribution," *Int. J. Distrib. Sens. Netw.*, vol. 12, no. 6, pp. 8972950, 2016. doi: 10.1155/2016/8972950.

[87] H. Khelifi, S. Luo, B. Nour, H. Moungla, and S. H. Ahmed, "Reputation based blockchain for secure ndn caching in vehicular networks," presented at the 2018 IEEE Conf. Standards Commun. Netw. (CSCN), Paris, France, 2018, pp. 1–6.

[88] H. A. Khattak, F. Z. Raja, M. Aloqaily, and O. Bouachir, "Efficient in-network caching in NDN-based connected vehicles," presented at the 2021 IEEE Global Commun. Conf. (GLOBECOM), Madrid, Spain, 2021, pp. 1–6.

[89] Z. Su, P. Ren, and X. Gan, "A novel algorithm to cache vehicular content with parked vehicles applications," presented at the 2014 IEEE Int. Conf. Commun. (ICC), Sydney, Australia, 2014, pp. 5665–5669.

[90] J. Zhou, J. Luo, L. Deng, and J. Wang, "Cache pollution prevention mechanism based on cache partition in V-NDN," presented at the 2020 IEEE/CIC Int. Conf. Commun. China (ICCC), Chongqing, China, 2020, pp. 330–335.

[91]   H. Tian, M. Mohri, Y. Otsuka, Y. Shiraishi, and M. Morii, "LCE in network caching on vehicular networks for content distribution in urban environments," presented at the 2015 Seventh Int. Conf. Ubiquitous Future Netw., Sapporo, Japan, 2015, pp. 551–556.

[92]   V. Sampath, S. Karthik, and R. Sabitha, "Position-based adaptive clustering model (PACM) for efficient data caching in vehicular named data networks (VNDN)," *Wirel. Pers. Commun.*, vol. 117, no. 4, pp. 2955–2971, 2021. doi: 10.1007/s11277-020-07208-2.

[93]   M. Amadeo, "A literature review on caching transient contents in vehicular named data networking," *Telecom*, vol. 2, no. 1, pp. 75–92, 2021.

[94]   M. Y. Khan, M. Adnan, J. Iqbal, N. ul Amin, B. H. Roh and J. Ali, "A hierarchal clustered based proactive caching in NDN-based vehicular network," *Comput. Syst. Sci. Eng.*, vol. 47, no. 1, 2023.

[95]   C. Liao, X. Liu, and H. Zhou, "Environment-adaptive dynamic caching for vehicular named data networks in dynamic network environments," *IEEE Trans. Veh. Technol.*, 2023.

[96]   S. Dhara, A. Majidi, and S. Clarke, "Revving up VNDN: Efficient caching and forwarding by expanding content popularity perspective and mobility," *Comput. Commun.*, vol. 212, no. 6, pp. 342–352, 2023. doi: 10.1016/j.comcom.2023.10.004.

[97]   K. Rashid, Y. Saeed, A. Ali, R. Alkanhel, and A. Muthanna, "An adaptive real-time malicious node detection framework using machine learning in vehicular Ad-Hoc networks (VANETs)," *Sensors*, vol. 23, no. 5, pp. 2594, 2023. doi: 10.3390/s23052594.

[98]   M. S. M. Shah, Y. B. Leau, M. Anbar, and A. A. Bin-Salem, "Security and integrity attacks in named data networking: A survey," *IEEE Access*, vol. 11, pp. 7984–8004, 2023. doi: 10.1109/ACCESS.2023.3238732.

[99]   A. Benmoussa, C. A. Kerrache, C. T. Calafate, and N. Lagraa, "NDN-BDA: A blockchain-based decentralized data authentication mechanism for vehicular named data networking," *Future Internet*, vol. 15, no. 5, pp. 167, 2023. doi: 10.3390/fi15050167.

[100] I. K. B. Cunha, J. C. Junior, M. P. Fernandez, A. Patel, and M. E. Monteiro, "VNDN-Fuzzy–A strategy to mitigate the forwarding interests broadcast storm problem in VNDN networks," presented at the 2023 Int. Conf. Inf. Netw. (ICOIN), Bangkok, Thailand, 2023.

[101] R. Glazkov *et al.*, "Provisioning of fog computing over named-data networking in dynamic wireless mesh systems," *Sensors*, vol. 24, no. 4, pp. 1120, 2024. doi: 10.3390/s24041120.