Adaptive Forwarding and Routing of Named Data Networking: A Literature Survey

Ade Aditya Ramadha[1], Leanna Vidya Yovita[2], Tody Ariefianto Wibowo[3], Ilmi Aulia Assyifatunisa[4]
[1,2,3,4]*Telkom University*
[1,2,3,4]*Bandung, Indonesia*
email : [1]ade.ramadha@gmail.com, [2]leanna@telkomuniversity.ac.id

ARTICLE INFORMATION

Received 21 July 2023
Revised 27 September 2023
Accepted 06 November 2023

Keywords:
Adaptive Networking
Artificial Intelligence
Forwarding
Named Data Networking
Routing

ABSTRACT

In recent years, Named Data Networking (NDN) has been one of the future internet network architectures by proposing it as a substitute for existing IP networks. NDN assigns addresses/names to data or content, whereas IP assigns addresses to devices. NDN router has a Content Store (CS) component to store requested packets and reducing the time for duplicate content requests. As network architecture, NDN uses several algorithms and strategies in the process. This paper will specifically discuss the forwarding and routing algorithm. Forwarding mechanisms play a significant role in packet delivery on the NDN system. Even though routing was not the main component of the NDN system, the routing mechanism and Routing Information Base remain important, considering Forwarding Information Base was generated after Routing Information Base was generated. Routing has significant control, whereas forwarding can give finer control over the delivery path. Considering the previous statement, the combination of routing and forwarding strategy must be considered to optimize the performance of the packet delivery system on NDN. In addition to getting information about the entire network, centralized routing, and adaptive networking are needed to distribute all network traffic fairly. The challenges and opportunities can also be a reference and a guideline for future Adaptive NDN research.

1. Introduction

Host-to-host communication has been the foundation of the internet network since 1960. Initially, this communication aims to solve the problem of resource sharing. For the modern internet user, who prioritizes data content over data origin in computer networks, The need for host-to-host communication has passed. To address this issue, numerous novel network architectural studies have been carried out (Jacobson et al., 2012). At first, Van Jacobson proposed new network architecture called Content-Centric Network (CCN) (Jacobson et al., 2012). In 2014, Named Data Networking (NDN) became the successor to CCN (L. Zhang et al., 2014). This paper also explains the basic concept of NDN, current NDN Development, and NDN research challenges. Furthermore, in 2018, NDN was declared one of the proposed internet architectures (Alex Afanasyev et al., 2019). In contrast to the IP network architecture, which assigns addresses/names to devices, NDN network architectures assign addresses/names to data or content. NDN changes the network paradigm from being based on the location of the content to the content itself. With this paradigm shift, clients can request content directly without knowing the location of the data or even the network topology itself.

One instance of NDN implementation is demonstrated in the (Ramadha et al., 2022). The authors utilized NDN to implement a video streaming system and compared it with a video streaming system using IP. The findings indicate that NDN can enhance Quality of Service and Quality of Experience during the second access because of its caching mechanism.

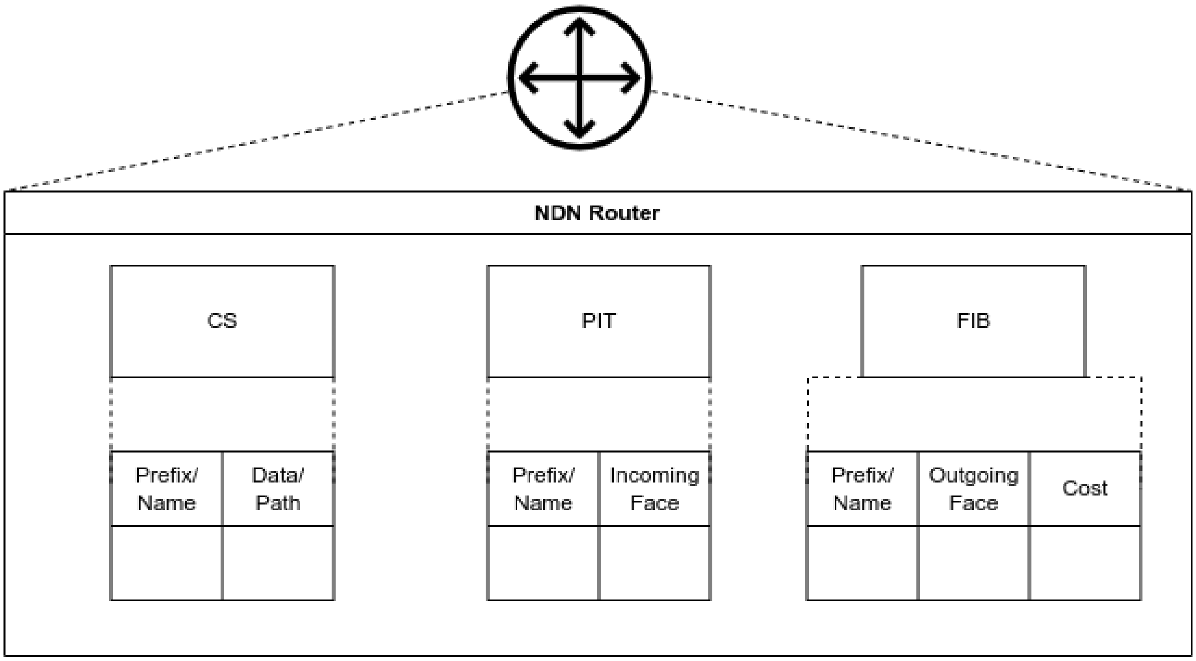

Figure 1. NDN Router Main Component.

NDN router has three main components to forward the interest packet and the data packet, as shown in Fig. 1. There are (i) Content Store (CS) to cache data packets and improve the time taken for duplicate content. (ii) Pending Interest Table (PIT) records interest received and their incoming interface to guide the packet of the data back to the client. (iii) Forwarding Information Base (FIB) contains a possible path or outgoing interface to the content.

The customer submits a content interest packet. First, the NDN router will search for matching prefixes/names from the interest packet in CS. If the requested content is in CS, the Incoming interface will receive the packet data from the router. Otherwise, the interest packet will process the interest packet to PIT. Next, the router will check for the same prefix on its PIT, and if there are any prefix records, the incoming interface will be added to the record by the router. Otherwise, the interest packet will process the interest packet to FIB. In FIB, the router will select a path route to the possible interface of the content. If no possible interface is found, the router will send nack packet back to the incoming interface.

Several algorithms and strategies are used in the process. On the forwarding side, the router selects a face from the FIB table based on the forwarding strategy. The FIB table was generated after the Routing Information Base (RIB) was successfully generated. Despite not being the main component of the NDN router, the routing mechanism and routing information base remain essential to discuss. Furthermore, the NDN paradigm is different from the IP paradigm, so new algorithms or modifications to existing algorithms need to be done.

According to the previous statement, forwarding and routing must be considered to provide an efficient and adaptive content delivery path on Named Data Networking. To understand the existing mechanisms and the combination of both, this paper will give a brief overview of previous research and map some potential future directions for NDN network mechanism research. Based on reference papers from (Ariefianto & Syambas, 2017; Rowshanrad et al., 2016; Sembati et al., 2022) have explained Centralized routing which has been of interest in recent years by researchers. Where the old paradigm becomes a major influence for the development of a new architecture, namely SDN (Software Defined Network). So referring to these papers as the main source of implementing NDN functionality using SDN, including consolidating special controller nodes for NDN. It's just that the methods they reviewed did not fully discuss centralized routing. For example in a paper survey (Ariefianto & Syambas, 2017; Sembati et al., 2022) describes research in the field of NDN routing and also classifies all types of routing protocols. Likewise with the paper survey (Rowshanrad et al., 2016) which only focuses on discussing the implementation of NDN using SDN. Not focused on centralized routing. Therefore, to the best of our knowledge, this is the first paper we have proposed that focuses on centralized routing.

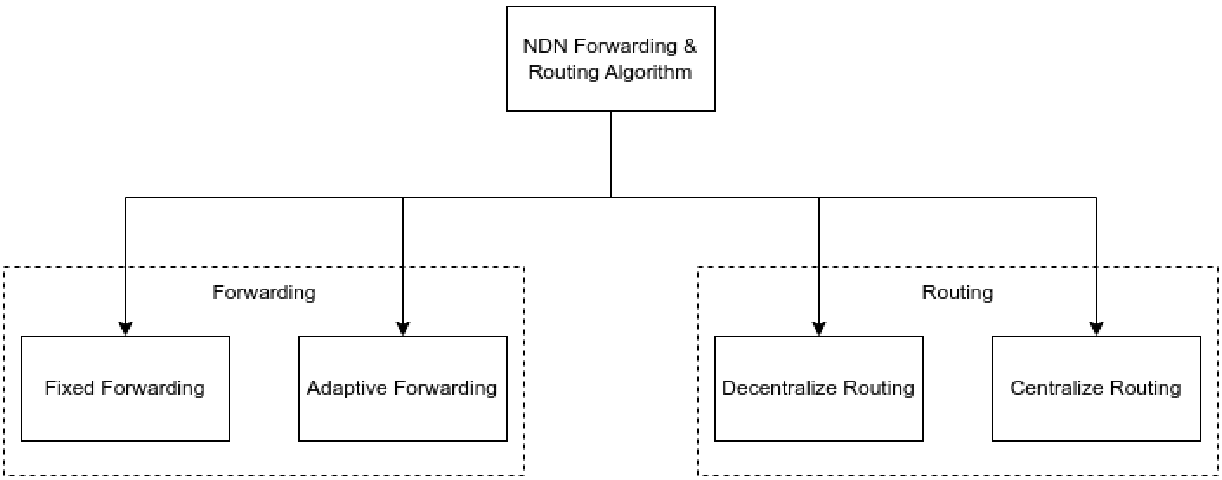

Figure 2. Classification of NDN Routing and Forwarding.

The structure of this paper is as follows: Section II shows a brief review of default forwarding and includes many proposed adaptive forwarding mechanisms. In section III, we classify several proposed NDN routing mechanisms into centralized and decentralized mechanisms. Section IV mainly discussed open research problems and opportunities. Lastly, section V will conclude the topic of this paper. The chosen research papers and methods for this study were searched using the keywords "Named Data Networking," "Adaptive Routing," "Adaptive Forwarding," "Routing," and "Forwarding." All of the research materials were indexed in Scopus. Furthermore, the selected research materials do not have a year limit as the aim of this paper is to examine the development of NDN forwarding and routing mechanisms from the very beginning.

2. NDN Forwarding

The forwarding mechanism plays a significant role in packet delivery on the NDN system; considering that the downstream path will be the same as the upstream path as shown in Fig. 2 we map the forwarding mechanism into two groups. First is fixed forwarding, which highly depends on routing cost without additional estimation of network condition. The second group is adaptive forwarding which has a mechanism to estimate network conditions and adapt accordingly.

2.1. Fixed Forwarding

The default forwarding defined on NDN Forwarding Daemon (NFD), such as Best Route, Multicast, and Access, is considered as fixed forwarding because their behavior highly depends on routing cost and does not have a mechanism to estimate network condition (Alexander Afanasyev et al., 2015).

2.1.1. Best Route

The Best Route forwarding strategy forwards interest to the lowest cost available interface based on RIB. The received interest would be suppressed if received within the suppression interval if the selected face has a pending interest with the same prefix, selectors, and link. The Best-Route method will transmit the interest to the lowest cost available face that has not been utilized before if the interest is received after the suppression interval, referred to as the retransmission interval. This forwarding mainly only uses one outgoing face. Although ensuring the best path, the Best Route strategy may lead to congestion in a saturated scenario.

2.1.2. Multicast

Multicast forwarding strategy forward incoming interest to all available routes in FIB. If the matched first received data packet on the router, the router will forward the data packet back to the client and drop all received data packets afterward. This forwarding gives higher overhead if several interests with different prefixes enter the router. Although ensuring the fastest path to retrieve the data, the Multicast strategy also leads to interest packet flooding on the NDN network.

2.1.3. Access

Access forwarding strategy uses the combination of Best Route and Multicast mechanism. Access forwarding strategy forwards interest packets to all available routes. The router will record the incoming face of the data packet when it has been received. Even when the FIB nexthops are inaccurate, this method tolerates them and can still find the correct path. The Access strategy is the best of the two worlds between the Best Route and Multicast strategies, although it can lead to interest packet flooding on the first request.

2.2. Adaptive Forwarding

Several proposed methods shown in (de Sena et al., 2020; Fu et al., 2017; Gong et al., 2016; Hao et al., 2021; Kerrouche et al., 2016; Lehman et al., 2016; Lei et al., 2015; Mekinda & Muscariello, 2016; Posch et al., 2017; Ren et al., 2019; Yao et al., 2017; Ye et al., 2017; G. Zhang et al., 2015) are considered as adaptive forwarding for their ability to estimate network conditions and adapt accordingly.

2.2.1. Adaptive Smoothed RTT-based Forwarding (ASF)

ASF strategy proposed by Lehman et al. (Lehman et al., 2016) to mitigate Hyperbolic Routing sub-optimal path selection. ASF strategy periodically measures Smoothed Round-Trip Time (SRTT) on all available interfaces for every interest requested. ASF strategy forwards interest packet to the highest rank face according to its cost, and SRTT for the same prefix requested. Even though the ASF strategy can improve the Hyperbolic Routing network and more balanced path selection on the Fixed Routing network, the probing mechanism of this strategy can lead to higher overall Round-Trip Time (RTT) and increased forwarding overhead.

2.2.2. Heterogeneous-Latency Adaptive Forwarding (HLAF)

The HLAF strategy proposed by Yuhang et al. (Ye et al., 2017) proposed HLAF to improve video streaming applications in NDN. HLAF forwards the packet according to the face congestion level and RTT. This mechanism led to efficient content sharing among nodes, with maximum throughput and minimum latency. HLAF keeps track of the percentages of interface allocations and adjusts the allocation based on NACK as explicit congestion signals and RTT simultaneously. With the improved mechanism for video streaming, HLAF reduces stalling time, increases the playback video quality, and outperforms other conventional strategies.

2.2.3. Maximizing Deviation Based Probabilistic Forwarding (MDPF)

MDPF strategy proposed by Kai et al. (Lei et al., 2015) and select forwarding interface based on probability. This method treats the NDN forwarding problem’s various characteristics as a MADM issue. First, MDPF builds a decision matrix from each attribute and calculates the optimal weight vector using the MD method. Next, MDPF computes the interface score using Simple Additive Weighting. To select the forwarding interface, MDPF uses a fitness proportionate selection algorithm. With this mechanism, MDPF can improve performance and swiftly detect and react to network changes.

2.2.4. Stochastic Adaptive Forwarding (SAF)

SAF strategy proposed by Daniel et al. (Posch et al., 2017) enables more scalable with requiring laxer convergence time requirements and completeness. Every router implements the SAF strategy to maintain a virtual face FD, which acts as an overpressure valve. Every interest forwarded to virtual face FD will be dropped. The SAF approach maintains a two-dimensional matrix forwarding table with rows for each of the accessible faces and columns for each of the contents that are accessible. Every element of the matrix indicates the probability for every face and prefix. With the improved mechanism, SAF can outperform existing algorithms such as iNRR, OMP-IF, and RFA.

2.2.5. POMDP Framework

This strategy proposed by Jinfa et al. (Yao et al., 2017) maximizes various connectivities and ensures adequate network performance. Based on the fundamental idea of an event, the Partly Observable Markov Decision Process (POMDP) is utilized to create the NDN request routing mechanism. This paper proposes an approximation algorithm with decreased complexity. As a result, This strategy achieves a superior network performance compared with other existing strategies, such as the optimum route forwarding strategy and uniform random forwarding method.

2.2.6. QoS-FS

QoS-FS proposed by Abdelali et al. (Kerrouche et al., 2016) to improve forwarding using Quality of Service (QoS) aware mechanism. In this way, forwarding depends on the routing mechanism and can adapt to various network conditions. As stated before, this strategy can adapt to various network conditions thanks to its real-time monitoring of the link QoS mechanism. This strategy use routing cost, bandwidth, and RTT as metric. The results show that the proposed QoS-FS offer more QoS to the end user and the lowest hop count as traffic volume rises.

2.2.7. Stochastic Adaptive Forwarding Strategy based on Deep Reinforcement Learning (SAF-DRL)

SAF-DRL proposed by Bowei Hao et al. (Hao et al., 2021) not only seeks to boost mobile video communication effectiveness but furthermore keep away from assaults to raise security. As the name implies, SAF-DRL uses Deep Reinforce Learning (DRL) to optimize the path selection. The state space has three essential components: 1) throughput, 2) delay, and 3) interface, for Action space defines the adaptive forwarding probability of each interface and content prefixes. The last, Reward space is defined as each node’s overall utility in the network. They use the TD3 algorithm as the DRL algorithm. With the implementation of DRL on SAF, When compared to SAF, SAF-DRL can achieve faster delivery times and fewer lost packets..

2.2.8. Luby Transform Code Aided Multiple Forwarding (LTCA-MF)

LTCA-MF strategy proposed by Guanghui Zhang et al. (G. Zhang et al., 2015) to improve forwarding using Luby Transform (LT) coded content chunks. LTCA-MF using SDN concept for routing management framework. In their implementation, the device is divided into intermediate routers, controllers, code-enabled servers, and end users. Controllers create the topology and determine the best multipath routing method based on the LTCA-MF approach. As multipath-establishing operations and content transmission, LTCA-MF employs discontinuous pathways and parallel content forwarding. LTCA-MF uses Alternative Multipath Algorithm (AM Algorithm) to detect disjoint paths. Their use of the AM algorithm allows them to avoid links with more weight and make better decisions than the greedy technique.

In the implementation, the end user sends requests to the controller. The AM Algorithm enables the controller to calculate and install the multipath in their FIB. Based on the multipath sent by the controller, the end user send a parallel content request on all paths. Upon receiving a content request, the server divides content into several chunks and codes by the LT code, and all LT-coded chunks are forwarded back to the end user. The end user will send a stop signal to the server once they have received enough chunks to decode. Based on their implementation, LTCA-MF outperforms earlier methods (OMP-IF and BPI).

2.2.9. Dynamic Multi-path Forwarding (DMF)

DMF proposed by Ren et al. (Ren et al., 2019) to improve forwarding using RTT at the initial phase and bandwidth at the saturated phase. According to their analytical model, RTT controls the receiving rate during the early phase, so the path with the smallest RTT was chosen for the metric. On the other hand, The bottleneck bandwidth determines the receiving rate at the saturated phase, so the most extensive bandwidth-available path was chosen for the metric. During testing, DMF was implemented in ndnSIM and outperformed the existing strategies like Fast Pipeline Filling (FPF).

2.2.10. DQN-AF : Deep Q-Network based Adaptive Forwarding Strategy for Named Data Networking

DQN-Adaptive Forwarding (DQN-AF) (de Sena et al., 2020) is an adaptive forwarding strategy based on the Deep Q Network (DQN) method. First, the router collects information about the available links from FIB. The DQN-AF Agent processes the information gathered to determine the status of the environment. The NDN router alerts the state for the DQN-AF Agent to determine the best next hop after receiving the interest packet. According to agent prediction, the router forwards the interest to the selected interface and records it to the PIT. Only two events will trigger an alert from the NDN router: (a) the arrival of the relevant data packet, and (b) loss detection, with an Acknowledgment (NACK) or timeout. This paper’s authors conduct trials on the best route forwarding strategy, ASF, DQ-Learning, Multicast, and DQN-AF. The goal is to determine which forwarding technique can send more data packets. In contrast to conventional regular NDN non-adaptive forwarding, DQN-AF chooses the matching hop based on information from the router link.

2.2.11. Reinforcement learning-based algorithm for efficient and adaptive forwarding in named data networking

This paper explained (Fu et al., 2017) by modifying the Q-Learning algorithm to solve the problem the writer designs and applies an IQ-Learning strategy (Interest Q-Learning) and DQ Learning strategy (Q-Learning data). The DQ-Learning strategy has the same algorithm as the IQ-Learning technique, with the sole variation being how information is updated, which does not require additional feedback packages and can save the energy of intermediary nodes. The initial flow of this strategy is the data package again following the way the coming interest package. The intermediary node receives the data package and unravels the data package to get the Q then updates the local Q table and sends back the data package that is packaged according to the notes on the pit. Likewise, the DQ-Learning strategy has several parts that are the same as Receiving a snack delivery, or old interest notes are examples of IQ-Learning. The authors demonstrate the benefits of techniques in terms of efficiency based on the simulation findings and adaptivity compared to flooding strategies and best route strategies.

2.2.12. Intelligent Forwarding Strategy Based on Online Machine Learning in Named Data Networking

This paper presents (Gong et al., 2016), a probabilistic binary tree-based forwarding strategy (PBTF). This technique is described as follows: (i) The appropriate alternative interface is extracted under FIB when the router gets interest that needs to be forwarded, (ii) With all interfaces included, binary probabilistic trees are constructed and initialized, (iii) According to various scenario demands, the learning process threshold's function is regulated. (iv) The cost of each interface is computed throughout the probabilistic tree learning process, (v) The interface with the cost and the smallest increase in the value of the non-leaf node is selected, (vi) The annealing algorithm is used to prevent the method from becoming a local optimum, (vii) According to the newly created probability tree, the selection procedure is carried out between the leaf node and the root node, (viii) Continuing interest received through this correspondence interface. As a result, PBTF performed better than other strategies with lower complexity because it benefited from using two common tactics simultaneously.

2.2.13. Supervised Machine Learning-Based Routing for Named Data Networking

In this paper, the author proposes (Mekinda & Muscariello, 2016) ask for Direction (AFFORD) as a new stochastic forwarding strategy using a supervised Machine Learning approach. In AFFORD, the probability of sending interest to the face of the way out is calculated with ANN-FIB, an artificial nerve network that replaces ordinary FIB. A synthetic neural network called Multilayer Perceptrs (MLP) divides neurons into inputs, one or more hidden layers, and output layers. In AFFORD, MLP reads a signal containing the name of the coded object as the input layer and calculates a probability vector as the output layer. Considering the size of the MLP and Dataset impacted the learning time, AFFORD strategy divides the namespace with Bitwise Trie systems. Each lower trie node refers to the ANN RIB responsible for the subset namespace, which begins with its path in Trie. The outcomes of AFFORD simulation are auspicious, and the results demonstrate that ANN-FIB performs significantly better than traditional FIBs.

2.2.14. An Efficient Location-Based Forwarding Strategy for Named Data Networking and LEO Satellite Communications

A new location-based forwarding strategy proposed in (Iglesias-Sanuy et al., 2022) specialized for LEO satellite networks. Their methods utilize knowledge of the relative position of satellite and formed by ISLs. Each satellite keeps four ISL for each four satellite neighbor: two intra-plane in same orbital plane, and two inter-plane closest satellite of both immediate adjacent planes. Futhermore, their consider satellite network as a grid with Cartesian coordinates. To solve mobile producer problem, producer reports its point of attachment (PoA) constantly.

NDN Link Adaptation Protocol version 2 (NDNLPv2) protocols supports GeoTag using 3-byte header to represent Cartesian 3D coordinates (GPS). Their implementation using the first two bytes to encode the two Cartesian coordinates. With this implementation, consumer obtains the PoA of producer and sends the coordinates on GeoTag Field of NDN. To forward the interest packet, satellite choose one from five possible nexthop: if GeoTag match the address of satellite, satellite forward the interest packet to ground link. Otherwise, satellite choose one neighboring satellite with closest index. As a result, their implementation of methods using NDNsim software shows the method works properly, 95% link usage at producer, and, maximise the network capacity.

2.2.15. IFS-RL: An Intelligent Forwarding Strategy Based on Reinforcement Learning in Named-Data Networking

This paper (Y. Zhang et al., 2018) proposes a reinforcement learning-based forwarding called IFS-RL. Reinforcement learning is a form of machine learning that learns continuously by analyzing the environment and determining the optimal action. Applying reinforcement learning, IFS-RL has the ability to continuously learn and select the optimal face based on network conditions without relying on pre-trained models. In addition, this paper also examines the granularity of learning and improvement for network topology changes. As a results, IFS-RL exhibits greater adaptability than the best-route forwarding strategy and has lower computational complexity than the EPF forwarding strategy. Furthermore, IFS-RL achieve a higher throughput and lower packet forwarding rate compared to another forwarding strategy.

2.2.16. Intelligent Forwarding Strategy for Congestion Control Using Q-Learning and LSTM in Named Data Networking

This paper (Ryu et al., 2021) proposes IFS-QLSTM. In contrast to IFS-RL, IFS-QLSTM using Q-Learning and LSRM for congestion control mechanism. The LSTM model utilized in this approach can identify traffic congestion by learning the rate of Pending Interest Table entries. Additionally, Q-Learning is employed to detect congestion in nearby nodes through analysis of the predicted PIT entry rate from the preceding LSTM model. As a results, IFS-QLSTM has a higher data rate and lower packet forwarding compared to best-route forwarding strategy and ASF.

2.2.17. Interest Forwarding in Named Data Networking Using Reinforcement Learning

The authors in this paper (Akinwande, 2018) proposed an adaptive forwarding strategy called NDNFS-RLRNN. NDNFS-RLRNN uses the RL method with RNN in the process. By utilizing the convergence of the RL method and the RNN algorithm to maximize the content store hit ratio. Initially, RNN only has information related to prefixes and path preferences contained in the FIB. RNN will continue to learn the network using RTT collected from probing mechanism as a reward value. The probing process use a different type of interest, normal interest will be sent using the best interface. Probe interest in other hand will be sent using random interface. The different type of interest can lead to better exploration process by NDNFS-RLRNN. As a result, NDNFS-RLRNN gets better delivery performance than ASF and Nearest Replica Routing (NRR).

2.2.18. DRL-based Forwarding Strategy in Named Data Networking

In this paper (Lv et al., 2018) the authors propose a DRL-based Forwarding Strategy. As its name suggests, the method presented here employs Deep Reinforcement Learning (DRL) to determine the optimal path. Along the way, a Deep Q Network (DQN) model is trained at each node to address the path selection issue. Subsequently, when the data packet enters the router, the DRL-based forwarding strategy learns the network conditions by analyzing the incoming data packet and using RTT as the DQN reward. With this method, the DRL-based forwarding strategy achieves better performance in terms of RTT and throughput.

2.2.19. A Forwarding Strategy based on Reinforcement Learning for Content-Centric Networking

This paper (Bastos & Moraes, 2016) proposes a packet forwarding strategy for Information Center Networks called Multi Armed Bandits Strategy (MABS). MABS aims to decrease the time taken for retrieving packet data. The MAB issue addressed in this paper concerns the chance that an interest packet cannot be fulfilled by the content store and is not present on the PIT, thereby requiring the router to choose one of the available interfaces. To overcome this challenge, MABS employs a greedy algorithm to identify the optimal route by recording the reception time of each available interface. The MABS method is implemented using

NDNsim and compared to other algorithms, including Best-Route and Smart-Flooding. The results show that the MABS method outperforms Best-Route and Smart-Flooding, reducing hops by 28% and requiring less effort.

2.2.20. AFSndn

ASFndn proposed in (M. Zhang et al., 2020) using Q-learning algorithm to improve forwarding strategy. To speed up the learning speed, AFSndn implement a heuristic knowledge to Q-Learning. AFSndn has two phase: (1) Exploration phase to collects the information by calculating the Q-value of prefix and interface pair based on the information included in the packets; (2) Exploitation phase to forward the interest packet based on the Q-value and select the lowest Q-value. Furthermore, to support the exploration phase, ASFndn add a new attribute Q-value to FIB, and adds departure time and minimum Q-value attribute to data packet. As a results based on their implementation on NDNsim, the algorithm shows better performance in term of average delay and number of packet loss compared to MFC and RFA algorithm.

3. NDN Routing

NDN uses routing to set up topologies and policies, handle long-term changes, and update forwarding tables. In NDN, the main distinction between routing and forwarding is that forwarding decides router priority and utilization based on performance/status, whereas routing determines router availability. IP and NDN use FIBs to hold routing-related data. On forwarding the packet to the target address, IP checks the FIB to determine the next hop, which may not always be the most direct route. The same is true for NDN, which searches the FIB using the name prefix to determine the next hop and obtain the data—not necessarily the next copy. To make NDN function without routing, routers must be able to efficiently: 1) When the network is stable, download the data; 2) handle connection Mistakes; 3) handle reconnection (Yi et al., 2014) as shown in Fig. 2 we also map the routing mechanism into two groups.

3.1. Routing Decentralized

To provide multipath forwarding, NDN routing systems must provide multiple next hops. The NDN project team suggests two link-state-based routing protocols: OSPFN and NLSR (Hoque et al., 2013; Wang & Hoque, 2012)

3.1.1. NLSR

The Named-data Link State Routing Protocol (NLSR) was proposed by Hoque et al. (Hoque et al., 2013), which runs on top of NDN. In other words, NLSR exchanges routing messages using NDN's Interest or Data packets. Next, NLSR uses adjacency LSAs and prefix LSAs, two different types of LSAs, to provide routing information. As the name implies, Adjacency LSAs act as broadcast messages to each active link an NDN router has with its neighbors. The Prefix LSAs used to advertise the available content as prefix/name on its node. To detect link and remote NLSR process failures, NLSR periodically sends ”Info” Interests to each neighbor node. When information of interest expires, NLSR tries to send it. In case of interest wanes, a few times at brief intervals. The neighbor is deemed down if there is no answer from neighbors or neighbors throughout this time.

NLSR offered effective update propagation, Integrated updates authentication, and native multipath forwarding functionality. NLSR mechanism uses a decentralized routing approach, so all routers are involved in path calculation. This mechanism can lead to higher routing overhead.

3.1.2. OSPFN

Open Shortest Path First for Named Data (OSPFN) (Wang & Hoque, 2012) inherits IP properties focused on Prefix propagation and routing. In order to provide extensibility for future use, OSPF supports Opaque LSA (OLSA). OLSA consisting a standard LSA header and a set of data fields that other applications can expand. OLSA is distributed throughout the network by OSPF. Network topology is unknown to OSPF nodes, and forwarding decisions are based on local perceptions of received area code announcements. When a node receives the same Display of multiple interfaces, they are printed interface per hop distance to the producer. There are still numerous restrictions on OSPFN, such as using IP addresses as NDN Router IDs and using Tunnel Encapsulation to go via TCP/IP networks, which limit their NDN advantages.

3.2. Routing Centralized

In NDN, although it supports forwarding and load balancing (Saxena et al., 2016), packet interest forwarding is usually only controlled by consumers or intermediate nodes, which makes it difficult to achieve global optimization. Global optimization here is the process of managing the entire NDN traffic so that congestion does not occur during the process of packet delivery as well as limit the increase in network performance. Therefore, information about the entire network is required to achieve optimal global display. This is what makes the introduction of controls Centralized network vital to make the entire network traffic in NDN more reasonably distributed. To design, manage, and control technology can be used Software Defined Network (SDN) architecture. SDN is a new technology network development paradigm to solve problems in network mechanisms. When the control and data planes are separated, SDN can execute centralized control (Q. Y. Zhang et al., 2018). The control-plane functions to handle the determination of forwarding traffic, while the data plane functions to move data between a single endpoint and another single endpoint. Several algorithms use SDN to improve the routing performance on NDN; meanwhile, the others only use the controller concept from SDN instead of the SDN itself. (Alhowaidi et al., 2018; Amadeo et al., 2020; Aubry et al., 2017; CHA et al., 2016; Engineering, 2018; Kalafatidis et al., n.d., 2022; Kalghoum, 2019; Kalghoum & Gammar, 2017; Liu, 2016; Mahmood et al., 2018; Syrivelis et al., 2012; Tariq & Rehman, 2020; Torres et al., 2012, 2017, 2019; van Adrichem & Kuipers, 2015; Q. Zhang et al., 2019) are considered as Routing Centralized.

3.2.1. CRoS-NDN

The CRoS-NDN (Torres et al., 2012, 2019; Vitor et al., 2017) utilizes the SDN paradigm, namely separating the control plane and data plane. The first dedicated node is combined (controller). So this scheme removes the router’s reliance on the IP address and only forwards identified data packets. In CRoS, there are two schemes, namely the bootstrap phase and the name data routing phase. During the bootstrap phase, the router finds a controller to register. The controller gets the topology and calculates all routers. After this phase, The controller has access to and is familiar with every router on the network.

The bootstrap phase has three sub-protocols: hello Protocols, Controller Discovery, and Router Registration. The next phase is the Routing phase; after the bootstrap phase, all routes can route to the controller. The controller has the topology and calculates the routes between all routers. The producer must register their prefix in the controller, and every router that does not know the route in their FIB must request the route to the controller. This plan ensures access to the necessary content and localization. This paper analysis shows CRoS-NDN can reduce routing overhead by limiting network interest flooding. CRoS-NDN also increases the efficiency of content mobility producers. However, this method did not consider caching.

3.2.2. SDN-Enabled control for Multipath Forwarding (S-MP)

The proposed S-MP (Alhowaidi et al., 2018) utilizes a controller in the SDN paradigm. Whenever a data transfer request is made, The SDN controller determines the per-router pipeline, computes the forwarding strategy, and configures many routers. In addition, the SDN controller is responsible for managing the status of the CS and off-path router (producer). The router will transmit a REST POST to the controller each time the CS state changes, giving the controller access to the most recent CS state information. While their strategy improves throughput over the current implementation of forwarding and routing, only the CS condition in each router is used to determine the best interest pipeline (on-path or off-path).

3.2.3. Name-based forwarding rules in Software-Defined NDN

This method proposed by Marica et al. (Amadeo et al., 2020) uses a modified FIB design to integrate with SDN using OpenFlow. The modified FIB design contains prefixes and actions. The action either forwards the interest to the output port or contacts the controller to retrieve the path for the requested interest. The second action (contact controller) is carried out by utilizing PACKET IN, the traditional OpenFlow message to request flow rule if the requested interest fails to match during the CS/PIT/FIB matching procedure. The Content Information Base (CIB) and the Network Information Base (NIB) keep track of the information in each node, the two primary data structures the controller maintains for routing purposes. Upon receiving PACKET IN, the controller will access CIB to get the location of the content and access NIB to get the path. With OpenFlow FLOW MOD, the controller inserts forwarding rules into the FIB of the path.

3.2.4. SDN-Based NDN over Wireless Mesh Networks

In this paper, Kalafatidis et al. (Kalafatidis et al., n.d.) use a real-time monitoring system to support the controller. In order to determine the dynamic best-routing strategy and create NDN configuration based on the chosen routes, the controller receives all changes from network nodes. For each destination in WMN, the controller determines the appropriate route depending on network performance and reliability. The proposed method contains two entities: (i) Controller and (ii) Network Nodes. The three primary goals of the controller entities are to build NDN routes, choose the optimum route, and gather data on network conditions to generate a global view of the entire network. The developers of FIB entries and per-hop NDN faces are included in the NDN route establishment. The WMN routing protocol is used for the network monitoring system, in this case, the B.A.T.M.A.N. routing protocol. In this system, B.A.T.M.A.N is only used for gathering/monitoring network status and changes; the controller itself generates all packet delivery systems (interest and data packet). The network node entities divide into the NDN plane and WMN plane. NDN planes have a role as content delivery systems. The controller node initiates face construction and prefix registration. The WMN plane has three main objectives: dynamic topology detection, receiving the required network data and updating the network state to the controller. The capabilities are enabled using the WMN routing protocol, as stated before. B.A.T.M.A.N. routing protocol collects the neighboring nodes and next hops on the best path in every node.

3.2.5. Logically-Centralized SDN-Based NDN Strategies for Wireless Mesh Smart-City Networks

This paper (Kalafatidis et al., 2022) adopts a decision-making strategy concerning the dynamic WMN situations. Dynamic WMN networks with frequently changing topologies require adaptive routing protocols. To overcome this, implementing an SDN system with a reactive or proactive approach can provide a prompt response to topology changes in dynamic WMN systems. A reactive NDN path selection technique that lines up NDN routes with the dynamic routing options of the WMN protocol. In order to set up the NDN nodes, this method relies on distributed decision-making data (each node selects the best route among its neighbors for each destination). This data is obtained via the SDN Controller. Their reactive approach are previously discussed on paper (Kalafatidis et al., n.d.). A proactive technique for choosing an NDN path based on analysis of the RSSI and delays of the available wireless lines. To choose the best NDN path, the inventors of this approach collect historical network monitoring data and categorize its performance. This paper also summarizes the benefits of each technique by considering a variety of use cases and wireless communication scenarios. Because real-time centralized management increases network overhead, a reactive strategy is preferred in wireless communications with frequent network changes. In contrast to reactive strategy, proactive strategy can select the optimal route, achieving both high performance and high reliability. Proactive solutions are suitable for implementation in more stable wireless communications, such as smart city networks.

3.2.6. SDAR

SDAR (Liu, 2016) is designed for intra-domain routing, which can manage routers across the network on NDN software. This scheme also uses adaptive forwarding features and centralized control management at SDN (Software Defined Network). SDAR consists of two types of components; controllers and nodes. Nodes are defined as NDN routers or end systems. In the SDAR communication mode, the NDN router is responsible for registering directly linked links/nodes, name prefixes, and packet forwarding to the destination. In this communication model, the approach method is carried out, where each node communicates with the controller to instruct it to deliver requests for particular data. The controller will get a request with an empty data packet to satisfy the interest. As soon as the controller receives the notification, it will receive a request with an empty data packet to satisfy the interest requests. Even though this SDAR creates an additional traffic load, it will be more efficient if the network does not experience link changes and topology updates. Thus, The network and controllers' traffic overhead can be decreased by multi-lane routing in SDAR. This mechanism will increase the response speed of the traffic diversion.

3.2.7. FCR-NS

FCR-NS (Kalghoum, 2019) is a technique that uses a bloom filter structure and a novel cache replacement strategy to gauge the switch's popularity of local data and speed up forwarding. As a result, FCR-NS separates the control and data planes due to its SDN network foundation. This technique creates a novel SDN architecture with a bloom filter to improve the performance of the NDN controller network and an SDN that can administer the NDN network from a central location. Moreover, the SDN separation of the data and control fields reduces network overhead and bandwidth consumption. In addition, the Bloom Filter (BF) is a relatively compact probabilistic data structure that enables quick verification of an element's membership in a group of data and quick name lookup NDN tables. This approach reduces data retrieval time and messages overhead. While compared to other NDN solutions, it uses a lot of RAM.

3.2.8. SRSC

SRSC (Aubry et al., 2017) is an SDN-based routing forwarding scheme for NDN. SRSC uses the SDN paradigm to separate controllers and nodes, where nodes function as forwarders and controllers function to seek routing decisions. SRSC relies on two

distinct phases to achieve its goals: Bootstrap and Forwarding. Bootstrap aims to pull information from a node to its controller by selecting the fastest route to the desired location, and the controller learns the shape of the network topology. The second phase is forwarding. A forward interest must be forwarded to the closest Content Store once the controller asks for the rules, The FIB node is updated with the rules, and the interest is transmitted. The goal of this SRSC is not to lower traffic overhead, but to increasing the Hit Cache on the network. The probability of content being found in the Hit Cache on the network, which is the probability of content being found in the cache as determined by Cache Hit results, reaches 47%, meaning that almost half of the requests have discovered Store content Node content, rather than on the original server. Also, this SRSC routing system can forward requests to the closest node so that it does not need to be on the path to the origin server; this can lessen server load and enhance Cache Hit performance.

3.2.9. Controller based Broadcast storm Avoidance forwarding Mechanism (CBAM)

(Tariq & Rehman, 2020) to resolve the Broadcast storm problem. CBAM controller maintains three tables: (i) Global Information Table (GIT) contains information about nodes including nodes id, neighbor nodes, and content it contains, (ii) Routing Information Table (RIT) to compute the shortest path with minimum count available, and (iii) Forwarding Information Base (FIB). Node elements maintain Content Store (CS), Pending Interest Table (PIT), and Local Information Table (LIT) containing information about the controller associated with the network range. CBAM utilizes four types of packets C id pkt, CBAM I pkt, CBAM D pkt, and L info pkt in the process. CBAM broadcasts C id pkt to start controller registration in every nearby node. Nearby node records controller ID in their LIT. Every node broadcasts its L info pkt to get information from the next hop node and record it in its LIT. Next, the node sends the information using L info pkt to the controller. When the controller receives L info pkt, it updates GIT. The actual packet delivery mechanism uses modified interest packet CBAM I pkt and modified data packet CBAM D pkt. Upon receiving CBAM I pkt, the controller computes the optimal path using the Dijkstra algorithm. The controller returns the computed optimal path to the consumer in CBAM I pkt. After knowing the optimal path, the consumer will return the interest packet to the producer and the information.

3.2.10. Towards new Information Centric Networking strategy based on Software Defined network

This paper proposed by Anwar et al. (Kalghoum & Gammar, 2017) proposed a Named Data Networking architecture (NDNS) based on Software Defined Networks that offers an NDN network routing mechanism based solely on NDN messages. In order to develop a routing strategy controlled by the controller and switch nodes, respectively, this work separates the control plane and data plane. This strategy can address significant NDN problems, like scalability, bandwidth usage, and increased latency. Moreover, it does away with TCP/IP architecture. The NDNS consists of four steps in its application. The first is topology management, the second is data management, the third is to produce a routing base, and the last is to produce a forwarding base. This algorithm's first phase involves exchanging LLDP packets across various network elements to gather topology data (switches, nodes, and controllers). The connected switches receive a data prefix name of each node in the second stage. The controller creates a local Routing Information Base (RIB) for each switch in the NDNS architecture in step three by using the Global Management Information Base (GMIB) table. The controller creates a unique Flof FIB for each switch in this architecture's last stage utilizing the Global Data Information Base (GDIB) table and the Routing Information Base (RIB) table. All data name prefixes and subsequent data fetching interfaces are listed in this table. In contrast to other well-liked solutions like SRSC and CRoS-NDN, the experimental results of this research enable quick and effective topology discovery without overtaxing the system.

3.2.11. Pursuing a Software Defined Information Centric Network

The work of Syrivelis et al. (Syrivelis et al., 2012) was one of the pioneers in fusing SDN with ICN. ICN architecture, as mentioned in The three primary network functions supported by each node, are rendezvous, topology maintenance, and forwarding. Two centralized nodes, the topology manager and the rendezvous server, perform the network topology and rendezvous management duties. The topology manager uses a bloom filter-based coding method to create source routing forwarding identifiers. The SDN controller has control over the forwarding network. The centralized method of packet forwarding creates serious scaling problems.

3.2.12. Named Data Networking over a Software-Defined Network Using Fixed Size

The architecture proposed in this paper (CHA et al., 2016) operates in two phases: interest forwarding and data packet forwarding. In the forwarding of interest in outline in the initial step, the consumer expresses interest in receiving the desired material. Then, based on the interest fields for the destination, source IP, and protocol, the OF (open flow) switch that has received the interest searches for a flow table. If no matching stream entries are detected, the OF switch notifies the NDN controller of an incoming packet. The SDN controller transfers the NDN protocol handler after determining that the packet relayed in the incoming packet is of type NDN. The next NDN node that should contain the requested content is determined by the content name-based routing module. Prior to the Data packet stream reaching the OpenFlow switch, It determines the Data Packet Flow Path to determine the interest-related response. A setup OpenFlow switch will reevaluate the flow table and pass the interest based on the flow entrance action. Until the NDN node finally receives the interest. The NDN node returns a Data packet if the requested material is located. If not, the NDN node notifies other NDN nodes that are anticipated to contain the necessary data of its interest. While the data forwarding operation is more straightforward than the interest forwarding operation because the data flow path is already configured on the OpenFlow switch. Where the NDN Node with the requested content responds with a Data Packet, the source IP address and the destination IP address of the Data packet are, respectively, the IP addresses of interest. Data packets are transmitted via pre-configured paths to content request nodes. The data packet is delivered to the NDN node that made the content request.

3.2.13. MTO : Multicast-based Traffic Optimization for Information Centric Networks

In this paper, (Q. Zhang et al., 2019) The authors emphasize effective traffic optimization for ICN through multicasting and provide an SDN-aided strategy. Also, the author suggests a multicast tree built in tunnels to aggregate content transmissions. In order to achieve optimal traffic distribution, this MTO balances link consumption with ICN multicast tunnels that are globally optimized. The essence of this system is that whenever a new content request arrives at the SDN controller, Using the content name, the Tunnel Information Base is searched by the Policy Management module for the tunnel request. In order to determine the shortest way to the source based on user needs, matched content requests are delivered to Tunnel Computing and treated as new members of the multicast tunnel. The Tunnel Information Base is then updated with the calculated information and tunnel names. If the tunnel is estimated already exists in the Tunnel Information Base, the content name is then added to the appropriate record. In addition, the Policy Management module monitors network statistics and performs a tunnel update process by recalculating all or part of the tunnel. It then updates the Tunnel Information Base according to a predefined optimization policy. All estimated pathways are transmitted via Tunnel Configuration to the Configuration Interpreter. The findings demonstrate that MTO accomplishes the load-balancing traffic optimization goal and steadily raises overall network use. MTO also lowers the price of multicast trees

3.2.14. NDNFlow

NDNFlow proposed by (van Adrichem & Kuipers, 2015) adds a second layer of application-specifics to OpenFlow. Parallel to the current OpenFlow, NDNFlow employs a different communication channel and controller module. With this mechanism, NDNFlow can prevent interdependencies on protocol versions and provide better deployment and future maintenance. NDNFlow using ICN-enabled switch and install ICN module to OpenFlow controller. The OpenFlow Controller must be able to interface with the ICN module.

ICN-enabled switches establish parallel communication channels with their existing OpenFlow communication channels. ICN channels request unreserved flows, announce ICN capabilities, and make information available. The ICN modules then determine the route for ICN flows and set up the legacy IP switch and the ICN-enabled switch to support ICN flows. NDNFlow is easier to extend and less difficult to install.

3.2.15. NDN Routing Strategy Based on SDN Centralized Topology Update

B.-B. Yin et al. (Engineering, 2018) use a controller for the topology update mechanism and keep the route calculation mechanism in the NDN node. In contrast, the current implementation of SDN on NDN counts heavily on the controller for topology update and route calculation. The first packet of the content request will be severely delayed due to the existing centralized routing solution. This method can ensure that the topology is updated synchronously and quickly thanks to the controller and shorten the duration of the initial content packet. The SDN controller’s load is lessened on the controller side. Furthermore, the distributed routing calculations placed in the NDN node can optimize the final path selection by using the Forwarding Information Base table in the NDN node. Their implementation uses the OVS module for basic SDN network-related service and uses NDNFlow communication to establish a connection with the NDN agent on the SDN controller side. The goal is to achieve a lower delay of the first packet, reduce the load SDN controller, and achieve efficient load balancing over the entire network.

3.2.16. Efficient Caching Through Stateful SDN in Named Data Networking

Stateful S/N-DN solutions (Mahmood et al., 2018) combine SDN and NDN technologies. The authors add a stateful SDN switch in this paper that allows the NDN approach. The results also show that it can make decisions regarding NDN inside the switch, not engaging with the SDN controller during runtime. This stateful S/N-DN is based on a single-domain SDN. Cached switches, SDN controllers, servers, and hosts make up the SDN domain itself. The cached switch connects to the server. The goal is to generate data and have complete data on the requested content. As mentioned earlier, Requests for content are first attempted to be fulfilled by this cached switch using its cache; If that does not work, the server receives the request. The server sends the requested content to the host when the switch answers and a copy of each is stored in the switch cache in case more requests are made. In this case, the server serves as an NDN generator, Whereas the host acts as an NDN consumer, responding to it with interest packets. To evaluate the effectiveness of these nodes, the author constructed stateful S/N-DN nodes using Open State and system-tested them with Ryu SDN controllers, Mininet, and Open State. After the open-state switch’s first configuration, the controller is not required in this stateful solution, which results in low latency and minimal overhead.

Table 1: Summary of NDN Forwarding and Routing Mechanism

Group	Section	Main Mechanism
Fixed Forwarding	2.1.1	Select lowest possible cost in FIB.
Fixed Forwarding	2.1.2, 2.1.3	Broadcast to get the best route.
Adaptive Forwarding	2.2.1, 2.2.2, 2.2.6, 2.2.9	Use probing mechanism to get the network quality and select face according to network quality.
Adaptive Forwarding	2.2.3-2.2.5	Using probability method such as Maximizing Deviation Based Probabilistic, Stochastic, Partially and Observable Markov Decision Proccess to select the best face.
Adaptive Forwarding	2.2.14	Using GeoTag as forwarding hint in LEO satellite networks.
Adaptive Forwarding	2.2.7, 2.2.10-2.2.13, 2.2.15-2.2.20	Using Artificial Intelligent and its derivative to improve face selection.
Adaptive Forwarding	2.2.8.	Using Luby Transform (LT) coded content chunks to reduce interest request repetition on chunked content.

Decentralized Routing	3.1.1, 3.12.	Using conventional routing mechanism such as OSPF and Link State Routing.
Centralized Routing	3.2.3, 3.2.7, 3.2.11-3.2.13, 3.2.16.	Using Modified SDN controller to support NDN Networks.
Centralized Routing	3.2.10, 3.2.14.	Implements SDN concept to separate Control Plane and Data Plane of the NDN Router.
Centralized Routing	3.2.1, 3.2.2, 3.2.4-3.2.6, 3.2.8, 3.2.9, 3.2.15.	These methods only implement the centralization concept of SDN on NDN networks using NDN packets.

4. Open Challenge

The existing mechanism discussed in this paper is summarized in Table 1. Next, we will discuss the challenge and problems of the current implementation. First, fixed forwarding has are relatively simple and easy to implement. The downside of fixed forwarding is the lack of adaptability and highly depends on the routing mechanism. Adaptive forwarding, on the other hand, can quickly adapt to network conditions and is less dependent on the routing mechanism thanks to its quality measurement. However, adaptive forwarding is more complex and harder to implement than fixed forwarding. Furthermore, adaptive forwarding such as (Lehman et al., 2016) mechanism actively probes the network can add some forwarding overhead. A forwarding mechanism with high adaptability and passively probe the network are preferred to get the best of two worlds.

Centralized routing can increase the Quality of Service and more robust adaptability, thanks to the global view topology on the controller side. Furthermore, the limited FIB problem on each router can be eliminated with the help of an enormous FIB on the controller side. The downside of the centralized system is vulnerable to single-point failure of the controller and can introduce some routing overhead if the controller is included in network topology. In addition, most of this research focuses on something other than caching. Caching is essential to reduce server load considering the content is located in the nearest router's content store. Furthermore, several centralized routing methods use another protocol such as SDN itself (Amadeo et al., 2020; CHA et al., 2016; Kalghoum, 2019; Mahmood et al., 2018; Syrivelis et al., 2012; Q. Zhang et al., 2019) or different layer routing protocol such as B.A.T.M.A.N. (Kalafatidis et al., n.d., 2022) can leads to increase the complexity of the implementation of the centralized routing method.

Decentralized routing is more resist single-point failure considering the routing table is installed on every router in the network, and the computing process is distributed to every router. Even though decentralized routing is harder to search optimal routing with CS installed, considering the router does not have the mechanism to check available content on another CS. A routing mechanism with global view topology, high adaptability, and resistance to single-point failure are preferred to get the best of two worlds.

The combination of routing and forwarding strategies must be considered to optimize the performance of both strategies. It is considered that centralized routing is used for macro control over the delivery path and combined with adaptive forwarding that actively measures network performance for finer control over the delivery path. Even though the combination mentioned above needs higher computational power than the combination of fixed and decentralized forwarding, the combination mentioned above gives higher adaptability and better overall performance.

Implementing Artificial Intelligent (AI) and its derivatives can also improve the routing and forwarding strategy. As shown in many papers, the proposed strategy with AI implementation outperform the existing strategy. Although the implementation of AI can improve routing and forwarding strategy, the computational load must be considered to ensure the availability and affordability of the device.

Considering the abovementioned challenges, an algorithm to overcome single-point failure on Routing Centralized can improve the mechanism's reliability. A new centralized routing protocol using only NDN packets can lower the implementation complexity. Passive probing mechanism also needed on the adaptive forwarding mechanism to overcome the forwarding overhead problems. The implementation of AI in the controller-side can reduce the need for availability and affordability of the router, considering path calculating and AI computation happens on the controller side.

5. Conclusion

This paper discusses the importance of routing and forwarding mechanisms for NDN systems. Several proposed methods were also discussed and classified into four different groups. We divide forwarding into two specific groups: (i) Fixed Forwarding, which highly depends on the routing mechanism and does not have a monitoring system or high adaptability. (ii) Adaptive Forwarding, which has a monitoring system or high adaptability but has higher overhead for measurement mechanism. We also divide routing into two groups: (i) Decentralized Routing, every router has its routing table and calculates the path based on received information from another router. (ii) Centralized Routing has a particular node called a controller to calculate the path. It has a global topology view, so centralized routing can easily select the best path according to network status.

Packet delivery in the NDN system highly depends on the forwarding mechanism. It does not mean that forwarding can operate alone without the help of routing. Routing is used for path selection and to register available faces and content to FIB. The combination of routing and forwarding strategy must be considered to optimize NDN system performance. Routing can be macro control, whereas forwarding can offer finer control for the packet delivery mechanism. Existing routing and forwarding in the IP system must be modified, or even a new routing and forwarding mechanism for NDN. Considering the different approaches of the network paradigm.

References

Afanasyev, Alex, Burke, J., Refaei, T., Wang, L., Zhang, B., & Zhang, L. (2019). A Brief Introduction to Named Data Networking. *Proceedings - IEEE Military Communications Conference MILCOM*, *2019-Octob*, 605–611. https://doi.org/10.1109/MILCOM.2018.8599682

Afanasyev, Alexander, Shi, J., Zhang, B., Zhang, L., Moiseenko, I., Yu, Y., Shang, W., Huang, Y., Abraham, J. P., Dibenedetto, S., Fan, C., Pesavento, D., Grassi, G., Pau, G., Zhang, H., Song, T., Abraham, H. Ben, Crowley, P., Amin, S. O., … Wang, L. (2015). *NFD Developer ' s Guide*. *9*, 1–56.

Akinwande, O. (2018). Interest Forwarding in Named Data Networking Using Reinforcement Learning. *Sensors*, *18*(10), 3354. https://doi.org/10.3390/s18103354

Alhowaidi, M., Nadig, D., Ramamurthy, B., Bockelman, B., & Swanson, D. (2018). Multipath Forwarding Strategies and SDN Control for Named Data Networking. *2018 IEEE International Conference on Advanced Networks and Telecommunications Systems (ANTS)*, 1–6. https://doi.org/10.1109/ANTS.2018.8710068

Amadeo, M., Campolo, C., Ruggeri, G., Molinaro, A., & Iera, A. (2020). Understanding Name-based Forwarding Rules in Software-Defined Named Data Networking. *ICC 2020 - 2020 IEEE International Conference on Communications (ICC)*, 1–6. https://doi.org/10.1109/ICC40277.2020.9149266

Ariefianto, W. T., & Syambas, N. R. (2017). Routing in NDN network: A survey and future perspectives. *2017 11th International Conference on Telecommunication Systems Services and Applications (TSSA)*, 1–6. https://doi.org/10.1109/TSSA.2017.8272942

Aubry, E., Silverston, T., & Chrisment, I. (2017). Implementation and evaluation of a controller-based forwarding scheme for NDN. *Proceedings - International Conference on Advanced Information Networking and Applications, AINA*, 144–151. https://doi.org/10.1109/AINA.2017.83

Bastos, I. V., & Moraes, I. M. (2016). A forwarding strategy based on reinforcement learning for Content-Centric Networking. *2016 7th International Conference on the Network of the Future (NOF)*, 1–5. https://doi.org/10.1109/NOF.2016.7810121

CHA, J.-H., HAN, Y.-H., & MIN, S.-G. (2016). Named Data Networking over a Software-Defined Network Using Fixed-Size Content Names. *IEICE Transactions on Communications*, *E99.B*(7), 1455–1463. https://doi.org/10.1587/transcom.2015EBP3464

de Sena, Y. A. B. L., Dias, K. L., & Zanchettin, C. (2020). DQN-AF: Deep Q-Network based Adaptive Forwarding Strategy for Named Data Networking. *2020 IEEE Latin-American Conference on Communications (LATINCOM)*, 1–6. https://doi.org/10.1109/LATINCOM50620.2020.9282301

Engineering, C. (2018). *基于 SDN 拓扑集中更新的 NDN 路由策略*. *61671086*, 1–8.

Fu, B., Qian, L., Zhu, Y., & Wang, L. (2017). Reinforcement learning-based algorithm for efficient and adaptive forwarding in named data networking. *2017 IEEE/CIC International Conference on Communications in China (ICCC)*, 1–6. https://doi.org/10.1109/ICCChina.2017.8330354

Gong, L., Wang, J., Zhang, X., & Lei, K. (2016). Intelligent Forwarding Strategy Based on Online Machine Learning in Named Data Networking. *2016 IEEE Trustcom/BigDataSE/ISPA*, 1288–1294. https://doi.org/10.1109/TrustCom.2016.0206

Hao, B., Wang, G., Zhang, M., Zhu, J., Xing, L., & Wu, Q. (2021). Stochastic Adaptive Forwarding Strategy Based on Deep Reinforcement Learning for Secure Mobile Video Communications in NDN. *Security and Communication Networks*, *2021*, 1–13. https://doi.org/10.1155/2021/6630717

Hoque, A. K. M. M., Amin, S. O., Alyyan, A., Zhang, B., Zhang, L., & Wang, L. (2013). NLSR: Named-data Link State. *Proceedings of the 3rd ACM SIGCOMM Workshop on Information-Centric Networking - ICN '13*, 15.

Iglesias-Sanuy, P., López-Ardao, J. C., Rodríguez-Pérez, M., Herrería-Alonso, S., Suárez-González, A., & Rodríguez-Rubio, R. F. (2022). An Efficient Location-Based Forwarding Strategy for Named Data Networking and LEO Satellite Communications. *Future Internet*, *14*(10), 285.

https://doi.org/10.3390/fi14100285
Jacobson, V., Smetters, D. K., Thornton, J. D., Plass, M., Briggs, N., & Braynard, R. (2012). Networking named content. *Communications of the ACM*, *55*(1), 117–124. https://doi.org/10.1145/2063176.2063204
Kalafatidis, S., Demiroglou, V., Mamatas, L., & Tsaoussidis, V. (n.d.). *Experimenting with an SDN-Based NDN Deployment over Wireless Mesh Networks*.
Kalafatidis, S., Skaperas, S., Demiroglou, V., Mamatas, L., & Tsaoussidis, V. (2022). Logically-Centralized SDN-Based NDN Strategies for Wireless Mesh Smart-City Networks. *Future Internet*, *15*(1), 19. https://doi.org/10.3390/fi15010019
Kalghoum, A. (2019). *FCR-NS : a novel caching and forwarding strategy for Named Data Networking based on Software Defined Networking*. *0123456789*.
Kalghoum, A., & Gammar, S. M. (2017). Towards New Information Centric Networking Strategy Based on Software Defined Networking. *2017 IEEE Wireless Communications and Networking Conference (WCNC)*, 1–6. https://doi.org/10.1109/WCNC.2017.7925536
Kerrouche, A., Senouci, M. R., & Mellouk, A. (2016). QoS-FS: A new forwarding strategy with QoS for routing in Named Data Networking. *2016 IEEE International Conference on Communications (ICC)*, 1–7. https://doi.org/10.1109/ICC.2016.7511378
Lehman, V., Gawande, A., Zhang, B., Zhang, L., Aldecoa, R., Krioukov, D., & Wang, L. (2016). An experimental investigation of hyperbolic routing with a smart forwarding plane in NDN. *2016 IEEE/ACM 24th International Symposium on Quality of Service (IWQoS)*, 1–10. https://doi.org/10.1109/IWQoS.2016.7590394
Lei, K., Yuan, J., & Wang, J. (2015). MDPF: An NDN Probabilistic Forwarding Strategy Based on Maximizing Deviation Method. *2015 IEEE Global Communications Conference (GLOBECOM)*, 1–7. https://doi.org/10.1109/GLOCOM.2015.7417024
Liu, Y. (2016). *SDAR : Software Defined Intra-Domain Routing in Named Data Networks*. *3*, 158–161.
Lv, J., Tan, X., Jin, Y., & Zhu, J. (2018). DRL-based Forwarding Strategy in Named Data Networking. *2018 37th Chinese Control Conference (CCC)*, 6493–6498. https://doi.org/10.23919/ChiCC.2018.8483989
Mahmood, A., Casetti, C., Chiasserini, C. F., Giaccone, P., & Härri, J. (2018). Efficient caching through stateful SDN in named data networking. *Transactions on Emerging Telecommunications Technologies*, *29*(1), e3271. https://doi.org/10.1002/ett.3271
Mekinda, L., & Muscariello, L. (2016). Supervised Machine Learning-Based Routing for Named Data Networking. *2016 IEEE Global Communications Conference (GLOBECOM)*, 1–6. https://doi.org/10.1109/GLOCOM.2016.7842307
Posch, D., Rainer, B., & Hellwagner, H. (2017). SAF: Stochastic Adaptive Forwarding in Named Data Networking. *IEEE/ACM Transactions on Networking*, *25*(2), 1089–1102. https://doi.org/10.1109/TNET.2016.2614710
Ramadha, A. A., Yovita, L. V., & Wibowo, T. A. (2022). Design and Implementation Named Data Networking-Based Video Streaming System. *2022 5th International Conference on Information and Communications Technology (ICOIACT)*, 66–70. https://doi.org/10.1109/ICOIACT55506.2022.9972215
Ren, Y., Li, Z., Li, J., Qin, Y., Wu, H., & Zhou, X. (2019). DMF: A Dynamic Multi-Path Forwarding Strategy for Information Centric Networks. *2019 IEEE 21st International Conference on High Performance Computing and Communications; IEEE 17th International Conference on Smart City; IEEE 5th International Conference on Data Science and Systems (HPCC/SmartCity/DSS)*, 2495–2501. https://doi.org/10.1109/HPCC/SmartCity/DSS.2019.00349
Rowshanrad, S., Parsaei, M. R., & Keshtgari, M. (2016). IMPLEMENTING NDN USING SDN: A REVIEW ON METHODS AND APPLICATIONS. *IIUM Engineering Journal*, *17*(2), 11–20. https://doi.org/10.31436/iiumej.v17i2.590
Ryu, S., Joe, I., & Kim, W. (2021). Intelligent Forwarding Strategy for Congestion Control Using Q-Learning and LSTM in Named Data Networking. *Mobile Information Systems*, *2021*, 1–10. https://doi.org/10.1155/2021/5595260
Saxena, D., Raychoudhury, V., Suri, N., Becker, C., & Cao, J. (2016). Named Data Networking: A survey. *Computer Science Review*, *19*, 15–55. https://doi.org/10.1016/j.cosrev.2016.01.001
Sembati, Y., Naja, N., & Jamali, A. (2022). A global review of routing mechanisms in the named data network. *ITM Web of Conferences*, *43*, 01006. https://doi.org/10.1051/itmconf/20224301006
Syrivelis, D., Parisis, G., Trossen, D., Flegkas, P., Sourlas, V., Korakis, T., & Tassiulas, L. (2012). Pursuing a Software Defined Information-centric Network. *2012 European Workshop on Software Defined Networking*, 103–108. https://doi.org/10.1109/EWSDN.2012.20
Tariq, A., & Rehman, R. A. (2020). CBAM: A Controller based Broadcast Storm Avoidance Mechanism in SDN based NDN-IoTs. *2020 3rd International Conference on Advancements in Computational Sciences (ICACS)*, 1–6. https://doi.org/10.1109/ICACS47775.2020.9055939
Torres, J. V., Alvarenga, I. D., Boutaba, R., & Duarte, O. C. M. B. (2017). An autonomous and efficient controller-based routing scheme for networking Named-Data mobility. *Computer Communications*, *103*, 94–103. https://doi.org/10.1016/j.comcom.2017.02.001
Torres, J. V., Alvarenga, I. D., Boutaba, R., & Duarte, O. C. M. B. (2019). Evaluating CRoS-NDN: a comparative performance analysis of a controller-based routing scheme for named-data networking. *Journal of Internet Services and Applications*, *10*(1). https://doi.org/10.1186/s13174-019-0119-6
Torres, J. V., Ferraz, H. G., Carlos, O., & Duarte, M. B. (2012). *Controller-based Routing Scheme for Named Data Network*. 1–6.
van Adrichem, N. L. M., & Kuipers, F. A. (2015). NDNFlow: Software-defined Named Data Networking. *Proceedings of the 2015 1st IEEE Conference on Network Softwarization (NetSoft)*, 1–5. https://doi.org/10.1109/NETSOFT.2015.7116131
Vitor, J., Drummond, I., Boutaba, R., Carlos, O., & Duarte, M. B. (2017). *An autonomous and efficient controller-based routing scheme for networking Named-Data mobility*. *103*, 94–103. https://doi.org/10.1016/j.comcom.2017.02.001
Wang, L., & Hoque, A. K. M. M. (2012). *OSPFN : An OSPF Based Routing Protocol for Named Data Networking*. 1–15.
Yao, J., Yin, B., Tan, X., & Jiang, X. (2017). A POMDP framework for forwarding mechanism in named data networking. *Computer Networks*, *112*, 167–175. https://doi.org/10.1016/j.comnet.2016.11.005
Ye, Y., Lee, B., Flynn, R., Murray, N., & Qiao, Y. (2017). HLAF: Heterogeneous-Latency Adaptive Forwarding strategy for Peer-Assisted Video Streaming in NDN. *2017 IEEE Symposium on Computers and Communications (ISCC)*, 657–662. https://doi.org/10.1109/ISCC.2017.8024603
Yi, C., Abraham, J., Afanasyev, A., Wang, L., Zhang, B., & Zhang, L. (2014). On the role of routing in named data networking. *ICN 2014 - Proceedings of the 1st International Conference on Information-Centric Networking*, 27–36. https://doi.org/10.1145/2660129.2660140
Zhang, G., Li, H., Zhang, T., Li, D., & Xu, L. (2015). A multi-path forwarding strategy for content-centric networking. *2015 IEEE/CIC International Conference on Communications in China (ICCC)*, 1–6. https://doi.org/10.1109/ICCChina.2015.7448593
Zhang, L., Afanasyev, A., Burke, J., Jacobson, V., Claffy, K., Crowley, P., Papadopoulos, C., Wang, L., & Zhang, B. (2014). Named data networking. *ACM SIGCOMM Computer Communication Review*, *44*(3), 66–73. https://doi.org/10.1145/2656877.2656887
Zhang, M., Wang, X., Liu, T., Zhu, J., & Wu, Q. (2020). AFSndn: A novel adaptive forwarding strategy in named data networking based on Q-learning. *Peer-to-Peer Networking and Applications*, *13*(4), 1176–1184. https://doi.org/10.1007/s12083-019-00845-w
Zhang, Q., Wang, X., Lv, J., & Huang, M. (2019). MTO: Multicast-Based Traffic Optimization for Information Centric Networks. *2019 IEEE 25th International Conference on Parallel and Distributed Systems (ICPADS)*, 259–266. https://doi.org/10.1109/ICPADS47876.2019.00045
Zhang, Q. Y., Wang, X. W., Huang, M., Li, K. Q., & Das, S. K. (2018). Software Defined Networking Meets Information Centric Networking: A Survey.

IEEE Access, *6*, 39547–39563. https://doi.org/10.1109/ACCESS.2018.2855135
Zhang, Y., Bai, B., Xu, K., & Lei, K. (2018). IFS-RL. *Proceedings of the 2018 Workshop on Network Meets AI & ML - NetAI'18*, 54–59. https://doi.org/10.1145/3229543.3229547