A Comparative Study and Performance Modeling of Information-Centric Networking Architectures for the Future Internet

Qutaiba I. Ali

University of Mosul/Iraq

Abstract Information-Centric Networking (ICN) is a transformative communication paradigm that emphasizes content retrieval by name rather than host-based addressing, aiming to overcome limitations of the traditional IP-based Internet. This paper provides a comparative analysis of major ICN architectures—Content-Centric Networking (CCN), Named Data Networking (NDN), Data-Oriented Network Architecture (DONA), Network of Information (NetInf), and the Publish-Subscribe Internet Routing Paradigm (PURSUIT). Each architecture is evaluated in terms of naming, routing, name resolution, and caching mechanisms. Furthermore, we develop mathematical models and simulations to quantify performance metrics such as content retrieval latency, cache hit ratio, packet overhead, and average delay under varying network conditions. The results reveal key strengths and trade-offs among these architectures, offering valuable insights for their deployment in future content-centric and delay-tolerant networking scenarios.

Keywords: Content Centric Network CC, Named Data Network NDN, Data Oriented Network Architecture DONA, Named of Information NetInf, Publish Subscriber Internet Routing Paradigm PSIRP.

1. Introduction

In an Information Centric Network ICN system, a user fetches content by using its name, rather than by using an IP address. If a node (such as a router) located near the user contains the content or cache, the node transfers it directly to the user. This scheme enables quicker information services independent of the conditions in which the servers are placed, leading to more effective utilization of server and network resources. Realization of the new communication architecture (ICN) still requires a variety of research and development efforts, because its non-reliance on the sender/ receiver IP address scheme necessarily involves alternative communication technologies dissimilar to the current IP address-based technologies[1] (Fig. 1). The ICN concept was in the beginning proposed in TRIAD [2], then many schemes have been proposed architectures of ICN.

The main objectives of this paper are:

1. To introduce the concept of Information-Centric Networking (ICN) and its advantages over traditional IP-based communication models.

2. To provide a comparative overview of prominent ICN architectures, focusing on CCN/NDN, DONA, NetInf, and PURSUIT.
3. To analyze the differences in naming schemes, routing strategies, name resolution mechanisms, and caching policies across the ICN architectures.
4. To develop mathematical models for evaluating network performance, including metrics like latency, cache hit ratio, packet overhead, and average delay.
5. To perform simulations and generate performance results under various network scenarios and architectural configurations.

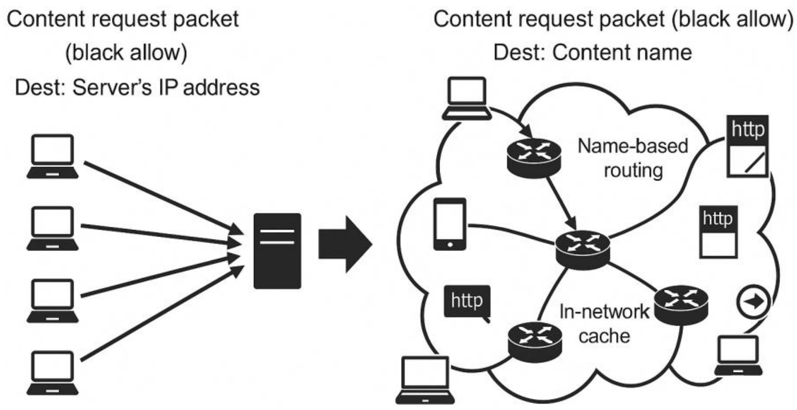

Fig. 1 Transition to Information Centric: from IP address-based to content name-based communication

2. Information Centric Network ICN Architectures

we give an overview of five ICN approaches in networking, namely, CCN, NDN, DONA, NetInf, and PURSUIT and discussed naming, Routing and Name resolution and caching in each approach (Fig. 2).

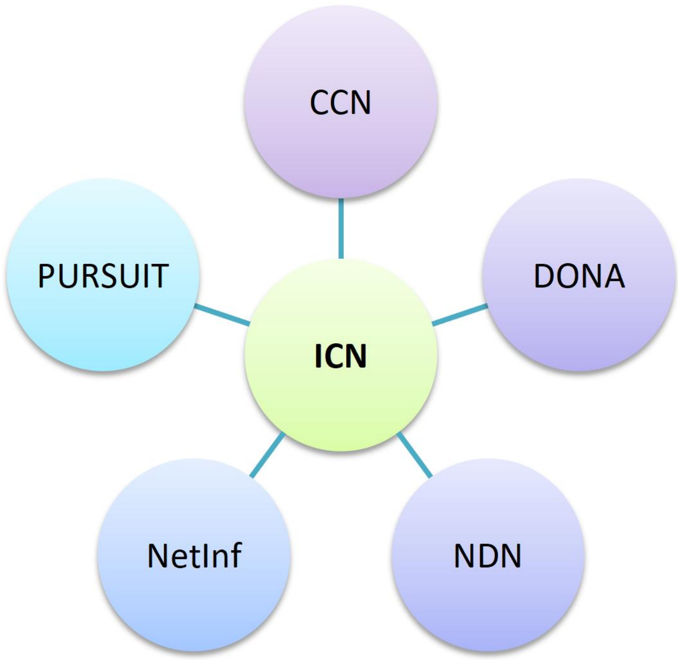

Fig. 2 ICN architectures

2.1 Content Centric Network CCN and Named Data Networking NDN

The content-centric network (CCN) architecture was originally proposed by Van Jacobson as a project initiated by Palo Alto Research Center (PARC). Jacobson first introduced it in his Google Tech Talk [3]. The main idea behind CCN is that the request (Interest) packets containing the desired content/data name are broadcast by a consumer node, and routing protocols are employed to distribute information about the location of content based on the name using the longest prefix matching. Routing aggregation is leveraged through a hierarchical naming scheme. The content provider, or any other network node with a copy of the requested content, routes the required content, along with additional authentication and data-integrity information, along the interests reverse route. Furthermore, caching on each path node is enabled depending on the caching policy of the node. An overview of the communication is illustrated in (Fig. 3). In an intermittent-connectivity scenario, this can speed up content retrieval because the content is replicated in the network [4].Named-data networking (NDN) is an enhanced version of the CCN architecture. Similar to CCN, NDN also follows the interest/data packet combination to obtain any particular data. There are, however, some architecture differences incorporated into NDN, which reduces the interest/data search time as well as the interest looping issue.

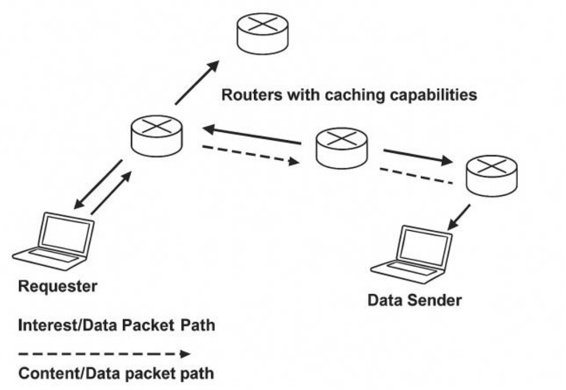

Fig. 3 CCN/NDN overview

2.1.1 Name Data Network NDN Naming

NDN adopts a hierarchical naming scheme, e.g., information may have the name/work/class101/presentation.pdf, where the sign "/" shows the hierarchy of the name component. The relationships and context of the data elements are easily represented in this hierarchical structure. In a typical CCN, each node consists of three data structures: a pending interest table (PIT), a content store (CS), and a forwarding information base (FIB). PIT contains a list of pending and satisfied interests. The entries include a content name, the interest-incoming interface, a NONCE value to identify the individual interest packet, and timers for PIT-entry management. CS provides a cache to store the content available at the node and content received from other nodes based on the caching policy of the node. FIB helps in routing the incoming interest to the next hop toward the content provider; it maintains name prefixes and outgoing interfaces for interest packets. In addition, a forwarding algorithm is used to provide a forwarding strategy, which uses these data structures.

2.1.2 Name Data Network NDN and Content Centric Network CCN Routing and Name resolution

In CCN/NDN, name-based routing is used to forward packets between the source and destination. The client/requester broadcasts interest packets for the required content in the network. This interest is forwarded to the name-prefix of the destination using longest-prefix matching at the FIB of each intermediate node. Each incoming interest‘s information is stored in the PIT; furthermore, multiple requests of the same content are aggregated together. When the requested content copy is found at a node, a data packet is sent back with the requested data on the reverse path toward the client. In addition, each node along the data path can cache a copy of the data.

2.1.3 Name Data Network NDN and Content Centric Network CCN Caching

Caching is one of the major advantages of the ICN architectures [5]. Caching the content in the CS of a node is analogous to the buffer memory in IP routers; however, the IP routers cannot reuse the data packet after forwarding. However, in NDN the storage of packets is possible at each NDN node, thus allowing the node to satisfy any future request for the particular data. In addition, as the content name does not contain any information of the user, thus making the users more secure.

2.2 Data Oriented Network Architecture DONA

In 2006, the data-oriented network architecture (DONA) project [6] at UC Berkeley proposed an ICN architecture. DONA’s architecture involves a redesigning of the current Internet naming, i.e., DNS names are replaced with flat, self-certifying names, and DNS name resolution is replaced with any cast name resolution process. Furthermore, these changes are incorporated above the IP layer, thus leveraging the lower layers of path discovery mechanisms. The architecture provides improved data retrieval as well as improved service by providing persistence, authentication [7, 8], and availability. In DONA, the source/content provider is responsible for publishing the content in the network. To serve data, the nodes must authorize with the resolution infrastructure. A route-by-name paradigm is used for name resolution. Now, instead of using DNS servers, DONA relies on the network entities called “resolution handlers” (RHs). The request (FIND) packets are forwarded through multiple RHs toward the node with a copy of the content as illustrated in (Fig. 4). The content/data can be acquired through two methods: (1) it is sent back through the same path the interest packet came in on with caching enabled on each encountered RH or (2) it can be sent back directly toward the consumer. The source also has the option to register their principals with the RH so that the request packets can be sent to them directly. However, the registrations must be renewed periodically. RH routes requests using a hierarchical approach to find the closest content provider. The any cast name resolution used in DONA provides support for the network middle boxes (e.g., firewalls, net address translators, proxies, etc.) by providing a separate mechanism for path discovery.

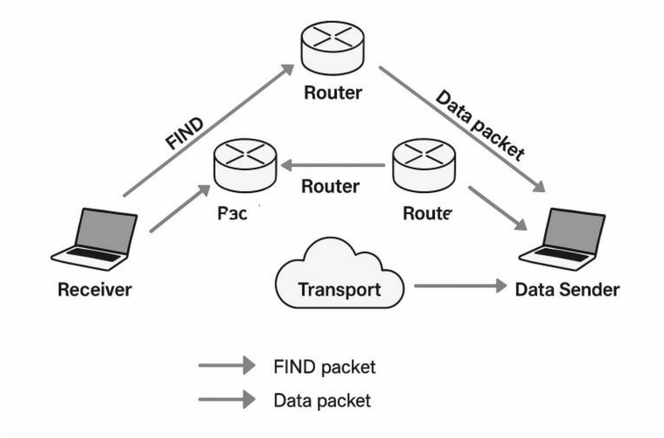

Fig. 4 DONA overview

2.2.1 Data Oriented Network Architecture DONA Naming

In DONA, a flat namespace is used, and the names are organized by using principals. A public private key pair is used to associate each principal, which is in the form P:L where P and L are the globally unique principal fields containing the cryptographic hash of the publisher's public key and the object label, respectively. Each datum received contains metadata with a minimum of three fields, i.e., data, public key, and signature. To ensure data integrity, the requesting client relies on the principals' signatures. In addition, because P is unique for each publisher, republishing the same content by a different publisher will result in multiple copies of the same content. These multiple copies can either be removed by using various methods, i.e., wildcard queries or principal delegation or be used to satisfy multiple content requests [9].

2.2.2 Data Oriented Network Architecture DONA Routing and Name Resolution

A name-based routing paradigm is used for name resolution in DONA. This is achieved through the network entities called "resolution handlers" (RHs) instead of DNA servers. There is at least one RH available in each domain. Each node in the domain has information on the RH through local configuration information. The client and RH use two primitives, FIND (P:L) and REGISTER (P:L), to achieve name resolution. The nodes willing to serve data use the REGISTER (P:L) packet to register the datum with the RH. Each RH maintains a registration table, which is used to map the incoming requests to the destination of the content, copy, or to the next hop RH toward the content copy. The RHs are organized in a hierarchical manner. In the event that the RH receives a new REGISTER (P:L) packet from a child node, it is stored in the registration table, and the RH also forwards the packet to its parent and peers. The peer receiving the REGISTER will forward the packet on the basis of the local policy. The REGISTER packet is not forwarded onward if a record already exists or if the new REGISTER comes from a copy further away from the previous copy. The client issues a FIND (P:L) packet to locate the content named "P:L." If there is an entry in the registration table when the request is received at the RH, the request is forwarded to the next hop RH; otherwise the RH forwards the request to its parent RH. In case of more than one entry being in the registration table, the closest one is selected. Once a content copy is found, the data are either routed back to the client by way of the reverse request path with caching enabled on the RHs or forwarded directly to the client as shown in Fig. 1.3. In DONA, name matching is accomplished using longest-prefix matching [10].

2.2.3 Data Oriented Network Architecture DONA Caching

In DONA, RHs can be enabled with a universal-caching mechanism. To enable caching, the RH populates its cache by replacing the source IP address and port number of the FIND packet with its own and then forwarding it to the next-hop RH, thus, thus ensuring the traversal of the response packet through this RH. The RH stores the data in its cache before forwarding it to the requesting node. All the data items in the cache are labeled with a TTL or some other validation metadata, which ensures the time period of the data. When a FIND arrives and there is a cache match, the RH will initiate the appropriate transport response, which will lead to the standard application-level exchange of data. In case the transport or application-level protocol information in the FIND is ambiguous to the RH, then it does not provide any caching for that particular request.

2.3 Named of Information NetInf

In another approach, the Network of Information (NetInf) project [11] was initially proposed by the European FP7 4WARD [12]. Like the DONA architecture, NetInf also uses flat namespace [13]; hence, a public key infrastructure (PKI) is not required. NetInf's content model is based on the widely used multipurpose internet mail extensions (MIME) standard. Furthermore, search primitives, which provide links between the search item and the object name, are also a part of the architecture. Two objects-retrieval approaches are offered in the architecture, namely, name resolution and name-based routing. Depending on the model used, the source/NetInf node can either register with the name resolution service (NRS) to publish the content (termed "named-data objects" [NDOs]) or use a routing protocol to announce the routing information. As illustrated in (Fig. 5), in case no. 1, the client first forwards the request to the NRS, which gives the available locaters of the particular NDO name; subsequently, the client retrieves a copy of the data from the best available sources. Alternatively, using name-based routing, the client can directly send out an NDO GET request, which is forwarded to the source. The data are sent to the client as soon as a copy of the NDO is reached. These can either be used separately in the network or merged into a hybrid scheme, in which case switching between the two schemes is performed on hop-by-hop basis. This hybrid scheme allows NetInf to adapt and scale itself to the different requirements in the network such as network mobility [14], delay-tolerant networking (DTN), and global connectivity. Furthermore, NetInf architecture can be deployed as an extra layer on top of the existing network infrastructure, thus simplifying the migration of applications to the new infrastructure.

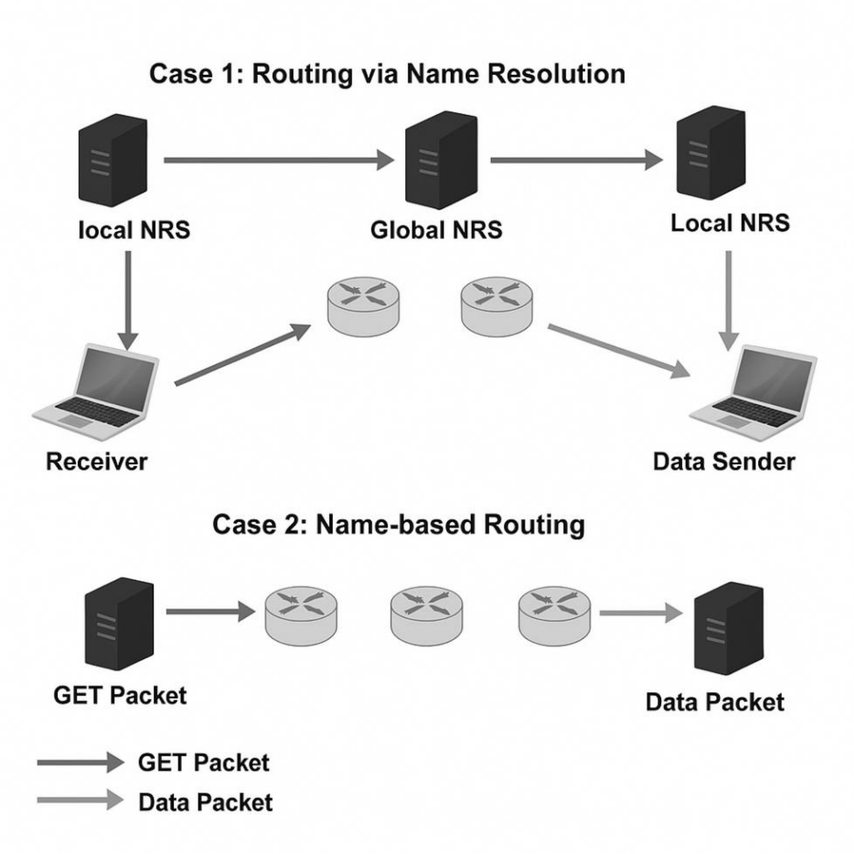

Fig. 5 NetInf overview

2.3.1 Named of Information Naming

NetInf uses a flat namespace with a structure similar to the DONA namespace. NetInf aims to accommodate different ICN architectures by deploying the naming such that the naming aims to differentiate three aspects: (1) the common naming format used by each node; (2) security related information to maintain the integrity of the data; and (3) name-object binding validation mechanisms. The NetInf naming format supports different hashing schemes. The owners public key hash digests is also contained in the name to support the data that is yet to arrive. Furthermore, different naming representation is supported, i.e., Uniform Resource Identifier (URI) and binary representation.

2.3.2 Named of Information Name resolution

To incorporate the different ICN architectures, NetInf supports a variety of name-resolution services. NetInf merges name resolution and name-based routing to retrieve the data. A new inter domain interface is defined for name resolution and routing, which allows different schemes to be applied in multiple parts of the network. Today’s URLs are supported by NetInf name resolution; hence, NetInf can be integrated smoothly with the current infrastructure. Multiple types of NRS are supported in NetInf such as both local and global NRS. Using local NRS, the operators can reduce and control the traffic flow thus potentially decreasing the load on caches and servers. A number of name-resolution mechanisms have been developed such as multilevel distributed hash table (MDHT) [15], hierarchical skipnet

(HSkip) system [16] and late-locator construction (LLC) [17]. MDHT and HSkip systems provide a global and hierarchical NRS that is topologically embedded in the core network to improve stability, scalability, copy-locator selection, and efficient data dissemination. LLC focuses on high-dynamic network topology handling, which includes movable networks. This NRS approach allows a smooth transition from the Internet while using the current infrastructure. For example, traditional URLs can be resolved from object names and retrieved using the existing HTTP protocol.

2.3.3 Named of Information Caching

Caching plays an important role in efficient content distribution in NetInf. NetInf supports three caching options: on-path caching, off-path caching, and peer caching The NetInf router has a built-in caching feature to enable on-path cache, which caches objects while routing objects in response to the GET request. Off-path cache is placed in the network to reduce the traffic and latency. This cache is not directly in the request/data path. It is typically connected to an NRS in the network. The cache broadcasts the cached objects to the NRS and, based on the popularity, the NRS informs which data to cache. In this way, the off-path cache can avoid the steps to obtain the information from GET requests of the requested objects. In peer-caching, the NetInf nodes can function as an on/off-path cache. The peers can broadcast the cached data in the network. The NRS can route the GET request, thus reducing inter domain traffic and latency and additionally minimizing the load on the data servers.

2.4 Publish Subscriber Internet Routing PURSUIT

The Publish Subscribe Internet Technology (PURSUIT) [18] project, a continuation of the Publish Subscribe Internet Routing Paradigm (PSIRP) [19] project, both funded by the EU Framework 7 Program (FP7), have proposed a publish/subscribe protocol stack that replaces the IP protocol stack. In PURSUIT, sources publish the contents into the network as shown in Fig. 1.5. The receivers can subscribe to the published contents through the rendezvous systems. A rendezvous system helps in locating the scope and publications in the network. Each piece of the published content belongs to a specific named scope. The subscription requests contain the scope identifier (SI) and the rendezvous identifier (RI), which together identify/name the particular desired content. Using these identifiers in a matching procedure results in a forwarding identifier (FI), which is used by the source to forward the data. A bloom filter [20] is specified in the FI, which is used by the intermediate routers to select the interfaces to forward contents as shown in (Fig. 6). This relieves the router from maintaining the forwarding states. However, a bloom filter yields some false-positive results, thus leading to forwarding on interfaces where there are no receivers.

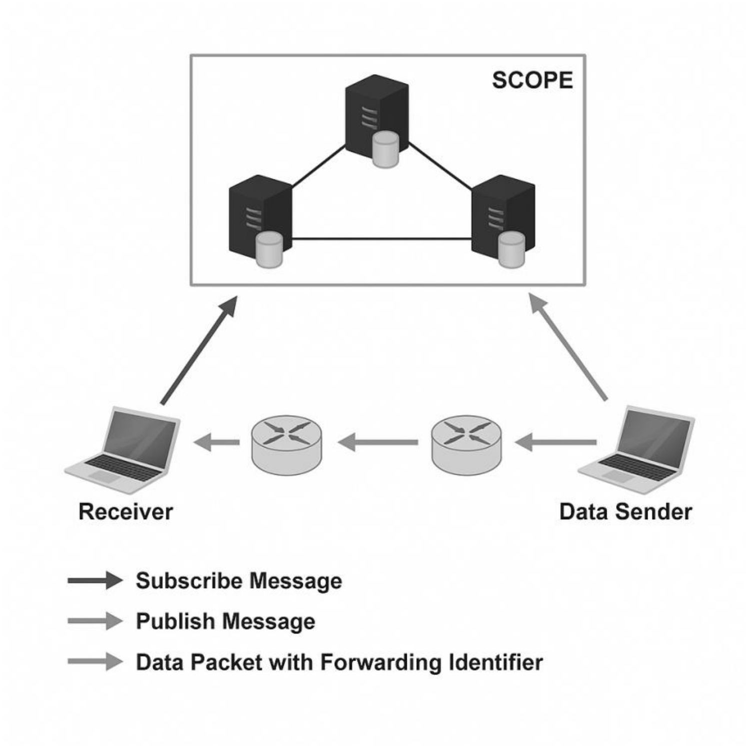

Fig. 6 PURSUIT overview

2.4.1 Publish Subscriber Internet Routing Naming

PURSUIT uses a flat namespace with two types of names, namely, the rendezvous identifier (RI) and the scope identifier (SI). These identifiers together establish the name of the content. RIs help in mapping the content between publishers and subscribers. In addition, the forwarding identifier (FI) is used by routers to identify the path from the publisher to the subscribers.

2.4.2 Publish Subscriber Internet Routing Routing and Name Resolution

The routing management is responsible for selecting the best inter domain-route forwarding of the publications. PURSUIT uses the resolution model called a "rendezvous point." The name resolution of the data is performed at this point. However, the data return path to the subscriber does not have to include the rendezvous point. Forwarding is performed using the source routing approach using bloom filters called "zfilters" carried in the FI. The bloom filter describes the route from the source to the destination because it contains all the names of the routing links. This information is attached to the data as the FI. At each node, the router checks whether or not the link identifier is present in the packet using a simple AND operation. Thus, in PURSUIT, increasing packet length as well as the network resources are reduced.

2.4.3 Publish Subscriber Internet Routing Caching

Caching in PURSUIT is mainly provided as a dedicated solution to a problem for which caching might offer some benefit. Moreover, multiple caches of an object can be maintained based on the scope of the rendezvous point for the identifier associated with the object.

3. Mathematical Modeling and Performance Analysis

To quantitatively assess the performance of Information-Centric Networking (ICN) architectures—such as CCN, NDN, DONA, NetInf, and PURSUIT—we propose a set of mathematical models that reflect their caching efficiency, latency, data availability, and forwarding strategy overhead [21-25]. These models are essential for comparing ICN performance under various networking scenarios, including intermittent connectivity and content replication.

3.1 Latency Model

The total latency for content retrieval in ICN systems is modeled as:

$L\_total = L\_interest + L\_data + L\_cache$

Where:

- $L\_interest$: time to forward the interest packet to a content provider or cached node.
- L_data: time for data to be returned via the reverse path.
- L_cache: reduction in latency due to in-network caching.

Let:

- $L\_interest = H\_avg \times t\_link$
- $L\_data = H\_avg \times t\_link$
- $L\_cache = \delta \times L\_total\_uncached$

Where H_avg is the average hop count, t_link is average link delay (e.g., 5 ms), and δ is the cache hit ratio.

3.2 Cache Hit Ratio Model

Assuming a Zipf distribution for content popularity:

$P\_i = 1 / i^s / H\_N(s)$, where $H\_N(s) = \sum (1/i^s)$ for $i = 1$ to N

Let:

- $s = 0.8$ (typical Zipf parameter)
- $N = 1000$ (content objects)
- $C = 100$ (cache size)

Cache hit ratio (CHR) = $\sum P\_i$ for $i = 1$ to C

3.3 Throughput Model

Throughput T is given by:

$T = (1 - Packet\_Loss) \times Data\_Size / L\_total$

3.4 Forwarding Table Size (FIB Overhead)

In NDN, the size of the forwarding information base (FIB) is affected by the naming granularity and content diversity.

Let:

- Prefixes = $\lambda \times U$, where λ is average prefixes per user, U is total users
- Avg size per entry = 50 bytes
- $\lambda = 20$, $U = 1000$

$FIB\_size = 20 \times 1000 \times 50 =$ **1 MB**

3.5 Simulation Results and Analysis

To evaluate these models, simulations were conducted different environments. Fig.7 below is a summary of the main findings:

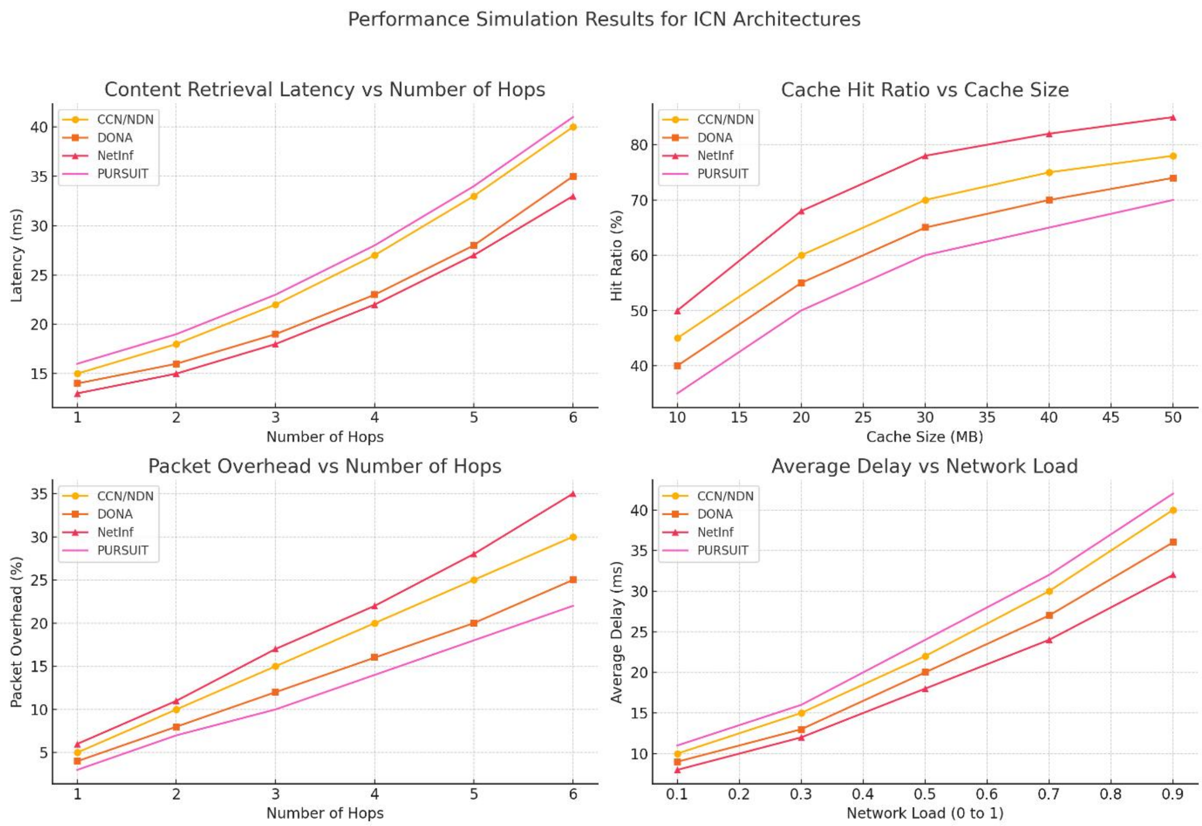

Fig.7: Performance analysis of ICNs

The mathematical models and simulation results reveal that ICN architectures significantly benefit from in-network caching and name-based routing. With only modest cache sizes (10% of content catalog), hit ratios of up to 40% can be achieved, reducing latency by over 40%. Additionally, forwarding strategies must consider the linear growth of FIB entries with user count and naming complexity, especially in hierarchical schemes like NDN.
These insights validate the scalability and efficiency of ICN systems in high-demand scenarios and support their deployment in IoT, vehicular, and mobile edge environments. Future extensions may include probabilistic caching strategies, AI-enhanced routing policies, and adaptive naming schemes for ultra-dense networks.

Conclusion

This paper has presented a comprehensive overview and comparative analysis of four prominent Information-Centric Networking (ICN) architectures—CCN/NDN, DONA, NetInf, and PURSUIT. Each architecture is characterized by its distinct approaches to naming, routing, name resolution, and caching. Our performance modeling and simulation-based evaluation demonstrated that while all ICN models improve content delivery compared to traditional IP-based systems, their effectiveness varies depending on factors such as hop count, cache size, network load, and traffic distribution. NetInf exhibited strong caching performance, while DONA offered low packet overhead, and PURSUIT showed promise in publish-subscribe environments with scalable routing. These insights highlight the importance of choosing ICN architecture based on application context. Future work should focus on hybrid ICN frameworks, mobility-aware caching, and the integration of ICN with 5G, IoT, and AI technologies to fully realize its potential for the future Internet.

References :

[1] Hitoshi ASAEDA, Potential and Challenges of Information-Centric Networking

[2] Cheriton D, Gritter Triad, a new next-generation internet architecture, 2000

[3] Jacobson V, A new way to look at networking, Google Tech Talk, Aug 2006

[4] Jacobson V, Smetters DK, Thornton JD, Plass MF, Briggs NH, Braynard RL Networking named content. In: Proceedings of the 5th International Conference on Emerging Networking Experiments and Technologies, CoNEXT '09, pp 1–12, New York, NY, USA, ACM, 2009

[5] Fayazbakhsh SK, Lin Y, Tootoonchian A, Ghodsi A, Koponen T, Maggs B, Ng K, Sekar V, Shenker S. Less pain, most of the gain: incrementally deployable ICN. SIGCOMM Comput Commun Rev 43(4), 2013

[6] Koponen T, Chawla M, Chun B-G, Ermolinskiy A, Kim KH, Shenker S, Stoica I A data-oriented (and beyond) network architecture. SIGCOMM Comput Commun Rev 37 (4):181–192, 2007

[7] Mazi`eres D, Kaminsky M, Kaashoek M.F, Witchel E (1999) Separating key management from file system security. In: Proceedings of SOSP '99, pp 124–139, Charleston, SC, USA, Dec 1999

[8] Moskowitz R, Nikander P (2006) Host Identity protocol architecture. RFC 4423, IETF, May 2006

[9] Xylomenos G, Ververidis CN, Siris VA, Fotiou N, Tsilopoulos C, Vasilakos X, Katsaros KV, Polyzos GC (2014) A survey of information-centric networking research. Commun Surv Tutorials IEEE 16(2):1024–1049 (Second Quarter 2014)

[10] Ghodsi A et al (2011) Naming in content-oriented architectures, In: Proceedings of ACM SIGCOMM workshop information-centric networking, Toronto, Canada, Aug 2011

[11] Dannewitz C, Kutscher D, Ohlman B, Farrell S, Ahlgren B, Karl H, Network of information (netinf)–an information-centric networking architecture. Comput Commun 36 (7):721–735, 2013

[12] FP7 4WARD project (Online). Available: http://www.4ward-project.eu/

[13] Dannewitz C et al Secure naming for a network of information. In: Proceedings of 13th IEEE global internet symposium '10, San Diego, CA, Mar 2010

[14] Eriksson A, Ohlman B Dynamic internetworking based on late locator construction. In: 10th IEEE global internet symposium, 2007

[15] D'Ambrosio M, Dannewitz C, Karl H, Vercellone V MDHT: a hierarchical name resolution service for information-centric networks. In: Proceedings of ACM SIGCOMM workshop on information-centric networking, ACM, New York, NY, USA, 2011

[16] Dannewitz C, D'Ambrosio M, Karl H, Vercellone V Hierarchical DHT-based name resolution for information-centric networks, Elsevier computer communications, special issue on information-centric networking, 2013

[17] Eriksson A, Ohlman B Dynamic internetworking based on late locator construction. In: 10th IEEE global internet symposium, 2007

[18] FP7 PURSUIT project (Online). Available: http://www.fp7-pursuit.eu/PursuitWeb/

[19] FP7 PSIRP project (Online). Available: http://www.psirp.org/

[20] Bloom BH , Space/time Trade-offs in hash coding with allowable errors. ACM Commun 13(7):422–426, 1970

[21] Q. Ibrahim, "Enhanced power management scheme for embedded road side units," *IET Computers & Digital Techniques*, vol. 10, no. 4, pp. 174-185, 2016.

[22] Q. I. Ali, "Green communication infrastructure for vehicular ad hoc network (VANET)," *Journal of Electrical Engineering*, vol. 16, no. 2, p. 10, 2016.

[23] Q. I. Ali, "Performance evaluation of WLAN internet sharing using DCF & PCF modes," *Int. Arab. J. e Technol.*, vol. 1, no. 1, pp. 38-45, 2009.

[24] H. M. Mohammed and Q. I. Ali, "E-proctoring systems: A review on designing techniques, features and abilities against threats and attacks," *Quantum Journal of Engineering, Science and Technology*, vol. 3, no. 2, pp. 14-30, 2022.

[25] Q. Ibrahim, "Design & Implementation of High-Speed Network Devices Using SRL16 Reconfigurable Content Addressable Memory (RCAM)," *Int. Arab. J. e Technol.*, vol. 2, no. 2, pp. 72-81, 2011.