Computer Networks 165 (2019) 106937

Contents lists available at ScienceDirect

Computer Networks

journal homepage: www.elsevier.com/locate/comnet

A functional taxonomy of caching schemes: Towards guided designs in information-centric networks

Faria Khandaker[a], Sharief Oteafy[b,*], Hossam S. Hassanein[a], Hesham Farahat[a]

[a] School of Computing, Queen's University, Kingston, ON, Canada
[b] School of Computing, DePaul University, Chicago, IL, USA

ARTICLE INFO

Article history:
Received 11 February 2019
Revised 6 September 2019
Accepted 30 September 2019
Available online 2 October 2019

Keywords:
Cache designs
Cache valuation
Guided designs
Information-Centric Networking (ICN)
ICN caching taxonomy
ICN performance evaluation

ABSTRACT

Information Centric Networking (ICN) is a developing paradigm, poised to transform the Internet's architecture. At its core, ICN focuses on efficient content dissemination and retrieval, regardless of storage location and physical representation of content. Thus, content caching schemes play a pivotal role in providing fast, reliable, and scalable content distribution and delivery. Given the multitude of caching schemes that have evolved over the past few years, recent developments have been often hampered by a fixed set of design primitives. In this paper, we present a functional-based taxonomy of ICN caching schemes, detailing each functional component, and depicting the functional mandates of these schemes to aid in contrasting their operations. The goal of this survey is to guide the design and development of ICN protocols, building on insights from caching schemes and their inherent tradeoffs. We present a comprehensive benchmark for future caching schemes, coupled with a quantitative as well as qualitative analysis of leading caching schemes, encompassing cross-scheme performance metrics. We highlight the impact of ICN caching schemes in the development of the Internet of Things (IoT) and Vehicular Ad-Hoc Networks (VANETs). This work concludes by presenting insights for future developments in ICN, with a dedicated discussion on guided development for researchers in this domain; building on the aforementioned taxonomy, as well as quantitative and qualitative analyses.

1. Introduction

The Internet was originally developed as a "network of hosts" to access distributed resources and connect terminals. The underlying protocols of the Internet were designed to support the exchange of information between well-identified nodes, and followed a mailing paradigm that coupled content with its sender and receiver(s). Today, the Internet has become a global infrastructure for the distribution of information with billions of connected devices and Exabytes (EB) of data transferred monthly. In fact, overall IP traffic is expected to grow from 122 EB/month in 2017 to 396 EB/month by 2022 [1].

Internet usage patterns are becoming increasingly bandwidth-intensive, with IP video traffic projected to dominate 82 percent of all IP traffic by 2022, an increase from 74 percent in 2017 [1]. Internet users are primarily interested in fast and reliable retrieval of information, rather than identifying the server where the information is stored.

While many have argued about the demise of the current Internet infrastructure, significant efforts in expanding Content Delivery Networks (CDNs), Mobile Edge Computing (MEC), cross-layer multicasting and other overlay protocols have been aiding Internet scalability [2]. Motivated by these growth projections, the research community is exploring key requirements for the future Internet [3], building newer models for traffic characterization [4], and proposing new architectures to address them, as the current Internet architecture is struggling to scale with projected demand and usage patterns [5].

Information-Centric Networking (ICN) has evolved as a promising candidate for a future Internet architecture [3,6]. The primary objective of ICN is to shift the current host-oriented network model towards a content-centric model by focusing on highly scalable and efficient information distribution and retrieval. In order to fulfill future Internet requirements, ICN uses unique names to address and route content on the network, decouples contents from their locations, deploys in-network caching for efficient content distribution and retrieval, applies security mechanisms

[*] Corresponding author.
E-mail addresses: khandake@cs.queensu.ca (F. Khandaker), soteafy@depaul.edu (S. Oteafy), hossam@cs.queensu.ca (H.S. Hassanein), hesham@cs.queensu.ca (H. Farahat).

https://doi.org/10.1016/j.comnet.2019.106937
1389-1286/© 2019 Elsevier B.V. All rights reserved.

directly to the content, and provides efficient support for handling mobility [7,8].

Caching is a fundamental building block in ICN, tasked with enabling fast, reliable, and scalable content dissemination. Every network router in ICN has the capability to cache named contents, and respond to requests for such contents. Although caching is extensively investigated in many fields such as web proxies, ICN caching has emerged as a new area of research in recent years because of the integration of caching as a key architectural component; in contrast to the overlay approach in the current Internet architecture[9,10]. We elaborate on the distinguishing factors in Section 3B.

We present seven contributions in this paper: 1) We elaborate on the main advantages of ICN in-network caching, investigate the main differences between traditional web-caching and ICN caching and summarize research challenges specific to ICN caching. 2) We present a benchmark taxonomy of existing caching schemes in ICN literature and elaborate on the choice of classifiers. We provide functional descriptions of the classified ICN caching schemes, noting the strengths and limitations of the schemes and present the comparisons in a tabular format. 3) We highlight recent research efforts that demonstrate the potential and high impact of ICN for the deployments of Internet of Things (IoTs), Vehicular Ad-Hoc Networks (VANETs), and newer manifestations of Mobile Ad-Hoc Networks (MANETs). 4) We present a comprehensive qualitative performance assessment and comparison of ICN caching schemes built on four core performance metrics. 5) We devise and present a quantitative performance assessment to contrast a number of well-known caching schemes, including at least one representative scheme under each classifier in our taxonomy. We build on our insights from these experiments, and correlate our quantitative performance analysis with the qualitative one to emphasize the impact of their design choices on ICN caching performance. 6) We elaborate on the prominent challenges in ICN caching research, detailing potential directions, and discussing our insights on developments that would guide the development of novel caching paradigms, in contrast to modifying/updating existing schemes. 7) We dedicate a discussion on guided designs of ICN schemes, building on the presented taxonomy and our performance analyses, to aid researchers in developing novel ICN schemes under varying constraints; highlighting the impact of design choices in cases of conflicting objectives.

Recent attempts have been made to classify ICN caching schemes in literature. We hereby elaborate on the major surveys on ICN caching schemes [11–17] which have proposed classifications of caching schemes, and contrast our contribution and methodology to them.

Zhang et al. [12] classify ICN caching schemes based on one classifier which is co-operation among caching nodes and only describe very few caching schemes. Ioannou et al. [11] propose a classification of ICN caching schemes also based on one criterion; namely the content delivery path between content source and consumer. Later, Ioannou et al. extend their taxonomy in [15] again based solely on content delivery path, with more detailed analysis of some existing caching schemes and forwarding mechanisms. In [14], Abdullahi et al. present some distinct functionalities and open research issues in four popular ICN architectures relating to caching, yet these schemes are classified and described based on content delivery path only. Bernardini et al. have only visited very few ICN caching schemes to present a comparative study for different network scenarios in [16]. The survey by M. Zhang et al. in [13] is more comprehensive than the aforementioned [11,12,14,16], however we adopt a broad taxonomy approach to encompass more recent directions and paradigms by providing functional descriptions of caching schemes under multiple classes and subclasses. Moreover, we highlight the main attributes of the caching schemes under each class. A recent caching survey [17] provides a detailed description of many of the newer caching schemes, under the traditional categories of ICN caching. While this is indeed valuable, our taxonomy is targeted at building on the fundamental design choices in caching parameters and constraints, to emphasize what can be built on them, rather than summarize what others have incorporated in their designs. This yielded a finer granularity in our taxonomy, to address specific design choices that have significant impact on performance, as demonstrated in Sections 6 and 7. Simply put, our goal in this survey and tutorial is to derive from caching primitives to explain what can be designed, rather than solely classify what has been presented. We aim to aid researchers undertaking caching challenges in ICN, to better build caching models that capitalize on their design choices, and avoid design pitfalls that would hinder their performance, as elaborated on in Section 9.

In doing so, we appreciate the insights gained from previous designs, as we provide a thorough functional description of caching schemes under each classifier, aiming for coherency and consistency to facilitate functional comparisons. Furthermore, we summarize the basic strategy of each class in our taxonomy by illustrating their functional mandates in consistent depictions. We cover more caching schemes compared to previous surveys, under more paradigms, and focus on the diversity of recent caching schemes to broaden our taxonomy.

One of the significant voids that this paper addresses is carrying out a thorough qualitative performance assessment and comparison of ICN caching schemes. While Ioannou and Weber present a qualitative comparison of some caching schemes in [15], they focus on the basic goals, advantages, and disadvantages of the caching schemes rather than comparing their functional mandates and ensuing impact on caching efficiency. In contrast, we identify several performance core metrics for qualitative performance assessment and comparative analysis of these caching schemes.

This survey presents a comprehensive quantitative performance analysis to analyze the sensitivity of ICN caching schemes while choosing at least one scheme representing each of our proposed class. Surveys [13] and [17] present a quantitative performance analysis, built at contrasting their performance. In this paper, we carry out experiments to correlate quantitative performance analysis with qualitative analysis, to provide grounded insights on the impact of design goals, and reveal the hindrances of design choices on caching performance.

To the best of our knowledge, our work is the first survey to explore the benefits and implications of ICN as an ideal candidate architecture for the deployments of IoTs, VANETs and MANETs, building on the in-network caching mechanisms embedded with other core features of ICN.

We also highlight that we have dedicated a detailed discussion on novel research directions in ICN caching, to guide ICN researchers in understanding the spectrum of caching paradigms, as well as addressing specific design choices in targeted caching schemes.

The remainder of the paper is organized as follows. In Section 2, we present the key components of ICN as an infrastructure and highlight the applicability of ICN for deploying IoTs, VANETs and MANETs mainly because of the in-network caching feature. In Section 3, we facilitate a clear discussion of the unique characteristics of ICN caching. In Section 4, we elaborate on the core research challenges in ICN caching, to build a thorough study of our taxonomy of caching paradigms, which we present in Section 5. We elaborate on our detailed qualitative analysis in Section 6, under four major metrics, and contrast the operation of leading caching schemes therein. Our comprehensive quantitative performance analysis is presented in Section 7. We delve into some of the major research directions of ICN caching in Section 8. In Section 9, we highlight the prominent research directions and we further address early researchers in ICN caching schemes, and present guided designs for future caching schemes, based on promising directions in the status quo, and summarize the mandates for five umbrella design goals. We end our discussion with a number of insights for future development in Section 10, shining the light on directions with potentially significant impact on ICN research, and conclude this paper in Section 11.

2. ICN architecture and key components

ICN has been researched under multiple aliases, chief among them are content-aware, content-centric, and data-oriented networking [7]. In contrast to the current host-centric Internet architecture, where the senders (content providers/producers/publishers/source) have control over the data exchanged; in ICN, data is not typically sent unless the receiver (consumer) explicitly requests it. Hence, ICN shifts the Internet's sender-driven end-to-end communication paradigm to a receiver-driven content-retrieval paradigm. In ICN, a content provider does not send content directly to the receiver. A content provider sends advertisement messages to inform the network that it has some content to disseminate without a priori knowledge of potential receivers who may be interested in this content. On the other hand, a receiver declares its interest in the content without a priori knowledge of potential providers which have published such content. When a receiver's interest matches published content, the network initiates a content delivery path from the sender to the receiver so that the content may be delivered.

Fig. 1 illustrates a typical content dissemination and retrieval process in ICN. A given content publisher P has some content to distribute, and it sends advertisement messages to inform the network that it has content to disseminate. Consumer C_1 declares its interest for some content by sending a subscription messages in the network. If C_1's interest matches the published content of P, the network initiates a content delivery path from P to C_1 so that the requested content can be delivered to C_1. While delivering the content to C_1, the content is cached at router R_4. Hence, later when another consumer C_2 sends a subscription message for the same content, the requested content is directly delivered to consumer C_2 from the cache router R_4 which is the nearest available copy of the desired content.

2.1. Building ICN architectures

The idea of shifting from a host-centric network design to an information-centric network design was investigated by the research community almost a decade ago. The TRIAD project in 2000 adopted the concept of name-based routing and is considered as a pioneering work in ICN [18]. Later, Carzaniga et al. described content-based networking and proposed routing scheme for content-based networking in their seminal papers [19–21]. There are several ongoing ICN research projects, such as the Data-Oriented Network Architecture (DONA) [22], Publish-Subscribe Internet Technology (PURSUIT) [23] and its predecessor Publish-Subscribe Internet Routing Paradigm (PSIRP) [24], Network of Information (NetInf) [25], Named Data Networking (NDN) [26] and its predecessor Content-Centric Networking (CCN) [27,28], Scalable & Adaptive Internet soLutions (SAIL) [29] and its predecessor 4WARD [30], COntent Mediator architecture for content aware nETworks (COMET) [31], CONVERGENCE [32], MobilityFirst (MF) [33] and other descendants.

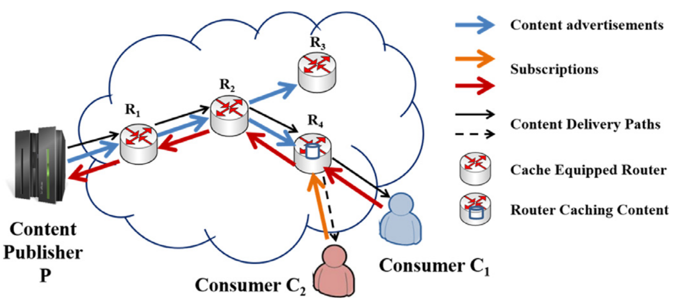


Fig. 1. Content dissemination and retrieval process in ICN.

These ICN projects are built on a set of key components to realize a more efficient architecture for content distribution and retrieval; designed to shift the Internet from a host-centric paradigm to an information-centric paradigm. A significant body of ICN literature focuses on realizing a predetermined set of key components in ICN projects [34–50] and many of these research works are detailed later in our paper in Sections 5–8. These key components of ICN projects are elaborated upon in the remainder of this section.

2.2. ICN key components

Our goal here is not to survey ICN literature, as this has been aptly addressed by George et al. in [7] and other surveys [8,9]. However, we aim to establish the common components on which diverse ICN schemes have been built, to facilitate a clear taxonomy of caching approaches and the impact of design choices on cache operation; as elaborated upon in Section 5.

2.2.1. Naming

Naming content is a pillar component in ICN infrastructures. Specifically, Named Data Objects (NDOs) are at the core of addressing and forwarding in ICNs [51]. All NDOs, including documents, web pages, videos, songs, and photos should have globally unique names. An NDO is independent of location, storage method, application, and transportation method; thus any two copies of an NDO are equivalent for all purposes. There are three basic naming schemes in ICN: flat naming, hierarchical naming and attribute-based naming [52].

In flat naming, also called self-certifying naming, the content name has two parts in the form P: L and associated with Meta-data. The first part of the name P is the cryptographic hash of the content publisher's or owner's public key and L is an arbitrary label assigned by the publisher. Meta-data contains the complete public key and digital digest signed by the publisher. Self-certifying names are globally unique, persistent as they lack semantics, easy for integrity checking and are not user friendly. Hierarchical names are formed by concatenating multiple hierarchical components where the components can be any descriptor-string of any length. Hierarchical names are user-friendly and have intrinsic semantics since their components or structures reveal information relating to the content itself such as property, version and so on but lack strong persistence [51]. Instead of identifying content by unique names, attribute-based naming identifies content by attribute-value pairs (AVP). Contents are requested by applying constraints over the attributes, where the constraints are called "predicates". This naming scheme does not guarantee uniqueness for content names [52].

2.2.2. Routing

In ICN, there are currently two main approaches to handle routing, either via name resolution or using name-based routing. In name resolution, there are two steps for routing content. In the first step, consumers send request messages to the name resolution system (NRS) with the name of the desired content and the NRS resolves the name with a single or set of addresses of content producers or cache nodes having the desired content.

In the second step, consumers send request messages to content producers or cache nodes and the desired content is routed back to the consumers.

In name-based routing, in a single step, the request message for a content is routed based on the name of the content from the consumer to the content producer or to any cache node which can serve the request. Some state information is stored along the way so that the requested content can be routed back to the consumer [9].

2.2.3. Caching

Any network router in ICN can cache content. When a network router (also called content router or cache router) receives a content request, it can take one of two actions, either it responds directly if it has the content in its local cache, or it sends a request message to its peers (or the content producer) and can thereafter cache the content when the request is fulfilled through it.

ICN caching is dubbed universal caching because of three main properties: it provides uniform caching for all content carried by any protocol, it democratizes content delivery by caching contents from any content provider and it provides pervasive caching as caching in ICN is implemented by all network routers instead of a few specialized cache nodes (as adopted in CDN or Web caches) [53].

2.2.4. Security

In ICN, content can be retrieved from any network router which has cached the content rather than the originating content server (or producer). The security model in ICN is not based on the originating content server or producer; instead, ICN designs have to secure the content itself [54]. Hence, ICN architectures implement a content-oriented security model where the content is signed by the content producer so that any network element and consumer can verify the validity of the content by verifying the signature of content provider.

2.2.5. Mobility

ICN architectures are based on the basic principle of a publish-subscribe communication model [11]. The strength of this model is rooted in the fact that publication and subscription operations are decoupled with respect to time and space [55] which allows ICN to provide efficient support for mobility. In ICN architectures, mobile consumers can send new subscriptions for contents after a handoff occurs. Publisher mobility is more difficult to support as the name resolution system or routing tables in name-based routing need to be updated [56].

2.2.6. Application programming interface (API)

API is used for requesting and delivering content. Content producers make named contents available to consumers by publishing it to the network using the operation called publish or register, and the consumer requests for the named content using the operation called subscribe or find or get or interest. Both of these operations (publish and subscribe/get) use the content name as the main parameter [51].

2.3. ICN in synergy with Edge technologies and IoT

Recent developments in managing services and data at the edge of the Internet, are mandating novel approaches to data dissemination and content handling. In light of rising demands for more responsive systems, many Edge technologies and networking domains have evolved to better address rising data access challenges. These evolvements span developments in MEC, IoT technologies [57], and VANETs [58].

We present recent milestones in research that explicitly address the integration of ICN based in-network caching schemes for IoT [59–65], VANETs [66–68] and MANETs [69–72]. As the benefits of building on ICN's content-based naming, security and mobility features are evident, we survey recent literature that explored the applicability of ICN architectures for deployments of IoT [73–81], VANETs [82–89] and MANETs [90–94]; focusing on the merit of adopting ICN features, before we specifically focus on in-network caching.

2.3.1. ICN in IoT

Several researchers have considered using an ICN architecture for IoT management, as ICN matches a wide set of IoT application demands that are information-centric in nature. IoT applications often target content regardless of the identity of the source that stores them. For example, road traffic or environmental monitoring applications are oblivious to the specific car or sensor that provides the information. ICN names can directly address heterogeneous IoT contents and services, such as vehicular services, home services, environmental data etc. The rich set of challenges and requirements of IoT placed over the current Internet provides an interesting and challenging ground for demonstrating the potential contribution of the ICN mechanisms for IoT deployment. In the following subsection, some of the research works that have explored the applicability of ICN architectures for deployments of IoT are highlighted.

Amadeo et al. in [73] present ICN-integration as an innovative potential networking paradigm in the IoT domain because of its improved data dissemination efficiency and robustness in challenging communication scenarios. The applicability of ICN principles in a challenging environment like IoT having high number of heterogeneous and potentially constrained networked devices, and unique and heavy traffic patterns has been critically analyzed and discussed. For critical analysis, the current literature has been surveyed and the major motivations, benefits, open challenges and opportunities to introduce the ICN in the context of IoT have been highlighted as future research guidelines. A comparison between the IP IoT and the ICN IoT is not provided in the article as ICN is still being progressed whereas IP has unfolded into multiple variants of IoT solutions, but rather IoT has been identified as an important deployment scenario for the utilization of ICN mechanisms. Based on existing research, Amadeo et al. have analytically shown that core ICN principles have the potential to fulfill many IoT requirements; such as addressing scalability challenges, fulfilling differentiated Quality of service (QoS) requirements, enabling security services, designing energy-efficient operations, supporting mobility and highly heterogeneous environment.

The increasing explosion of data and signaling packets generated by billions of connected devices can provide stringent scalability challenges in IoT environments, and the typical IP-based content retrieval mechanisms such as peer to-peer (P2P) and CDN pose complex issues, such as suboptimal peer selection or their incapability to leverage in-network storage in such scenarios. Although not being specially aimed to be deployed in IoT scenario, the inherent attributes of ICN offer promising scalability aspects for its deployment capability in IoT environments. ICN naming mechanisms can associate IoT contents to names enabling contents to be structured into scopes and allowing users to specifically request the content that they really want. So, this naming flexibility exploits the higher addressing potential of ICN by allowing a name in the IoT context to identify not only a content, but also a service or a device function. ICN also has the potential to reduce the signaling footprint in IoT deployments as it can offer name resolution at the network layer and forward content by content's name. ICN nodes have the ability to identify requests for the same named content, avoiding the need to forward them differently on the same path. ICN can prevent source over-querying and support connectionless scenarios as contents can be cached in traversing nodes allowing requests to be satisfied by the first available copy. Moreover, content can be transmitted to multiple consumers by using native anycasting and multicasting in ICN.

Unfortunately, routing and naming capabilities of ICN may face a much more difficult scalability problem when compared to the current global routing and Domain Name Service (DNS) resolution services, as the amount of content names can be orders of magnitude larger than the number of hosts connected to the current Internet. In this regard, the ICN research community has already proposed solutions such as the utilization of distributed hash tables (DHTs), late-binding mechanisms, and routing information aggregation to solve the problem at the expense of increased memory and processing costs.

QoS requirements vary significantly across IoT applications, as their utility span a highly-heterogeneous spectrum of use cases and services. In IP networks, QoS requirements are fulfilled by the execution of different extensions done over the base protocol, such as multiprotocol label switching (MPLS) and Resource Reservation Protocol (RSVP) under integrated/differential service (IntServ/ Diffserv) paradigms. So, resources are reserved at each hop between the source and the content requester which require extensive signaling, flow identification, and queue processing at the forwarding entities.

Ultimately, the mechanisms of content handling in ICN can significantly leverage performance under the projected explosion of IoT traffic, due to an unprecedented number of connected nodes, varying device characteristics, and traffic requirements [95]. ICN has the potential to manage different QoS demands, and improve the quality of content retrieval, because of its in-network caching, anycasting, and multicasting which all together contribute to speed up data retrieval and reduce traffic congestion. Additionally, ICN designs excel in advanced and efficient forwarding mechanisms.

Enabling security provenance in IoT is a fundamental requirement, since most IoT applications can affect our personal daily lives. IP-based security protocols have limitations to provide the security services as they are not conceived by design, depend on the location identification of nodes and secure the communication channel rather than the content. On the other hand, ICN provides security support at the network layer, facilitates content sharing between the nodes since the content authentication and integrity can be verified locally while removing the need for trusting in intermediary nodes. Additionally, ICN secures the content itself and restricts data access to a specific user or a group of users.

Energy-efficient operation design is very essential for any IoT networking solution as IoT devices can have severe limited power and computing capabilities because of resource constraints and most embedded devices spend a major part of their lifetime in sleep mode and only wake up when they need to exchange data. Current energy efficiency approaches in IP network are not handled at the network layer, target a limited class of applications, require devices to support a full web stack implementation and can imprint processing requirements over low-powered devices. In this context, the receiver-driven communication model of ICN can help to retrieve contents even in constrained networks with low duty-cycle providers because of its anycasting and in-network caching capabilities. Distributed caching can avoid the massive data access to constrained devices while saving energy resources. Moreover, native multicasting can also help to reduce the amount of traffic and interactions with energy-constrained nodes.

One of the key requirements for IoT devices is to provide mobility support but the IP mobility management solutions (e.g., Mobile IP) are commonly have scalability problems. Because of the receiver-driven nature, ICN supports consumer mobility by simply reissuing any unsatisfied request and serving the request by a different node when a consumer relocates. Additionally, ICN natively supports host multi-homing and as a result content requests or content delivery can use any of the interfaces (or even all simultaneously) available at the device. Although producer mobility entails additional signaling in ICN as it requires updates in intermediate forwarders while generating delays and disruption periods, anycasting, in-network caching, and multi-homing of ICN can greatly help to handle these issues.

A key challenge in IoT lies in adapting to its highly heterogeneous environment consisting of huge variety of devices, technologies, and services involving different stakeholders and manufacturers. In spite of the flexibility of the narrow-waist design of IP and its ability to maximize interoperability, it is complex to apply common network functionality to the explosive number of technologies involved in the upcoming IoT. To hide the heterogeneity in the underlying networks and devices and to facilitate the interoperability among the different network players, standardized ICN naming schemes for IoT can allow abstraction of services and contents. Moreover, ICN can interconnect information, devices, and services under heterogeneous network scenarios by decoupling consumers and producers and delivering self-consistent data packets.

Moreover, Amadeo et al. have also surveyed [73] the design considerations that are critical in ICN architectures to rise up to the challenges posed in IoT deployments. Although ICN matches many IoT design considerations, the complexity of IoT deployments mandate a number of adaptations in the design of ICN protocols. First, an ICN naming scheme for IoT should be highly expressive and customizable and it should expose service (e.g., sensing and action) and data features. Hierarchical naming scheme has been mainly considered in the literature to support such properties of IoT by defining a hierarchy of name components to identify the IoT application and the attributes to describe the related contents or services [96,97]. Hierarchical names can also facilitate request generation for dynamic contents by specifying the naming conventions during the system configuration. The variable-length names in hierarchical naming can make line-speed name lookup extremely challenging where the challenge can be addressed by sharing a common name prefix for multiple contents or services.

Current ICN security mechanisms are generally applied over contents, and rarely support request authentication; whereas some IoT applications require authenticated queries from consumers. The existing literature suggests to embed security information in content request packets as the last name component at the expense of complex security framework and increased name length [96]. Moreover, IoT devices having low processing and memory capabilities hardly use resource-intensive public key cryptography and require lightweight solutions for encryption [97]. Symmetric cryptography has been mentioned useful in this regard but requires pre-distribution of keys. To obtain a good tradeoff between complexity and resource saving, elliptic curve cryptography has been mentioned as a good candidate in the literature.

In-network caching has special significance in IoT domains because it speeds up content retrieval and increases content availability but increases the expenses for caching and related replacement operations. To address the research question whether caching should be enabled in any IoT device or only in powerful nodes has become a research question. A deterministic design choice should forbid constrained devices in IoT to cache contents [98], but caching is also proved to be highly beneficial even when enabled in IoT devices with small storage capacity [62]. In case of IoT, caching decision policies should focus on improving dissemination speed rather than long-lasting caching and off-path caching can be used to alleviate the load on constrained IoT devices and proactively cache contents in specific locations by preventing data redundancy at the cost of additional overhead for cache management. ICN offers name-based routing (NBR) and lookup-based resolution systems (LRS) for content discovery and both of these solutions may suit specific IoT scenarios mainly depending on the content characteristics and network features and can complement each other. An effective discovery and delivery platform for an IoT environment can be provided by leveraging cloud computing, multi-level DHT, name-prefix aggregation and adaptive forwarding.

Shang et al. [74] have also explored how NDN can build on the IoT vision in a secure, straightforward, and innovation-friendly manner; as the current IP-dependent solutions for IoT deployment have felt short on several significant networking challenges. It has been shown that the semantics of NDN naturally fit the inherent requirements of IoT applications. Along with the analysis of the achieved significant benefits acquired by applying NDN for the IoT deployment, the authors have also discussed some of the most major existing open problems for realizing IoT over NDN requiring urgent attention of the research community. NDN enables applications to name things and their contents and forward the content requests in the network directly based on those names, addressing the core challenges of the IoT vision by closing the gap between application and network semantics. Instead of building up new layers to achieve request-response communication of named content like IP networks, NDN implements this functionality at the network layer.

Instead of struggling to define and manage security in terms of subnetworks, channels, and sessions that are largely orthogonal to application security requirements, NDN directly secures the named contents at the network layer. This content-centric security mechanism ensures that the content consumers can validate a content packet independently of where and how they obtain it and the only authorized consumers can access the content. NDN's content-centric security solutions provide granular, packet-level authentication and access control which supports the realistic IoT scenarios where devices use a variety of ways to communicate in the networks supporting heterogeneous applications. NDN's name-based stateful forwarding can be useful for realizing other important features for IoT, such as a delay-tolerant style of communication and fast local recovery from losses.

Securing content directly in NDN enables even simple NDN applications to benefit from the in-network caching capability. NDN cache routers can opportunistically cache the contents while forwarding enabling efficient dissemination of popular contents and facilitating local recovery. Heterogeneous classes of devices in IoT applications can adjust the cache size and management policy based on the available storage, power, and processing capabilities.

Although the applications and usage of the IoT often imply information-centric usage patterns, using an ICN to design an IoT architecture is not always beneficial and depends on the applications and usage of the IoT network. Lindgren et al. [75] describe the contexts and applications for which the IoT architecture may benefit from using an ICN and helps to find the right tradeoff between using an ICN or a host-centric network for IoT deployment depending on the context. Some fundamental design choices and possible additions to ICN functionalities are also proposed in order to make the overall IoT solutions more effective, efficient and scalable using ICN. The key advantages of using ICN in IoT architecture have been identified as: naming of content and service independently from the device providing that content or service, reduced energy consumption because of fewer wireless transmissions and increased duty cycling possibilities along with reductions in information access latency due to distributed caching, and decoupling between publisher and consumer of content in the network, resulting into improved performance in networks and the possibility for increased sharing of content between applications.

To build on ICN architectures, many challenges and tradeoffs in IoT design need to be specifically considered, specifically towards scalable and efficient IoT operation. Firstly, creating and formatting names in an efficient and feasible way for the contents created in huge numbers and accessed in real-time by a large number of IoT devices are very challenging. The widely used hash-based content naming in ICN is suitable for systems with large contents where content verifying is important, but it is a challenge for IoT systems to use hash-based naming where content is dynamically generated. Second, maximizing the benefits of distributed in-network caching depending on the content consumption patterns, frequency of occurrence of the overlay CDN servers, device requirements and capabilities are challenging. Third, decoupling the content publisher and consumer arises security and application design challenges. Some major fundamental architecture-agnostic design choices and guidelines for effective, efficient, and scalable handling of IoT applications without requiring new functionality to be added to the ICN architecture are finally suggested in [75].

There are several research projects that have applied ICN to implement various IoT scenarios to explore the feasibility, advantages, and challenges of using ICN-based approaches in the IoT scenarios. We hereby survey prominent projects that target such feasibility considerations.

Rayes et al. have emphasized ICN architectures as the most likely architecture to be commonly used for IoT deployment in their work [76]. The authors have considered specifically the security implications while developing their IoT architectural framework and presented the ICN performance and the security requirements of IoT networks, together with a case study of management of an Information Technology- based network.

To set up simple smart home networks, a development toolkit named Named Data Network Internet of Things Toolkit (NDN-IoTT) is proposed in [77]. NDN-IoTT provides an experimental platform running on Raspberry Pi devices equipped with a number of simple sensors. NDN-IoTT contains templates for two types of nodes: controllers and devices to implement the basic bootstrapping and discovery mechanism.

A special version of the NDN client library called NDN−CPP Lite [78] has been developed for the IoT applications where sensors and actuators have to run on a wide-area infrastructure-less environment having constrained hardware platforms such as the agricultural fields.

An access control framework named NDN-ACE [79] is developed for securing actuation operations using constrained IoT devices where the devices themselves may not have enough storage or computation power for supporting complex security mechanisms such as executing expensive public key cryptography. To secure actuation operations, NDN-ACE adopts a protocol architecture where the constrained actuators offload authorization and key management tasks to a trusted third party that runs on more powerful platforms.

The design and implementation details of transporting Constrained Application Protocol (CoAP) *observe* [99] traffic for POINT ICN architecture [100] are presented in [80]. This design aims at improving communication overhead, state management and latency; specifically, in the situations where multiple consumers are interested in subscribing to the same resource hosted in constrained IoT devices.

CoAP [101] is an HTTP-like protocol operated in IP networks for applications intended to run on constrained devices in IoT, and CoAP observe is an extension to the CoAP specification that allows CoAP clients to observe a resource through a simple publish/subscribe mechanism. As the CoAP observe protocol is based on the publish/subscribe paradigm, it can benefit from the POINT architecture in terms of latency, state management, communication overhead, security and privacy. Hence, ICN has been transparently deployed within the domain of a network provider to provide enhanced CoAP services and the inherent multicast capabilities of ICN and caching at the edge have been exploited in observing similar resources hosted in IoT devices by multiple CoAP clients.

To highlight the benefits of POINT architecture in another extension of CoAP protocol i.e. CoAP group communication [102], a POINT-enabled building management system is proposed by Fotiou et al. [81]. For CoAP group communication, the proposed building management system enables issuing requests to groups of CoAP servers that implement the standard version of the CoAP protocol. *Things* management becomes much easier as the CoAP servers are oblivious to group names and the names are handled by the Network Access Points (NAPs). The ICN core of the management system makes group name administration easier as the new attributes are easily added to the namespace without affecting already deployed NAPs and group names do not have to be mapped a priori to a lower layer network address.

In the remainder of this section, we highlight promising research that explicitly addressed ICN in-network caching mechanisms for IoT deployments.

Vural et al. have stated their research work in [59] where they have argued for systematically analyzing the feasibility and benefit of using content routers to cache transient contents generated by IoT applications**.** As IoT contents are often transient [103], the contents expire in a certain time period, called content lifetime. So, the research in [59] has taken the challenge of addressing the content freshness issue considering the content lifetime while making caching decision. To address the challenge, in-network caching of transient data at content routers, considering the key temporal content property: content lifetime has been studied and methods to determine the quantifiable benefits of caching transient contents considering their lifetimes have been provided. By defining the freshness of a transient content and considering the tradeoff between content freshness and multi-hop communication costs to retrieve content from the content source, an analytical model is proposed for content routers to adapt their caching strategy. Caching benefits of transient contents are verified through simulations and the results demonstrate that if caching strategy can adapt to the two key system variables: a content's lifetime and received rate of requests for the content, caching transient contents is indeed possible while making it possible to fetch transient contents directly from content routers with acceptable reduction in content freshness, depending on content lifetime.

Quevedo et al. have analyzed in [60] how the in-network caching mechanisms of ICN, specifically caching mechanisms implemented in the CCN architecture can contribute in IoT environments, particularly in terms of energy consumption and bandwidth usage. The simulation results comparing IP and the CCN architecture in IoT environments have demonstrated that CCN leads to a considerable reduction of the energy consumed by the content producers and to a reduction of bandwidth requirements. The results have also highlighted the flexibility for adapting current ICN caching mechanisms while targeting specific requirements of IoT.

NDN-BMS [61] is an application-driven project that has proposed a content-centric building management system (BMS) design and has implemented an NDN-based BMS to be used by facility management personnel. The prototype system has been deployed on the University of California at Los Angeles (UCLA) campus testbed that captures, archives, and visualizes time-series data generated by industry standard sensors located in the campus buildings. In NDN-BMS, the sensor data namespace is based on naming the things that are measured, such as electrical current and chilled water flow, according to the physical hierarchy of the building structure.

NDN has been used on a real IoT deployment scenario for the purpose of building automation in [62] where there are resource constrained nodes spreading over tens of rooms on several floors of a building. Experiment results have shown that ICN is indeed applicable in the IoT deployment in terms of energy consumption.

A cooperative caching side-protocol for NDN, named as NDN+CoCa has been designed and implemented in [63] to enable distributed cooperative caching of IoT contents while offering low energy consumption. Relevant IoT contents can be available at any time even the IoT devices remain in deep-sleep mode most of the time because of the local in-network caching of NDN. The proposed protocol NDN+CoCa exploits both the content names and interplay between deep-sleep capabilities and content caching on IoT devices. Using a theoretical model, auto-configuration mechanisms to enable practical ICN deployments on IoT networks with NDN+CoCa are also designed and implemented.

To develop unified IoT platforms demonstrating the potential of using ICN networks to support IoT applications, a unified IoT platform is proposed in [64], named as ICN-IoT, leveraging the salient features of ICN architectures. Two ICN architectures – MF and NDN have been explored in details to support IoT in two different realistic application scenarios such as a smart building scenario consisting of stationary IoT devices and a smart campus bus scenario focusing on mobile IoT devices. A detailed performance comparison of NDN and MF in their capabilities of supporting IoT scalability and mobility has revealed that both the architectures have comparable delays and throughput, while MF incurs less overhead in terms of both routing table size and the number of control messages.

In order to quantify the benefits from hierarchical content naming, transparent in-network caching and other ICN characteristics in a sensor environment, a home automation system that supports wireless sensors for various purposes, such as temperature, humidity and energy consumption measuring is implemented using CCN in [65].

2.3.2. ICN in VANETs

Host-centric IP-based protocols face significant challenges in adapting to vehicular applications. Among many challenges, VANETs require the distribution of a significant amount of data among heterogeneous players, often with poor and intermittent connectivity, under high mobility, harsh signal propagation, and sparse roadside infrastructure support. The ICN paradigm particularly suits the vehicular ecosystem by natively privileging content rather than nodes, and inherently adjusts with mobility and sporadic connectivity issues by leveraging its in-network content caching mechanism [82].

Although the basic features of ICN can be potentially beneficial to address the peculiarities of VANETs, there exist several research efforts discussing about the proper ICN deployment in VANETs while addressing the scope of ICN namespaces matching, ICN routing and forwarding strategies together with in-network caching and identifying the related open research challenges for ICN-based VANETs. Some of the research works addressing these challenges for deploying ICN-based VANETs are described in the following.

Early research works of ICN-based VANETs [83,84] have shown the benefits of ICN comparing with the IP-based solutions in VANETs to effectively handle vehicle mobility, intermittent connectivity and data security. The research in [83] has explored the NDN to design new protocols for vehicle data collection and designed a highly efficient, reliable and secure vehicle data collection system, named as DMND which performs better than the Mobile IP solutions while demonstrating high efficiency for content collection and resiliency while handling mobility.

For ensuring reliable and low-overhead content discovery and delivery in VANETs, the work in [84] is a pioneer in proposing a framework, dubbed as CRoWN for deploying ICN in VANETs.

The applicability of NDN as a replacement of IP for direct Vehicle-to-Vehicle (V2V) networking has been investigated in

[85] specifically for the vehicular applications that share real time traffic information for safety purposes. The proposed named-based approach allows vehicles to collect and disseminate traffic information among themselves using content names that are defined a priori during the application development and understood by all the vehicles so that the request reply model of data exchange can be best suited in the ad hoc environments.

TCP/IP protocol stack has been mentioned as inefficient and not scalable for inter vehicular communication (IVC) in a vehicular information network that implements a myriad of applications related to vehicles, traffic information, drivers, passengers, and pedestrians in [86]. While addressing the efficiency and scalability issues of the IVC, NDN model has been mentioned as highly suitable for the IVC scenario because of its hierarchical content naming scheme, flexible content retrieval and caching support. A novel vehicular information network architecture is proposed aiming to improve content naming, addressing, data aggregation, and mobility for IVC in the vehicular information network. To support pull-based and push-based communications, a hierarchical content naming scheme is proposed. To make NDN more suitable to handle the large number of vehicles and large-scale information involved in the vehicular information network environment, a data packet aggregation scheme and an interest packet segregation scheme have been proposed. Finally, a distributed mobility management scheme is proposed that relies on the content name and the vehicle's moving information.

Another NDN-based architecture, named as Vehicular Named-Data Network (V-NDN) has been proposed in [87] that enables the vehicles to effectively communicate through any available channels while making the system highly resilient to any disruption.

A prototype of V-NDN has been designed and implemented later in [88]. The conducted experimentations via both demonstration and simulation have revealed that, V-NDN is able to bring considerable benefits to vehicular communication by removing the isolation between applications and network transport, allowing forwarding nodes to handle content based on application needs.

Amadeo et al. have discussed and scrutinized the potential of the ICN as a networking solution for connected vehicles [82] reviewing the core functionalities of ICN; such as named content retrieval, innate multicast support and in-network content caching. Their analysis has shown that ICN-based VANETs promise enhancements in the areas of application, mobility, and security. Vehicular applications are content-oriented in nature as they address contents and do not care about the producer identities of the contents. The generated contents in VANET are relevant to a given location and/or to a given time interval and ultimately intend for groups of recipients. ICN matches the described vehicular applications' pattern better than the current Internet because of its named content and routing by name features. Content discovery becomes simpler because ICN does not require name-to-IP-address resolution and the continuous connectivity with the producer. Additionally, ICN simplifies data retrieval from multiple consumers by aggregating requests intended for the same named content. The usage of named content in ICN also simplifies mobility support. The anycasting and in-network caching mechanisms of ICN allow vehicles to retrieve content from the most convenient producer or storage point reducing the latency and network traffic. In ICN-based VANET, vehicle can also serve as a link between disconnected areas and enable communications even under intermittent connectivity because of the store-carry-and-forward mechanism supported by ICN. The content-based security in ICN can natively provide security supports in VANETs as the trustworthiness in VANETs should be based on the content instead of the reputation of the content providing entities because of the ephemeral nature of vehicular communications. The major challenges for ICN deployment in upcoming VANETs are listed as: interoperability of ICN with existing and underway connected vehicle standards and technologies, meeting the delivery requirements of vehicular applications such as short latency and high reliability for providing QoS support, supporting in-network processing operations and novel naming schemes for the tremendous amount of generated contents and developing business models to define the economic incentive mechanisms among the involved stakeholders.

To address the challenges that the current vehicular networks face to increase capacity, support mobility, and improve Quality of Experience (QoE), a novel framework for a content-centric vehicular network (CCVN) is proposed in [89]. A content- centric unit (CCU) is introduced to work with the conventional onboard units (OBUs) and roadside units (RSUs) and a group of CCUs are distributed in the network to store the replicas of vehicular contents. The CCU delivers vehicular contents from OBUs and RSUs in the network. Vehicles request contents based on their interests and the contents are managed by their naming information. An integrated algorithm is presented for caching contents in the content stores, keeping pending interests, updating forwarding information and delivering contents to vehicles with the help of CCUs. Contents are stored according to their priorities determined by the vehicle density and content popularity and the pending interests are updated based on the analysis of the transmission ratio and network topology. The proposed framework increases network capacity, supports vehicle mobility and demonstrates supremacy comparing to the conventional scheme (where the contents are stored based on request times in all the vehicular networks without considering the different popularity in the coverage area of different CCUs) in terms of hit rate and delay.

In recent years, the native in-network caching capabilities and the mobility support of ICN have mainly attracted the attention of researchers in the context of ICN-based VANETs. Some of the research that mainly focus on ICN-based VANETs deployments centering the caching capabilities of ICN are surveyed in the following.

The proactive content caching scheme in [67] uses transportation systems to provide high-quality and highly reliable video delivery services for mobile users, especially for the train passengers. Content servers are placed with cache capabilities at every train and station. The scheme mainly works in three steps. At the first step, the requested content is divided into several segments that are delivered separately to relay points, such as the train stations and/or bus stops, according to the vehicle's schedule. At the second step, the transportation vehicle receives the distributed content segments at the relay points through high-speed wireless transport schemes, such as wireless LANs. Finally, the transportation vehicle streams the received content segments to mobile users inside the vehicle via wireless LANs. A delivery scheduler, called as a "smart scheduler" determines the content quality and the amount of content segments and also selects delivery locations and timing. Prototype systems are also developed based on hypertext transfer (HTTP) protocol and CCN/NDN protocol and evaluated in two field experiments that use actual trains comparing with the traditional video streaming over cellular networks. The performance comparison shows that the developed prototypes perform better than the traditional video streaming while serving high quality video without any interruption to 50 mobile users simultaneously.

The design alternatives of ICN-based VANETs are investigated in [58]. The most important ICN-specific design options such as caching policies and content source selection policies have been analyzed and a multi-hop broadcast, name-based forwarding mechanism for urban VANETs has been proposed here. The performance gain achieved from the pervasive caching in the VANETs has been found significantly better while comparing to the edge caching that offers low performance improvement.

The research in [68] handles the challenge of placing the right content at the right node in-time for VANETs. To address this challenge, a PRoactive Caching approach, named as PeRCeIVE, has been proposed to cache contents within Information-Centric VANETs proactively. PeRCeIVE enables an efficient content distribution mechanism facilitating RSU-based caches while considering current velocity and the heading direction of an exemplary vehicle. PeRCeIVE has demonstrated that direct placement of content can improve network performance by distributing a minimal number of content replicas one-hop away from the requesting consumer.

2.3.3. *ICN in MANETs*

The TCP/IP stack has witnessed many modifications and supporting protocols to cater to the unique challenges of MANETs. Over the past thirty years, significant research efforts have been invested in improving MANET operation under signal interference, multipath fading, frequent movement of ad hoc nodes causing dynamic change of topology and intermittent connection between nodes. In this regard, the ICN research community has investigated the gain in adopting content-centric principals in the management of MANETs. As such, ad hoc and mobile networking would benefit from separating contents from their hosts, reducing the requirement of host location-awareness and enhancing the mobility of terminals [90]. ICN's receiver-driven model of communication, along with the location independent naming scheme and in-network caching mechanism, have extended the deployment scope of ICN from fixed networks to dynamic ad hoc networks. There are several research efforts that analyzed the Information-Centric Mobile Ad Hoc Networks (ICMANETs) in the context of naming, routing, caching, transport and security mechanisms while taking NDN as the candidate architecture of ICN.

ICMANET has been mentioned as a new, cross-cutting, gradually forming research architecture in [90] as the ICN is capable of offering a superior architectural support for MANETs. Considering routing as the most promising feature of ICMANET, a conceptual routing model is proposed for content routing that comprises of four tuples such as receiver-driven mode, caching, node and content retrieval. The content routing schemes of the ICMANET literature has been surveyed and also classified into three categories from the perspective of content discovery which are: proactive, reactive and opportunistic routing. Finally, the existing general research challenges of content routing in ICMANET are summarized.

The CCN architecture [27] has been extended in [92] to propose a reliable and secure content distribution approach for disruptive networks having high mobility and lossy channels. The proposed approach aims to achieve scalability, security, and efficient network resource management in large scale disaster recovery and battlefield networks. Security has been increased by using digital signature and public key infrastructure (PKI) data encryption. Additionally, network resources such as channel bandwidth and power are saved by taking advantage of efficient in-network caching of CCN.

The applicability of NDN model in the military communication networks to conserve bandwidth, avoid looping, provide message security, and support interoperability has been briefly analyzed in [93].

The social network-based security scheme [94] has solved both the authenticity and integrity problem for the ICMANET while demonstrating scalability and practicability being deployed in a large social network. The social network (and associated the graph) allows the content consumers to retrieve the public-key of the content producer using a trusted chain that leads to a cached identity bundle. The scheme has been evaluated in a large social network and reported its performance in terms of scalability and practicability. For providing local and efficient content retrieval, a content-centric routing protocol for MANETs, named as SCALE [91], prioritizes routing based on content names.

Research in ICN-based MANETs has paid significant attention to explore the applicability of in-network caching schemes in MANETs because the caching and mobility in ICN play the major roles for ICN based MANETs deployments. One of the early research works in ICMANET proposes a NDN-based caching strategy [69] that uses an extra parameter, named as content interval whose value does the selection of caching nodes in order to distribute contents in the cache routers along the content delivery path.

To leverage the broadcast nature of radio channel to improve the cooperative caching gain of content-centric ad hoc networks, the broadcasting-based neighborhood cooperative caching (BNC) strategy [70] selects cache nodes among the neighborhood and tries to make the best use of the broadcast nature of radio channel without occupying extra wireless channel resource.

Aiming at reducing the cache redundancy in MANETs, the caching strategy named *l*ess space *s*till faster (LF) [71] selects the cached contents based on content popularity instead of indiscriminate caching while showing efficiency in terms of response time, network traffic, and cache hit ratio.

The energy efficient content distribution scheme in [72] delivers contents in a unicast manner while minimizing the flooding overhead claiming that in case of energy efficiency, unicast content delivery outperforms broadcast delivery in MANET unless topological changes in the network incur too much packet flooding overhead.

3. Unique challenges of caching in ICN

Content caching and distribution are intrinsic ICN capabilities, potentially performed by all network routers. Before elaborating on the unique aspects of caching in ICN, it is important to define each of its components. ICN Caching is often named in-network caching and Information-Centric Networks (ICNets) are typically considered as networks of caches [52,104], spanning the following basic components as illustrated in Fig. 2:

(1) **Content publishers** are network nodes which publish named content such as servers, tablets, sensors and RFID tags.
(2) **Cache routers** are any network routers in ICN capable of caching content, often referred to as content routers.
(3) **Content consumers** are network nodes which subscribe to (or request) named content. Content consumers can be personal computers, servers, smartphones, sensors, etc.

3.1. *Merits of ICN caching*

Inherent in-network caching in ICN has several advantageous features, chief among them are:

3.1.1. *Leveraging QoS for users*

The goal of in-network caching is to improve network performance through efficient content dissemination and retrieval. This

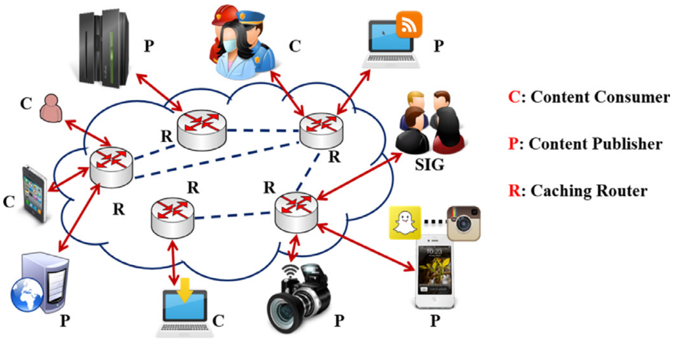


Fig. 2. An Information-Centric Network and the interplay of consumers, producers and caching routers.

Table 1
Comparison Between Traditional Caching And ICN Caching.

Factors	Traditional Caching	ICN Caching
Caching layer	Caching occurs at the application layer as an overlay [7,11]	Caching occurs at the network layer, thus called in-network caching
Location and topology	Caching nodes are predefined and fixed, whether as surrogate servers in CDNs, or web proxies for web caching. The cache topology is usually built as a linear structure or hierarchical tree structure [105]	Cache nodes in ICN are ubiquitous, and caching points are neither fixed nor predetermined. Thus, ICN caching topologies are often arbitrary
Granularity of cache content	Mostly macro-caching approaches, as they aim to cache whole content such as web pages or entire files [108] with some exceptions, such as video content, which are divided into chunks over HTTP in DASH implementations	Content is divided into pieces, named content chunks in ICN, and thus caches store chunks instead of the whole content resulting in finer content granularity. Hence, ICN adopts a micro-caching approach [108]
Cache transparency and redundancy	Traditional caching systems are application dependent. Different copies of the same content have different names in different domains as web content follows domain-based naming conventions	In ICN, content has a unique name and content caching is based on this unified and consistent content name. Different applications can use the same cached content. Hence, ICN caching is transparent and independent of applications

performance gain is achieved by reducing the delay experienced by users in retrieving content, in an infrastructure that moves data closer to regions of interest.

3.1.2. Reducing load on the publisher

Multiple copies of the same content are available in ICNets since any router can directly respond to content requests if it has cached the content, and the content request does not have to traverse to the publisher of the content. Furthermore, this reduces the criticality of “single point of failure” challenges, both in terms of security and reliability.

3.1.3. Reducing network traffic and congestion

Since any network router can cache content, overall network traffic is reduced as requests do not need to be routed to the content publisher or a specific surrogate server, thereby alleviating network congestion.

3.2. Traditional caching vs ICN caching

Caching is not unique to ICN architectures, and has been heavily investigated and implemented under the current Internet paradigm. However, there are fundamental differences between caching in ICN in contrast to the Internet, which are summarized in Table 1. Most notably, caching has been heavily adopted in CDNs to distribute request bottlenecks from the main publisher to designated surrogate servers, which are typically static [105]. On the other hand, web caching is heavily used in the current Internet to reduce traffic flow beyond a local network for “repeatedly” visited/requested web content.

Today, many advancements in Mobile Edge Computing (MEC) are leveraging computing power at the edge of the network to aid user-centric caching mechanisms. In these architectures, caching in Cloudlets and Fog-nodes is presenting new opportunities to leverage responsiveness and bring more popular content towards the edge [106].

However, there has been quite some contention on the merits of caching at the edge vs network core, with significant debate on the underlying network technologies used, and the mere potential of rolling out a new networking paradigm to replace/augment the current Internet. Fayazbaksh et al. have presented a well-cited study [107] on the benefits of caching at the edge, and argued for avoiding inherent complexity in caching at the core. However, most of their study and conclusions are built on LRU caching, which is quite dated and has been improved on in many caching schemes. Sections 5–7 cover many of the newer caching schemes in depth.

More importantly, the argument for building on static allocation of caching, that is somehow mandated by centralized monitors/controllers, brings significant complexity and time delay in network operation, that defeats the Internet’s philosophy in keeping all network intelligence at the far edge of the network.

In contrasting intrinsic caching paradigms, as envisioned in ICN, versus controlled/mandated caching that is rarely close to real-time, is an ongoing research challenge that requires significant quantification and exploration. There is a significant push from the ICN research community to standardize the management of caching, across the hierarchy of network entities, to enable a seamless adoption and operation across heterogeneous networks. While the argument is not settled to either research “camps”, it remains a highly investigated topic that will yield more insights into the potential gains in adopting inherent caching mechanisms.

Furthermore, we present in Section 7 a primer quantitative assessment of caching schemes, and contrast it to static caching operations to highlight the gain and potential yield in improving User experience and network operation.

4. Inherent challenges in ICN caching

The design of caching on a network that is centered on content, significantly enhances the spectrum of design choices. In ICN caching, there are three phases in caching to setup, coordinate and manage in-network caching system: designing cache placement policies, content replacement policies, and modelling the topologies of cache networks [52,108].

Cache placement policies face two challenging issues: which content should be stored in ICN nodes (i.e. what to cache, since content is the pivotal factor in connections) and which cache nodes of the network should be selected as caching points (i.e. where to cache, since most nodes inherently hold caching capacity). The most cited and criticized cache decision policy in ICN literature is Cache everything everywhere (CEE) [27], where cache content and cache node are selected indiscriminately along the content delivery path between content source and consumer.

Many caching schemes adopt policies based on content popularity [34,36,37,39,109–112], some follow collaborative approaches [37,39,112–116] while others use topology related metrics [112,117–119] for improving caching performance; all of which are detailed in Section 5.

Content replacement policies evict content whenever the cache reaches capacity and new content arrives to be stored. Least Recently Used (LRU) is the most often used cache replacement policy in ICN literature, where cache eviction is accomplished by using a queue such that the item which is queried least recently has to be removed to make room for a new arriving item [35,38,48,108,112,120–122]. A few schemes use Least Frequently Used (LFU) cache replacement policy which discards contents used less frequently [109,122–124], and recent efforts have contrasted their performance gains and detailed models [125].

Analyzing caching network models in ICN presents new challenges since all routers are cache equipped. ICNets are thus mostly represented as a large scale network of caches [52]. Although previous studies have considered the network of caches in ICN under hierarchical tree-structured networks [108–110,120,121,123], the topology of ICN cache networks should no longer be confined to hierarchical trees.

Hence, the high dynamicity of general cache network topologies in ICN obsoletes the fixed parent-child relationship of cache networks [119], necessitating a transition to larger arbitrary graphs.

5. Taxonomy of caching paradigms in ICN literature

We hereby present a taxonomy of caching paradigms in ICN, with two main goals. First, distinguishing the main factors that impact functional operations and ensuing performance goals across caching systems. Second, enabling a thorough discussion of cache design strategies for future models, especially for cross-paradigm schemes that aim for specific operational goals such as improving QoE or reducing overall content redundancy.

Our taxonomy is presented in Fig. 3. We adopt four main classifiers for mainstream caching schemes: popularity of cache content, location of the cache nodes in the network topology, operational collaboration among cache nodes, and finally caching decisions based on content delivery paths. We also define the distinguishing factors of the main classes to further categorize the classes into subclasses Furthermore, we demonstrate notable caching schemes under each sub-class, to guide early researchers in this area, which is elaborated upon in Section 9.

5.1. Content popularity-based caching schemes

Caching content based on content popularity is a crucial strategy for improving performance in ICN caching schemes [34–39,109–112,115,116,126]. Many schemes propose caching strategies based on the popularity of contents and we classify these schemes as popularity-based caching schemes.

Popularity of a content is highly correlated with the requesting frequency of that content made by consumers. We divide the content popularity-based schemes into two categories: static (consistent) content popularity-based schemes or dynamic (inconsistent) content popularity-based schemes. Caching schemes based on static content popularity are required to define a specific threshold value for consumer requests, beyond which content is considered popular [127–129]. Static popularity approaches are unrealistic as they usually generate out of date calculations, incapable of reflecting the most recent histories of content request statistics and consequently misuse available cache capacity while providing lower overhead.

On the other hand, dynamic content popularity is defined by the number of consumer requests for content during a typically short time interval. In this case, caching algorithms are invoked at regular time intervals based on the most recent histories of content request statistics while properly utilizing cache capacity [34]. As content popularity varies over time, dynamic popularity approaches keep only the most recent access histories to reduce the space complexity. The invocation interval for a caching scheme is chosen based on the total arrival rates of content requests at the access cache routers connected to the requesting consumers and the changing frequency of the content popularity. A disadvantage of dynamic popularity approaches is the constant comparison of content request rates. A shorter invocation interval to calculate dynamic content popularity can incur a higher overhead but a longer interval is suited only to stable access patterns.

In Fig. 4(a), the basic idea of content popularity-based caching schemes is depicted. The figure shows the strategy of many popularity-based schemes to cache most popular content at access routers ($R_{1,}R_{2,}R_3$) near end users and gradually less popular content near the server at intermediate (R_4) and core router (R_5) aiming to increase cache hit rates, and reduce both the delay to retrieve requested content and publisher load [34,36,39].

Fig. 4(b) illustrates the caching strategy to cache diverse Internet content; comprised of web pages, files, user generated content

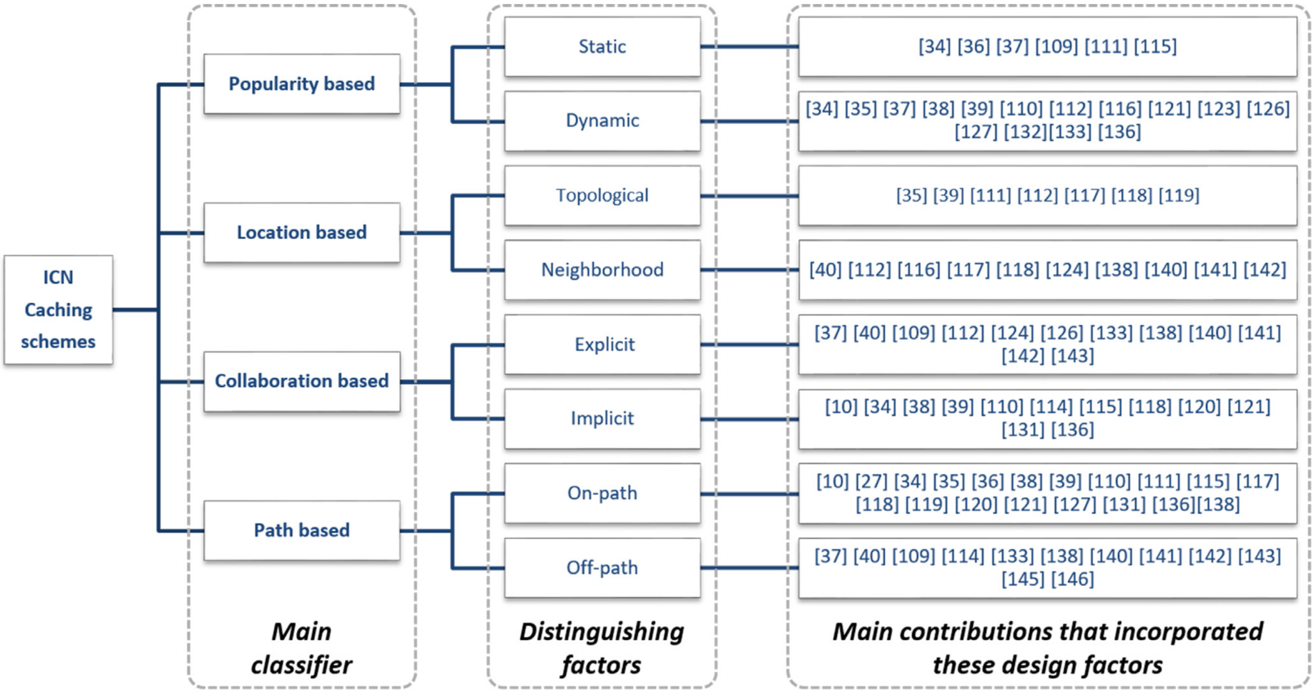


Fig. 3. A functional taxonomy of caching paradigms in ICN literature, with notable caching schemes under each category.

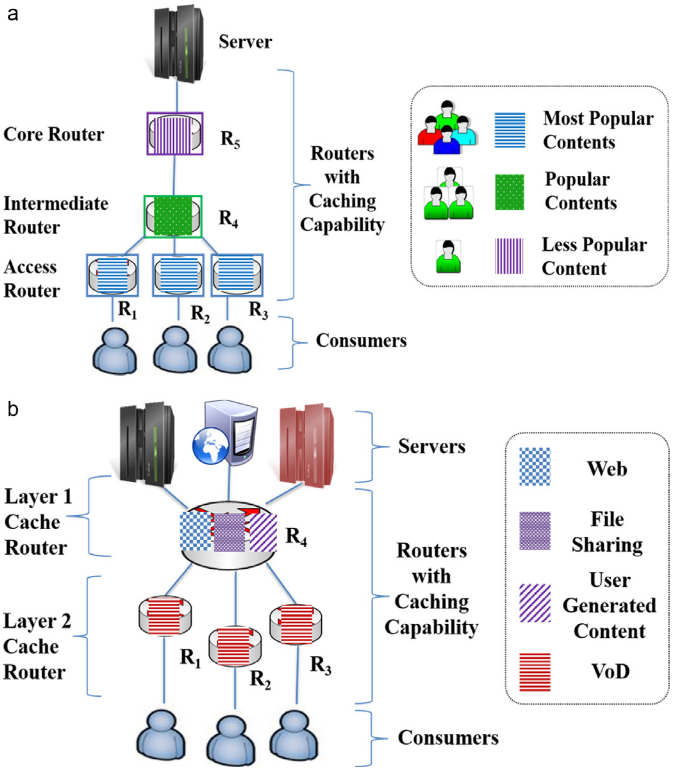


Fig. 4. (a) Content popularity-based caching schemes – Gauging popularity. (b) Content popularity-based caching schemes – Diversity of content.

(UGC) and video on demand (VoD) content. Such caching schemes are typically based on content popularity distributions, traffic patterns, population size and size of content in a generic two-layer hierarchical cache network [123]. The performance evaluation of hierarchical caching scheme reveals that caching performance mainly depends on the content popularity distribution, and such caching schemes suggest caching VoD content at access routers ($R_{1,}$ $R_{2,}$ R_3) near consumers and the remaining three types of contents (web pages, file sharing & UGC) in the core router (R_4). This is also guided by the large storage capacity at the core, to achieve efficient memory-bandwidth tradeoff through caching.

In the following, we describe some of the core popularity-based caching schemes. The caching schemes which cache contents based on both static and dynamic content popularity are described under the static content popularity-based schemes.

5.1.1. Static content popularity-based schemes

A collaborative caching and forwarding scheme for CCN which adopts popularity-based content ranking is proposed by Guo et al. in [37]. Content popularity guides the caching scheme, where popularity is measured in a distributed manner by content routers. Each router has a new component named Availability Information Base (AIB) for caching contents and forwarding content requests.

To deal with the inconsistent popularity ranking produced by different routers in an autonomous system (AS) of realistic networks, the authors design a distributed, self-adaptive algorithm for content store division in the content routers. The proposed scheme performs better than other schemes, named as hierarchical caching scheme and local collaborative caching and forwarding schemes, even though it has a high percentage of popularity inconsistency in terms of average access cost.

Focusing on minimizing inter-ISP traffic and average access latency, Li et al. propose two popularity driven dynamic caching schemes for CCN in [34]. As a baseline, the authors first solve the optimal replica-placement problem and then propose two popularity-based algorithms, named TopDown and AsympOpt, where the router makes dynamic caching decisions according to content popularity, measured by the aggregated request statistics of the sub-tree rooted at that router, and caches the most popular content near end users. Both schemes achieve performance comparable with the optimal solution and show superiority compared to popularity-blind indiscriminate caching [27] and uniform probability caching. Moreover, the proposed schemes show effectiveness, stability and scalability for several important caching design issues such as network topology, cache size, access pattern and content popularity. However, the performance gains of the popularity based schemes over the baseline algorithm tend to decline for real trace input as the authors analyze web requests only as their inputs.

Draxler et al. [109] propose three content popularity-based caching strategies for efficient usage of cache space and avoiding redundant content caching. The proposed popularity-based caching strategies, named Basic, Adapted and Stacked work in a hierarchical fashion using a tree-topology based on content popularity. In Basic, the topmost cache node of a tree stores the most popular items and children cache nodes store other items in a distributed way according to their popularity, such that the less popular items are stored down to the lower levels of the tree. In Adapted, among four levels of a tree topology, the two lowest levels store the most popular items and then the popular items go to the topmost cache node and after that, on the level below of the topmost node. Stacked strategy is a combination of Basic and Adapted and forms a bundle of cache nodes, where the bundle consists of one node from a higher level in the tree and all its children nodes. In Stacked inside a bundle, the most popular items are stored on the topmost cache node and less popular items are distributed among the children cache nodes in the level below. These bundles are stacked to incorporate all the levels of the tree hierarchy. The proposed popularity-based schemes perform better than LRU and LFU policies in terms of cache hit efficiency, mean hop delay and power consumption while considering abstract model of network topology only.

The node-content pass probability (NCPP) caching scheme [111] considers content popularity as well as the node utilization ratio (NUR) for calculating the NCPP values of cache node that are used to take caching decision. NUR value of a cache node defines the traffic load distribution characteristics of the network and the frequency of a cache node to be selected for relaying contents [130]. Requested contents are ranked based on their popularity values and the NCPP scheme caches the most highly ranked content (content having highest popularity) at the node having the largest NCPP value and the second most highly ranked content (content having the second highest popularity) at the node having the second largest NCPP value and the same procedure is repeated for rest of the contents.

An in-network video caching policy, named StreamCache [115] aims to improve users' QoE in terms of average throughput for dynamic adaptive video streaming over ICN. For better cache utilization and higher throughput, StreamCache caches those video segments which are frequently requested as caching popular video contents can increase the probability of cache hits.

StreamCache operates in a distributed way along the cache routers while caching video contents with different sizes and bit rates and makes caching decisions locally at each router based on the aggregated video request statistics. In order to cache frequently requested video contents, the edge cache routers start to collect request statistics on each round of the caching scheme and makes caching decisions based on the collected content popularity-based statistics and proceeds to the next round. Edge routers and intermediate routers calculate the caching utility of video contents to derive the importance of caching on improving the average throughput using a utility function where the utility function is designed considering the probability of requesting a video content, bit rate of the content, average round trip time of video contents and the length of the video content. StreamCache relies on the summarized statistics aggregated from the downstream cache routers to calculate the caching utility and makes greedy choices to make caching decisions to fill the cache spaces of the routers. The StreamCache popularity-based caching policy performs better compared to indiscriminate scheme CEE [27] and ProbCache [131] in terms of average throughput as StreamCache better distinguishes between popular and unpopular contents but does not consider any cache replacement policy.

To improve data dissemination in ICN, PopCache [36] caching scheme allows each router along the data forwarding path to apply a caching probability to cache content according to the popularity distribution of content. The main motivation of the proposed scheme is the observation that allocating only few cache spaces to store highly skewed popular contents and large cache spaces to store contents with flat popularity distributions produce higher cache hit ratios and reduce server hit rates. PopCache is built on two principles: (1) routers closer to end users should store the most popular content and other routers should store unpopular content, and (2) content should be cached as distributed as possible across content routers. This scheme performs better than indiscriminate caching [27] and path-capacity based caching [131], in terms of expected round-trip time and server hit rate.

5.1.2. Dynamic content popularity-based schemes

A value-based cache replacement policy named Least Valuable First (LVF) proposed by Al-Turjman et al. [132] considers user input to decide the value of content according to the content's popularity, delay sensitivity, and age. The demand for a content is measured by the frequency of requesting that content within a specific geographical vicinity and operational cycle. The delay sensitivity of content is measured by the specification of the requesting consumer indicating how long the consumer is willing to wait for the content. The age of content is measured by its Time-To-Live (TTL) count either assigned by the publisher or updated by any intermediate node of a network.

In the LVF approach, the authors devise a novel dynamic utility function that sets the value of each content based on users' requests such that, the least valuable content is dropped first while cache eviction occurs. LVF approach outperforms First-in-First-out (FIFO) and LRU approaches, in terms of cache hit ratio, time to hit cache data, average in network delay and publisher load for varying content popularity and connectivity degree of the network nodes.

A simple, decentralized and incrementally deployable content caching scheme, named WAVE [38], dynamically adjusts the number of chunks to be cached for efficient content delivery based on content popularity. In WAVE, an upstream router suggests the number of chunks to be cached at its downstream router which exponentially increases as the popularity increases for a content file. WAVE uses the LRU approach for content replacement. Although the caching unit in WAVE is a chunk, to find a victim chunk for replacement, WAVE maintains the access history of content on the file level, to reduce control overhead. WAVE performs better than several in-network caching schemes, such as indiscriminate caching [27], UniCache (content chunk is cached at one router along the data delivery path) and Fixed probability caching in terms of average hop count to content delivery, link stress, inter-Internet Service Provider (ISP) traffic reduction, cache hit ratio and cache replacement count. WAVE does not consider content having multiple-sources, which is a somewhat common abstraction.

A popularity guided age-based caching scheme is presented by Ming et al. in [39] to reduce network delay and publisher load by increasing aggregate cache-hit probability. The scheme dynamically configures content's age. Content location and popularity guide the content age following the two rules: the more popular a content replica is and the closer it is to the network edge; the longer age it has. The age-based scheme spreads popular content to the network edge longer which ultimately reduces network delay and publisher load and additionally removes content replication at intermediate network routers for efficient utilization of network storage. This scheme performs well compared with the FIFO and LRU schemes under realistic network topologies, as it requires less access latency and reduces server load.

The collaborative caching scheme, named as RPC [110] caches popular video contents on the edge routers closer to users based on the routers' position or level in the network topology. The caching scheme can adapt itself dynamically according to the user request thus requiring no prior knowledge of video content's popularity. The cache routers keep track the access counts for each content locally and stores all the access counts of all the requested contents as a paired key-value structure (content name; access count).

The caching decision of a content at a cache router depends on three factors: the access counts of the requested content which determines the popularity of content, the topology value of the cache router and the caching threshold value of the cache router determined by a specific method. The caching threshold value of the root router is assumed to be precomputed and the caching threshold value of each cache router is calculated considering its topology level and the root cache router's threshold value. Content is cached in a cache router either when the access counts of a content becomes larger the caching threshold value of the cache router or there is enough cache space left at the cache router to cache the content. RPC shows superiority comparing to the popularity blind scheme CEE [27] and age-based popularity caching scheme [39] while reducing server load and latency for small cache size.

Statistical methods of recording content popularity and ranking contents according to the number of historical requests for calculating chunk-level video content popularity such as in [115] has limitations especially for video applications. In the worst case, the former requested video chunks with high number of past requests can be judged as popular and selected to be cached, but are no longer required in the future. So, the cache replacement method, named Popularity Prediction Caching (PPC) [116] predicts the chunk popularity of the video contents of a same stream using the potential future requests other than the historical requests.

The PPC method uses two processes to predict the future popularity of the video chunks: Assist-Predict process that uses the request information of neighboring chunks for prediction and Self-Predict process that is based on the historical requests when there is no reference chunk to assist the prediction process. There are five system modules in a single cache node to support PPC scheme. The Request Handling Module (RHM) records all the incoming requests and maintains the prediction-benefit entries. The request recorder component in the RHM records the name, incoming time, and the incoming face of the request. The updater component in HRM periodically triggers the request recorder component to erase the historical records of chunks with request time gap larger than a predefined time threshold value.

The Popularity Prediction Module (PPM) is the core module to realize the PPC scheme and accepts two types of inputs; the incoming content name and the cached content name and produces the predicted future popularity of the incoming content and the cached content as outputs. Whenever a video chunk reaches the cache node and the cache space is full, PPM module is visited to make prediction. On the other way, the cached contents periodically visit the PPM module to update their future popularity. The prediction classification component in the PPM module classifies the input content name into self-predict or assist-predict based on the historical request records in the request recorder component. The predicted future popularity calculated by the PPM works as the input of the Replacement Decision Module (RDM). The cached popularity component in RDM is used for keeping record of the future popularity of cached content, and the incoming popularity component is used to keep record of the future popularity of the newly incoming content. The RDM compares the least popular record in the cached popularity with the record in incoming popularity and the chunk (either the least popular cached one or the incoming one) having the least future popularity is evicted. There is also a module named Deletion Handling Module (DHM) that notifies an upstream cache node the decrease of future requests of a chunk from an interface. The PPC shows superiority comparing to the FIFO, LRU and LFU in terms of average cache hit ratio and average delay as it caches the future most popular video chunks while increasing the system complexity linearly.

To remedy cache utilization challenges, the Chunk Caching Location and Searching scheme (CLS) [121] addresses two objectives: reducing the impact of cache replacement errors and achieving cache exclusivity. Cache replacement error refers to the eviction of a more popular content to cache less popular content, and cache exclusivity can be achieved by reducing unnecessary repetitious caching of the same content at multiple levels in a cache hierarchy. To meet these objectives, CLS pulls-down and pushes-up content chunks according to content popularity. Whenever a chunk gets a cache hit at a cache router in a certain level, the hit content chunk is pulled-down to a router in the preceding level (immediately below) along the content delivery path. If a content chunk becomes more popular, ultimately CLS brings the content chunk to a leaf router, with each request pulling-down the chunk one hop closer to the consumer.

On the other hand, when a content chunk is evicted from a cache router from a certain level, the chunk is pulled-back to the cache router in the upper level. Thereby, a least popular content chunk can be pushed back up until it reaches the content server. During the content pull-down and push-up strategy, a caching trail is created to assist content searching. This content pull-down and push-up scheme avoids cache replacement errors and achieves cache exclusivity by ensuring that there is always a copy of a content chunk cached on the path between a content server and a leaf cache router.

For better utilization of cache space, Intra-domain Cooperative Caching (IDCC) scheme [133] caches more varied types of popular contents hierarchically in a local AS by reducing cache redundancy. IDCC uses probabilistic caching depending on inlet traffic speed, caching time and capacity, and conducts intra-domain advertisements for popular contents. Cache routers cooperate in a distributed way and store a number of hierarchical cached replicas reducing redundancy and caching more popular content within a local AS. Router load constraints are satisfied for load balancing by storing multiple replicas of very popular contents at nearby routers. Moreover, IDCC sets content age and increases the age every time a content is accessed so that more popular contents can stay at the cache nodes for longer time and avoid being replaced too quickly. Although IDCC outperforms CEE [27] and ProbCache [131] in terms of cache hit ratio, user delay and cross-AS traffic, yet it suffers from scalability for highly popular content, as the number of stored replicas in the AS depends on content popularity.

A generic two-layer hierarchical cache network in [123] caches realistic Internet traffic consisting of four different types of contents such as web pages, file sharing, user generated content (UGC) and video on demand (VoD) content based on the popularity distributions, traffic shares, population size and size of the contents. For evaluating the performance of the proposed cache hierarchy, the Independent Reference Model (IRM) [134] is adopted to analyze the significant differences between different characteristics of the different contents. In the independent reference model, the probability of a content request depends only on that content's popularity, not on the sequence of content requests that came before. The performance evaluations of the hierarchical cache network demonstrate that, web page, file sharing and UGC roughly behave in the same way requiring larger storage space and follow a Zipf popularity distribution law [135] with a low exponent. VoD content demonstrates contrasting behavior to other Internet contents, as it requires less cache space and dominates a larger traffic share. To significantly reduce bandwidth and publisher loads, it has been suggested that large cache capacity can be economically provided in the network core, at the upper-layer caches to cache web pages, file sharing and UGC whereas VoD content should be cached beneficially at the lower-layer caches near requesting hosts. Additionally, results on cache hit rates suggest that caching performance crucially depends on content popularity distribution, and it is important to establish the popularity distribution law of VoD contents, since these contents do not necessarily follow the Zipf popularity distribution.

An on-path caching strategy named Least Unified Value (LUV)-path [136] caches new contents with different probabilities to push popular contents to the network edge, near consumers, to improve in-network storage. In LUV-path, upstream routers near content providers have a higher probability than downstream routers to cache new contents. As downstream routers are less likely to cache new contents, they cache popular contents having greater requesting frequencies near consumers. According to LUV-path, if content cached at upstream routers gets increased consumer requests, later resulting in possible cache misses at downstream routers, then they are moved downward if they are actually popular. Hence, cache contents in upstream are more likely to be replaced and are usually less popular. Cache weight is assigned in LUV-path reflecting content popularity along with cache cost, where cost reflects the distance between the cache router and the content provider, reflecting the significance of that cache router. LUV-path is shown to be superior to FIFO and LFU under different network topologies and cache sizes, while it reduces average content retrieval latency and network traffic, and alleviates pressure on the content provider.

Another probabilistic caching scheme, named Prob-PD [127] uses a dynamic-content popularity approach to efficiently deal with on-path caching. Prob-PD considers three factors in caching decision: (a) content's popularity ratio on a cache node and (b) the distance ratio of the same cache node and (c) the content source. Any content's popularity is compared only once against other contents within a considered time interval (time between the arrival and first request for a content) to minimize the complexity of calculating up-to-date dynamic content popularity. Although Prob-PD shows superiority comparing to ProbCache [131], CCE/LCE (Leave copy everywhere) [27], Fixed probability caching, LCD [137] and Degree centrality scheme [119], attaining higher cache hit rates and lesser cache replacement rates depends on the nature of workload such as the total number of contents because of the method of content popularity calculation.

The Hierarchical Cluster-based Caching (HCC) scheme [112] considers the content popularity in the edge cluster in a two-layer hierarchical cluster-based cache topology. The two- layer hierarchical structure are called core layer and edge layer where the core layer does content routing and the edge layer does content caching based on cache router importance value and the dynamic content popularity value in a probabilistic way. The cluster head is elected by the other two types of router, namely gateway and member router using Weighted-based Clustering Algorithm (WCA). The WCA considers three different factors based on the different network scenarios and use cases to determine the cluster head router. The factors are: the reciprocal of the neighbor router degree of a router, the average transmission time to all reachable routers from a router and the weighted based number of hops of a router. The final score of a router is a weighted sum of all three factors and finally the WCA elects the cache router having the smallest overall weight value as a cluster head. The cluster head router forms the hierarchical cluster with a certain number of cache routers based on the transmission times required to send messages from the cluster head to the other routers. The major roles of the cluster head router are assigning node importance value to the cache router based on the betweenness centrality value and measuring the content popularity of the requested contents in the edge cluster. Access routers in the edge cluster are connected to the content requesters and responsible for counting and providing the request rate of contents to their elected cluster head. Based on the content popularity information reported by the access routers, cluster head makes a summary of requested content names and their popularity values within the cluster in descending order while making popularity classes using Zipf-Mandelbrot power-law distribution [135]. Each requested content belongs to a popularity class based on its popularity score and the highest popularity class is named as Class 1 and the next popularity class is named as class 2 and so on. The contents lying in some the top percent of popularity classes are considered more popular, the remaining contents are considered less popular and these two groups of contents are treated differently while considered for caching.

Finally, the cluster head router computes a probability matrix considering the router importance and content popularity and sends the probabilistic caching decisions to the important cache routers so that the more important routers cache more popular contents and less popular routers cache the less important contents. The HCC scheme performs better than the LCE [27] and LCD [137] in terms of average hop reduction and average router hit while having extra communication overhead.

In order to make optimal caching and routing decisions, the network coding-based cache management (NCCM) algorithm applies linear network coding (LNC) in [126]. The collaborated content routers report content request statistics periodically within a fixed time period to a software-defined networking (SDN)-based controller. There are two modules in the controller to execute the NCCM scheme: the first one calculates the popular content set and the second one decides the caching strategy and content routing based on the network status. In case of using LNC in ICNs, each content is firstly divided into data chunks of fixed size and the LNC-enabled servers generate the coded data chunks of each content. So, the data chunks cached in the routers and transmitted in the network are linear combinations of the original data chunks. The requesting users for contents only need to acquire a sufficient number of coded data chunks from any set of content routers to recover the original content as different coded data chunks are linearly independent. To facilitate the popularity prediction of the contents, each content router (which is an open flow switch) maintains a counter for retrieving the local content request statistics, sends the request statistics collection to the controller and the content popularity is inferred directly from these statistical information using the measurement module of the controller. Hence, the controller determines the local popular content request set for each of the router and set of popular contents for the whole network. Assuming that most traffic in the network belongs to popular contents downloading, the NCCM scheme tries to optimally cache the coded data chunks of popular contents within a threshold value to minimize both the network bandwidth cost and cache cost by jointly optimizing caching strategy and content routing. In case of the unpopular contents, the controller or edge content routers route the content requests to the original servers.

An analytical and simulation based study is conducted to evaluate the caching performance of CCN considering several aspects such as content popularity, cache size, catalog size (number of files and their respective average sizes), network topologies, single-path and multi-path routing strategies of the content requests to the cache points and several cache decision and cache replacement policies in [35]. The simulation results demonstrate that the impact of network topology is quite limited as the replacement policies almost show indistinguishable performance over all the topologies. For single-path routing scenario, the considered decision and replacement policies perform almost the same. Multi-path routing worsens overall system performance playing against CCN efficiency, possibly causing pollution of caches as missing data travelling back to the requester which cause cache evictions on multiple caches, offsetting the advantage of reaching a higher number of caches. Finally, catalog and popularity settings play crucial roles becoming the key factors in caching performance as the network-centric performance metrics such as cache diversity, cache hit and user-centric metric such as path stretch (number of CCN hops that a data chunk actually travels normalized over the path length until the content originator) are greatly influenced by these factors.

Another analytical model to characterize the performance of chunk based data transfer in CCN is proposed by Giovanna et al. in [104]. The authors develop their analytical model to capture both the single cache and multi cache scenario.

The findings of the analytical framework reveal that the system throughputs mainly depend on content popularity, cache size and cache content size. They also show that content request aggregation can have a significant impact on the cache miss rate for a given cache node in CCN. Table 2 highlights the main attributes of the Content popularity-based caching schemes.

5.2. Location-based caching schemes

Location-based caching schemes select specific routers in the network as cache nodes or cache routers, which have higher probabilities of attaining cache hits compared to the other routers of the network. Location-based caching strategies are mainly divided into two categories: topological-based or neighborhood-based.

Topological-based caching schemes consider several standard graph-centrality metrics as the allocation criteria of the cache nodes such as betweenness centrality measured by the number of times a node lies on the data delivery path between all pairs of nodes in a network topology or degree centrality (the number of links incident upon a node) [112,117–119]. Several topological-based approaches propose to cache popular contents at the access network and less popular contents near the core network or suggest that, access and core networks should cache contents based on the types of contents [39,123]. These approaches reduce delay and publisher load because of caching popular contents near end users.

Several approaches propose caching schemes to provide enhanced mobility support, reduce average content retrieval latency and facilitate efficient caching for dynamic network topologies [40,112,117,124,138]. We name these approaches as neighborhood-based schemes because, the content routers can explore and exchange nearby cache contents utilizing the neighbor cache spaces. Figs. 5 and 6 show topological-based and neighborhood-based caching schemes.

In Fig. 5, a betweenness centrality-based topological-based scheme [117] is depicted. In this scheme, the highest betweenness centrality valued cache node (R_4) only caches the content instead of indiscriminate caching at all nodes as it has the highest probability of getting cache hits along the content delivery paths.

Table 2
Overview of Content Popularity-Based Schemes.

Caching Schemes	Content Popularity-Based
Collaborative forwarding & caching [37]	Considers static(consistent) and dynamic(inconsistent) popularity to rank contents
Popularity-driven [34]	Considers static and dynamic popularity where content popularity is measured by the aggregated request statistics
StreamCache [115]	Considers consistent popularity where popularity is measured by the aggregated request statistics collected and forwarded by downward routers
NCPP [111]	Static content popularity along with node utilization ration value take caching decisions
Efficiency of on- and off-path [109]	Considers consistent popularity to efficiently utilize the available caches to store more contents
PopCache [36]	Cache routers cache probabilistically based on consistent content popularity
RPC [110]	Compares dynamic popularity value of content with the caching threshold value of cache router to take caching decision
HCC [112]	Cluster head router considers dynamic content popularity value to take caching decisions in the edge cluster
PPC [116]	Future popular video chunks are predicted to take caching decisions dynamically
CLS [121]	Considers dynamic popularity to push up and pull down content along content delivery path
Value-based [132]	Considers dynamic popularity, delay and age of content to take caching decisions
Towards on-path [127]	Considers dynamic popularity and distance from the content source to the cache node
WAVE [38]	Considers locality of content requests to adjust the number of content chunks to be cached dynamically
Age-based [39]	Dynamic content popularity is used to guide content age
Impact of traffic [123]	Content popularity distribution is considered to characterize different types of contents while considering locality of content requests
Caching performance of CCN [35]	Reveals catalog and popularity settings as the key factors in caching performance of CCN by thorough simulation campaign considering different popularity settings
Content hierarchical intra-domain [133]	Content popularity dynamically determines content age and number of replicas of content to cache content probabilistically within a time interval
Caching from content delivery path [136]	Considers content popularity to assign cache weights dynamically and avoids bookkeeping of popularity of large set of contents
NCCM [126]	Software-defined networking (SDN)-based controller calculates content popularity periodically (dynamically) based on statistical information

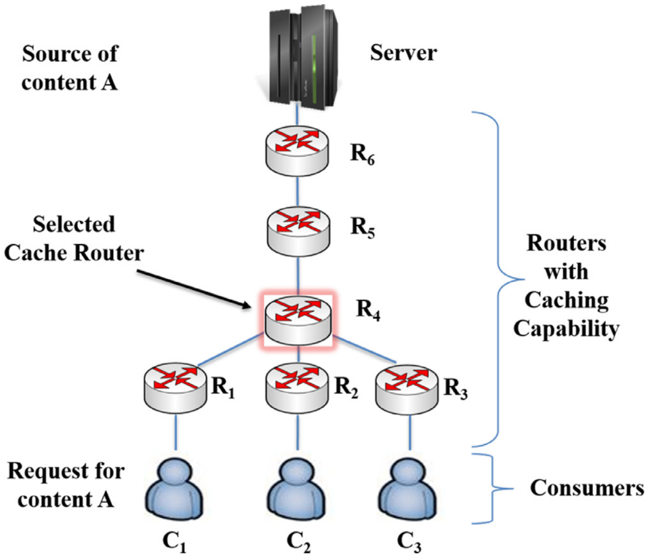


Fig. 5. Topology-based (Betweenness centrality-based) caching scheme.

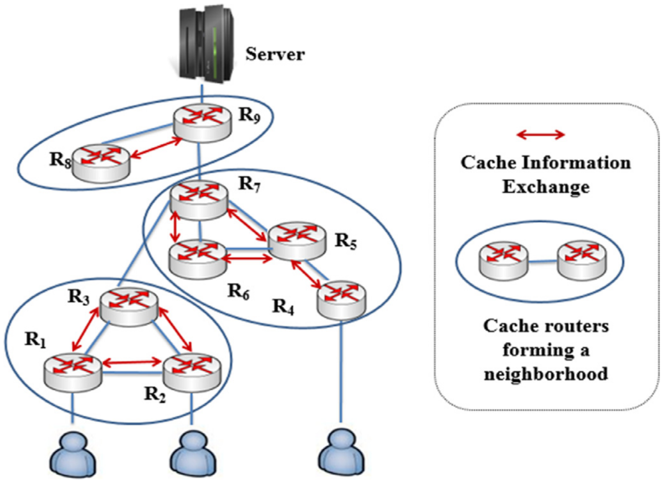

Server
R9
R8
R7
R5
R6
R4
R3
R1
R2
Cache Information Exchange
Cache routers forming a neighborhood

Fig. 6. Neighborhood-based caching scheme.

Hence, when a content request is routed from consumer C_1 to the server for content A and the desired content A is routed back to the consumer C_1, only the node R_4 having highest betweenness value is selected as a cache node for caching content A.

Fig. 6 shows the basic idea of neighborhood-based caching schemes, where a neighborhood of cache routers is selected based on the cache decision policy and the neighbor cache routers explore and exchange cache information with nearby cache routers to utilize the neighbor cache spaces. For an example, in Fig. 6, content routers R_1, R_2 and R_3 form a neighborhood to make caching decisions by exploring and exchanging cache information with one another to improve caching performance.

In the following section, we describe core location-based caching schemes.

5.2.1. Topological-Based schemes

Instead of caching indiscriminately, Chai et al. propose a centrality based-scheme in [117] aiming to investigate whether caching less can actually achieve more by only caching contents at some specific subset of cache nodes. The authors propose a betweenness centrality-based scheme that caches contents only at the cache nodes with higher betweenness values, these having the highest probability of getting cache hits along the content delivery paths. The authors also propose an approximation of their betweenness centrality scheme, based on the concept of ego network betweenness [139] for dynamic network environments. The proposed schemes perform better than the universal caching [27] in case of hop reduction ratio and server hit reduction ratio.

Betweenness centrality value is also used in [112] to calculate the importance of cache routers by measuring the rates or the frequencies of the cache router involved in the shortest path routing in a cluster. Betweenness centrality gives high rating to those cache routers that are likely to handle majority of traffic and hence selects these routers as important cache routers. To improve the cache hit ratio, clusters of cache routers are formed in a two-layer hierarchy consisting of cluster head, gateway and member routers and the selected cluster head router collects the information of betweenness centrality value of all routers and content popularity values of the requested contents and then calculates the probability matrix of caching as a pair by producing non-zero caching values where the important routers cache the popular contents and less important routers cache the less popular contents in a probabilistic way.

Aiming to improve the transmission efficiency of content dissemination, the node-content pass probability (NCPP) caching scheme [111] uses the topological attribute of cache node such as the node utilization ratio (NUR) along with the content popularity to take caching decision. The importance of a node is measured by the concept of NUR that depends on the network topology and the content routing algorithm. NUR has been defined as an effective metric for characterizing the traffic load distribution and how frequently a cache node is chosen to relay content packets in the network while considering user distribution, server distribution and different content delivery paths [130].

The NCPP value is computed considering both the NUR value of cache node and the popularity of content. In case of caching, the contents are ranked based on their popularities and the most popular content is cached at the cache node having the largest NCPP value, the second most popular content is cached at the cache node having the second largest NCPP value and the rest of the contents are cached in the same manner. So, eventually the most popular content gets cached at the cache node having the highest NUR value. As the NUR metric can be used for effective resource allocation in communication networks [130], a NUR–based resource allocation mechanism is also proposed to allocate the transmission capacity and the cache capacity to the cache nodes.

The NUR–based resource allocation mechanism allocates resources proportionally to the NUR values to the cache nodes. Hence, the cache nodes with higher NUR values can have larger transmission rate and cache size eventually caching more contents, improving the cache hit ratios and alleviating the server load. The NCPP scheme performs better than the random caching scheme by providing smaller server hit ratio but does not provide any theoretical analysis of its performance and highly depends on the number of contents.

To provide efficient content dissemination, the ICN architecture, named Cache Aware Target Identification (CATT) [118] focuses on two major components: caching and routing. For caching, CATT uses Topology Aware Caching mechanism, where content is cached at the cache node having the highest degree along the data downloading path since a large degree node can be reached from the other nodes with less latency than a smaller degree node. Compared to Traffic Aware Caching [117], the proposed scheme performs better and achieves availability, adaptability, diversity, and robustness.

The caching performance of CCN, emphasizing several graph-related centrality metrics as allocation criteria of cache space heterogeneously across the network has been studied in [119]. To allocate or distribute cache space heterogeneously among the nodes, degree, betweenness value, closeness (distance between a node to all the other nodes), graph (distance from a node to the farthest node) and eccentricity (the maximum distance from a node to all other nodes) centralities for the cache nodes are considered. Simulation results show that, performance gain by heterogeneity of the cache spaces is very limited compared to homogeneous allocation, and degree centrality is the most robust cache allocation metric for different network topologies and popularity settings.

To cache a realistic Internet traffic mix, consisting of four main types of contents such as web pages, file sharing, UGC and VoD, a two-layer cache hierarchy is proposed in [123]. The cache hierarchy reduces bandwidth requirements and server load for CCN. In the simple hierarchical cache network, the lower layer or first layer of caches near to the requesting end users are located in access routers, consisting of a large number of similarly sized content stores. The second layer typically consists of a set of co-ordinated storage facilities situated within the network core. The performance evaluations of the hierarchical cache network suggest that, web pages, file sharing and UGC, require large cache space and should be cached in the upper layer, near the core network. On the contrary, VoD content requires less cache space, has high and increasing traffic share and should be cached advantageously in the lower layer at access router caches, near the end users.

Contents are distributed hierarchically in such a way that the most popular contents are cached at the access network or network edge near end users and least popular contents are cached near the server in [39]. Content ages are dynamically configured such that, the popular contents are spread to the network edge reducing network delay and publisher load. Redundant content replications are removed at the intermediate routers for efficient utilization of network storage. This age-based hierarchical scheme is shown to be more effective than FIFO and LRU schemes, but is incapable of maintaining highly dynamic contents.

5.2.2. Neighborhood-based schemes

The Selective Neighbor Caching (SNC) approach in [40] selects an appropriate subset of neighboring proxies that are one hop away from the proxy to which a mobile is currently connected to for proactively caching information requests and the corresponding information for enhancing seamless mobility in ICN architectures. Here, the neighbor corresponds to the mobile's sequence of attachment points, not the geographic proximity, and the proxies are considered as special caches having additional functionalities of handling information requests on behalf of mobiles and can pre-fetch and cache information, matching a mobile's requests while the mobile is in handover phase or disconnected from the network. Aiming at minimizing the target cost function, the authors propose finding a subset of neighboring proxies, where an individual proxy is selected to proactively cache content matching the mobiles' subscriptions, if this caching is beneficial in terms of expected average delay and proactive caching cost.

The cache replacement method named PPC [116] has analyzed and derived the relationship among the neighboring video chunks in the same video stream from the viewpoint of video user watching behavior to predict and cache the future most popular content chunks and evict those with least future popularity while using neighborhood-based popularity prediction process. Based on the analysis results, PPC utilizes the known request information of the neighboring video chunks to predict the popularity of the following incoming video chunks and this neighborhood-based prediction process is named as Assist-Predict process. When there is no available neighboring chunk to assist the prediction that can occur when the first incoming request arrives, or a user fast forwards a video by skipping part of the video, as an alternative prediction process, Self-Predict process is used where a video chunk uses its own historical requests to predict the future popularity. A system model (described earlier) comprising of five modules is built in the cache nodes to support PPC replacement scheme. Among the five modules of the cache node, the PPM module executes the neighborhood-based assist-prediction process or the historical-based self-prediction process to calculate the predicted future popularity of incoming content and the cached content.

The two-layer HCC scheme [112] aims to improve in-network caching efficiency by grouping the network into several multiple autonomous groups named as clusters and then a centralized cluster head is nominated for each of the cluster to make caching decision in a centralized way based on a caching probability matrix. In the tow-layer hierarchical clustering architecture, core layer routers do not cache any content and only focus on content routing. In edge layer, routers are dedicated to cache contents while responding promptly to requesting users. Edge cluster routers provide connectivity to content providers and users and can be of three types, namely cluster head, gateway and member. Gateway is a border router between two clusters that can participate in the cluster head election and forwards content interests and Data packets among different clusters. Member is the router having the right to vote for the cluster head and cache content temporarily. Cluster head is the cluster controller elected by other cluster members based on Weighted-based Clustering Algorithm (WCA) while considering the neighbor node degree, the average transmission time and the weighted based hops of a node.

The cluster head node selects the top nodes as the cluster members forming a cluster based on the rank of transmission time from cluster head to other nodes through exchanging messages. The cluster head then calculates the node importance of the node inside the formed cluster based on the betweenness centrality of a node and all the members, gateways are ranked based on their importance inside the cluster. A probability matrix combines the node importance along with the content popularity to form a pair indicating a non-zero caching probability such as more important nodes have higher probability for caching higher popularity class of contents and the less important nodes have higher probability to cache less popular contents while having LRU replacement policy.

For infrastructure-less networks having dynamic topologies such as self-organizing, ad hoc and mobile networks, a selective caching strategy based on the concept of ego network betweenness [139] selects higher betweenness-centrality valued nodes as caching nodes within immediate neighborhoods in [117]. For each node, the ego-network consists of that node along with its immediate neighbors and all links among those neighbor. The proposed ego network betweenness scheme outperforms indiscriminate caching scheme [27] and random caching scheme in terms of hop reduction ratio and server hit reduction ratio for large scale real world Internet topology.

A neighborhood-based cooperative caching strategy for time-shifted TV for CCN is proposed in [140]. In the proposed policy, cache routers do not cache all content chunks routed by them, but only a subset following a designed modulo function-based technique and exchange caching information within their 2 hops neighbors. This scheme performs better than LRU, in terms of increased caching diversity and reduced cross domain traffic.

Dong et al. propose a caching scheme [141], where the cache router broadcasts its cached content information, within its vicinity and directs content requests to explore nearby cached copies to avoid the low neighborhood cache utilization problem. The region of neighborhood is increased sequentially hop-by-hop basis in the proposed scheme. The authors also develop a rigorous mathematical model to formulate the caching decision problem as an optimization problem. To solve the optimization problem, an Independent Allocation algorithm is proposed for providing optimal cache replacement policy while keeping the routers Content Broadcast (CB) enabled to their neighbors.

For locating or routing to the requested cache contents, Potential Based Routing (PBR) scheme [118] defines a scalar value named potential value for every content provider node and the provider node floods this potential value through advertisement messages within a limited scope of a neighborhood such as within a couple of hops. The nodes receiving this message, calculate their own potential values for a content based on the information carried in the advertisement message under the assumption that the potential value increases in proportion to the distance. After a potential field is defined within a limited scope of a neighborhood, whenever a node receives a user request from a requesting host, it simply forwards the request to one of its neighbor content provider nodes having the lowest potential value.

The cooperative caching strategy for CCN in [142] is especially designed for large scale video stream delivery in an intra-domain environment (an ISP). In the caching strategy, a small group of cache routers along the nearest neighborhood exchanges local information to take caching decision and request routing based on hash-based and directory-based cooperative scheme. A distributed algorithm is presented for assigning levels to the content routers based on neighborhood and a hash function is used to take co-operative caching decision among the small neighborhoods. This proposed strategy outperforms LRU and LFU schemes in terms of caching diversity and reduced inter-domain traffic revealing ISP-friendliness for VoD and time-shifted TV within an administrative domain. However, as the neighborhood or cluster of cooperative content routers grows, the inter-communication overhead among the caches increases.

The collaborative caching scheme MuNCC [124] aims to utilize neighborhood caching capacity of content routers to provide better cache utilization to improve user QoE as well as reduce costs by reducing the need to transit other networks while having negligible communication overhead. There are two basic building blocks of the neighborhood-based scheme which are the construction and exchange of Attenuated Bloom Filter and the coordinated cache eviction. MuNCC employs attenuated Bloom filters (BFs) to aggregate cache states within a particular level of the neighborhood of each cache router that allows neighbors' caches to be utilized to serve content requests with low latency and overhead. Any content request is directed first to the neighbor cache routers based on the aggregated cache state information stored as a set of attenuated BFs to get better cache hit incurring low latency.

In MuNCC, a cache router never caches a content if the content is found in a neighborhood cache router to reduce content redundancy. A cache router caches a content if the content request is not satisfied by the neighboring routers and forwarded to the content source while using LFU replacement policy. A two-layer Content Store (CS) is deigned to collaborate the cache eviction where the primary layer of the CS ensures the availability of advertised contents among the neighbor routers and the candidate evicted item is transferred to the secondary layer of the CS instead of being directly deleted. The effectiveness of MuNCC has been demonstrated comparing against LCD [137] and ProbCache [131] in terms of cache hit ratio and latency.

A scalable content routing scheme, dubbed SCAN [138] realizes content aware networking among neighboring cache routers. In SCAN, cache router first performs default IP routing ensuring reachability of the requested content and additionally does scanning to locate the nearby multiple cache copies of the requested content in its vicinity by looking up its local cache information and neighbor cache information for efficient content delivery. Table 3 highlights the main attributes of the Location-based caching schemes.

5.3. Collaboration-based caching schemes

Collaboration among cache nodes can improve caching efficiency where the cache nodes can cooperate with one another to make caching decisions. Many studies investigated the effectiveness of collaboration in ICN caching and propose collaboration-

Table 3
Overview of Location-Based Schemes.

Caching Schemes	Location-Based	
	Topology	Neighborhood
Cache “less for more” [117]	Betweenness centrality-based value is used to select the cache node	Ego network based on immediate neighborhood of nodes for large dynamic network topology
CATT [118]	Topology aware caching is used	Content provider advertises potential value of content along its neighbor hops
HCC [112]	The importance value of cache router is determined by the betweenness value of the cache router	Cluster head, gateway and the member cache routers form the edge cluster as a neighborhood to cache contents
Age-based [39]	Content age is determined by location of content router	
On sizing CCN content stores [119]	Heterogeneous cache allocation based on graph- centrality metrics	
Caching performance of CCN [35]	Considers generalized network topologies and a standard binary tree topology	
NCPP [111]	Node utilization ratio (NUR) value selects the cache node	
PPC [116]		Neighborhood-based prediction is used to calculate the future popularity of incoming video content and the cached content
MuNCC [124]		Neighbor cache routers collaboratively take caching decisions and cache eviction decisions
Time-shifted TV [140]		Content routers exchange information within closest neighbors
SCAN [138]		Content routers exchange content routing information among neighbors
Proactive caching [40]		Proactive selective-neighborhood caching to support enhanced seamless mobility
Optimal caching with content broadcast [141]		Addresses low neighborhood cache utilization problem
Cooperative Caching [142]		Small group of routers exchange messages for caching decisions within an ISP

based caching schemes [34,37,39,109,120,121]. The main advantage of collaborative caching is that caching redundancy is reduced, since the same content is not unnecessarily copied at multiple cache nodes. Moreover, it improves cache diversity by ensuring that, huge diverse contents are cached by the cache nodes which ultimately results in minimized access latency for content retrieval and improved response time [12]. However, collaborative caching suffers from notable drawbacks, mainly in terms of communication overhead due to exchange of control messages among cache nodes necessary for co-operation, and the extra latency occurring for exchanging such control messages.

Collaboration among caching nodes is typically either explicit or implicit. In case of explicit collaboration, all caching nodes in an AS or the cache nodes in a neighborhood can exchange their states or summaries of states, in addition to information on content popularity, content access patterns, and cache network topology. Using this exchanged information, caching nodes can make caching decisions and control resource allocation [120]. The main limitations of explicit schemes are the additional cost incurred for the communication overhead and the complexity of the coordination algorithms. On the other hand, implicit collaboration remedies the requirement of exchanging elaborate information among caching nodes, and depends on local cache management policies by exchanging limited information among caching nodes; typically restrained to the content delivery paths. This happens at the expense of lower cache diversity, as such implicit policies cannot fully utilize the cache spaces available from other neighboring caches [120,121].

In Fig. 7, we show the basic concept of explicit and implicit collaborated caching schemes using the same network topology. Fig. 7 shows that, for the explicit scheme, cache nodes R_1, R_2, R_3, R_4 and R_5 exchange caching information with their 1-hop neighbors beyond the content delivery paths to make caching decisions. On the other hand, for the same network topology, cache nodes R_6, R_8 and R_{10} or R_7, R_9 and R_{10} collaborate by exchanging information in a limited extent along the content delivery paths only in a hierarchical fashion for the implicit scheme.

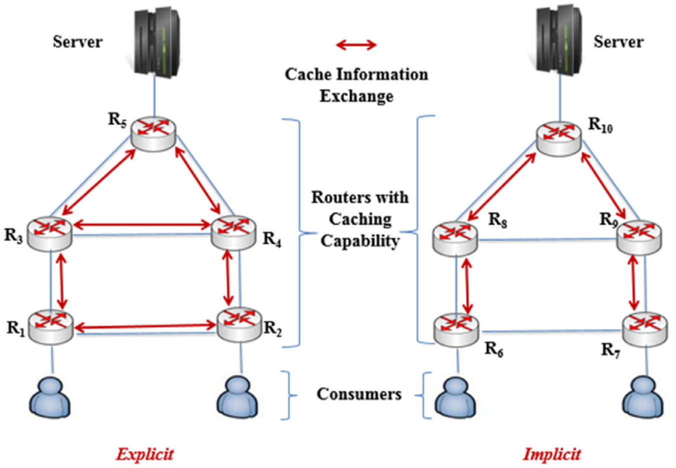


Fig. 7. Collaboration-based caching schemes.

5.3.1. Explicit collaboration-based schemes

The main objectives of the collaborative forwarding and caching scheme for CCN in [37], where popularity is ranking-based, are to use global collaboration among routers in an AS for improving caching performance. Additionally, the scheme coordinates caching with content request routing to further optimize network performance. Collaboration is guided by the popularity ranking sequence of content generated by each participating router. The authors introduce a collaborative caching and forwarding table and propose a self-adaptive dual segment cache design algorithm to deal with the dynamic popularity inconsistency experienced by the different routers. The proposed scheme outperforms the hierarchical scheme as hierarchical caching only uses the on-route hierarchical caches instead of utilizing all the available cache spaces from other nearby routers' caches.

Location selection and content popularity are jointly considered in a cooperative model for making caching decisions in the HCC scheme [112]. HCC uses collaborative caching approach by allowing devices to exchange information among a group or a cluster for more effective caching and less redundancy. To manage the communication overheads, communication messages are exchanged in a two-layer hierarchical cluster-based architecture. The core layer routers perform only content routing and the edge layer routers perform content caching depending their importance values calculated by the cluster head node in a cluster. In generally, more important routers cache more popular contents, while less important routers cache less popular contents, and non-important routers perform no caching. Cache routers exchange information collaboratively and dynamically to adjust their caching roles and the management of caching and message exchange is done by clustering. The communication overhead of the HCC occurs both in the initialization phase and update phase of formation of a hierarchical cluster when the cluster head router and other cache routers exchange messages to form a cluster. Moreover, cluster head also collects the information of betweenness centrality value of other routers to define the importance value of the routers as caching candidates and also collects the content popularity values from the access routers. After the collections, cluster head calculates the probability matrix to make caching decisions considering the router importance and the content popularity values. The cluster head also periodically updates the probability matrix and popularity class list and exchanges these updated information with the cache routers as the content popularity changes from time to time. All the important cache routers start to cache contents upon receiving the probability matrix result from the cluster head.

Three explicit collaborated schemes: Basic, Adapted and Stacked in [109] cache contents according to content popularity and work hierarchically. The Basic scheme utilizes all cache nodes in the tree to store contents resulting in a better cache hit efficiency but poor mean hop delay to get cache hit. The Adapted scheme requires more coordination than the Basic scheme, but offers the same cache hit efficiency while offering minimum possible mean hop delay to get cache hit. Stacked scheme requires less coordination than the other two variants at the expense of increased cache redundancy. All three schemes perform better than uncoordinated on-path popularity based scheme accompanied with LRU and LFU replacement policies, in terms of cache hit efficiency, mean hop delay, and power consumption.

An explicit collaborated in-network caching scheme guided by traffic engineering, named TECC, is proposed in [143]. The authors formulate a joint optimization problem of content caching and routing, and further decouple the problem as two sub problems. The network model is a non-structured flat network where all cache nodes collaborate to fetch content. TECC outperforms an implicit coordinated hierarchical scheme in terms of average link utilization, server load reduction and user experienced latency. This is because, in TECC, explicit coordinated peer nodes largely absorb cache misses.

The caching strategy in [140] aims to handle large video streams with on-demand access for CCN to minimize cross-domain traffic for ISP. The content routers minimize the amount of queries which are handled by the origin servers outside the ISP network for time-shifted TV. In the proposed strategy, a content router does not cache all the content chunks routed through it, but only a few of those which can satisfy the designed modulo function with LRU policy. Each content router collaborates with its neighbors to distribute and search content chunks by maintaining two new tables: Collaborative Router Table (CRT) and Collaborative Content Store (CCS). This proposed scheme outperforms the LRU policy by achieving increased caching diversity and reduced cross-domain traffic of an ISP, incurring insignificant increased average response time.

The neighborhood-based explicit collaborated caching scheme in [141] minimizes the average content retrieval latency by solving the low neighborhood cache utilization problem. To collaborate with neighbors, instead of remaining silent after contents are cached, a content router explicitly advertises or broadcasts its caching information with its neighbors. The proposed scheme outperforms other Content Broadcast (CB) enabled policies, such as CB-LRU and CB-LPFO (Least-Popular-First-Out) in terms of average content retrieval latency while incurring communication overhead.

In the SNC approach [40], selected one-hop away neighbor proxies of a proxy to which currently a mobile is connected to collaboratively take proactive caching decisions to minimize the mobility costs while ensuring seamless mobility. The decision of joining a proxy in a subset to proactively cache information requests and the corresponding information items matching a mobile's requests is taken locally by the neighbor proxies based on a designed cost function. The cost function tells whether the proxy's joining to the subset of proxies results into caching gain greater than the caching cost or not. SNC attains better caching gain comparing with full proactive caching approach where all the one-hop away neighbors are selected for proactive caching and no proactive caching approach.

For every content router, the SCAN [138] scheme maintains a local content table (LCT) consisting of its own cache information and a content routing table (CRT) consisting of the cache information of neighbor routers. Whenever a router on the data delivery path gets a content request from a requesting host, it performs default IP routing to the original server of the requested content and then scans by looking up its LCT and CRT to search multiple cache copies of the requested content. In SCAN, content routers exchange content routing information periodically using their LCTs and CRTs with their neighbor routers. Bloom filters [144] are used for information compression to achieve scalability. Additionally, information decaying process is used to mitigate the negative effect of false positive decisions.

A hash-based, and directory-based collaborative caching strategy caches large video-streams with on demand access to reduce cross-domain traffic in [142]. Selected content routers based on latency collaborate within a small nearest neighbor to efficiently cache contents by eliminating caching redundancy among adjacent routers within a single administrative domain. The cooperative scheme assigns labels to the content routers in a distributed way and uses a modulo function to determine the caching location of a content. Additionally, the scheme requires two new tables integrated in a content router, named Cooperative Router Table (CRT) and Cooperative Content Store (CCS) for request routing. The cooperative scheme improves caching diversity and especially reduces inter-domain traffic in an intra-domain environment for two popular applications: VoD and time-shifted TV under realistic network conditions comparing with non-cooperative LRU and LFU schemes.

If more accesses can be done within an AS to retrieve content instead of going outside, the better caching performance can be achieved by saving cross-AS traffic. Based on this idea, IDCC [133] uses probabilistic caching based on cache capacity, caching time and traffic speed and avoids caching unnecessary replicas of contents by intra-domain cooperation among the cache routers exchanging content advertisement messages within the AS. Within a local AS, cached contents are advertised throughout the whole AS.

To reduce caching redundancy, cache routers cooperate with one another using content advertisements guaranteeing that contents are cached only once in the AS unless any router becomes overloaded caching too many popular contents. The most popular contents can be replicated more than once by the cooperating nearby routers for load balancing. IDCC performs better than CEE [27] and ProbCache [131] in terms of cache hit ratio, user delay and cross-AS traffic unless the popular contents are very concentrated as number of replicas cached in the AS can be extremely large in this case.

In exploring the impact of varying traffic patterns, a caching mechanism named MuNCC [124] aims to provide a scalable and efficient collaborative caching strategy by achieving not only better cache utilization but also minimum communication overhead to minimize coordination cost. In MuNCC, a cache router can forward content request leveraging explicit information exchanged with its direct neighbors to assess whether another router in its neighborhood can satisfy the request from its local cache or not. Attenuated Bloom Filters (BFs) filers are used to aggregate the exchanged cache states information within the neighborhoods of each cache router and utilizing these aggregated information, content requests can be served with low latency and overhead. If a content request cannot be satisfied from the neighborhood of the requester, then it is forwarded towards the content producer in the normal manner. To collaboratively cache contents and satisfy content requests, the CS of the cache router is partitioned in two layers named as primary cache and secondary cache, additional forwarding states are stored in the Pending Interest Table (PIT) and the forwarding algorithm is altered in MuNcc. The primary cache space holds 90% cache space of CS and caches the contents which are advertised in the cache summaries in the exchanged cache information. A coordinated cache eviction scheme is also designed to increase cache diversity and consequently the overall cache-hit ratio while ensuring that evicted contents remain available nearby. In case of cache eviction, if a content needs to be evicted from the primary cache, instead of being immediately deleted, the evicted content is transferred to the secondary cache of CS and LFU replacement policy is used when the secondary cache becomes full. MuNCC uses two algorithms named Request Handler and Request Helper to collaborate content caching and request routing within a neighborhood. Request Handler algorithm is invoked when a content router receives incoming content requests and the Request Handler algorithm invokes Request Helper algorithm as required to route the content requests within a particular level of the neighborhood.

To enable cooperation among distributed content routers and make caching and routing decisions, the cache management framework in [126] collects cooperation related information such as content request rates and the current cache status using a software-defined networking (SDN)-based controller. In the proposed framework, an ICN consisting of distributed content routers, a controller, and LNC-enabled servers has been considered. The cooperating content routers monitor content requests from its end local users and send content request statistics periodically to the central controller. The controller gathers content request statistics from each content router and determines the local popular content request set for each content router and pre-configures the route to these set of contents. The controller predicts a popular content set from all the gathered statistical information from the collaborated routers, configures the content routers to cache popular contents and route content requests and deliver contents.

5.3.2. Implicit collaboration-based schemes

An implicit collaborated scheme called CLS [121] for CCN aims to improve network performance by increasing cache exclusivity by caching more diverse contents in the network. The CLS scheme ensures that there is at most one single copy of a content chunk cached on the content delivery path between a content server and a leaf cache router. This single copy of each content chunk is pulled down one level towards the leaf router by each request and pushed up one level towards the server by the cache eviction. The idea behind this pull-down and push-up is to avoid amplification of cache replacement and achieve cache exclusivity.

Additionally, a caching trail of content chunk, which stores the chunk caching history, is built up during the chunk cached up and down, which can direct the searching policy of that content chunk in the future which eventually reduces the server workload and file download time. CLS performs slightly better than other schemes such as indiscriminate caching [27] and another implicit collaborated caching scheme named Leave Copy Down (LCD) [137] in terms of average cache hit ratio and server load reduction. The authors justify the slight supremacy of their scheme because of the considered three-level test-bed, the performance of CLS should be validated in a larger architecture setting with many hierarchical levels to prove its supremacy over others.

A simple, transparent and best-effort content caching policy, named Breadcrumbs is developed in [120] to forward queries to search contents. The proposed approach is best-effort in that, forwarded content requests may or may not locate the content while being routed among the cache nodes. There is a minimal amount of per-file information regarding content caching history termed as “breadcrumbs”, which is stored at the cache spaces of the collaborating content routers.

The proposed query routing policy, named Best Effort CONtent Search (BECONS) uses Breadcrumbs for locating contents. BECONS query routing policy needs implicit coordination among the cache nodes along the content delivery path to forward content queries in a hierarchical infrastructure. Although Breadcrumbs is an implicit best-effort policy, it performs better by locating cached contents more frequently and reduces the load on the source more than the considered two explicit collaborated schemes when the cache size is relatively small. The Breadcrumbs scheme though focuses only on content searching without considering the cache placement policy.

Although every content router makes caching decisions independently, there exists implicit collaboration among the content routers in WAVE [38]. This collaboration exists in a hierarchical manner and avoids inefficient caching situations, such as redundant caching, resulting in efficient content delivery. WAVE uses implicit cache coordination in such a way, that an upstream content router recommends caching of a content chunk to its downstream router by marking the chunk when it is forwarded from the server to the requesting host. If the cache space of the downstream router is full, or it follows its own caching policy, the recommendation may be not followed and caching recommendation can be forwarded further to the downstream routers. WAVE pushes the chunks for caching in the direction of the incoming requests in a hop-by-hop manner and selects the victim chunks for replacement using LRU policy. WAVE shows superiority comparing with several other uncoordinated caching schemes such as AllCache [27], and Fixed probability caching.

Content age is set in a manner such that, the further a content is from a server, the longer age it has and the more popular a content is, the longer age it has by the age-based cooperative caching scheme in [39]. The cache routers dynamically set content age in an implicit cooperated manner such that, the popular contents are spread near to the end users at network edge for longer and unnecessary caching is eliminated at the intermediate routers. The collaborative routers replace contents when the age of content expires and the cache space of a content router becomes full. The implicit coordinated age-based scheme performs better than the non-coordinated FIFO and LRU approaches in terms of aggregated network delay, end user delay, and server load reduction evaluated under real network topologies.

To efficiently utilize the cache router's storage capacity, the router position-based cooperative caching scheme RPC [110] considers the position of routers as the main caching parameter. The main idea of RPC is to push popular contents closer to the customers (users or enterprises) and less popular contents to near the core network to reduce the publisher load and the network delay. The routers along the content delivery paths collaboratively take the caching decisions based on the topology values of the routers.

In RPC, the topology level value of a router is calculated by adding 1 to the value of its immediate upstream router's topology level. This topology level value and the root router's caching threshold value are transmitted from the upstream routers to the downstream cache routers in a collaborated way along the content delivery path. This collaboration among the cache routers is light-weight as this happens only once during the procedure of determining the caching threshold value incurring low overhead. RPC assigns a caching threshold value to each router based on the position of a router (topology level) and the root router's caching threshold value. The root router's caching threshold is preconfigured, and caching threshold value of other cache router is calculated by the proposed caching threshold decision policy. This caching threshold and the storage capacity of the cache router determines whether a content should be cached at a content router or not.

For minimizing inter-ISP traffic and average access latency, the cache routers implicitly coordinate to make caching decisions within an ISP in [34]. Content replica placement problem is formulated as an optimization problem.

The proposed dynamic caching strategies for CCN use popularity-based caching, where caching is coordinated implicitly by sharing caching information hierarchically to avoid redundant caching, but the routers make caching decisions dynamically and independently. The popularity-based dynamic strategies are shown to be effective enough to achieve near optimal performance for small scale networks.

To direct user requests toward a matching content file considering the proximity and quality of the content file, in Potential Based Routing (PBR) scheme [118], the content provider broadcasts the potential values of contents with their neighborhoods using advertisement messages. Hence, the nodes within the scope of the neighborhood form a potential field and can route received content requests to the content providers having minimum potential values in an implicated collaborated way of ensuring efficient content retrieval along the content delivery paths.

A caching strategy, dubbed as Least Unified Value (LUV)-path is proposed in [136] where routers implicitly coordinate in selecting cache contents and assigning value to each cache content along the content delivery path. LUV-path assigns different caching probabilities between the upstream routers and downstream routers to reduce caching redundancy based on the number of hops from the requesting client to the cache router and total number of hops from the content provider to the client along the content delivery path in a coordinated way.

The probability of caching new content gets higher at upstream routers, and contents which are not cached there get higher chance to be cached at downstream routers using accumulated probabilities. To assign weight for cache content, LUV-path assigns greater value or weight to the downstream routers near to the client and lesser weight to the upstream routers and considers content popularity.

For reducing cache redundancy along the content delivery path, the cache routers in ProbCache [131] collaborate by considering the caching path as a shared pool of resources to do a fair multiplexing of the resources among all the incoming request flows. Any cache router on the delivery path caches contents probabilistically, by considering the remaining cache capability of the caching path required for other requested contents. This is done by defining two values, namely the Time Since Inception (TSI) that indicates the hop distance from the requesting consumer to the cache router and Time Since Birth (TSB) that indicates hop distance from the content source to the cache router. These two values define the distance of a cache router from the requesting consumer while assigning a weight value to the router. Caching decision is made probabilistically while considering the cache capacity of the delivery path and the weight value of a cache router in a collaborative way among the on-path cache routers.

In order to operate over a large-scale network, the StreamCache [115] popularity-based video caching policy improves users' QoE by maximizing average throughput, makes distributed caching decisions without relying on a centralized controller to synthesize the global network information. StreamCache does not incur frequent control information exchanges among the cache routers as video streaming itself is a delay-sensitive application incurring high-volume and continuous data transmission and inevitably large amount of coordination signals and explicit information exchange among the cache routers can interfere with the normal video delivery while degrading users' experience significantly.

On each round of the scheme, edge routers collect the request statistics of the video contents which reside at the video server at the root of the cache hierarchy and at the end of each round each cache router makes local decisions about selecting video contents based on the aggregated video request statistics while requiring minimal coordination while proceeding to start the next round of the scheme. The size of records of video requests can become considerably large after a long run but these records are stored locally on edge routers and never passed or aggregated with records of other routers over the network directly.

To efficiently utilize cache capacity, at the end of each round, the summarized versions of statistics tables created by each edge router are delivered along the forwarding path, as well as its local caching decisions to the parent node. After parent node receives the summarized statistics and caching decisions from all its children nodes, it then creates its own tables and passes forward to the upstream routers. This implicit collaborated procedure of caching continues until all cache routers finish updating their own local caching decisions based on the caching utility values, from the edge routers toward the root of routing topology.

In capitalizing on hashing techniques, Wang et al. present CPHR [10], a cooperative caching scheme where hashing techniques have been used for partitioning the cache space. Content routers also use hash techniques for caching and request routing to maximize the overall cache-hit-ratio of an ISP network by eliminating the content duplications. CPHR partitions the cache space in such a way that contents originating from the same egress router are assigned in the same partition and subsequently uses hash function to assign the cache contents in these cache partitions and request routing to the content. The problem of assigning partitions to caches has been formulated as an optimization problem aiming to maximize the overall cache hit ratio and a heuristic algorithm is also proposed to solve the partition assignment problem.

To collaborate, ingress routers (routers that receive requests from their clients or other ASes) add a new column to their Forwarding information based (FIB) table for listing the egress router (router that is connected to the external links) through which an interest packet for a specific content leaves the domain toward the content source. Additionally, while request routing, ingress router prefixes the name of the assigned cache with a content name, giving the content a new name and adds the name of the cache routers to store the caching location of the requested content and the name of the egress router for the content in the

Table 4
Overview of Collaboration-Based Schemes.

Caching Schemes	Collaboration Method	Collaboration Model
Efficiency of on- and off-path [109]	Global collaboration in hierarchical way	Explicit
SCAN [138]	Neighbor routers exchange routing information for request routing	Explicit
Collaborative forwarding and caching [37]	Global collaboration, Caching coordinated with forwarding	Explicit
TECC [143]	Joint optimization of collaborative caching and traffic engineering	Explicit
Time-shifted TV [140]	Neighborhood-based collaboration within a domain	Explicit
Proactive caching [40]	Proxies exchange information for proactive caching within the neighborhood	Explicit
Optimal caching with content broadcast [141]	Addresses low neighborhood cache utilization problem	Explicit
Cooperative caching [142]	Neighborhood- based, small group of routers exchange control messages to take caching decisions	Explicit
Content hierarchical intra-domain [133]	Hierarchical Collaboration, allows popular contents to be cached longer	Explicit
MuNCC [124]	Neighbor cache routers collaborate to take caching and eviction decision by exchanging cache state information within a certain level of neighborhood	Explicit
HCC [112]	The cluster head router periodically exchanges the probability matrix values and the content popularity classes with the important cache routers	Explicit
NCCM [126]	Content routers periodically send content request statistics to the SDN-based controller for caching and routing decisions	Explicit
CLS [121]	Hierarchical collaboration, content caching location and searching scheme	Implicit
Breadcrumbs [120]	Hierarchical collaboration, content searching scheme	Implicit
CATT [118]	Collaboration based on potential value of content	Implicit
WAVE [38]	Hierarchical collaboration, upper level routers recommend lower level routers to cache contents	Implicit
Age-based [39]	Hierarchical collaboration, upstream and downstream routers dynamically configure content age	Implicit
Popularity-driven [34]	Hierarchical collaboration, caching decision takes place from top to bottom	Implicit
Caching from a content delivery path [136]	Upstream and downstream routers coordinate to assign probability of caching new content	Implicit
CPHR [10]	Routers collaborate using cache partitioning and hash routing	Implicit
StreamCache [115]	Downstream routers forward the upstream routers the summarized versions of content request statistics tables and local caching decisions to coordinate	Implicit
RPC [110]	Topology level values of the cache routers are transmitted from the upstream routers towards the downstream routers	Implicit
Probabilistic [131]	Cache routers collaboratively share the caching resources in a probabilistic way	Implicit
Efficient cache availability [114]	Hash function-based content caching and request routing	Implicit
Caching performance of CCN [35]	Caching in uncoordinated and uncooperative fashion	Non-Cooperative
Towards on-path [127]	No inter-coordination among cache routers	Non-Cooperative
Hash routing [145]	Hash function is computed in a distributed manner	Non-Cooperative
Cache “less for more” [117]	No inter-coordination among cache routers	Non-Cooperative
Value-based [132]	No inter-coordination among cache routers	Non-Cooperative
Impact of traffic [123]	No inter-coordination between the two layers of caches	Non-Cooperative
PopCache [36]	No inter-coordination among cache routers while taking cache decisions based on content popularity	Non-Cooperative
On sizing CCN content stores [119]	No inter-coordination among cache routers	Non-Cooperative
PPC [116]	No inter-coordination among cache routers for predicting future popularity values of the incoming requested contents and the already cached contents	Non-Cooperative
NCPP [111]	No inter-coordination among the neighbor cache routers	Non-Cooperative

interest packet. CPHR significantly outperforms both LRU and LFU policies in terms of byte-hit ratio for P2P file sharing and VoD contents with an added propagation delay.

The lightweight locally cooperative content caching and request routing policy in [114] uses a hash function-based technique. It maintains a controlled distribution of contents within an AS. The hash function maps the content to a local cache router based on the content ID inside an AS assuming that AS supports content ID-based caching and routing.

The selected “designated router” for a specific content is the only cache router which caches the content. Hence, a local simple hash function assigns cache routers with the responsibilities of caching a certain range of contents for distri- buting contents in an AS. In case of request routing to the cached contents, whenever an AS receives a request for a content ID within its range of cached contents, the request is forwarded through the responsible cache router.

If the request does not get any cache hit, it is forwarded further towards the next AS. The cooperation between the Autonomous systems (ASes) is not mandatory in the scheme, but the more ASes express an interest of caching to cooperate, the better cache spread the caching scheme can achieve. The proposed caching scheme outperforms CEE [27] and ProbCache [131] in terms of cache storage efficiency (ability to cache unique contents and avoid duplicate caching), server load reduction with increased latency or hop-count to retrieve requested content. Table 4 highlights the main attributes of the Collaboration-based caching schemes.

5.4. Path-based caching schemes

ICN caching schemes can be classified according to the location of cache nodes with respect to the content delivery path from content producer to consumer. We name these schemes as Path-based caching schemes. There can be two types of Path-based caching schemes: on-path or off-path. In the former, content is cached at intermediate cache nodes along the direct forwarding path from producer to consumer.

Under this approach, content request is forwarded only to the next cache node lying in the direct path from producer to consumer. As a result, only the cache nodes along the forwarding path have the chance to response to the content request using their cached data. However, under off-path caching, content can be cached away from the direct path from the content source to the consumer. It is also possible to forward queries or content requests to the cache nodes deviating from the direct path and retrieve requested contents [109].

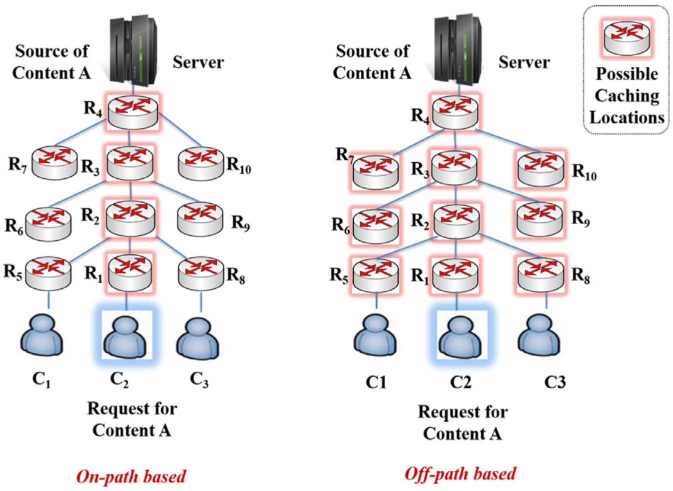


Fig. 8. On vs Off Path-Based caching schemes.

Fig. 8 depicts the contrast between on-path and off-path-based schemes. When a content request is generated from consumer C_2 for content A, the request will be forwarded only to the caches along the direct content delivery path between the consumer C_2 and the source for on-path caching. As a result, when the desired content A is routed back to the consumer C_2, the cache nodes along the forwarding path, i.e., cache routers R_1, R_2, R_3 and R_4 have the chance to cache the content and respond to the content request in future.

In off-path caching, when the content request is routed from the consumer C_2 to the source and the content is routed back to the consumer C_2, any router deviating from the content delivery path between consumer C_2 and source, i.e., cache routers R_1, R_2, R_3, R_4, R_5, R_6, R_7, $R_{8,}$ R_9 and R_{10} have the options of caching the content A and responding to content request in future based on the caching decision policy.

5.4.1. On-path based schemes

The most cited and criticized caching scheme in ICN literature is the CEE scheme presented by Jacobson et al. for their proposed ICN architecture named CCN [27] is an on-path scheme. In CEE, content is cached indiscriminately in every intermediate node along the content delivery path, from the cache hit node to the requesting client.

This approach was criticized by [117,121,131] for its unnecessary caching redundancy, content eviction and resource consumption. This indiscriminate universal caching can provide fast content delivery [117] and is mostly useful for recovery from data packet losses and tackling flash-crowds, where many consumers request the same content in close succession [7]. Whether caching only at a specific subset of cache nodes in the content delivery path can achieve better gain than indiscriminate caching, and if it can which are these cache nodes, and how can these cache nodes be identified are investigated in [117].

The proposed betweenness centrality-based scheme suggests that, if a node lies along a high number of content delivery paths, then it is more likely to get a cache hit. Hence caching only at this type of "important" nodes should reduce the cache replacement rate. For dynamic topology, the proposed caching scheme uses ego network betweenness concept [139]. Both betweenness and ego network betweenness schemes perform better than the indiscriminate caching scheme [27]; in terms of hop reduction ratio and server hit reduction ratio for different network topologies.

The cache node having the higher node utilization ratio (NUR) value along the content delivery path caches contents considering the content popularity value in [111]. NUR value is defined by considering user distribution, server distribution, and different content delivery paths while defining the importance of a node as a candidate cache node. A node-content pass probability (NCPP) value is calculated for each of the cache node along the content delivery path based on its NUR value and the popularity value of the requested content. Based on this NCPP value, the on-path cache node caches the contents where the contents are ranked based on their popularity.

The CLS scheme presented by Li et al. [121] considers content caching together with request to cache routing. This scheme ensures that, there is at most one copy of each content chunk in the content delivery path aiming to provide cache exclusivity by only caching diverse contents. The key idea of the on-path scheme is pulling down and pushing up content chuck one level for increased content requests and cache evictions and searching that chunk using the created caching history stored along the content delivery path for the push and pull strategy. CLS outperforms other on-path schemes such as indiscriminate caching [27] and the LCD scheme [137] when the majority of content requests are for fewer distinct contents.

The simple, on-path scheme of [120] is a best-effort approach to search requested contents among the cache nodes which may or may not locate content using the minimal amount of per file of content routing history. The routing history or "breadcrumbs" means the most recent direction and time that a file is forwarded in the past along the content delivery path. The policy only focuses on the content searching policy without considering content placement along the caches.

The content provider selects the node having the highest degree along the data delivery path as a cache node, while content is routed from content provider to consumer in Cache Aware Target Identification (CATT) [118]. For locating cache content, the Potential Based Routing (PBR) approach is proposed. The key idea of PBR is that, all content provider nodes have scalar potential values, forming a potential field within a scope, then the content request is forwarded to the provider having the minimum potential value. The scheme outperforms other on-path and random schemes in terms of average access latency to retrieve content.

WAVE [38] caches content chunks on the content delivery path in the direction from which chunk requests come considering the locality of the content requests and uses LRU policy to determine victim chunk to be replaced. WAVE outperforms several other on-path schemes such as indiscriminate caching [27], UniCache, and Fixed probability caching.

The on-path scheme in [39] called Age-based caching, uses content location and content popularity to configure the content's age dynamically along content delivery path following the two rules: 1) the closer a content is to the network edge near end users, the longer age it gets, by ensuring that contents are pushed near network edge, 2) the more popular a content is, the longer age it has which ensures that popular contents have higher priority to be cached by the routers and cached at network edge.

The router position-based on-path caching scheme RPC [110] determines whether a video content should be cached in a cache router or not depending on the position of the content router along the content delivery path. In RPC, the root cache router is directly connected to the content publisher, while the edge routers are directly connected to the users. The scheme caches content ensuring that the frequently requested contents are cached at the edge routers while eliminating the unnecessary content duplication at the cache routers. The on-path routers keep track of the access counts of the contents locally and calculate the caching threshold value based on their topology level values (transmitted from the upstream cache routers to the downstream routers starting from the root cache router) and the caching threshold value of the root cache router which has been assumed to be precomputed. The topology level and the caching threshold have positive correlation, i.e., a higher topology level value of a cache router always generates a higher caching threshold value for a router. The cache router compares the access counts of a content to its caching threshold value before caching and use LRU scheme for cache replacement. The RPC scheme performs better than other on-path proposed schemes [27,39], but does not mention the method of calculating the root cache router's threshold value.

A probabilistic caching scheme, named ProbCache, is proposed in [131], with the objective of eliminating caching redundancy. ProbCache solves the problem of content placement within the on-path caches from the path caching capability point of view, where the path is a pool of caching resources from which ProbCache tries to find optimal ways of distributing contents in these caches. Content routers aim to probabilistically cache contents to do fair multiplexing of the path capacity among different content flows per unit of time while achieving significant reductions in cache evictions and proper utilization of caching resources. ProbCache performs well in terms of server hit reduction ratio, hop reduction ratio, and reduction of cache-evictions compared to other on-path caching schemes [27,137] at the expense of increased computational cost.

Two on-path popularity driven caching schemes, named Top-Down caching and AsymOpt in [34] dynamically place the content replicas in the cache routers along the content delivery path. As guidance, the optimal replica placement problem is solved first and then based on the solution, the on-path dynamic caching schemes are developed. In the schemes, cache routers along the path make caching decisions independently but the caching is implicitly coordinated among the routers to increase caching diversity while reducing inter-ISP traffic and average access latency. Cache routers cache contents based on the latest accumulated content requests of their subordinates and cache popular contents near requesting hosts hierarchically.

The popularity driven schemes demonstrate superiority comparing to several other on-path schemes such as LCE [27], LCUnip (Leaving copies with uniform probability), LCProb (Leaving copies with probability) in terms of various network topologies, content access patterns, content popularities and cache capacities.

The distributed video caching policy in [115] takes real time cache placement decisions in a distributed manner along the content delivery paths to improve the average throughput of users ultimately enhancing the QoE of users. The caching scheme assumes that all the requested videos by users have only one replica that resides in the single server at the top of the content delivery path and request routing follows the shortest path routing scheme. To make caching decisions, the caching scheme works in rounds and at each of the round, the edge routers collects the request statistics of the video contents. Edge and intermediate routers calculate the caching utility based on the aggregated request statistics of the contents and these aggregated statistics along with the local caching decisions are forwarded from downstream to upstream cache routers along the content delivery paths to take caching decisions.

The on-path caching scheme PopCache [36], brings the idea that an individual ICN cache router can apply a simple probability to cache content more or less along the content delivery path according to content popularity for improved data dissemination. An analytical model is developed to evaluate the probabilistic scheme comparing it with other improved data dissemination benchmark schemes. The proposed probabilistic scheme outperforms other on-path caching schemes such as indiscriminate caching [27], caching with a certain probability and probabilistic path capacity-based scheme [131] in terms of expected round-trip-time and server hit rate for cascading and binary network topologies.

Cache routers cache contents along the content delivery path with different probabilities in the Least Unified Value (LUV)-path caching strategy [136]. Disparity of caching probabilities is increased along the content delivery path between upstream and downstream routers and the cache routers near the content providers (upstream routers) have higher probability to cache new content than the routers near the content consumers (downstream routers). Additionally, contents get higher caching probability near the consumers if they are not cached near the provider by means of accumulated probability. A weight value is composed for each content in a router based on the content popularity combining LFU and LRU strategies in a unified form and a cost for each of the cache router considering the distance of the router from the content provider.

In order to fully exploit the built-in caching capability of ICN, the content space is partitioned and the content name is hashed to content routers to select caching location and forward request routing along the content delivery path in the collaborative caching scheme named CPHR [10] while assuming symmetric paths for the request routing and content delivery. In CPHR, the content space is mapped into a hash space, and then contents are partitioned evenly into a number of caches in the network according to their corresponding hash values. A content request is first forwarded to the nearest cache assigned with its corresponding partition for achieving a cache hit before going outside of the network. While forwarding the request, the local caching tables of other cache routers along the content delivery path do not need to be checked. On the way back along the reverse path to the content requester, the corresponding requested content coming from its originating server is cached only in the assigned cache router.

Routing content requests to the assigned cache, before routing request to the content provider, can inevitably result in path stretch that can incur considerable extra propagation latency and significant increase in bandwidth consumption of the internal link. CPHR controls this negative impact on performance by constraining the worst path stretch threshold value during the cache assignment process to provide a flexible tuning knob for the network operator to make a favorable trade-off between the overall cache hit ratio and the incurred path stretch. It has been demonstrated by simulations that CPHR can significantly increase the overall cache hit ratio while having an acceptable increased of propagation latency.

To decide whether content should be cached or not along a content delivery path, the probabilistic algorithm, named Prob-PD [127] aims at achieving reduced content retrieval latency. For latency reduction, Prob-PD considers the distance ratio of each cache node and the content source serving content as well as the content's popularity ratio observed on the same cache node. The main goal of Prob-PD is to investigate how the combination of these distance and popularity factors benefits ICN on-path caching problem. Prob-PD outperforms several other on-path schemes such as CEE [27], ProbCache, [131], Fixed probability caching, LCD [137], and Degree centrality scheme [119] in terms of cache hit rates and cache replacement rates but produces anomalies for content delivery time. Moreover, Prob-PD algorithm needs a better way of approximating content popularity as it depends on the nature of workload.

A thorough evaluation of the on-path caching performance of CCN is performed in [119]. Extensive simulation is conducted to assess whether heterogeneous cache sizes can provide performance gain over a homogenous cache allocation and if the performance gain is consistent across all network topologies. Several graph-related centrality metrics are considered for proportionately allocating cache spaces heterogeneously and comparing the performance with homogeneous cache allocation. Simulation results depict that, very limited performance gain is achievable for heterogeneous cache allocation in comparison with homogenous allocation, and the simplest degree centrality can be the best choice for heterogeneous cache allocation.

The performance evaluation of the data transfer model, along the on-path caches in CCN is done in the analytical study of [104]. The authors propose analytical models for both a single cache and a network of caches and demonstrate that, content popularity, content size and the cache size are the key system parameters to tune caching performance of the chunk based communication of CCN.

5.4.2. Off-path based schemes

Five different hash routing techniques are devised in [145] aimed at having a viable and efficient caching approach in a domain-wide ICN deployments without requiring the cache nodes to maintain per-content state information. The proposed schemes require edge-domain routers and cache nodes to implement a hash function. The hash function is used by the cache nodes to identify what contents they can cache, and by edge routers to deal with the content placement to the relevant cache nodes, and request routing to the desired cache nodes. The differences among the five proposed techniques exist in the routing and replication process followed for requests and contents. The hash-based schemes make use of all available cache spaces within a domain and outperform two on-path caching schemes [27,131] in terms of cache hit ratio and inter-domain traffic loads on the links. The proposed techniques have not considered the increased latency occurred by the detouring required to look up the responsible caches while limiting their applicability's in small domains.

Hash-routing techniques in [145] improve the caching performance in terms of cache hits but at the expense of increased latency caused by the travelling of the extra number of hops to find the cached content in the network. This incurred latency because of the detouring can be negligible for a small network but can become remarkable even increasing to the prohibitive levels for a large size network. In order to deal with the extensive detour delays occurred in [145], nodal/domain clustering techniques have been investigated in [146] to split the large domain in clusters to apply hash-routing in the subset of cache nodes of each cluster instead of the whole network. Hence, the hash- routing techniques of [145] have been extended in [146] by providing a practical way to bound latency while making the hash techniques suitable for network topologies of arbitrary size.

To propose the domain clustering/partitioning techniques, detailed mathematical analysis have been done to find the evidence of existing trade-off between the probability of finding a requested content cached within the domain and the incurred latency for the retrieval of that content. The proposed domain clustering techniques are applied in an offline manner and split the whole network domain in smaller clusters and consequently each cluster is in the charge of a separate hash-function which only applies to the specific nodes in that cluster. The clustering algorithms partition the domain into clusters based on a hybrid similarity metric that is comprised of the actual topological latency/distance and the pairwise Euclidean distance of the content popularity between two domain nodes.

The usage of the Euclidean distance of content popularity transforms the Euclidean space of popularity into a metric space that is combined with the topological distance of the nodes in the network. Hence, the domain clustering algorithm forms clusters where the nodes of each cluster are not only in close proximity to each other, but also consumers with similar request patterns. The extended clustering technique [146] with the original hash-routing proposal [145] can reduce the excessive latency due to hash-routing while achieving considerable cache hit rates compared to the original hash-routing scheme [145] applied in the whole domain.

The distributed collaborative forwarding and caching scheme in [37], uses a new component, named Availability Information Base (AIB) for caching and request forwarding, based on the popularity ranking sequence of contents, produced by all the participating cache routers in an AS. A dynamic adaptive content store division scheme is applied on the cache spaces of the routers for real network scenarios. The distributed and the dynamic schemes improve content access cost and cache miss rate than the compared on-path schemes for different network settings by taking full advantage from peer cache nodes, but unfortunately they lack feasibility and complexity analysis.

Three off-path caching strategies, Basic, Adapted and Stacked in [109], aim to efficiently use the cache space by avoiding redundant caching of data items. All three off-path schemes store contents deviating from the direct path between content source and consumer in three different ways hierarchically considering content popularity. The proposed off-path schemes outperform on-path popularity-based scheme with LRU and LFU policies, in terms of cache-hit efficiency, mean hop delay to get requested content and power consumption in terms of energy efficiency for various cache sizes while considering non-realistic network topologies only.

TECC [143] is an off-path collaborative caching scheme which jointly and consistently provisions content routing and caching capabilities to obtain the benefits of both for content centric networks. TECC considers content caching as an inherent underlay capability same as underlay routing. The network is modeled as a flat structure where any node can be the caching parent of any other node and managed by a single administrative domain. TECC outperforms hierarchical on-path caching scheme by sending fewer requests to the content servers as the collaborating off-path peer caches largely absorb cache misses and fetch more contents within the network increasing caching diversity. TECC improves both network link utilization and user perceived performance comparing to on-path scheme.

The SNC [40] off-path caching approach supports enhance seamless mobility for ICN. SNC is based on proactive caching of information requests and the corresponding information items to an optimal subset of proxies which are one hop away from the proxy to which currently a mobile is connected to. Hence, when a mobile connects to one of these selected proxies in this subset, it can immediately retrieve the information it could not receive due to its disconnection from the current proxy. The authors design a target cost function to capture the tradeoff between the average delay for a mobile to retrieve an information item and the corresponding caching cost and aim to minimize this function. SNC performs better than full proactive caching when caching is done proactively at all neighbor proxies, and no proactive caching approach, without considering the influence of network topology, information size, and duration of disconnection period.

On account of the unawareness of the neighbor content routers' caching information, content requests can miss the content router caching the desired information if it is not on the routing path from the requester to the original server, even though the desired router may reside closer to the requester than the original server. Hence, the caching scheme, named Independent allocation algorithm in [141] promotes the content routers to advertise or broadcast their caching information to the neighbor caches within their vicinity deviating from the direct content delivery path to significantly reduce the content retrieval latency.

SCAN [138] exploits nearby multiple off-path and on-path cache copies of content to provide efficient scalable content delivery. SCAN addresses the inefficiencies of conventional IP routing and proposes that every content router caches contents while maintaining local and neighbor routers' cache information. SCAN performs both default IP routing to ensure reachability to the requested content and routing called scanning to efficiently

Table 5
Overview of Path-Based Schemes.

Caching Schemes	Path-Based Caching Model	Caching Locations
Indiscriminate [27]	All cache routers cache all contents indiscriminately	On-path
Cache “less for more” [117]	Higher betweenness centrality- valued nodes selected as cache nodes	On-path
Caching from a content delivery path [136]	Content routers are assigned weight (significance) relating distance from the source of content	On-path
CPHR [10]	Content name is hashed to the cache content router for caching and request routing	On-path
StreamCache [115]	Downstream routers passes the summarized aggregated video content requests and local caching decisions to the upstream routers along content delivery path	On-path
RPC [110]	Caching decision is made based on the topology value of the cache router along the content delivery path	On-path
NCPP [111]	Cache node having higher Node utilization ratio (NUR) value along the path caches content	On-path
Towards on-path [127]	Distance from source to cache node is a decision making factor to select cache node	On-path
CLS [121]	Contents pushed up and pulled down along content delivery path	On-path
Breadcrumbs [120]	Content routing history is stored at cache routers along content delivery path	On-path
CATT [118]	Node having largest degree along content delivery path cache contents	On-path
WAVE [38]	Caching suggestion is given by the upper level routers to the lower level routers	On-path
Age-based [39]	Upstream and downstream routers dynamically set content age	On-path
ProbCache [131]	Cache routers probabilistically cache contents for fair sharing of available cache capacity of a path	On-path
Popularity-driven [34]	Cache routers cache content based on the recent aggregated content requests of its subordinates	On-path
PopCache [36]	Cache router applies caching probability according to popularity distribution of content	On-path
On Sizing CCN content stores [119]	Cache routers cache contents indiscriminately	On-path
Caching performance of CCN [35]	Considers indiscriminate, fixed probabilities and Leave copy down caching approaches	On-path
TECC [143]	Traffic engineering guided collaborative caching on flat network model	Off-path
Efficiency of on- and off-path [109]	Cache allocation occurs hierarchically according to popularity-based caching in three different ways	Off-path
Time-shifted TV [140]	Content routers cache and forward queries based on neighborhood collaboration	Off-path
Proactive caching [40]	Selected one-hop neighbor proxies proactively cache contents	Off-path
Optimal caching with content broadcast [141]	Content broadcast-based among neighborhood	Off-path
Cooperative caching [142]	Content routers use hash-based and directory- based cooperative schemes	Off-path
Content hierarchical intra-domain [133]	Cache contents are registered to border routers for request routing and advertised in the whole AS	Off-path
Efficient cache availability [114]	Content ID is hashed to select cache router and for request routing to cache content	Off-path
Hash routing [145]	Hash function does content placement and request routing to the cache nodes	Off-path
Efficient hash routing [146]	Hash-routing based domain clustering technique is used for cache content placement and request routing	Off-path
Collaborative forwarding and caching [37]	Contents are ranked based on popularity for caching	Off-path
SCAN [138]	Content routers perform scanning to locate additional multiple copies of cache contents along with default IP routing	On and Off-path

search close and multiple cached copies of the requested content in the neighboring routers. SCAN is superior to IP routing in terms of average hop count for content delivery, network traffic reduction, load reduction on the original server and load balancing among the links of the network. On the other hand, SCAN neither considers any mechanism to select content routers for optimized performance nor investigates the effects of caching policies.

The collaborative caching scheme in [140], caches and searches content chunks deviating from the direct content delivery path within a nearest neighborhood to deal with large video streams with on-demand access for CCN. The scheme reduces cross-domain traffic becoming ISP-friendly by utilizing nearby cache resources to increase caching diversity based on a modulo function and performs better than on-path LRU policy having insignificant communication overhead.

To increase cache efficiency in terms of increased content availability and reduced content redundancy, the content caching and routing scheme of [114] uses hash-function. The scheme controls distribution of contents across the network in order to control the number of redundantly duplicated contents in an AS. The hash function maps the content to the designated cache routers for caching across a local AS based on content ID and later performs the request routing to the cache content when needed. Rather than placing the content in an optimal position, the caching scheme spreads the contents sufficiently to retain the maximum amount of contents in the cache routers. If adequate numbers of ASes cooperate for caching and request routing, the amount of cache distribution gets bigger by achieving higher content availability and lower redundancy. As the hash-based scheme utilizes the cache space efficiently, it performs better than on-path schemes in terms of cache hit ratio, reduced server hit ratio, cache eviction ratio since these on-path schemes try to cache contents on the content delivery paths only. The proposed scheme incurs increased hop count ratio for content retrieval due to the detour of the content requests deviating from the content delivery paths.

Adjacent content routers along a small group of nearest neighbor routers use a hash-based system to take collaborative caching decision and a directory-based system to forward the request routing for CCN in [142]. To achieve reduced redundant caching, adjacent routers exchange control messages within the neighbor to efficiently cache large video streams within a domain where the routers are assigned unique labels in a distributed style. Caching gain is improved comparing with LRU and LFU schemes by increased total caching diversity, increased per-video caching diversity, significantly reduced inter-domain traffic within an ISP for VoD and catch-up TV services while causing increased communication overhead.

To eliminate caching redundancy and fully utilize the cache space within an AS, the IDCC caching scheme [133] caches content probabilistically by intra-domain content advertisements and co-

Table 6
Summarized Taxonomy and Comparison of ICN Caching Schemes.

Caching Schemes	Popularity-Based		Location-Based		Collaboration-Based			Path-Based	
	Static	Dynamic	Topology	Neighbor hood	Explicit	Implicit	Non-Cooperative	On-Path	Off-Path
Age-based [39]		√	√			√		√	
Breadcrumbs [120]						√		√	
Cache “less for more” [117]			√	√			√	√	
Collaborative forwarding and caching [37]	√	√			√				√
CATT [118]			√	√		√		√	
CLS [121]						√		√	
Cooperative caching [142]				√	√				√
Content hierarchical Intra domain [133]		√			√				√
Caching from a content delivery path [136]		√				√		√	
CPHR [10]						√		√	
Caching performance of CCN [35]		√	√				√	√	
Efficiency of on-path and off-path [109]	√				√				√
Efficient cache availability [114]						√			√
Efficient hash routing [146]							√		√
Hash routing [145]							√		√
HCC [112]		√	√	√	√				
Indiscriminate [27]							√	√	
Impact of traffic [123]		√	√				√		
LCD [137]						√		√	
MuNCC [124]				√	√				
NCPP [111]	√						√	√	
NCCM [126]		√			√				
On sizing CCN content stores [119]			√				√	√	
Optimal caching with content broadcast [141]				√	√				√
PPC [116]		√		√					
ProbCache [131]						√		√	
Proactive caching [40]				√	√				√
Popularity-driven [34]	√	√				√		√	
PopCache [36]	√						√	√	
RPC [110]		√				√		√	
SCAN [138]				√	√			√	√
StreamCache[115]	√					√		√	
Towards on-path [127]		√					√	√	
Time-shifted TV [140]				√	√				√
TECC [143]					√				√
Value-based [132]		√					√		
WAVE [38]		√				√		√	

operation among the cache routers. IDCC makes a caching decision considering cache capacity, cache time and traffic speed and stores contents, having different popularities, in the AS. Hence, IDCC keeps only one replica for each cached content using content advertisements among the cooperated routers and avoids router overloading situation by keeping multiple replicas of the very popular contents in a hierarchical manner. IDCC shows superiority to on-path schemes such as CEE [27] and ProbCache [131], but cannot handle highly skewed popular contents as numbers of replicas become increased in this situation. Table 5 highlights the main attributes of Path-based caching schemes.

Finally, it is important to note that major schemes couple different operational mandates across caching systems. More importantly, the tabular summary in Table 6 aids ICN caching researchers in establishing the merit of each of these schemes as a basis for their contribution, based on their adopted functional approaches.

6. Qualitative assessment and comparison of the ICN caching schemes

We present a detailed qualitative assessment and comparison of ICN caching schemes, encompassing mainstream and hybrid approaches under a multiplicity of objectives. As presented in Sections. 4 and 5, many design objectives are inherently contradicting, and caching schemes often attempt to meet only a subset of general ICN caching objectives as highlighted in Section 3. Therefore, we have adopted four performance metrics in our assessment: communication overhead, scalability, availability, and diversity of caching schemes.

6.1. Communication overhead

We consider the periodic exchange (or broadcast) of cache related information and control messages among cache routers which are mandated for taking caching decisions as additional communication overheads of the caching schemes. We assess the communication overhead as High, Medium and Low for our qualitative performance assessment. When in a caching scheme, the cache related information is exchanged or communicated along all the cache routers periodically in a global fashion such as within an AS or ISP or within a neighborhood of cache routers, we consider the communication overhead as High. When the caching information is exchanged along the content delivery path only as a hierarchical fashion within a limited number of cache routers, we consider the communication overhead as Medium.

Finally, when a caching scheme does not exchange any control information or exchanges very little cache related information among the cache routers and the routers take caching decision independently, we consider the communication overhead as Low for the scheme.

Explicit collaboration-based schemes [37,40,112,124,133,138] produce high communication overhead as they periodically exchange content caching related information or request routing information globally within an AS (an ISP) or within a neighborhood to collaboratively take caching decision. On the other hand, implicit collaborated schemes [10,34,39,110,121,127,136] have medium communication overhead as these schemes advertise or flood or exchange content caching or routing related control messages within a small number of cache routers most of the time hierarchically along content delivery paths instead of globally for taking collaborative caching decision. Non-cooperative schemes [35,116,119,127,145,146] produce low communication overhead as for these schemes cache routers take content caching and routing decision independently without collaborating with one another either exchanging very little or no control message.

In on-path based schemes [35,36,111,117,127,131] no information or very little caching information is exchanged or broadcasted. In this case, collaborating cache related information is exchanged locally among the cache routers along content delivery paths only [10,39,115,118,120] for caching decision. Hence, they produce low or medium communication overhead. On the other hand, off-path schemes globally in an administrative domain [37,133,143] or within an extending neighborhood [138,140,141,142] exchange cache related control messages to coordinate caching decisions while producing high communication overhead.

6.2. Scalability

The factors which can limit the scalability of a caching scheme can be different such as network topology, cache size, scope of a caching scheme, number of producers, number of content requests, types of content, mobility prediction, structure of a cache network, popularity distribution of contents etc.

Explicit collaborated schemes and off-path-based schemes are usually effective within a single administrative domain (an ISP) [37,133,142,146], an extended neighborhood [40,112,124,138] or only consider abstract network topologies to show their effectiveness [109] whereas, implicit collaborated schemes, non-cooperative schemes and on-path schemes suffer from other scalability issues such as they depend on certain pattern of content popularity distributions [36,38,115,121], aggregated content request statistics collected from downstream cache nodes [115], network topologies [10,117,123,136], cache size [35,120], network size [34,116] and number of content requests [111,127]. These schemes also demonstrate local scalability along the content delivery paths only [10,118,131] and also consider non-congested network environment only [35,119] to show their efficiencies.

6.3. Diversity

By diversity, we mean the number of distinct or different cache contents which are cached or stored in a network [133,140,142]. In other words, cache diversity expresses the ratio of the unique contents stored across all the caches without any repetition over the whole network cache size [35]. Hence, if a caching scheme reduces or eliminates caching redundancy among the cache routers for a fixed total cache capacity of a network, the caching diversity is increased.

The more distinct content chunks can be cached in a network, the more caching diversity a caching scheme can achieve and the better caching performance can be attained. We assess the metric diversity as High and Low. We consider several issues. to measure the diversity of a caching scheme such as types of cache content, popularity of content, the scope of a caching scheme such as a single administrative domain or AS (an ISP), or a neighborhood of cache routers or a content delivery path.

The collaborated caching schemes can achieve greater caching diversity than the non-cooperating ones while reducing caching redundancy as they consider all the cache routers in an AS (an ISP) or in a neighborhood or in a content deliver path as a big caching pool to improve caching performance [10,39,113,121,133,140,143].

Because of the cooperation among the cache routers, a single copy of cache content can be kept within a local AS or a neighborhood or a path to reduce the caching redundancy to minimum. As a result, collaborated approaches are capable to cache more diverse contents in the network and achieve better cache diversity and caching performance than the non-cooperative schemes. Off-path caching schemes achieve greater caching diversity than on-path schemes as all cache routers in off-path schemes can cache contents by better utilizing the caching resources [109,113,114,142,145]. Additionally, caching schemes can target to increase caching diversity reducing caching redundancy while focusing on some specific issues such as different types of contents [10,123,132,140,142], popularity of contents [36,111,116,136] and caching only at some specific cache nodes [117,118].

6.4. Availability

To define the availability of requested content, we consider several issues of the caching schemes such as the scope of the caching scheme: a single administrative domain or a neighborhood of cache routers or the content delivery path, types of cache content and popularity distribution of content. We assess the availability of the caching schemes as High and

Low while considering the issues. The same trend of caching diversity is also observed in case of availability of cache contents. Collaborated and off-path schemes achieve better global or local availability of cache contents because of better utilization of available caching resources attaining improved performance than the non-cooperative and on-path schemes [40,109,113,118,121,136,141]. Moreover, caching schemes can aim to achieve greater caching availability considering different issues such as different content flows, different types of cache contents and popularity distributions of content [36,39,111,115,123,131]. In Table 7, we summarize our qualitative assessment and comparison of ICN caching status quo.

7. Quantitative assessment and comparison of ICN caching schemes

In this section, we evaluate the performances of in-network ICN caching schemes via extensive simulations. To emphasize comprehensiveness, we selected at least one representative caching scheme from each of the categories of our proposed classification system. A core design principal in this performance evaluation was adopting realistic performance metrics. We correlated our quantitative analysis with our qualitative performance analysis to build a progressive guide for research directions and development for the ICN research community, which are presented in Section 9.

7.1. Simulation environment

To evaluate the performances of the caching schemes, we have considered the Transit-Stub network model [147] as our network topology. The reason behind the selection of the Transit-Stub model is that this model follows the standard Internet routing policy [147]. Hence the sensitivity analysis of the caching schemes while deployed in the Transit-Stub model can give us realistic analysis. Our considered Transit-Stub topology has 5 domains and consists of 40 cache routers where the cache routers have the same cache capacities. Our performance evaluation scenarios consider total 100 users where there are 75 content producers and 25 consumers. Each of the producer produces 1000 contents where the size of each content is 1 KB. The producer produces the content, receives the Interest packet from the consumer and responses to the requester consumer by Data packet. The consumers issue the Interest packets for their desired contents with a rate of 20 Interest packets per second. Content popularity value follows the well-known Zipf-like popularity [135] and the Zipf parameter value is set to 0.95. We have assumed that the Interest

Table 7
Qualitative Assessment and Comparison of The ICN Caching Schemes.

Caching Schemes	Communication Overhead	Scalability	Availability	Diversity
Age-Based [39]	Medium as upstream and downstream content routers exchange messages to collaboratively configure content age along the content delivery path	Scalable without maintaining highly dynamic contents as cache routers far from the content server take relatively long time to refresh their contents	High availability for popular contents along the content delivery path	High as contents are distributed along the whole network based on their popularities aiming reducing redundancy
Breadcrumbs [120]	Medium as cache routers exchange routing history of cache content (caching trail) along the content delivery path	Not scalable as proposed scheme is efficient when the cache size is relatively small	Medium as Best Effort content searching scheme does not guarantee of locating content	Low as all cache routers cache all contents indiscriminately
Cache "Less for More" [117]	Low as for static topology, no exchange of information. Message propagation overhead is one hop only for dynamic topology	Depends on betweenness centrality distribution of the network topology which skews the load balance	High locally along the content delivery paths as contents are cached at the important cache nodes having higher betweenness centrality values	High locally along content delivery paths as the higher betweenness-valued nodes only cache contents
Collaborative forwarding and caching [37]	High because of periodic exchange of cache related information within an AS	Efficient for a single AS	High intra-domain availability as cache routers take full advantage of cache available from other routers	High within an AS as the cache routers explicitly coordinate for taking caching decision
CATT [118]	Medium as content provider floods potential value of content among neighbors using advertising message along content delivery path	Not scalable as PBR can work within a limited scope only because of flooding advertisement messages. PBR is scalable along with Random walk algorithm	High availability of content only locally as PBR explores multiple copies of queried content within a limited scope such as within a couple of hops along content delivery path	High locally along content delivery path as node with the largest degree only caches contents
CLS [121]	Medium as caching information is exchanged along the data delivery path in a hierarchical fashion	Scalable while majority clients require few contents (highly skewed exponent value of Zipf popularity distribution)	High for both popular and unpopular contents locally along content delivery paths as contents pushed up (while eviction) and pulled down (for increased requests) one level	High as reduces unnecessary repetitious caching at multiple levels of hierarchical topology ensuring a single copy of each content along the content delivery path
Cooperative caching [142]	High as the content routers exchange control messages within their closest 2 hops neighborhoods	Effective for video stream delivery within a single administrative domain	High availability of large-scale video streams with on demand access and Time-shifted TV contents among neighborhoods in a single domain	High diversity for VoD and time-shifted TV in a neighborhood as redundancy is eliminated among adjacent cooperative routers
Content-hierarchical intra-domain [133]	High as advertisement messages are exchanged within a domain for caching decision and load balancing of routers while caching popular contents	Scalable within a small single domain as long as the popular contents are not very concentrated	High within a domain as contents of different popularities are cached	High within a domain as only single replica exists for each content unless the content is very popular
Caching from a content delivery path [136]	Medium as control messages are communicated among a small fraction of content routers	Scalable for different network topologies and cache sizes while the topology knowledge is known in advance	High availability for popular and unpopular contents along the content delivery path as upstream and downstream routers cache new contents with different probabilities	High diversity along the content delivery path as caching redundancy is reduced while increasing disparity of caching probability of content routers for caching new contents
CPHR [10]	Medium as caching information is exchanged among cache routers along the content delivery paths within an ISP	Effective within a small sized ISP for file sharing and VoD contents only	High for file sharing contents and VoD contents along the content delivery paths within a small network domain	High for file sharing content and VoD contents locally along content delivery paths within a small network domain as caching redundancy is reduced
Caching performance of CCN [35]	Low as no cache related information is exchanged among the cache routers	Scalable as considers larger cache sizes, realistic popularity distribution and real network topologies while considering operating in a non-congested regime	Low for multi-path routing as it induces cache pollution along multiple paths resulting into cache evictions on multiple caches	Low for highly skewed popular contents and High for contents having flat popularity
Efficiency of on-path and off-path [109]	High as content popularity information is exchanged among cache routers globally in hierarchical way deviating from the content delivery path.	Effective while considering abstract network topology only, not considering real network topologies	High availability for popular contents as available cache space is utilized efficiently to store popular contents for abstract network topology only	High for popular contents as caching redundancy is eliminated by hierarchical global collaboration among the cache routers for abstract network topology only
Efficient cache availability [114]	Low within a local AS as hash function is used but High when a set of ASes propagate updates about their interests about caching	Depends on the number of the cooperating ASes and the policies among the ASes	High availability of content locally within an AS as hash-function does controlled distribution of contents	High diversity of content locally within an AS as hash-function does controlled distribution of contents

(*continued on next page*)

Table 7 (*continued*)

Efficient hash routing [146]	Low as hash function is computed in a distributed manner by edge routers and cache nodes	Scalable within a single domain as the hash function is computed in a distributed manner and applied to the specific nodes of a specific cluster or partition of the network	High availability for distinct contents as the hash function places one content once only within a single domain or network	High diversity within a single domain or network as the hash function places a content at most once in a specific cluster or partition based on the clustering approach
Hash routing [145]	Low as hash function is computed in a distributed manner by edge routers and caches require minimal inter-cache coordination for content placement and content request routing	Not scalable as proposed hash routing cannot be applied to the entire amount of traffic of a large network as the stretch of the detour path can become very big	High availability for distinct contents as the hash function places one content once only within a small network	High diversity within a small network (an ISP) as the hash function places a content at most once preventing redundant content caching
HCC [112]	High as communication messages are exchanged for initialization and updating a cluster, for calculating router importance value, for collecting popularity information and forwarding the probability matrix values and the popularity class lists among neighborhood	Not scalable because of the periodic exchanges of messages between the cluster head routers and all other cache routers for creating cluster, calculating node importance value, collecting popularity information	High availability for popular contents locally along the neighborhood or cluster consisting of the cluster head router, gateway routers and member cache routers	High availability for popular contents locally along the neighborhood or cluster consisting of the cluster head router, gateway routers and member cache routers
Impact of traffic [123]	Low as no caching information is exchanged between the two layers of caches as the first and second layer caches work independently	Effective while considering the cache network as generic hierarchical network and the precision level of the scheme is not considered	High availability for diverse Internet contents such as web, file sharing, UGC and VoD	High as the proposed two-layer caches cache different types of contents based on content popularity distribution, content population size and content size
LCD [137]	Medium as cache suggestion bit is passed from the cache hit point towards the requesting client along the content delivery paths	Scalability depends on the popularity distributions of the requested contents	High availability for different popularity leveled contents locally along the content delivery paths	High locally along the content paths for different popularity leveled contents
MuNCC [124]	High within a neighborhood as content caching and eviction require periodic cache information exchange among neighbor routers	Scalability depends on the total number of cached contents and the size of the formed neighborhood	High locally around the neighborhood as cached routers are coordinated to reply content requests forwarded from their neighbor routers	High locally among neighbor routers because of the neighborhood-based caching and collaborated cache eviction
NCPP [111]	Low as no cache related information is exchanged among the cache routers	Scalability depends on number of content requests	High along the content delivery paths locally for popular video contents for small number of content requests	High along the content delivery paths locally for popular video contents for small number of content requests
NCCM [126]	High as the content request rates and the current cache status exchanged periodically among the content routers and SDN-based controller	Scalability depends on the number of popular contents	High along the administrative domain for fewer popular contents only	Low as the controller caches the popular contents only in the distributed content routers
On-sizing CCN content stores [119]	Low as no cache related information is exchanged among the content routers	Scalable while considering non-congested environment where links have infinite capacities	Low as indiscriminate content caching results into huge cache evictions	Low as cache nodes cache contents indiscriminately resulting caching redundancy
Optimal caching with content broadcast [141]	High as cache content information is exchanged among the region of neighbor caches and the region of neighborhood is increased sequentially hop-by-hop basis	Scalable only while content requests from an end user belong to contents originated from the same stub	High locally around neighborhood as cached contents of the neighbor cache routers are used to fulfill query	High locally among neighbors as neighbors cache spaces are utilized
Probabilistic [131]	Low as no cache related information is exchanged among the cache routers	Scalable locally considering only content delivery path	High along content delivery path as content routers probabilistically cache contents to fairly multiplex path caching capacity among different content flows	High along content delivery path as the cache contents are distributed probabilistically along the path eliminating caching redundancy
Proactive caching [40]	High as the proxies within the neighborhood exchange information several times to take proactive caching decisions for mobile node's subscriptions when hand-off occurs	Scalable within a neighborhood while assuming transition probabilities of mobile nodes connecting to the neighbor proxies are known	High availability of cache contents for mobile node's subscriptions within selective one-hop neighbor proxies	High along neighborhood as contents are cached proactively at an appropriate subset of neighbor proxies matching mobile's subscription

(*continued on next page*)

Table 7 (*continued*)

Caching Schemes	Communication Overhead	Scalability	Availability	Diversity
Popularity-driven [34]	Medium as caching information is exchanged among the cache routers along the content delivery path in a hierarchical way	Not scalable as proposed algorithms achieve near optimal performance for small scale networks	High availability within an AS (an ISP) for popular web requests contents only	High for web requests contents within an AS as both consistent and inconsistent popularity are considered to minimize redundancy
PopCache [36]	Low as only the information of hop count from the server is needed to be exchanged among cache routers	Scalable while considering only certain range of exponent value of Zipf content popularity distribution	High locally along content delivery paths for contents having different popularities	High locally along content delivery paths as cache routers cache probabilistically both popular and unpopular contents in a distributed way
PPC [116]	Low as no caching information is exchanged among the cache routers to predict the future popularity of the incoming video content and the cached content	Scalability depends on the number of the video chinks passing through the cache nodes	High availability of popular video chunks only within a network depending on the number of passing video chunks through the cache nodes	High diversity for popular video chunks only within a network depending on the number of passing video chunks through the cache nodes
RPC [110]	Medium as the topology level values are exchanged among the cache routers from upstream towards the downstream routers in a hierarchical way	Scalability depends on the number of the topology levels of the cache routers and the popularity pattern of the video chunks	High availability of popular video chunks only locally along the content delivery paths	High diversity for popular video chunks locally only along the content delivery paths
SCAN [138]	High because of the periodic exchange of content routing information among the neighbor content routers	Scalable as bloom filters compress content routing information and use probabilistic cache content information decaying to mitigate false positivity	High availability of the contents as scanning (proposed routing scheme) locates nearby additional multiple copies of the requested contents in the neighborhood along with mandatory IP routing where IP routing guarantees reachability to requested content	Low as content routers cache all contents indiscriminately
StreamCache[115]	Medium as summarized statistical information and caching decisions information are exchanged among the content routers	Not scalable as cache allocation table is aggregated from child cache node to parent node along the content delivery paths	High availability of contents along the content delivery paths locally as it caches both popular and unpopular contents based on the cache utility function as there is no coordination about caching among the edge routers	High diversity of contents along the content delivery paths only if the intermediate and edge cache routers utilize their cache spaces properly, otherwise not
Towards on-path [127]	Low as no control message is exchanged among the content routers	Not scalable as capable to handle only few content requests	High availability of highly skewed popular contents locally along the content delivery path for only small number of contents	Low as caches only highly skewed popular contents along content delivery paths
Time-shifted TV [140]	High as the content routers exchange content caching and routing information within their closest neighborhoods	Effective within a single administrative domain (an ISP)	High availability of large-scale video streams with on demand access and Time-shifted TV contents within a single administrative domain (an ISP)	High diversity within a domain for large-scale video streams with on demand access and Time-shifted TV as cache routers take collaborative caching decisions within their closest neighborhoods to avoid redundant content caching
TECC [143]	High as the peer nodes exchange cache information to collaborate with one another to fetch content	Scalable for a single administrative domain while considering non-realistic network environment	High within a single administrative domain as the peer cache nodes collaborate with each other to fulfill content request	High within a single administrative domain as cache nodes collaboratively take caching decisions to avoid redundancy
Value-based [132]	Low as no information is exchanged among caches and caching decision is taken independently	Efficient while considering random graph topologies only, not considering realistic network topology	High for different types of contents as considers several content attributes while cache replacement	High for different types of contents
WAVE [38]	Medium as content caching suggestion information is exchanged among the cache routers hierarchically along the content delivery path	Not scalable as popularity counter is associated with content file, not with content chunk which is unrealistic	High availability for the most popular contents locally along the content delivery paths	High locally along content delivery path for popular contents as caching redundancy is reduced using cache suggestion bit in a hierarchical way

packets arrival process follow a Poisson distribution process [148]. For cache replacement, we have used the most widely used LRU policy for all the evaluation scenarios. We have used the ndnSIM tool which is an NS-3 based simulator for Named Data Networking (NDN), developed at UCLA [149] for our performance analysis.

7.2. Performance metrics

There are two metrics that have significant impact on ICN performance, and are in direct correlation with efficient content dissemination and discovery in ICNs. We hereby detail our experiments with cache hit ration, and average consumer delay, and elaborate on their impact on ICN performance.

7.2.1. Cache hit ratio

A cache hit means that a requested content can be retrieved from the cached copy of the content whereas a cache miss occurs when the content request cannot be satisfied by the cached copy and consequently the content request has to be redirected to the content producer. We measure the cache hit ratio as the ratio of the total number of contents retrieved from the caches to the total number of requested contents successfully delivered. A higher cache hit ratio value is desirable for any caching scheme as it reduces the content producer's load and the latency or delay to retrieve the requested contents.

7.2.2. Average consumer delay

Consumer delay means the total delay that a content consumer experiences to request a content and receive the corresponding requested content. Specifically, it is the time difference between the first attempt of sending the Interest packet and receiving the Data packet. Consumer delay includes the total timeout as well as the delay of retransmitted Interest and Data packets. We measure the consumer delay in real time instance instead of considering the number of hops to retrieve the requested content to evaluate the schemes in a more realistic way. We measure the average consumer delay as the ratio of the total time lapsed to retrieve the requested contents from the caches to the total number of the contents retrieved from the caches in the unit of Millisecond (ms).

7.3. Simulation scenarios

As we have already mentioned that to make our performance evaluation as a complete one, we have taken at least one representative caching scheme from our categories. We have chosen CEE [27], Prob (p) [137] and the Cache "less for more" [117] scheme (mentioned as CLFM in our performance evaluations scenarios) as the representatives of the non-cooperative, on-path caching schemes. We have selected CEE as this scheme is the mostly cited, compared and criticized in the ICN literature. Prob (p) [137] is a randomized version of the CEE scheme where each of the cache node from the location of the cache hit down to the requesting consumer can cache content for storing a copy of the requested content along the content delivery path. Prob (p) scheme has been found as an efficient caching scheme if the random caching probability value is selected appropriately. The CLFM scheme is a representative of not only a non-cooperative and on-path scheme but also a location-based scheme because it caches contents at the important network locations to efficiently utilize the cache capacity.

ProbCache [131] and LCD [137] schemes have been selected to represent the implicit collaborative schemes as they can be considered as the most cited implicit collaboration caching schemes for performance comparisons of ICN caching schemes. LCD [137] is a very well-known web caching scheme as it can avoid the amplification of replacement errors and provides caching exclusivity by caching different sets of contents at the multiple levels of the cache hierarchy.

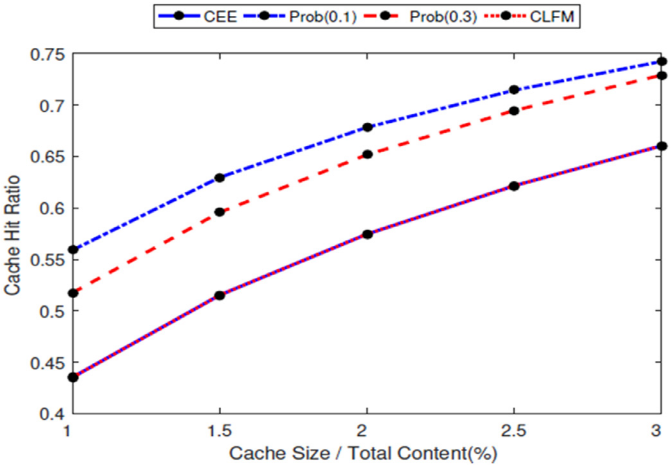


Fig. 9. Cache Hit Ratio of Non-Cooperative Caching Schemes.

Finally, we have selected the Collaborative forwarding and caching scheme [37] which we named as CFC in our simulations, as a representative of explicit collaborated, off-path scheme to make a comprehensive performance evaluation while covering all our proposed categories.

To do a clear and concise analysis, we have considered three different scenarios for our performance analysis as the following: Scenario 1 evaluates only the non-cooperative caching schemes, Scenario 2 compares the performances of the collaborative schemes and the Scenario 3 compares the performances of non-cooperative and the collaborated caching schemes while covering all the considered caching schemes.

7.4. Simulation results for different scenarios

In our results, the x-axis represents the cache size which is defined as a portion or percentage of the total content bytes and the cache size ranges from 1% to 3% of the total contents while the y-axis value represents the values of the considered performance metrics.

7.4.1. Scenario 1

In Fig. 9, the cache hit ratio for all the non-cooperative caching schemes increase along with the increment of cache size as more contents can be cached at larger sized caches. The Prob (p) scheme expectedly performs the best among all the non-cooperative schemes as it caches content in a probabilistic way rather than exhaustively caching every content at every cache like CEE or caching content only at the centrality located caches as CLFM. Smaller probability value produces higher cache hit ratio because the cache replacement errors get reduced if contents are cached with a low probability value. The betweenness-centrality based CLFM scheme performs the same as CEE because of the Transit-Stub topological issue as CLFM performs well in regular network topologies like tree topologies where the cache routers are not located sparsely.

The average consumer delay has expectedly coincided with the cache hit ratio in Fig. 10. If the consumer's requests can be satisfied by the caches resulting high cache hit ratio, the consumers experience less delay to retrieve their contents. As the Prob(p) scheme can achieve the largest cache hit ratio among all the non-cooperative schemes, it takes the smallest time to satisfy the consumer's request. Between CEE and CFLM, CFLM performs better in terms of average consumer delay as it selects caches in selective ways while having less cache replacements comparing to the CEE.

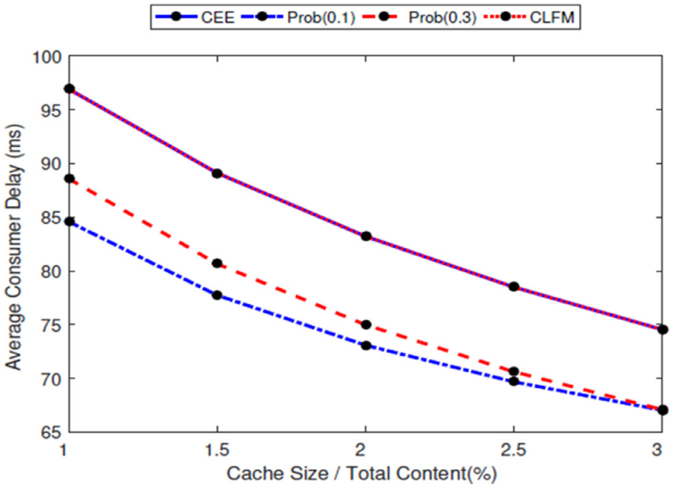


Fig. 10. Average Consumer Delay of Non-Cooperative Caching Schemes.

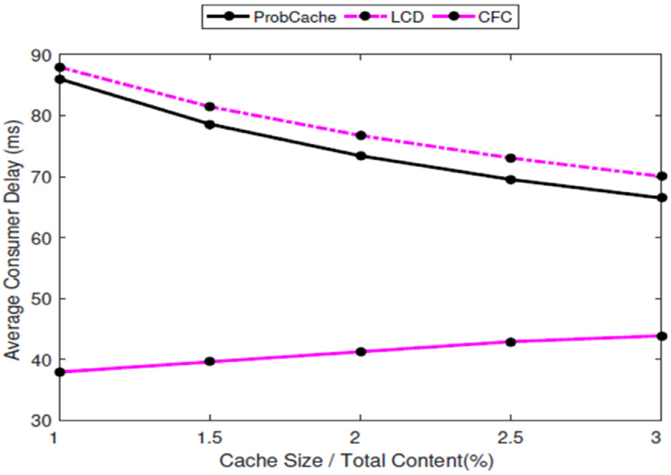


Fig. 12. Average Consumer Delay of Collaborative Caching Schemes.

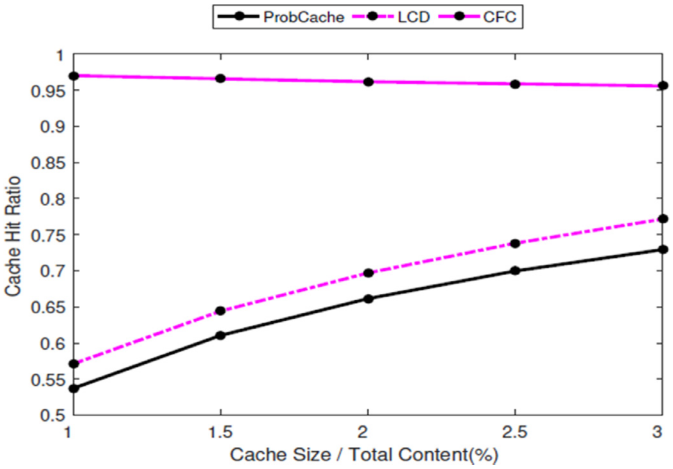


Fig. 11. Cache Hit Ratio of Collaborative Caching Schemes.

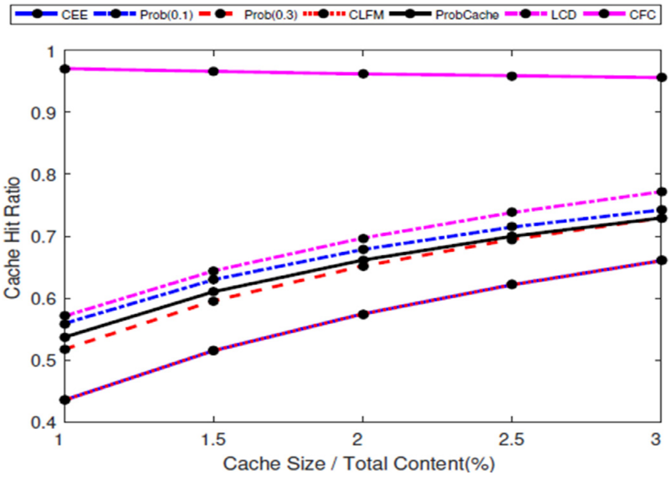


Fig. 13. Cache Hit Ratio of Non-Cooperative and Collaborative Caching Schemes.

7.4.2. Scenario 2

Fig. 11 compares the performances of the representative collaborative caching schemes in terms of cache hit ratio. The cache hit ratio increase as the cache sizes increases for all the schemes because of larger cache capacity.

We can observe the performance trends of the explicit and implicit collaborative schemes in Fig. 11 match with our qualitative performance analysis. The explicit collaborated, off-path caching scheme CFC generates larger cache hit ratio comparing to the implicit collaborated, on-path caching schemes ProbCache and LCD as CFC can utilize the caches more efficiently deviating from the content delivery paths whereas ProbCache and LCD utilize the cache spaces collaboratively along the content delivery paths only. Between the two on-path, implicit collaborated schemes, LCD shows better performance than the ProbCache. The reason is that the ProbCache assigns more weights or values to the nodes only near to the consumers along the content delivery paths to be selected as cache nodes while not fully utilizing the cache resources whereas LCD caches contents at multiple levels of the content delivery paths based on their popularity values satisfying multiple consumers while pushing the more popular contents sequentially near the consumers. Hence, cache exclusivity is achieved in LCD while achieving more cache hits comparing to the ProbCache.

Fig. 12 compares the performances of the collaborative schemes in terms of average consumer delay and coincides with the performance of the schemes in terms of cache hit ratio. As the explicit collaborated CFC achieves more cache hits comparing to the implicit ProbCache and LCD, consumers experience less latency or delay to retrieve their requested contents that also confirms our qualitative analysis. As ProbCache preferably caches the popular contents near the consumers, consumers need less time to get the requested contents than LCD as we have chosen high skewness value for the popularity distributions of content.

7.4.3. Scenario 3

Fig. 13 demonstrates the overall performance comparisons of all the representative caching schemes to show the basic trend of the performances of the schemes. Non-cooperative schemes achieve less cache hits as the cache nodes do not collaborate with each other to take caching decisions for efficient usage of caching resources resulting into unnecessary caching redundancy.

Collaborated schemes achieve more cache hits utilizing the cache resources efficiently but require extra overhead to achieve this by exchanging communication messages. Implicit collaborated scheme achieves better cache hit ratio than non- cooperative schemes for utilizing the cache resources locally along the content delivery paths whereas explicit collaborated scheme achieves the maximum cache hit ratio while having more communication overhead than the implicit, on-path schemes.

Fig. 14 depicts the performance comparisons of all the selected schemes to show the performance trend in terms of average consumer delay. As we already have seen that non-cooperative CEE,

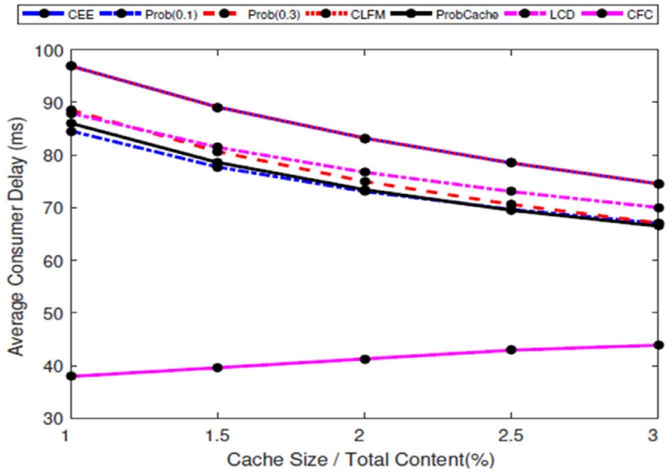


Fig. 14. Average Consumer Delay of Non-cooperative and Collaborative Caching Schemes.

Prob (p) and CLFM have less cache hits, so consequently increasing the delay to retrieve contents. Moreover, explicit collaborated CFC scheme provides the least consumer delay by achieving the maximum cache hit ratio among all. The implicit collaborated ProbCache and LCD give the consumers better experience in terms of delay but perform worse than the CFC because of partial local utilization of cache spaces among the content delivery paths.

8. Novel research directions in ICN caching

In recent ICN developments, newer models have incorporated measures that have been thus far ignored. We hereby highlight some of these design factors, and how they impact ICN caching. There are many research challenges in ICN caching, mostly oriented at catering for variation in content demand, measures for realistically tracking popularity, and improving caching schemes in an ICN-oriented design; transcending traditional caching systems which have more static mandates.

8.1. Heterogeneous and homogenous cache allocation

One of the key research challenges of ICN caching is the cache allocation problem. This problem addresses how cache storage should be distributed appropriately among all caching routers for a given set of finite cache storage and a network topology [150]. Most of the caching literature of content-centric networking consider homogeneous cache deployment, where all cache routers have same cache size [35,36,39,109,117]. However, a few studies address cache allocation problem and have considered heterogeneous cache allocation to allocate different cache sizes for the cache routers based on certain criteria [119,131,150,151]. Interestingly these studies often presented contradicting designs regarding homogeneous and heterogeneous cache allocation, which calls for further investigations.

Rossi and Rossini [119] take the first attempt to address cache allocation problem for CCN proposing to deploy cache capacity proportionally according to the centrality values of the cache routers such as degree, stress, betweenness, closeness and graph to heterogeneously distribute a given amount of cache to the cache routers under realistic network topologies. The authors conclude that, firstly, degree centrality, i.e. allocating cache space proportionally to the number of links of a cache router is the most robust heterogeneous allocation criterion under a large number of topologies and popularity settings and secondly, the performance gain brought by heterogeneous cache allocation is actually very limited over homogeneous cache allocation.

In contrast to this initial work, Psaras et al. [131] show that, extra cache capacity added towards the edge of the network can improve caching gain in ICN over homogeneous cache allocation. Later Wang et al. [150] investigate cache allocation extensively while proposing an optimal cache allocation algorithm aiming at traffic reduction under a total cache capacity constraint. Moreover, the authors utilize their optimal solution exploring the key factors which can impact the performance of cache allocation such as network topology, content popularity and cache replacement strategies. The authors argue that homogeneous allocation is suboptimal as it requires multiple times more cache capacity than heterogeneous allocation to achieve same traffic savings.

Arguing with the findings of [119], the authors demonstrate that, the benefits of the heterogeneous cache allocation only become apparent with larger topologies such as considering 2000 nodes, but not for the topologies considering 68 nodes or fewer considered at [119]. They emphasize that CCN should be considered as larger networks or even on Internet-scale, hence heterogeneous cache placement should be highly relevant in this regard.

In the context of core and edge cache allocation, defining the high centrality valued nodes as core and the low centrality valued nodes as edge, the authors also argue with [131] while demonstrating that, for more decentralized topologies (ISP-type networks) and highly skewed popular contents, more cache capacity should be added to the network edge and for centralized topologies (inter-AS type topologies) and less skewed popularity distribution, cache capacity should be pushed into the core as much as possible. The authors finally suggest that, different deployments require entirely different cache allocation approaches, and a one-size-fits-all approach is completely inappropriate. But their proposed optimal approach does not scale to Internet-wide deployment and provides a basis to explore the cache performance of content centric networking only and needs further research for practical deployment.

As an extension, later Wang et al. [151] propose a suboptimal heuristic method for approximate optimal cache allocation based on node centrality for practical dynamic networks with frequent content publishing. The proposed heuristic is based on the observation that, the top centrality nodes are good candidates for content caching and independent of the content server information. The authors conclude by suggesting that, in practice, network operators should make the decision of cache allocation based on the available information of network topology and content access patterns.

8.2. Caching along with request routing to content

In ICN literature, in-network caching with request routing to cached contents has been emphasized in several schemes because of the close correlation between content caching and routing to the cache content [112,113,117,121,142,143,145].

Content discovery or request to cache routing in ICN can follow one of the two approaches. One of the approach is opportunistic on-path routing where the requested content is queried on-path as the content request travels from the requester to the content source. The other approach is the coordinated off-path routing, where requests are forwarded off the shortest path to some designated cache node that is likely to cache the requested content [152]. Routing in ICN relies on ICN-specific design factors such as data in-network caching, node mobility, delay constraints and distortion tolerance for providing the most appropriate Rendezvous Points (RPs) from which consumers can pick up their requested data [153].

The collaborative caching and request forwarding scheme for CCN in [37] is content popularity guided and distributed. A new component, named AIB is introduced in each cache router to deal with content caching and forwarding content requests. The scheme named TECC [143] also proposes a traffic-engineering guided collaborative caching scheme to jointly and consistently provisioning content caching and request routing capabilities to be benefited from of these for content centric networks.

The ICN routing protocol of [114] ensures better network-wide cache space utilization by maintaining controlled assignment of contents within ASes. The caching and request routing work in two hierarchical ways using hash function where the hash function is based on content ID. In the elementary level, the ASes modify their internal routing to increase the internal cache efficiency and in the second level, the AS-path is selected to find the requested content in the network. The routing domain can have a set of interconnected ASes where the ASes propagate their interests regarding content caching. The more ASes express their interest and cooperate for content caching and request routing, the better cache space utilization is achievable.

The HCC scheme [112] has proposed a cluster-based routing protocol to efficiently find cached and non-cached contents and deliver them over the edge cluster in the two layers hierarchical clustered network. The elected cluster head router receives the content request through the access router and then checks the popularity class of the requested content. If the popularity class of the requested content maps with more important or less important cache routers (importance is based on centrality value), the cluster head then floods the interest to these routers in its cluster accordingly while implementing flooding strategy in a limited range. Each router receiving the request checks its CS and responds to the request directly if the requested content is cached. If the content is not cached, the content request is forwarded to the content provider directly according to the shortest path routing algorithm.

Li et al. have used a hash-based scheme for collaborative content caching and a directory-based scheme for content request forwarding for CCN in [142]. Here two new tables are integrated in a content router named Cooperative Router Table (CRT) and Cooperative Content Store (CCS) for request routing to the corresponding cache contents. Hash function is also used in [145] by the cache routers to select cache content and by edge routers for contents placement to the relevant cache nodes and request routing to the desired cache nodes. Another on-path, implicit collaborative caching scheme CPHR [10] uses hash-based scheme to combine content caching and routing into a joint mechanism to propose an efficient caching scheme. For caching purpose, content space is partitioned and mapped into a hash space and the caches are assigned to the partitions according to their corresponding hash values. A heuristic algorithm is proposed to assign caches to the partitions. Hash-based routing forwards the content request along the content delivery path to the assigned cache router before traveling to the content provider and delivers the requested content back to the content requester while caching the content in the assigned cache router.

Breadcrumb [120] is a best-effort content caching policy for searching requested contents among the cache nodes along content delivery path. Collaborating content routers store content caching history by storing a minimum amount of per-file information in their cache spaces to forward content request queries. Another collaborated content caching location and searching scheme named CLS [121] stores the content caching history while caching new content and cache evictions by building a caching trail along the content delivery path to direct the content searching policy in future.

SCAN [138] is a scalable content routing scheme where every router maintains a local content table (LCT) and a content routing table (CRT) for scanning to locate nearby cached copies of the requested contents. An ICN architecture, named CATT [118] uses Potential Based Routing (PBR) scheme for locating or routing requested cache contents along content delivery paths. CATT infers how many copies of a content need to be distributed in order to achieve a certain routing performance.

The cache-aware routing scheme in [146] utilizes hash-based routing techniques and requires edge- domain routers and cache nodes to implement a hash function to map content identifiers to cache nodes. The cache nodes use the hash functions to identify the set or range of the content identifiers or names that they are responsible for and the edge routers use the hash functions to route requests to the corresponding cache nodes. Hence, the edge routers can forward content requests to the designated cache directly without performing any lookup. Additionally, as the hash function can be computed in a distributed manner by edge routers and caches, the intra-domain forwarding procedure does not require any sort of inter-cache co-ordination.

To decrease the delivery latency of requested contents and increase the cache hit rate while exploring the caching capabilities of nearby cache routers, Ascigil et al. have enhanced the NDN architecture [27] by proposing an opportunistic off-path content discovery mechanism in ICN [152] that introduces a new component called Downstream Forwarding Information Table (D-FIB) in the cache router to track the directions of content packets that head towards the users.

The newly introduced D-FIB is combined with scoped interest forwarding techniques and reduces the delivery latency by fetching contents from cache routers located closer to the users that can significantly improve cache hit rate performance while maintaining the traffic overhead at reasonable levels. The functionalities of the remaining NDN router components such as CS, the Pending Interest Table (PIT) and the FIB remain the same. A content Interest packet matching with a router's D-FIB entries can be forwarded to both upstream towards the content source following the FIB entries and also to the downstream following the D-FIB entries towards the direction of users who successfully retrieved similar content interest in the past. A total forwarding counter (TFC) state in the packets limits the number of replicas of a request sent to the network.

The opportunistic off-path content discovery mechanism in [152] can perform the best assuming the exhaustive LCE [27] caching strategy although the LCE can minimize the achieved gain of the opportunistic mechanism. Hence, Ascigil et al. have proposed a more sophisticated content discovery mechanism in [154] that coordinates forwarding and caching opportunistically while keeping track of the trails towards the caches where the contents are cached. The proposed mechanism allows the usage of better cache placement schemes than LCE such as particularly assuming that each Data packet is cached exactly once along the delivery path eventually minimizing the inter-AS traffic without incurring unacceptable latency or overhead. In order to augment NDN architecture [27] with a content discovery mechanism, an Ephemeral FIB (EFIB) table is added to the original NDN content router design that utilizes a very small portion of the content store and maps name prefixes to a set of next hops, similar to the FIB table. The EFIB table is a temporary storage, triggered by a returning Data packet and forwards the information required to forward the content interests towards the direction in which content chunks were temporarily cached in the recent past among the local intra-AS cache routers. A Total Forwarding Budget (TFB) counter is introduced at the Interest packet to control the total number of initiated interests for content discovery.

In order to reveal the role of the Network Coding in information disseminating in ICN, an architecture named Network Coding for CCN (NC3N) is proposed in [155] for efficient routing to the cache content. The proposal adds some additional metadata in each content chunk of CCN along with the existing metadata such as one extra field into the Interest packet and the Data packet semantics. A flag is suggested to be inserted in the Selector field in the header of the Interest packet allowing the transmission of network coded chunks in response to the Interest carrying this flag. This additional flag can be set for instances where multiple Data packets can be received in response to the interest. A Data packet issued in response to such an Interest should have a modified Signed Info header field where the coefficients of the linear combinations of the content chunk will be carried and the Data itself would carry the encoded content. These integers are chosen randomly from a large set to avoid generating linearly dependent combinations at different content routers. Hence, the receiver can request new combinations and receives independent content chunk with high probability. In NC3N, the sender of the Interest packet sets the flag in the Interest if it supports network coding. Upon forwarding a response to the Interest, each content router looks up the number of coefficients and generates an encoded version while forwarding a response for the Interest. The requesting receiver of the content gets new degrees of freedom with each chunk it receives as every response to an Interest generates a new encoded version. NC3N can bring benefits in bandwidth reduction and delay comparing to CCN [27] with some added overhead for the additional bits in the content chunk.

The cache management framework in [126] is based on software-defined networking (SDN), where a controller is responsible for determining the optimal caching strategy and content routing through linear network coding (LNC). An optimization problem of minimizing network bandwidth cost is formulated by jointly considering caching strategy and content routing with LNC under the proposed cache management framework and a network coding based cache management (NCCM) algorithm is proposed to solve the problem aiming to obtain a near-optimal caching and routing solution for ICN. In the proposed cache management framework, at each content router, the content request is classified into two types, popular and non-popular content. The SDN controller finds the optimal route for each popular content at each content router and configures flow tables in the content routers on the route accordingly. The flow entry for a content has a list of outgoing interfaces, each of which leading to a content router caching the coded data chunks of the requested content. The controller or the edge content routers route the content requests to the original servers for requests of the unpopular contents.

CodingCache [156] is a multipath-ware CCN cache strategy which utilizes network coding and random forwarding to improve caching efficiency under multipath forwarding. The main idea of CodingCache is to achieve diverse cached contents in the network using Random Linear Network Coding (RLNC) [157] and recoding. Instead of caching every single content chunk, CodingCache caches a coded block of several content chunks. Content chunks are grouped into segments where the number of chunks in a segment is a design parameter. The content server codes the content chunks of a segment using RLNC. The coefficient vectors of two blocks corresponding to the same segment are likely to be linearly independent because of the randomization of RLNC. Cache routers recode the linearly independent coded blocks constituted by the same original chunks into a new coded block to achieve diverse cached contents in the network and the consumers decode the coded blocks to get the original requested content chunks. To maximize average cache efficiency, CodingCache exploits a random forwarding strategy to uniformly select forwarding interfaces. As CodingCache is orthogonal to existing caching strategies focusing on cache decision and replacement, any caching scheme such as LCD [137] can be incorporated with it for gaining better caching performance.

9. Towards guided designs in ICN caching

One of the main goals of this paper is guiding early researchers in ICN caching; both in understanding the spectrum of caching paradigms, as well as pointing to specific design choices that will aid targeted caching schemes. That is, realizing that there is no silver bullet, each caching scheme must identify a set of performance goals, and accordingly employ design choices – in terms of specific functional components that will be addressed – to reach these goals.

To present these design combinations, we depict in Fig. 15 a guided design plan, which builds on the taxonomy in Fig. 3. We detail in the remainder of this section five major design goals (DG), and the potential gain in choosing a specific combination of operational design factors.

These design choices are also annotated on Fig. 15, to offer a clear comparison between design choices pertinent to different performance goals. Each design goal is given an acronym below, and annotated respectively on the taxonomy in Fig. 15.

It is important to note that this is not a comprehensive list for caching performance metrics or design goals, but rather a number of umbrella goals which will allow further fine tuning under each direction.

9.1. DG1: access latency for general content

One of the most pressing mandates of ICN infrastructures is expediting content dissemination and retrieval. Hence, many caching schemes were designed to bring more popular content closer to potential requesters (at the network edge) to improve average access delay. In doing so, it is beneficial to consider the popularity of specific chunks in any given file, since demand patterns may often vary across the chunks.

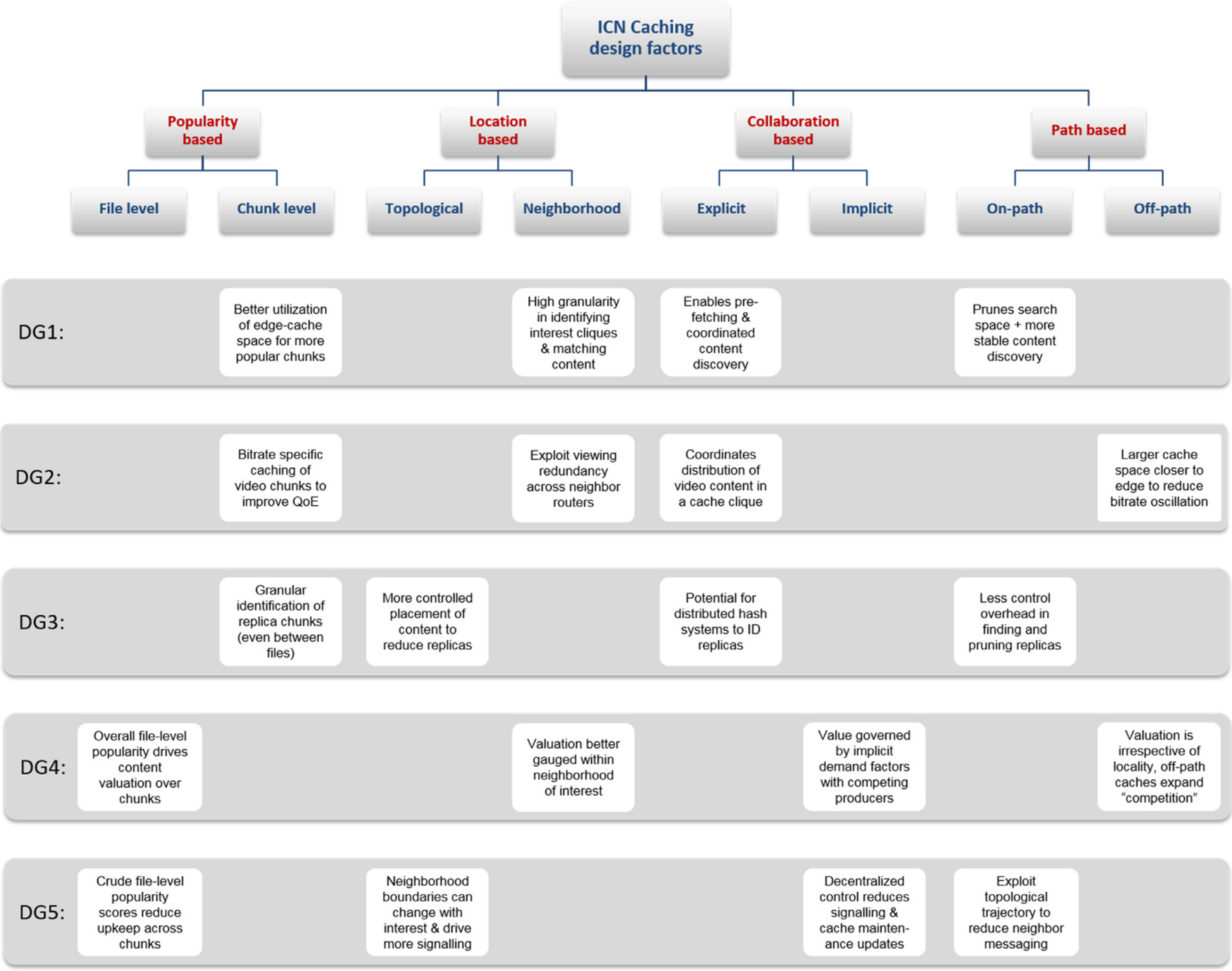
Similarly, it is easier to gauge content demand and popularity by abstracting neighborhoods of nodes into interest groups, thereby exploiting similar interests in expediting access to common content. Explicit collaboration is thus imperative, as exchanging meta-data about content and projected interest can expedite pre-fetching and access latency. Moreover, exploiting on-path caching reduces access time for files, as they are on the original trajectory towards content providers, and reduces search time in neighboring nodes.

9.2. DG2: QoE for video content

Many metrics govern QoE for video, prime among them are perceived throughput and video quality fluctuation due to variations in requested bitrates (i.e. bitrate oscillation), often as a result of bandwidth variations. To leverage QoE over ICN, caching schemes need to exploit chunk level popularity, as it considerably varies over the duration of a video (mostly a decaying curve), which often resembles a Geometric distribution [115]. In addition, neighborhood-based caching will improve video chunk availability among ICN peers, allowing faster access to popular videos. It is imperative to couple this design with explicit collaboration to maintain video availability, and leverage coordination between caching nodes to smoothen viewing by ensuring contiguous video chunks are pre-fetched at neighboring caches. Finally, off-path caching will expand the available cache capacity to store more (popular) video content closer to the edge to reduce bandwidth load and improve QoE.

9.3. DG3: reducing cache replicas

This is intrinsically a difficult problem, often due to the naming convention adopted in the ICN architecture, which might not



	Popularity based		Location based		Collaboration based		Path based	
	File level	Chunk level	Topological	Neighborhood	Explicit	Implicit	On-path	Off-path
DG1:		Better utilization of edge-cache space for more popular chunks		High granularity in identifying interest cliques & matching content	Enables pre-fetching & coordinated content discovery		Prunes search space + more stable content discovery	
DG2:		Bitrate specific caching of video chunks to improve QoE		Exploit viewing redundancy across neighbor routers	Coordinates distribution of video content in a cache clique			Larger cache space closer to edge to reduce bitrate oscillation
DG3:		Granular identification of replica chunks (even between files)	More controlled placement of content to reduce replicas		Potential for distributed hash systems to ID replicas		Less control overhead in finding and pruning replicas	
DG4:	Overall file-level popularity drives content valuation over chunks			Valuation better gauged within neighborhood of interest		Value governed by implicit demand factors with competing producers		Valuation is irrespective of locality, off-path caches expand "competition"
DG5:	Crude file-level popularity scores reduce upkeep across chunks		Neighborhood boundaries can change with interest & drive more signalling			Decentralized control reduces signalling & cache mainten-ance updates	Exploit topological trajectory to reduce neighbor messaging	

Fig. 15. Guiding the design of future caching schemes. Each design goal (DG), explained in Section 9, is addressed with specific design considerations.

uniquely assign names to content (e.g., in attribute based naming). However, considering more granular chunks of files can elicit redundancies, even across files with partial overlapping content (e.g., documents with media files). Popularity, as an indicator for relative demand for content, might even elicit attributes of uniqueness across files (e.g., a certain media file that is often embedded). Coupling this strategy with a topological view of caching nodes, will enable faster identification of replicas, as we transcend the restrictive view of in-neighborhood replicas. More advanced systems for collaborative caching could be developed to cross-check content for redundancy, and identify replicas that are superfluous in the network setting. However, to reduce the search space and potential intractability of identifying and managing replicas across the network, off-path caching should be sacrificed.

9.4. DG4: enabling cache valuation

Future ICN will be able to determine the value of content, and introduce new caching methods that are valuation-driven. To capture content demand, it is important to gauge overall file demand, and infer therein the projected popularity of its chunks, to adopt a more tractable approach. This is coupled with neighborhood-based caching schemes that exploit locality of interest and demand, rather than topological associations between caching nodes (and their serviced users on the edge).

While cooperative caching schemes can adapt to content valuation schemes, avoiding collusion drives an implicit model whereby caches (especially from competing providers or ISPs) can draw decentralized computation of value indicators, thereby better opening up a market for cache auctioning and aggregated cache valuation. Finally, opening up the competition scope dictates expanding to off-path caching reservoirs, whereby neighboring nodes can also cache and drive demand for different content, as well as increase supply of cache space to reduce the cost of caching.

9.5. DG5: reducing control overhead

Since scalability is a major concern for the future Internet, it is important to emphasize design factors in ICN that will leverage it. First, adopting chunk level popularity introduces significant overhead in computing and updating scores, especially in video content; this outweighs the potential gain in access latency if we only cache more popular chunks (closer). Also, maintaining neighborhood information and virtual cliques of caching node incurs significant overhead, especially as network sizes scale. Moreover, neighborhood boundaries change with interests and potentially with new users joining at the network edge, thereby inducing more control overhead in identifying and maintaining neighborhood boundaries and recalculating measures of centrality and betweenness. Also, adopting decentralized caching decisions reduces coordination overhead, and allows for low-overhead cache eviction and replacement schemes, cutting down the messaging and control overhead of explicitly coordinated caching schemes. Finally, it is more prudent to adopt a topological-based approach that capitalizes on the inherent trajectories of request forwarding towards content providers, exploiting on-path only caching to reduce overhead of content retrieval, larger network traversal, and leaving trails for backtracking to original requester.

10. Insights for progressive ICN caching development

10.1. The elephant in the room: is the ICN paradigm needed?

ICN is often presented as a clean slate technology, aiming to replace the current Internet infrastructure. Understandably, some researchers skeptically argue against the proposed gains [107], noting how the Internet has managed to scale well beyond its original operational mandates, and how current overlay paradigms (e.g., P2P, CDN, and IP multicast) are presenting practical and realizable remedies [105] in scaling with the undeniable projected growth. While such optimistic arguments hold merit [2], a new trend of minimalistic introduction of ICN at the edge, is gaining momentum as a likely middle ground [107].

It is important to investigate the underlying arguments and challenges in supporting such assertions. First, major reviews of ICN performance have considered a rather simplistic view of caching schemes (e.g., LRU based caching in [107]) and thereby generalized their findings on a relatively novel research direction. Moreover, many studies about Internet user behaviour and content request patterns are based on the current Internet architecture [158], where locations of surrogate servers and their access latencies play a major impact on user request patterns. In one dimension alone, Gao et al. attempt to quantitatively compare network reaction and content dissemination as mobile users request content [159]. Furthermore, Kurose argued in [160] that the Independent Reference Model (IRM), long used for studying caches (memory, web, CDNs, and even ICN) no longer fits the ICN caching paradigm.

The study of data request patterns, and their "arrivals" at edge nodes in ICN is a challenging problem that requires significant investigation. Even if we can assume that request arrive according to a Poisson model, or any memoryless process, this assumption no longer holds as requests are aggregated or filtered as they traverse up routing trees.

10.2. Cache valuation

Current caching schemes are re-cycling archaic measures of content value, mostly to guide cache eviction and replacement strategies. These started off with simple measures of content size, frequency of requests, and time stamps on requests. By far, LRU has been mostly adopted for the low-overhead of implementation using a simple queue. More recent models adopt a Zipf-like distribution to predict content popularity over time (named after George K. Zipf's work in 1929), based on its earlier adoption in modeling Web caching [135], after varying its skewness factor (α). However, its applicability in modelling popularity in ICN requires further experimentation and rigorous evaluation.

However, all of the above measures are user centric. That is, the publisher has little control on keeping their content "alive" in the network to improve its access latency and thereby potential spread. As more stakeholders take part in ICN, the model of cache replacement will need to adapt to enable a producer-aware paradigm. That is, the producer might be able to pay certain caching nodes to keep their "premium" content in their caches to facilitate faster access latencies. Moreover, caching routers will be able to publicize their measures of "connectivity" and "betweenness" to gauge their pricing models and attract better (or more profitable) content. Our earlier work in [132] addressed a primer direction for establishing value for content, but much needs exploration in terms of heterogeneously assessing the value (and underlying demand) for content, both in real time and over rounds of requests.

10.3. Granularity of popularity indices

In most caching schemes that are popularity based, the underlying assumption is that chunks of a given file share equal popularity. While this might hold for a fragmented text file or other static content that is interesting in its entirety, this assumption is less realistic when considering video files. With projections of video traffic dominating future internet infrastructures, our interpretation and modelling of popularity must resemble more realistic assumptions. For one, many users would watch the beginning of a video, and then decide if they want to proceed with the rest. In retrieving news or game replays, specific timelines (e.g. time of a goal) are more popular than the remainder of the video.

A clear distinction between chunk-level popularity and file-level popularity must be studied and modeled rigorously. At the heart of every cache eviction and replacement scheme is a measure of popularity that must reflect that of the specific chunk in question, not simply its entire file. In our earlier work [115], a model adopted chunk-level popularity is presented, yet much work remains in generalizing such models over popularity caching schemes. Furthermore, it is important to distinguish user-inferred popularity (based on requests received for a given content) and network dynamics that might encourage producers to artificially request their content to revive its replicas in ICN caches.

10.4. Pre-emptive popularity-based dissemination

Another dimension of gauging popularity can be classified under popularity prediction. That is, if a given content is known to typically generate much interest, then a reactive ICN infrastructure will attempt to cache that content more rapidly and increase its dissemination rate. A typical example arises from following the trends and "following" indices in social media. If a known celebrity has millions of followers, chances are whatever media they post (e.g., via Snapchat or Twitter) is known to yield significant traffic for accessing it. Future mechanisms for predicting popularity trends can thereby react to this by pushing replicas or geographically distributing content to interest regions.

This paradigm, if presented in a scalable framework, will enable the next generation of multicasting for live video streaming applications, especially for major sporting and entertainment events. The scale of growing demand, and the rising expectations for access latencies, mandate novel approaches in predicting popularity and pre-emptively tuning caching nodes to improving overall QoE for users.

10.5. Inter-ISP content management

There are multiple visions to how ICN will be deployed, ranging from the clean-slate replacement of the Internet, to the sheer adoption of overlay schemes on a subset of routers (mostly close to the edge). Having that said, many protocols neglect content management across ISPs or ICN silos. That is, many paradigms assume that each ICN model follows an arbitrary tree structure [38,39,109,120,121], with the main content provider positioned as the root. In that view, it is difficult to manage and mitigate content redundancy, adaptive interest forwarding schemes and potentially utilizing caching nodes from co-existing ISPs. This presents a core challenge, and must instigate a more encompassing approach of arbitrary graphs interconnected with potentially conflicting operational mandates. While some overhead is expected in the management of multiple ISP-based ICN, it is important to study the benefits of assessing content variation, redundancy, utility and popularity across ICN-based ISPs.

10.6. Addressing latency in ICN content retrieval

Caching protocols in ICN are prone to frequent changes, due to placement (e.g., most popular) and replacement (e.g., LRU & LFU) schemes, in addition to fluctuations in popularity and other factors. Hence, when an ICN paradigm adopts a Name Resolution System (described in Section 2.B.2), the initial phase of name resolution and translation to potential producers (or caching nodes) would hinder access latency and potentially burden the network with a stringent mandate for frequent updates. Chiocchetti et al. have presented in [161] a hybrid approach to forwarding requests that attempt to remedy challenges with "extremes" in ICN content retrieval; instead of either exclusively flooding requests to "explore" the network or "exploit" a strict path to a location known to have held a valid copy. In remedy, caching schemes in NRS-based ICN infrastructures should incorporate topological meta-data to prune search and updates based on the locality of requests. Moreover, cache replacement schemes must incorporate measures of outweighing cache eviction in light of the overhead of updating the NRS.

10.7. The challenge in using attribute-value pairs

In ICN infrastructures, attribute-based naming does not always provide a unique identification for every cached content (described in Section 2.B.1) [52]. To identify content, attribute-value pairs (AVP) are assigned to content. Instead of requesting and identifying content by providing an explicit name, contents are requested applying sets of logical constraints (predicates) over the AVP; which may match multiple contents. As the attribute-based naming scheme does not guarantee uniqueness for content names, a major challenge will arise in reducing content replicas across the network. Moreover, a major challenge manifests in cache content valuation and eviction policies if the demand for given content is not clearly gauged. That is, since there is no mandate for uniqueness of names, we cannot adopt caching schemes that rely on the premise of a unique NDO in cache replacement strategies. Hence, it is important to incorporate measures for cache valuation for content only identifiable via AVP predicates.

11. Conclusions

The drive for adapting the future Internet to scale and grow with projected demands, has led many efforts in the past decade to evolve its operational mandates. More recently, researchers in ICN led significant efforts to re-design a clean-slate technology to replace the Internet. As we overview the major challenges in adapting the status quo and realizing an efficient dissemination of ICN, more efforts are converging to a middle-ground based approach; one that considers rolling-in the ICN as an evolutionary phase in the Internet's progression. At the heart of ICN architectures, caching content plays a pivotal role in efficient dissemination and retrieval of content across the heterogeneity of evolving users and networks.

We presented a taxonomy to encompass the status quo in ICN caching paradigms, focusing on a comparative approach to enlist the major developments in caching schemes, and elaborate on the evolution of ICN caching schemes and their operational mandates. In hindsight, most ICN caching schemes in the status quo are focusing on content popularity as the major factor in cache decision policies. We argue for encompassing more attributes, along with content popularity while designing the caching policies for attaining better caching gain. Location-based caching schemes such as the centrality-based schemes work well by selectively choosing cache nodes in the network rather than by indiscriminate selection, yet can suffer from the load balancing problem. Off-path caching schemes typically perform better than on-path schemes because of better utilization of the caching resources yet require more co-ordination and communication overhead than the on-path to direct the content requests to the desired cache locations.

Our comprehensive quantitative and qualitative performance assessments of ICN caching schemes, and the correlation between both analyses, presents unique insights into ICN caching primitives and their impact on performance. For example, we demonstrate how non-cooperative caching schemes typically yield unnecessary cache redundancy. On the other hand, among non-cooperative schemes, probabilistic caching schemes shine in performance if the random caching probability value is tuned to minimize cache replacement errors. Location-based scheme can perform better than exhaustive indiscriminate scheme but depends on network topology. Off path, explicit collaborated schemes achieve better caching performance than the on-path implicit collaborated schemes by producing more cache hits and requiring less latency to retrieve requested contents because of efficient collaborative usage of caching resources. Caching scheme that caches contents near requesting consumers can perform well for high popularity skewness as the majority content requests concentrate on a smaller set of popular contents for high popularity skewness value.

While many tradeoffs in ICN caching are inevitable, we elaborated on a number of future directions with significant promise, and presented a detailed guide for developing novel caching schemes, underlining the challenging conflicts in design for the future ICN research community.

Funding

This research is supported by grants from the Natural Sciences and Engineering Research Council of Canada (NSERC) under grant numbers: RGPIN-2017-06902 and RGPIN-2019-05667.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] CISCO, Cisco Visual Networking index: Forecast and Methodology, 2017-2022, CISCO Whitepapers, February 27, 2019.
[2] M.E.M. Campista, M.G. Rubinstein, I.M. Moraes, L.H.M.K. Costa, O.C.M.B. Duarte, Challenges and research directions for the future internetworking, IEEE Commun. Surv. Tutor. 16 (February) (2014) 1050–1079.
[3] J. Pan, S. Paul, R. Jain, A survey of the research on future internet architecture, IEEE Commun. Mag. 49 (July) (2011) 26–36.
[4] A. Marnerides, D. Pezaros, D. Hutchison, Internet traffic characterisation: third-order statistics & higher-order spectra for precise traffic modelling, Comput. Netw. (April) (2018) 183–201.
[5] G. Papastergiou, G. Fairhurst, D. Ros, A. Brunstrom, K.-J. Grinnemo, P. Hurtig, N. Khademi, M. Tüxen, M. Welzl, D. Damjanovic, S. Mangiante, De-Ossifying the internet transport layer: a survey and future perspectives, IEEE Commun. Surv. Tutor. 19 (1) (2017) 619–639.
[6] D. Trossen, A. Sathiaseelan, J. Ott, Towards an information centric network architecture for universal internet access, ACM SIGCOMM Comput. Commun. Rev. 46 (January) (2016) 44–49.
[7] X. George, C.N. Ververidis, V. Siris, N. Fotiou, C. Tsilopoulos, X. Vasilakos, K.V. Katsaros, G.C. Polyzos, A survey of information-centric networking research, IEEE Commun. Surv. Tutor. 16 (July) (2014) 1024–1049.
[8] C. Fang, F.R. Yu, T. Huang, J. Liu, Y. Liu, A survey of green information-centric networking: research issues and challenges, IEEE Commun. Surv. Tutor. 17 (February) (2015) 1455–1472.

[9] F. Almeida, J. Lourenço, Information centric networks- design Issues, principles and approaches, Int. J. Latest Trend. Comput. (IJLTC) 3 (September) (2012) 58–66.

[10] S. Wang, J. Bi, J. Wu, CPHR: in-Network caching for information-centric networking with partitioning and hash-routing, IEEE/ACM Trans. Netw. 24 (October) (2016) 2742–2755.

[11] A. Ioannou, S. Weber, A taxonomy of caching approaches in information-centric network architectures, Elsev. J. (March) (2013).

[12] G. Zhang, Y. Li, T. Lin, Caching in information centric networking: a survey, Comput. Netw. 57 (November) (2013) 3128–3141.

[13] M. Zhang, H. Luo, H. Zhang, A survey of caching mechanisms in information–centric networking, IEEE Commun. Surv. Tutor. 17 (August) (2015) 1473–1499.

[14] I. Abdullahi, S. Arif, S. Hassan, Survey on caching approaches in information centric networking, J. Netw. Comput. Appl. 56 (October) (2015) 48–59.

[15] A. Ioannou, S. Weber, A survey of caching policies and forwarding mechanics in information-centric networking, IEEE Commun. Surv. Tutor. 18 (May) (2016) 2847–2886.

[16] C. Bernardini, T. Silverston, A. Vasilakos, Caching strategies for information centric networking: opportunities and challenges, CoRR (June) (2016) 54–61.

[17] I. Ud Din, S. Hassan, M.K. Khan, M. Guizani, O. Ghazali, A. Habbal, Caching in information-centric networking: strategies, challenges, and future research directions, IEEE Commun. Surv. Tutor. 20 (May) (2018) 1443–1474.

[18] D.R. Cheriton and M. Gritter, TRIAD: a Scalable Deployable NATbased Internet architecture, Stanford University, Technical Report, June 2000. [Online]. Available: https://stanford.edu/triad/triad.ps.gz (accessed 16 October 2019).

[19] A. Carzaniga, A.L. Wolf, Content-based networking: a new communication infrastructure, in: NSF Workshop on an Infrastructure for Mobile and Wireless Systems, Springer, 2001, pp. 59–68.

[20] A. Carzaniga, A.L. Wolf, Forwarding in a content-based network, in: Proceedings of the 2003 conference on Applications, technologies, architectures, and protocols for computer communications, Karlsruhe, Germany, 2003, pp. 163–174. August.

[21] A. Carzaniga, M.J. Rutherford, A.L. Wolf, A routing scheme for content-based networking, in: IEEE International Conference on Computer Communications (INFOCOM), 2004, pp. 918–928. March.

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

[23] FP7 PURSUIT project. [Online]. Available: https://cordis.europa.eu/project/rcn/95665/factsheet/en (accessed 16 October 2019).

[24] D. Lagutin, K. Visala, S. Tarkoma, Publish/subscribe for internet: psirp perspective, in: Towards the Future Internet Emerging Trends from European Research, 4, Amsterdam, The Netherlands: IOS Press, 2010, pp. 75–84.

[25] C. Dannewitz, D. Kutscher, B. Ohlman, S. Farrell, B. Ahlgren, H. Karl, Network of information (NetInf) - An information-centric networking architecture, Comput Commun 36 (April) (2013) 721–735.

[26] NSF Named Data Networking project. [Online]. Available: http://www.named-data.net/ (accessed 16 October 2019).

[27] V. Jacobson, D.K. Smetters, J.D. Thornton, M.F. Plass, N.H. Briggs, R.L. Braynard, Networking named content, in: Proceedings of the 5th ACM International Conference on Emerging Networking Experiments and Technologies (CoNEXT 2009), Rome, Italy, 2009, pp. 1–12. December.

[28] CCNx project. [Online]. Available: https://wiki.fd.io/view/Cicn (accessed 16 October 2019).

[29] FP7 SAIL project. [Online]. Available: https://sail-project.eu/ (accessed 16 October 2019).

[30] FP7 4WARD project. [Online]. Available: https://cordis.europa.eu/project/rcn/85316/factsheet/en (accessed 16 October 2019).

[31] FP7 COMET project. [Online]. Available: http://www.comet-project.org/ (accessed 16 October 2019).

[32] FP7 CONVERGENCE project. [Online]. Available: http://www.ict-convergence.eu/ (accessed 16 October 2019).

[33] NSF Mobility First project. [Online]. Available: http://mobilityfirst.winlab.rutgers.edu/ (accessed 16 October 2019).

[34] J. Li, H. Wu, B. Liu, J. Lu, Y. Wang, X. Wang, Y. Zhang, L. Dong, Popularity–driven coordinated caching in named data networking, in: Proceedings of the 8th ACM/IEEE symposium on Architectures for networking and communications systems (ANCS), Austin, Texas, USA, 2012, pp. 15–26. October.

[35] D. Rossi, G. Rossini, Caching Performance of Content Centric Networks Under Multi-Path Routing (and more), Relatório técnico, Telecom ParisTech, 2011 November.

[36] K. Suksomboon, S. Tarnoi, Y. Ji, M. Koibuchi, K. Fukuda, S. Abe, N. Motonori, M. Aoki, S. Urushidani, S. Yamada, PopCache: cache more or less based on content popularity for information centric networking, in: IEEE Conference on Local Computer Networks (LCN), Sydney, 2013, pp. 236–243. October.

[37] S. Guo, H. Xie, G. Shi, Collaborative forwarding and caching in content centric networks, in: Proceedings of the 11th International IFIP TC 6 Networking Conference, Prague, Czech Republic, 2012, pp. 41–55. May.

[38] K. Cho, M. Lee, K. Park, T.T. Kwon, Y. Choi, S. Pack, Wave: popularity-based and collaborative in-network caching for content-oriented networks, in: IEEE International Conference on Computer Communications Workshops(INFOCOM WKSHPS), Orlando, FL, 2012, pp. 316–321. March.

[39] Z. Ming, M. Xu, D. Wang, Age-based cooperative caching in information-centric networks, in: Workshop on Emerging Design Choices in Name-Oriented Networking, Shanghai, 2012, pp. 1–8. August.

[40] X. Vasilakos, V.A. Siris, G.C. Polyzos, M. Pomonis, Proactive selective neighbor caching for enhancing mobility support in information-centric networks, in: Proceedings of the second edition of the ICN workshop on Information-centric networking, Helsinki, Finland, 2012, pp. 61–66. August.

[41] G. Xylomenos, X. Vasilakos, C. Tsilopoulos, V.A. Siris, G.C. Polyzos, Caching and mobility support in a publish-subscribe internet architecture, IEEE Commun. Mag. 50 (July) (2012) 52–58.

[42] K. Katsaros, G. Xylomenos, G.C. Polyzos, MultiCache: an incrementally deployable overlay architecture for information-centric networking, in: IEEE Conference on Computer Communications (INFOCOM) Workshops, San Diego, CA, USA, 2010, pp. 1–5. March.

[43] K.V. Katsaros, G. Xylomenos, G.C. Polyzos, GlobeTraff: a traffic workload generator for the performance evaluation of future internet architectures, in: 5th International Conference on New Technologies, Mobility and Security (NTMS), Istanbul, Turkey, 2012, pp. 1–5. May.

[44] D. Raychaudhuri, K. Nagaraja, A. Venkataramani, MobilityFirst: a robust and trustworthy mobility-centric architecture for the future internet, ACM SIGMOBILE Mob. Comput. Commun. Rev. 16 (July) (2012) 2–13.

[45] A. Baid, T. Vu, D. Raychaudhuri, Comparing alternative approaches for networking of named objects in the future internet, in: IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS), Orlando, FL, USA, 2012, pp. 298–303. March.

[46] F. Zhang, C. Xu, Y. Zhang, S. Mukherjee, K.K. Ramakrishnan, R. Yates, T. Nguyen, EdgeBuffer: caching and prefetching content at the edge in the mobilityfirst future internet architecture, in: IEEE 16th International Symposium on A World of Wireless, Mobile and Multimedia Networks (WoWMoM), Boston, MA, USA, 2015, pp. 1–9. June.

[47] K.V. Katsaros, N. Fotiou, X. Vasilakos, C.N. Ververidis, C. Tsilopoulos, G. Xylomenos, G.C. Polyzos, On inter-domain name resolution for information-centric networks, in: International Conference on Research in Networking, Berlin, Heidelberg, 2012, pp. 13–26. May.

[48] L. Muscariello, G. Carofiglio, M. Gallo, Bandwidth and storage sharing performance in information centric networking, in: Proceedings of the ACM SIGCOMM workshop on Information-centric networking (ICN '11), Toronto, ON, 2011, pp. 26–31. August.

[49] W.K. Chai, N. Wang, I. Psaras, G. Pavlou, C. Wang, G.G. de Blas, F.J. Ramon-Salguero, L. Liang, S. Spirou, A. Beben, E. Hadjioannou, Curling: content-ubiquitous resolution and delivery infrastructure for next-generation services, IEEE Commun. Mag. 49 (3 March) (2011) 112–120.

[50] N. Blefari Melazzi, A. Detti, M. Pomposini, S. Salsano, Route discovery and caching: a way to improve the scalability of information-centric networking, in: IEEE Global Communications Conference (GLOBECOM), Anaheim, CA, USA, 2012, pp. 2701–2707. December.

[51] B. Ahlgren, C. Dannewitz, C. Imbrenda, D. Kutscher, B. Ohlman, A survey of information-centric networking, IEEE Commun. Mag. 50 (July) (2012) 26–36.

[52] G.M. de Brito, P.B. Velloso, I.M. Moraes, Information centric networks: A new paradigm for the Internet, ISTE Ltd and John Wiley & Sons, Inc., 2013.

[53] A. Ghodsi, S. Shenker, T. Koponen, A. Singla, B. Raghavan, J. Wilcox, Information-centric networking: seeing the forest for the trees, in: Proceedings of the 10th ACM Workshop on Hot Topics in Networks, Cambridge, MA, USA, 2011, pp. 1–6. November.

[54] D. Wendlandt, I. Avramopoulos, D.G. Andersen, J. Rexford, Don't secure routing protocols, secure data delivery, in: Proceedings of the 5th ACM SIGCOMM Workshop on Hot Topics in Networking (Hotnets-V), Irvine, CA, 2006 November.

[55] Y. Huang, H. Garcia-Molina, Publish/subscribe in a mobile environment, Wirel. Netw. 10 (November) (2004) 643–652.

[56] H. Farahat, H. Hassanein, On the design and evaluation of producer mobility management schemes in named data networks, in: Proceedings of the 18th ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems, Cancun, Mexico, 2015, pp. 171–178. November.

[57] O. Ascigil, S. Reñé, G. Xylomenos, I. Psaras, G. Pavlou, A keyword-based ICN-IoT platform, in: Proceedings of the 4th ACM Conference on Information–Centric Networking (ICN), Berlin, 2017.

[58] Y.T. Yu, M. Gerla, Information-centric VANETs: a study of content routing design alternatives, in: International Conference on Computing, Networking and Communications (ICNC), Kauai, HI, 2016.

[59] S. Vural, N. Wang, P. Navaratnam, R. Tafazolli, Caching transient data in internet content routers, IEEE/ACM Trans. Netw. 25 (April) (2017) 1048–1061.

[60] J. Quevedo, D. Corujo, R. Aguiar, A case for icn usage in iot environments, in: IEEE Global Communications Conference (GLOBECOM), 2014, pp. 2770–2775. December.

[61] W. Shang, Q. Ding, A. Marianantoni, J. Burke, L. Zhang, Securing building management systems using named data networking, IEEE Netw. 28 (May) (2014) 50–56.

[62] E. Baccelli, C. Mehlis, O. Hahm, T.C. Schmidt, M. Wählisch, Information centric networking in the iot: experiments with ndn in the wild, in: Proceedings of the 1st International Conference on ACM Information-centric Networking, Paris, France, September 24, 2014, pp. 77–86.

[63] O. Hahm, E. Baccelli, T.C. Schmidt, M. Wählisch, C. Adjih, L. Massoulié, Low-power internet of things with ndn & cooperative caching, in: Proceedings of the 4th ACM Conference on Information-Centric Networking, September 26, 2017, pp. 98–108.

[64] S. Li, Y. Zhang, D. Raychaudhuri, R. Ravindran, A comparative study of mobilityfirst and NDN based ICN-IoT architectures, in: IEEE 10th International Conference on Heterogeneous Networking for Quality, Reliability, Security and Robustness (QShine), Rhodes Island, Greece, August 18, 2014, pp. 158–163.

[65] O. Waltari, J. Kangasharju, Content-centric networking in the internet of things, in: 13th IEEE Annual Consumer Communications & Networking Conference (CCNC), Las Vegas, NV, USA, January 9, 2016, pp. 73–78.
[66] Y. Rao, H. Zhou, D. Gao, H. Luo, Y. Liu, Proactive caching for enhancing user–side mobility support in named data networking, in: IEEE 7th International Conference in Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS), Taichung, Taiwan, July 3, 2013, pp. 37–42.
[67] K. Kanai, T. Muto, J. Katto, S. Yamamura, T. Furutono, T. Saito, H. Mikami, K. Kusachi, T. Tsuda, W. Kameyama, Y.-J. Park, T. Sato, Proactive content caching for mobile video utilizing transportation systems and evaluation through field experiments, IEEE J. Sel. Areas Commun. 34 (August) (2016) 2102–2114.
[68] D. Grewe, M. Wagner, H. Frey, PeRCeIVE: proactive caching in ICN-based VANETs, in: IEEE Vehicular Networking Conference (VNC), Columbus, Ohio, USA, December 8, 2016, pp. 1–8.
[69] Y. Zeng, X. Hong, A caching strategy in mobile ad hoc named data network, in: IEEE 6th International ICST Conference on Communications and Networking in China (CHINACOM), Kunming, China, August 17, 2011, pp. 805–809.
[70] L. Zhou, T. Zhang, X. Xu, Z. Zeng, Y. Liu, Broadcasting based neighborhood cooperative caching for content centric ad hoc networks, in: IEEE/CIC International Conference on Communications in China (ICCC), Shenzhen, China, November 2, 2015, pp. 1–5.
[71] L. Zhang, J. Zhao, Z. Shi, LF: a caching strategy for named data mobile ad hoc networks, in: Proceedings of the 4th International Conference on Computer Engineering and Networks, Cham, Springer, 2015, pp. 279–290.
[72] D. Kim, J.-h. Kim, C. Moon, J. Choi, I. Yeom, Efficient content delivery in mobile ad-hoc networks using CCN, Ad Hoc Netw. 36 (January) (2016) 81–99.
[73] M. Amadeo, C. Campolo, J. Quevedo, D. Corujo, A. Molinaro, A. Iera, R.L. Aguiar, A.V. Vasilakos, Information-centric networking for the internet of things: challenges and opportunities, IEEE Netw. 30 (March) (2016) 92–100.
[74] W. Shang, A. Bannis, T. Liang, Z. Wang, Y. Yu, A. Afanasyev, J. Thompson, J. Burke, B. Zhang, L. Zhang, Named data networking of things (Invited paper), in: IEEE 1st International Conference on Internet-of-Things Design and Implementation (IoTDI), Berlin, Germany, April 4, 2016, pp. 117–128.
[75] A. Lindgren, F.B. Abdesslem, B. Ahlgren, O. Schelén, A.M. Malik, Design choices for the iot in information-centric networks, in: 13th IEEE Annual Consumer Communications & Networking Conference (CCNC), Las Vegas, NV, USA, January 9, 2016, pp. 882–888.
[76] A. Rayes, M. Morrow, D. Lake, Internet of things implications on icn, in: IEEE International Conference on Collaboration Technologies and Systems (CTS), Denver, CO, USA, May 21, 2012, pp. 27–33.
[77] Named data network internet of things toolkit (NDN-IoTT). [Online]. Available: https://github.com/remap/ndn-pi (accessed 16 October 2019).
[78] NDN Client Library for C++ and C. [Online]. Available: https://github.com/named-data/ndn-cpp (accessed 16 October 2019).
[79] W. Shang, Y. Yu, T. Liang, B. Zhang, L. Zhang, NDN-ACE: access control for constrained environments over named data networking, NDN Project, Tech. Rep. (December 21, 2015) NDN-0036, Revision 1.
[80] H.M.A. Islam, D. Lagutin, N. Fotiou, Observing iot resources over icn, in: IEEE IFIP Networking Conference (IFIP Networking) and Workshops, Stockholm, Sweden, June, 2017.
[81] N. Fotiou, G. Xylomenos, G.C. Polyzos, H. Islam, D. Lagutin, T. Hakala, E. Hakala, ICN enabling coap extensions for IP based IoT devices, in: Proceedings of the 4th ACM Conference on Information-Centric Networking, Berlin, Germany, September 26, 2017, pp. 218–219.
[82] M. Amadeo, C. Campolo, A. Molinaro, Information-centric networking for connected vehicles: a survey and future perspectives, IEEE Commun. Mag. 54 (February) (2016) 98–104.
[83] J. Wang, R. Wakikawa, L. Zhang, DMND: collecting data from mobiles using named data, in: IEEE Vehicular networking conference (VNC), Jersey City, NJ, USA, December 13, 2010, pp. 49–56.
[84] M. Amadeo, C. Campolo and A. Molinaro, CRoWN: content-centric networking in vehicular ad hoc networks, IEEE Commun. Lett., vol. 16, no. 9, pp. 1380–1383.
[85] L. Wang, R. Wakikawa, R. Kuntz, R. Vuyyuru, L. Zhang, Data naming in vehicle-to-vehicle communications, in: IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS), Orlando, Florida, USA, March 25, 2012, pp. 328–333.
[86] Z. Yan, S. Zeadally, Y.-J. Park, A novel vehicular information network architecture based on named data networking (NDN), IEEE Internet Thing J. 1 (December) (2014) 525–532.
[87] G. Grassi, D. Pesavento, L. Wang, G. Pau, R. Vuyyuru, R. Wakikawa and L. Zhang, Vehicular inter-networking via named data, in *ACM HotMobile 2013 Poster*:.
[88] G. Grassi, D. Pesavento, G. Pau, R. Vuyyuru, R. Wakikawa, L. Zhang, VANET via named data networking, in: IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS), Toronto, ON, Canada, April 27, 2014, pp. 410–415.
[89] Z. Su, Y. Hui, Q. Yang, The next generation vehicular networks: a content-centric framework, IEEE Wirel Commun 24 (February) (2017) 60–66.
[90] X. Liu, Z. Li, P. Yanga, Y. Dong, Information-centric mobile ad hoc networks and content routing: a survey, Ad Hoc Netw 58 (April) (2017) 255–268.
[91] M. Varvello, M. Schurgot, J. Esteban, L. Greenwald, Y. Guo, M. Smith, D. Stott, L. Wang, SCALE: a content-centric manet, in: IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS), Turin, Italy, April 14, 2013, pp. 29–30.
[92] S.Y. Oh, D. Lau, M. Gerla, Content centric networking in tactical and emergency MANETs, in: IEEE IFIP Wireless Days (WD), Venice, Italy, October 20, 2010.
[93] B. Etefia, L. Zhang, Named data networking for military communication systems, in: IEEE Aerospace Conference, Big Sky, MT, USA, March 3, 2012, pp. 1–7.
[94] L. You, Z. Wang, Y.-T. Yu, R. Fan, M. Gerla, Social network based security scheme in mobile information-centric network, in: IEEE 12th Annual Mediterranean Ad Hoc Networking Workshop (MED-HOC-NET), Ajaccio, France, June 24, 2013, pp. 1–7.
[95] S. Oteafy, H. Hassanein, Big sensed data: evolution, challenges, and a progressive, IEEE Commun. Mag. (2018).
[96] J. Burke, P. Gasti, N. Nathan, G. Tsudik, Securing instrumented environments over content-centric networking: the case of lighting control and ndn, in: IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS), Turin, Italy, April 14, 2013, pp. 394–398.
[97] J. Burke, P. Gasti, N. Nathan, G. Tsudik, Secure sensing over named data networking, in: IEEE 13th International Symposium on Network Computing and Applications (NCA), Cambridge, MA, USA, August 21, 2014, pp. 175–180.
[98] L. Anders, A.F. Ben, A. Bengt, S. Olov, M. Adeel, Applicability and tradeoffs of information-centric networking for efficient iot. [Online]. Available: https://datatracker.ietf.org/doc/draft-lindgren-icnrg-efficientiot/ (accessed 16 October 2019).
[99] H. Klaus, Observing resources in the constrained application protocol (CoAP), Internet Engineering Task Force (IETF), RFC 7641, September 2015. [Online]. Available: https://www.ietf.org/rfc/rfc7641.txt (accessed 16 October 2019).
[100] D. Trossen, M.J. Reed, J. Riihijärvi, M. Georgiades, N. Fotiou, G. Xylomenos, IP over ICN - The better IP? in: IEEE European Conference on Networks and Communications (EuCNC), Paris, France, June 29, 2015, pp. 413–417.
[101] Z. Shelby, A.K. Hartke, C. Bormann, The constrained application protocol (CoAP), Internet Engineering Task Force (IETF), RFC 7252, June 2014. [Online]. Available: https://tools.ietf.org/html/rfc7252 (accessed 16 October 2019).
[102] R. Akbar, E. Dijk, Group Communication for the Constrained Application Protocol (CoAP), Group, October 14, 2014 [Online]. Available: https://tools.ietf.org/rfc/rfc7390.txt.
[103] W. Wang, S. De, R. Toenjes, E. Reetz, K. Moessner, A comprehensive ontology for knowledge representation in the internet of things, in: IEEE 11th International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom), Liverpool, UK, June 25, 2012, pp. 1793–1798.
[104] G. Carofiglio, M. Gallo, L. Muscariello, D. Perino, Modeling data transfer in content-centric networking, in: Proceedings of the 23rd International Teletraffic Congress (ITC 23), San Francisco, USA, September 6, 2011, pp. 111–118.
[105] P. Andrea, A survey on content-centric technologies for the current internet: CDN and P2P solutions, Comput. Commun. 35 (January) (2012) 1–32.
[106] C. Li, L. Toni, J. Zou, H. Xiong, P. Frossard, QoE-driven mobile edge caching placement for adaptive video streaming, IEEE Trans. Multimed. 20 (April) (2017) 965–984.
[107] S.K. Fayazbakhsh, Y. Lin, A. Tootoonchian, A. Ghodsi, T. Koponen, B.M. Maggs, K.C. Ng, V. Sekar, S. Shenker, Less pain, most of the gain: incrementally deployable icn, ACM SIGCOMM Comput. Commun. Rev. 43 (October) (2013) 147–158.
[108] I. Psaras, R.G. Clegg, R. Landa, W.K. Chai, G. Pavlou, Modelling and evaluation of ccn-caching trees, in: proceedings of the 10th International IFIP TC 6 Networking Conference, Valencia, Spain, 2011, pp. 78–91. May.
[109] M. Dräxler, H. Karl, Efficiency of on-Path and off-path caching strategies in information centric networks, in: IEEE International Conference on Green Computing and Communications (GreenCom), 2012, pp. 581–587. November.
[110] Z. Zhang, C.-H. Lung, I. Lambadaris, M.S. Hilaire, S.S.N. Rao, Router position-based cooperative caching for video-on-demand in information-centric networking, in: IEEE 41st Annual Computer Software and Applications Conference (COMPSAC), 1, Turin, Italy, 2017, pp. 523–528. July.
[111] M. Zhang, P. Xie, J. Zhu, Q. Wu, R. Zheng, H. Zhang, NCPP-based caching and NUR-based resource allocation for information-centric networking, J. Ambient Intell. Humaniz. Comput. (2017) 1–7.
[112] H. Yan, D. Gao, W. Su, C.H. Foh, H. Zhang, A.V. Vasilakos, Caching strategy based on hierarchical cluster for named data networking, IEEE Access 5 (April) (2017) 8433–8443.
[113] S. Saha, A. Lukyanenko, A.Y. Jaaski, Cooperative caching through routing control in information-centric networks, in: IEEE International Conference on Computer Communications (INFOCOM), Turin, Italy, April 14, 2013, pp. 100–104.
[114] S. Saha, A. Lukyanenko, A.Y. Jaaski, Efficient cache availability management in information-centric networks, Comput. Netw. Int. J. Comput. Telecommun. Netw. 84 (June) (2015) 32–45.
[115] W. Li, S.M. Oteafy, H.S. Hassanein, StreamCache: popularity-based caching for adaptive streaming over information-centric networks, in: IEEE International Conferenve on Communications (ICC), Kuala Lumpur, Malaysia, 2016, pp. 1–6.
[116] Y. Zhang, X. Tan, W. Li, PPC: popularity prediction caching in ICN, IEEE Commun. Lett. (July) (2017).
[117] W.K. Chai, D. He, I. Psaras, G. Pavlou, Cache less for more in information–centric networks (extended version), Comput. Commun. 36 (April) (2013) 758–770.
[118] S. Eum, K. Nakauchi, Y. Shoji, N. Nishinaga, M. Murata, CATT: cache aware target identification for ICN, IEEE Commun. Mag. 50 (December) (2012) 60–67.

[119] D. Rossi, G. Rossini, On sizing ccn content stores by exploiting topological information, in: IEEE International Conference on Computer Communications (INFOCOM) NOMEN Workshop, 2012, pp. 280–285. March.

[120] E.J. Rosensweig, J. Kurose, Breadcrumbs: efficient, best-effort content location in cache networks, in: IEEE International Conference on Computer Communications (INFOCOM), Rio de Janeiro, 2009, pp. 2631–2635. April.

[121] Y. Li, T. Lin, H. Tang, P. Sun, A chunk caching location and searching scheme in content centric networking, in: IEEE International Conference on Communications (ICC), Ottawa, ON, June 2012, pp. 2655–2659.

[122] M. Diallo, S. Fdida, V. Sourlas, P. Flegkas, L. Tassiulas, Leveraging caching for internet-scale content-based publish/subscribe networks, in: IEEE International Conference on Communications (ICC), Kyoto, Japan, 2011, pp. 1–5. June.

[123] C. Fricker, P. Robert, J. Roberts, N. Sbihi, Impact of traffic mix on caching performance in a content-centric network, in: IEEE International Conference on Computer Communications (INFOCOM) NOMEN Workshop, Orlando, FL, 2012, pp. 310–315. March.

[124] T. Mick, R. Tourani, S. Misra, MuNCC: multi-hop neighborhood collaborative caching in information centric networks, in: Proceedings of the 3rd ACM Conference on Information-Centric Networking (ICN), Kyoto, Japan, 2016, pp. 93–101. September.

[125] M. Garetto, E. Leonardi, V. Martina, A unified approach to the performance, ACM Trans. Model. Perform. Evaluat. Comput. Syst. 1 (May) (2016) Article 12.

[126] J. Wang, J. Ren, K. Lu, J. Wang, S. Liu, C. Westphal, A minimum cost cache management framework for information-centric networks with network coding, Comput. Netw. 110 (December) (2016) 1–17.

[127] A. Ioannou, S. Weber, Towards on-path caching alternatives in information–centric networks, in: IEEE Conference onLocal Computer Networks (LCN), Edmonton, AB, 2014, pp. 362–365. September.

[128] H. Che, Y. Tung, Z. Wang, Hierarchical web caching systems: modeling, design and experimental results, IEEE J. Sel. Ar. Commun. 20 (September) (2002) 1305–1314.

[129] T. Janaszka, D. Bursztynowski, M. Dzida, On popularity-based load balancing in content networks, in: Proceedings of the 24th International Teletraffic Congress (ITC'12), Krakow, Poland, 2012, pp. 1–8. September.

[130] J. Wu, C.K. Tse, F.C. Lau, I.W.H. Ho, Analysis of communication network performance from a complex network perspective, IEEE Trans. Circu. Syst. Regul. Pap. 60 (December) (2013) 3303–3316.

[131] I. Psaras, W.K. Chai, G. Pavlou, Probabilistic in-network caching for information-centric networks, in: Proceedings of the second edition of the ICN workshop on Information-centric networking, Helsinki, Finland, 2012, pp. 55–60. August.

[132] F.M. Al-Turjman, A.E. Al-Fagih, H.S. Hassanein, A value-based cache replacement approach for ICN, in: IEEE 38th Conference on Local Computer Networks Workshops (LCN Workshops), Sydney, NSW, Australia, 2013, pp. 874–881. October.

[133] J. Ji, M. Xu, Y. Yang, Content hierarchical intra-domain cooperative caching for information-centric networks, in: Proceedings of the 9th International Conference on Future Internet Technologies, Tokyo, Japan, 2014 Article 6. June.

[134] S. Vanichpun, A.M. Makowsk, The output of a cache under the independent reference model: where did the locality of reference go? ACM SIGMETRICS Perform. Evaluat. Rev. 32 (June) (2004) 295–306.

[135] L. Breslau, P. Cao, L. Fan, G. Phillips, S. Shenker, Web caching and zipf-like distributions: evidence and implications, in: IEEE International Conference on Computer Communications (INFOCOM), 1999, pp. 126–134. March.

[136] C. Xiaohu, Q. Fan, H. Yin, Caching in information-centric networking: from a content delivery path perspective, in: the 9th International Conference onInnovations in Information Technology (IIT), Abu Dhabi, 2013, pp. 48–53. March.

[137] N. Laoutarisa, H. Cheb, I. Stavrakakisa, The lcd interconnection of LRU caches and its analysis, Perform. Evaluat. 63 (July) (2006) 609–634.

[138] M. Lee, K. Cho, K. Park, T. Kwon, Y. Choi, Scan: scalable content routing for content-aware networking, in: IEEE International Conference on Communications (ICC), Kyoto, Japan, 2011, pp. 1–5. June.

[139] M. Everetta, S.P. Borgattib, Ego network betweenness, Soc. Netw. 27 (January) (2005) 31–38.

[140] Z. Li, G. Simon, Time-shifted tv in content centric networks: the case for cooperative in-network caching, in: IEEE International Conference on Communications (ICC), Kyoto, Japan, 2011, pp. 1–6. June.

[141] L. Dong, D. Zhang, Y. Zhang, D. Raychaudhuri, Optimal caching with content broadcast in cache-and-forward networks, in: IEEE International Conference on Communications (ICC), Kyoto, Japan, 2011, pp. 1–5. June.

[142] Z. Li, G. Simon, Cooperative caching in a content centric network for video stream delivery, Netw. Syst. Manag. 23 (July) (2015) 445–473.

[143] H. Xie, G. Shi, P. Wang, TECC: towards collaborative in-network caching guided by traffic engineering, in: the 31st Annual IEEE International Conference on Computer Communications (INFOCOM), Orlando, FL, 2012, pp. 2546–2550. March.

[144] A. Broder, M. Mitzenmacher, Network applications of bloom filters: a survey, Internet Math. 1 (4) (2005) 485–509.

[145] L. Saino, I. Psaras, G. Pavlou, Hash routing scheme for information centric networking, in: Proceedings of the 3rd ACM SIGCOMM workshop on Information-Centric Networking, Hong Kong, China, 2013, pp. 27–32. August.

[146] V. Sourlas, I. Psaras, L. Saino, G. Pavlou, Efficient hash-routing and domain clustering techniques for information-centric networks, Comput. Netw. 103 (July) (2016) 67–83.

[147] K.L. Calvert, M.B. Doar, E.W. Zegura, Modeling internet topology, IEEE Commun. Mag. 35 (June) (1997) 160–163.

[148] Kingman, J.F. Charles, Poisson Processes, John Wiley & Sons, Ltd, 1993.

[149] A. Afanasyev, I. Moiseenko and L. Zhang, ndnSIM: NDN simulator for NS-3, NDN, Technical Report NDN-0005, October 2012. [Online]. Available: http://named-data.net/publications/techreports/trndnsim/ (accessed 16 October 2019).

[150] Y. Wang, Z. Li, G. Tyson, S. Uhlig, G. Xie, Optimal cache allocation for content-centric networking, in: 21st IEEE International Conference on Network Protocols (ICNP), Göttingen, Germany, 2013, pp. 1–10. October.

[151] Y. Wang, Z. Li, G. Tyson, S. Uhlig, G. Xie, Design and evaluation of the optimal cache allocation for content-centric networking, IEEE Trans. Comput. 65 (January) (2016) 95–107.

[152] O. Ascigil, V. Sourlas, I. Psaras, G. Pavlou, Opportunistic off-path content discovery in information-centric networks, in: IEEE International Symposium on Local and Metropolitan Area Networks (LANMAN), Rome, Italy, 2016, pp. 1–7. June.

[153] F.M. Al-Turjman, H.S. Hassanein, Enhanced data delivery framework for dynamic information-centric networks (ICNs), in: IEEE 38th Annual Conference on Local Computer Networks (LCN), Sydney, NSW, Australia, 2013, pp. 810–817. October.

[154] O. Ascigil, V. Sourlas, I. Psaras, G. Pavlou, A native content discovery mechanism for the information-centric networks, in: Proceedings of the 4th ACM Conference on Information-Centric Networking, Berlin, Germany, September 26, 2017, pp. 145–155.

[155] M.J. Montpetit, C. Westphal, D. Trossen, Network coding meets information–centric networking: an architectural case for information dispersion through native network coding, in: Proceedings of the 1st ACM workshop on Emerging Name-Oriented Mobile Networking Design - Architecture, Algorithms, and Applications, Hilton Head, South Carolina, USA, 2012, pp. 31–36. June.

[156] Q. Wu, Z. Li, G. Xie, CodingCache: multipath-aware ccn cache with network coding, in: Proceedings of the 3rd ACM SIGCOMM workshop on Information–centric networking, Hong Kong, China, 2012, pp. 41–42. August.

[157] T. Ho, M. Médard, J. Shi, M. Effros, D.R. Karger, On randomized network coding, in: Proceedings of the Annual Allerton Conference on Communication Control and Computing, 2003, pp. 11–20. October.

[158] M. Zink, K. Suh, Y. Gu, J. Kurose, Characteristics of youtube network traffic-measurements, models, and implications, Comput. Netw. 53 (March) (2009) 501–514.

[159] Z. Gao, A. Venkataramani, J. Kurose, S. Heimlicher, Towards a quantitative comparison of location-independent network architectures, ACM SIGCOMM Comput. Commun. Rev. 44 (October) (2014) 259–270.

[160] J. Kurose, Information-centric networking: the evolution from circuits, Comput. Netw. 66 (June) (2014) 112–120 June.

[161] R. Chiocchetti, D. Rossi, G. Rossini, G. Carofiglio, D. Perino, Exploit the known or explore the unknown?: hamlet-like doubts in ICN, in: Proceedings of the 2nd edition of the ICN workshop on Information-centric networking (ICN '12), Helsinki, Finland, 2012, pp. 7–12. August.

Faria Khandaker (S'16) received her first M.Sc. degree from University of Dhaka, Bangladesh in 2008 and second M.Sc. degree from Ryerson University, Canada in 2012. She is currently working toward the Ph.D. degree with the School of Computing, Queen's University, Kingston, ON, Canada. She is a Member of the Telecommunication Research Laboratory Group, School of Computing, Queen's University. Her current research interests include Information-Centric Networks and economic prospects of Named Data Networks.

Sharief M. A. Oteafy (S'08–M'13) is an Assistant Professor at the School of Computing, DePaul University. Dr. Oteafy received his PhD in 2013 from Queen's University, ON, Canada, focusing on adaptive resource management in Next Generation Sensing Networks, introducing the notion of Organic sensor networks that adapt to their environment and scale in functionality with resource augmentation. Dr. Oteafy's current research focuses on Information Centric Networks, as well as dynamic architectures for enabling large scale synergy with the Internet of Things; encompassing dynamic resource management across multiple platforms, in addition to managing the proliferation of Big Sensed Data. He is actively engaged in the IEEE Communications Society (ComSoc), and is a Professional member in both IEEE and ACM, having joined them since 2008. Dr. Oteafy is the ComSoc AHSN Standards Liaison, and on the ComSoc Tactile Internet standards Working Group. Dr. Oteafy has co-authored a book on "Dynamic Wireless Sensor Networks", published by Wiley, and presented over 50 peer-refereed publications and delivered multiple ComSoc tutorials in Sensing systems and IoT. Dr. Oteafy co-chaired a number of IEEE workshops, in conjunction with IEEE ICC and IEEE LCN conferences, and served on the TPC of numerous IEEE and ACM symposia. He is currently an Associate Editor with IEEE Access, and on the editorial board of Wiley's Internet Technology Letters.

Hossam S. Hassanein (S'86–M'90–SM'06—F'17) is a Professor at the School of Computing, Queen's University, Canada. He received the Ph.D. degree in computing science from the University of Alberta, Edmonton, AB, Canada, in 1990. He is a leading authority in the areas of broadband, wireless and mobile networks architecture, protocols, control, and performance evaluation. His record spans more than 500 publications in journals, conferences, and book chapters, in addition to numerous keynotes and plenary talks at flagship venues. He is also the Founder and Director of the Telecommunications Research Lab, School of Computing, Queen's University, Kingston, ON, Canada, with extensive international academic and industrial collaborations. Dr. Hassanein is an IEEE Communications Society Distinguished Speaker (Distinguished Lecturer 2008–2010). He is a past Chair of the IEEE Communication Society Technical Committee on Ad hoc and Sensor Networks. He has received several recognitions and best papers awards at top international conferences, and led a number of symposia in flagship ComSoc conferences.

Hesham Farahat (S'12–M'18) received his B.Sc. and M.Sc degrees from Kuwait University in 2006 and 2009, respectively. In 2017, he earned the Ph.D. degree from Queen's University in Computer Engineering. Within the last 2 years, he has been granted teaching fellowships with the Electrical & Computer Department of Queen's University.His research interests include Mobility management in Named Data Networks in addition to networks performance evaluation. He is currently working in the R&D division at Huawei Technologies, Canada.