CONTENTS

DECLARATION OF AUTHORSHIP

This is to certify that the work presented in this thesis is the outcome of the analysis and investigation carried out by Biozid Bostami and Seyed Mosayeb Alam under the supervision of Md. Sakhawat Hossen in the Department of Computer Science and Engineering (CSE), IUT, Dhaka, Bangladesh. It is also declared that neither of this thesis nor any part of this thesis has been submitted anywhere else for any degree or diploma. Information derived from the published and unpublished work of others has been acknowledged in the text and a list of references is given.

Authors:

Biozid Bostami
Student ID - 114440

Seyed Mosayeb Alam
Student ID - 114442

Supervisor:

Md. Sakhawat Hossen
Assistant Professor
Department of Computer Science and Engineering
Islamic University of Technology (IUT)

ABSTRACT

The Information-Centric Networking (ICN) architecture exploits a universal caching strategy whose inefficiency has been confirmed by research communities. Various caching schemes have been proposed to overcome some drawbacks of the universal caching strategy but they come with additional complexity and overheads. Besides those sophisticated caching schemes, there is a probabilistic caching scheme that is more efficient than the universal caching strategy and adds a modest complexity to a network. The probabilistic caching scheme was treated as a benchmark and the insights into its behavior have never been studied despite its promising performance and feasibility in practical use. Here we study the probabilistic caching scheme by means of computer simulation to explore the behavior of the probabilistic caching scheme when it works with various cache replacement policies. The simulation results show the different behavioral characteristics of the probabilistic caching scheme as a function of the cache replacement policy.

1. INTRODUCTION TO ICN

1.1 What is ICN

The information-centric networking(ICN) concept is a significant common approach of several future Internet research activities. The approach leverages in-network caching, multi party communication through replication, and interaction models decoupling senders and receivers. The goal is to provide a network infrastructure service that is better suited to today's use (in particular. content distribution and mobility) and more resilient to disruptions and failures. The increasing demand for highly scalable and efficient distribution of content has motivated the development of future Internet architectures based on named data objects (NDOs), for example, web pages, videos, documents, or other pieces of information. The approach of these architectures is commonly called informationcentric networking (ICN). The ICN approach is being explored by a number of research projects.

1.2 ICN and Traditional Internet

1.2.1 Traditional Internet

The traditional Internet came from the concept interconnected networks.To understand the Internet, it helps to look at it as a system with two main components. The first of those components is hardware. That includes everything from the cables that carry terabits of information every second to the computer sitting in front of you. Other types of hardware that support the Internet include routers, servers, cell phone towers, satellites, radios, smartphones and other devices. All these devices together create the network of networks. The Internet is a malleable system – it changes in little ways as elements join and leave networks around the world. Some of those elements may stay fairly static and make up the backbone of the Internet. Others are more peripheral.These elements are connections. Some are end points – the computer, smartphone or other device you're using to read this may count as one. We call those end points clients. Machines that store the information we seek on the Internet are servers. Other elements are nodes which serve as a connecting point along a route of traffic. And then there are the transmission lines which can be physical, as in the case of cables and fiber optics, or they can be wireless signals from satellites, cell phone

or 4G towers, or radios. All of this hardware wouldn't create a network without the second component of the Internet: the protocols. Protocols are sets of rules that machines follow to complete tasks. Without a common set of protocols that all machines connected to the Internet must follow, communication between devices couldn't happen. The various machines would be unable to understand one another or even send information in a meaningful way. The protocols provide both the method and a common language for machines to use to transmit data.

1.2.2 Problems of Traditional Internet

The current problems of the Internet are a natural consequence of its architecture, which was designed to address the communication needs of a time when a network was needed for sharing rare and expensive resources, such as peripherals, mainframe computers, and long distance communication links. The basic requirement from the Internet at that time was merely that of forwarding packets of data among a limited number of stationary machines, with well-established trust relationships. The key design principles of the Internet made it very simple to link new networks to the Internet and enabled a tremendous growth in its size. In parallel to the Internets growth, an unprecedented number of innovations, in both the applications and services running on top of it, as well as in the technologies below the (inter-)network layer, have emerged. This is attributed to the hourglass approach followed by the Internets protocol architecture: the network layer forming the waist of the hourglass is transparent enough, so that almost any application can run on top of it, and simple enough, so that it can run over almost any link-layer technology.

1.2.3 Emergence of ICN

The tremendous growth of the Internet and the introduction of new applications to fulfill emerging needs, has given rise to new requirements from the architecture, such as support for scalable content distribution, mobility, security, trust, and so on. However, the Internet was never designed to address such requirements and in order to help it evolve a vicious cycle of functionality patches began appearing, such as Mobile IP. Most of those patches increased the complexity of the overall architecture and proved to be only temporal solutions[1].In addition, many current and emerging requirements still cannot be addressed adequately by the current Internet. This has raised the question of whether we can continue patching over patches, or whether a new clean-slate architectural approach for the Internet is actually needed[2].Along these lines, a research community has been formed which, having identified the limitations of the current Internet, is

discussing the key requirements and objectives of the Future Internet, and is proposing new architectures and paradigms to address them. In this context, Information-Centric Networking (ICN) has emerged as a promising candidate for the architecture of the Future Internet. Inspired by the fact that the Internet is increasingly used for information dissemination, rather than for pair-wise communication between end hosts, ICN aims to reflect current and future needs better than the existing Internet architecture. By naming information at the network layer, ICN favors the deployment of in-network caching (or storage, more generally) and multicast mechanisms, thus facilitating the efficient and timely delivery of information to the users. However, there is more to ICN than information distribution, with related research initiatives employing information-awareness as the means for addressing a series of additional limitations in the current Internet architecture, for example, mobility management and security enforcement, so as to fulfill the entire spectrum of Future Internet requirements and objectives.

1.2.4 key concepts and principles of ICN

In this section we introduce the key concepts and principles of ICN and discuss how each one of them aims to address some of the current Internets problems and limitations.

A. Focus on Information Naming

Users are more and more interested in receiving information/content/data1 wherever it may be located, rather than in accessing a particular computer system (host or server). However, the fact that the Internet is still based on an underlying host-centric communication model requires the user to specify in each request not only the desired information, but also the specific server from which it can be retrieved from. Unless add-on functionality is used, the Internets native network-layer mechanisms cannot locate and fetch the requested information from the optimal location where it is hosted, unless the user somehow knows and includes the optimal location in the request.

The ICN approach fundamentally decouples information from its sources, by means of a clear location-identity split.The basic assumption behind this is that information is named, addressed, and matched independently of its location, therefore it may be located anywhere in the network [3],[4].In ICN, instead of specifying a source-destination host pair for communication, a piece of information itself is named. An indirect implication (and benefit) of moving from the host naming model to the information naming model, is that information retrieval becomes receiver-driven. In contrast to the current Internet where senders have absolute control over the data exchanged, in ICN no data can be received unless it is explicitly requested

by the receiver. In ICN, after a request is sent, the network is responsible for locating the best source that can provide the desired information. Routing of information requests thus seeks to find the best source for the information, based on a location-independent name.

B. Focus on Information Delivery

The shift towards content-centric bandwidth-demanding applications requires the Internet to efficiently deliver massive amounts of information and handle large spikes or surges in traffic, commonly referred to as flash crowds. However, the data-agnostic Internet architecture lacks native mechanisms for handling flash crowd events and for enabling efficient information delivery. In the current Internet, data in transit are treated by network elements as a series of bytes that have to be transferred from a specific source to a specific destination and, as such, network elements have no knowledge of the information they transfer and hence cannot realize optimizations that would otherwise be possible (e.g., smart in-network caching, information replication at various points, information-aware traffic engineering).

In ICN the network may satisfy an information request not only through locating the original information source, but also by utilizing (possibly multiple) in-network caches that hold copies of the desired information (or pieces of it). This can be accomplished without resorting to add-on, proprietary and costly overlay solutions (e.g., CDNs), since the network layer in ICN operates directly on named information. ICN-based architectures see non-opaque data packets, in the sense that these are named based on the information they carry. Therefore, information fragments (packets in current terms) can be cached and retrieved easily, unlike in the current Internet.

C. Focus on Mobility

The addressing scheme of the Internet was designed with fixed hosts in mind, since a hosts IP address must belong to the network where the host is currently attached. However, statistics show a constantly increasing number of non-fixed hosts accessing the Internet, with forecasts saying that by 2015, traffic from wireless terminals will exceed traffic from wired ones [7]. Wireless and mobile devices may easily switch networks, changing their IP address and thus introducing new communication modes based on intermittent and, possibly, opportunistic connectivity. However, such an approach does not achieve continuous connectivity while on the move, which is becoming an increasingly important requirement. On the other hand, the Mobile IP protocol, a patch to remedy the problem of locating moving hosts, imposes triangular routing: packets first need to be routed to a home agent, representing the mobile host at its home network, and

from there to the current location of the mobile node via a tunnel. This is a major inefficiency, since traffic has to travel along a path longer than the optimal, a problem significantly aggravated when the mobile node, its home agent, and the third party that the host is communicating with are all located in distant Autonomous Systems (AS). Even traffic originating from a mobile node may need to be tunneled via its home agent, since many routers on the Internet exercise ingress filtering, i.e., they check that incoming traffic comes from the actual network it claims to originate from, meaning that the mobile node may not be able to directly send traffic from its current location using its permanent home address. Mobile IP, just like overlay networks [25], also tends to violate the usual valley free Border Gateway Protocol (BGP) routing policies, since packets are first routed to the mobile nodes home agent and from there re-routed to its currently hosting network. This leads to (a) valley routing, i.e., a client AS (where the home agent is located) serves traffic for a provider AS, and (b) exit policy violation, i.e., traffic exiting from an exit point different than the one it was supposed to, according to the BGP rules for a given traffic destination.

In ICN, host mobility is addressed by employing the publish/ subscribe communication model [26]. In this model, users interested in information subscribe to it, i.e., they denote their interest for it to the network, and users offering information publish advertisements for information to the network. Inside the network, brokers are responsible for matching subscriptions with publications i.e., they provide a rendezvous function. It is important to note that the publish/subscribe terminology used in the context of ICN (e.g., [27]) differs from that of traditional publish/subscribe systems (e.g., [11], [12], [13], [26]). In traditional publish/subscribe systems, publish involves the actual transmission of data while subscribe results in receiving data published in the future, with the ability of receiving previously published data being optional. In ICN, on the other hand, publish involves only announcement of the availability of information to the network, whereas subscriptions by default refer to already available information, leaving the option of permanent subscriptions (i.e., receiving multiple publications matching a single subscription) as optional. The strength of the publish/subscribe communication model stems from the fact that publication and subscription operations are decoupled in time and space [28]. The communication between a publisher and a subscriber does not need to be time-synchronized, i.e., the publisher may publish information before any subscribers have requested it and the subscribers may initiate information requests after publication announcements. Publishers do not usually hold references to the subscribers, neither do they know how many subscribers are receiving a particular publication and, similarly, subscribers do not usually hold references to the publishers, neither do they know how many publishers are providing the information [26]. These properties allow for the efficient support of mobility: mobile nodes can simply reissue sub-

scriptions for information after handoffs and the network may direct these subscriptions to nearby caches rather than the original publisher.

D. Focus on Security

The Internet was designed to operate in a completely trustworthy environment. User and data authentication, data integrity and user privacy were not a requirement; indeed the focus was on openness and flexibility in allowing new hosts to join the network. Moreover, the Internet was designed to forward any traffic injected in the network, resulting in an imbalance of power between senders and receivers. These characteristics allow spammers, hackers and attackers in general to launch Denial of Service (DoS) attacks against the Internet infrastructure or against Internet hosts and services, while easily covering their tracks. In order to cope with such malicious and/or selfish behavior, add-on security patches and trust mechanisms have been developed, such as firewalls and spam filters, as well as new security protocols that complement the existing (inter)networking protocols (e.g., IPSec and DNSSec). However, such solutions do not penetrate deep into the network and bad data still gets forwarded, clogging systems and possibly fooling filtering mechanisms [3]. The required processing overhead and the Internets end-to-end philosophy have so far prevented placing security and trust mechanisms deeper into the network, where it would be most effective in avoiding or identifying and stopping attacks.

Many of the security problems of the Internet are largely due to the disconnection between information semantics at the application layer and the opaque data in individual IP packets. This places a significant burden on integrating accountability mechanisms into the overall architecture. Point solutions like DPI or lawful interception try to restore this broken link between the actual information semantics and the data scattered in individual packets. However, this is achieved at a relatively high cost and is therefore only applicable to critical problems, such as law enforcement. As a result, while secure end-to-end connections are prevalent, the overall Internet architecture is still not self-protected against malicious attacks and data is not secure. At the same time, the lack of an accountability framework which would allow non-intrusive and non-discriminatory means to detect misbehavior and mitigate its effects, while retaining the broad accessibility to the Internet and ensuring both data security and communication privacy (i.e., hiding from non-authorized parties that a communication between two parties took place) is a crucial limitation to overcome [20].

ICN architectures are in contrast interest-driven, i.e., there is no data flow unless a user has explicitly asked for a particular piece of information. This is expected to significantly reduce the amount of unwanted data transfers (such as spam) and also facilitate the deployment of accountability and

forensic mechanisms on the network points that handle availability and interest signaling. Moreover, for ICN architectures that use self-certifying names for information, malicious data filtering will be possible even by in-network mechanisms. Finally, most ICN architectures add a point of indirection between users requesting a piece of information and users possessing this piece of information, decoupling the communication between these parties. This decoupling can be a step towards fighting denial of service attacks, as requests can be evaluated at the indirection point, prior to arriving to their final destination. Indirection can also benefit user privacy, as a publisher does not need to be aware of the identities of its subscribers.

1.3 ICN APPROACHES

The various existing ICN initiatives focus on designing an Internet architecture that will replace the current hostcentric model and will directly address the problems and limitations identified in the previous section. ICN oriented projects (see Figure 1) include the DONA [29] project at Berkeley, the EU funded projects Publish-Subscribe Internet Technology (PURSUIT) [30] and its predecessor Publish- Subscribe Internet Routing Paradigm (PSIRP) [31], Scalable & Adaptive Internet soLutions (SAIL) [32] and its predecessor 4WARD [33], COntent Mediator architecture for contentaware nETworks (COMET) [34], CONVERGENCE [35], the US funded projects Named Data Networking (NDN) [36] and its predecessor Content Centric Networking (CCN) [37] and MobilityFirst [38], as well as the French funded project ANR Connect [39] which adopts the NDN architecture.

Although they are still under active development, these ICN architectures address a set of key functionalities, albeit with different approaches. Below we identify these key functionalities:

- **Naming:** The structure of the name assigned to a piece of information (or service) that can be communicated over the network is one of the main characteristics of each ICN architectural proposal. In all ICN architectures information names are location-independent.On the other hand, depending on the approach, names may range from flat to hierarchical and may or may not be humanreadable.
- **Name resolution and data routing** Name resolution involves matching an information name to a provider or source that can supply that information, while data routing involves constructing a path for transferring the information from that provider to the requesting host. A key issue is whether these two functions are integrated, or coupled, or are independent, or decoupled. In the coupled approach, the information request is routed to an information provider, which subsequently

sends the information to the requesting host by following the reverse path over which the request was forwarded. In the decoupled approach, the name resolution function does not determine or restrict the path that the data will use from the provider to the subscriber. For example, an independent data routing module may send to the provider a source route to the requesting host.

- **Caching** We distinguish between on-path and off-path caching. In on-path caching the network exploits information cached along the path taken by a name resolution request, while in off-path caching the network exploits information cached outside that path. In ICN architectures with decoupled name resolution and data routing, off-path caching must be supported by the name resolution system, which handles caches as regular information publishers. If name resolution and data transfer are coupled, off-path caching must be supported by the routing system used to forward the requests for information.
- **Mobility** Subscriber mobility is intrinsically supported in ICN architectures, since mobile subscribers can just send new subscriptions for information after a handoff. Publisher mobility is more difficult to support, since the name resolution system (in the coupled approach) or the routing tables (in the decoupled approach) need to be updated.
- **Security** This aspect is tightly related to the naming structure [40]. On the one hand, human-readable names require a trusted agent or a trust relationship with the name resolution system to verify that the returned information corresponds to the requested name. On the other hand, flat names can support self-certification, but are not-human readable, thus requiring another trusted system to map human-readable names to flat names.

It is important to note that these are not ICN-specific functionalities, but rather the common core of all the ICN architectures considered. As such, this list simply aims to assist in shaping the presentation of each individual ICN architecture.

1.4 Open Issues in ICN

In this section we identify a series of issues and problems that have either not been satisfactorily addressed or have not even been tackled by the ICN research community so far.

1.4.1 Naming

There is no clear consensus yet on whether hierarchical or flat names should be used. Hierarchical names can be human-readable and are easier to ag-

gregate in principle, but it is unclear whether they can scale to Internet levels without turning into DNS names due to aggregation. On the other hand, flat names are easier to administer, they do not impose processing requirements for longest prefix matching, they can be self-certifying and they can be easily handled with highly scalable structures such as DHTs, but it is unclear whether DHTs can offer satisfactory performance.

There has been practically no research on incorporating versioning, deletion and revocation of information objects to the naming structure, and only preliminary work on the optimal granularity of information objects (i.e., an object could correspond to a packet, to variable-sized information chunks or to entire application-level objects). Indeed, some work argues that performing signature checks on individual packets may have excessive overhead [68], while other work argues that this is feasible with hardware-level implementations [57]. Searching for information has also not received much attention in ICN research, something rather peculiar, given that most projects rely on flat names that have to be somehow discovered by human users. Information-awareness may provide the means for efficient searching, possibly taking into account meta-information such as contextual parameters, location, information type, language, etc. For example, SAIL envisions an extended name resolution system that integrates meta-information to the resolution process [60]. As information is the primary entity in ICNs, it is possible for this meta-information to co-exist with the actual information inside the network, thus allowing the intelligent manipulation of traffic for other purposes, such as for enabling geocasting and flow prioritization. However, the availability of such meta-information also raises significant concerns regarding network neutrality. Earlier attempts to throttle certain types of traffic (e.g., P2P) were based on DPI techniques. With ICN, the identification of traffic types (and of any other metainformation related to a flow) may constitute standard network functionality, thus unveiling sensitive information not only to ISPs, but also to potential attackers.

1.4.2 Name Resolution

The vast size of the naming space poses a significant scalability challenge for name resolution. DHT based designs have attracted the attention of researchers due to their logarithmic scalability. The routing policy violations and inflated path lengths of DHTs have resulted in hierarchical schemes that try to adapt the structure of the name space to the underlying inter-domain network topology [76], but the routing efficiency of these approaches is still lacking [51]. Moreover, recent studies on the structure of the inter-domain graph suggest that the increase of peering relationships between ASs gradually leads to a mesh-like inter-domain graph [77], [78], therefore, employing a strictly hierarchical structure for the organization of the name space does not seem to reflect reality. Another recently pro-

posed approach is to use hashing to map names directly to IP addresses and rely on IP routing to find the resolvers [70], but this requires global participation in the name resolution system. Hence, a flexible and practical approach, able to express the dynamically evolving routing relationships between ASs, is still lacking.

1.4.3 Data Routing

While a lot of effort has been devoted to the design of routing mechanisms for the intra-domain level, e.g., [53], little attention has been paid to the inter-domain level. Inter-domain routing is strongly affected by business relationships between the involved parties and is an area of active research even in the context of the current Internet architecture [79]. In the ICN area, the main issue is scaling the proposed solutions to Internet sizes. As shown in [80], the content routers in NDN face serious scalability limitations at the inter-domain level, something that also applies to some extent to COMET, which also installs forwarding state at routers.
In the PURSUIT architecture which uses in-packet Bloom filters for source routing, the most obvious issue is that longer paths (or larger multicast trees) lead to many false positives, i.e., wasted packet transmissions [53]. Since larger Bloom filters would introduce much higher overhead, ideas such as Bloom filter switching [81] and variable-sized Bloom filters [82] have been explored. But the real problem is establishing inter-domain paths, since it is unrealistic to expect topology managers to have a global view of the network, due to both the size of the Internet and the limited information exchanged between ASs. This means that a hierarchical decomposition of the inter-domain routing problem is required, coupled with Bloom filter switching between the ASs, to keep topology management local and path lengths short [81], [83].
On the other hand, in the architectures where source routes are accumulated during name resolution, such as CONVERGENCE and the coupled variants of DONA and SAIL, the main issue is the amount of overhead introduced in both request and data packets as these routes grow larger. Mobility- First and the decoupled variants of DONA and SAIL basicallyrely on IP routing, with the possibility of additional resolution steps in MobilityFirst and the hybrid variant of SAIL. This means that they do not introduce any new problems, but they, at least partially, inherit the existing problems of IP routing.

1.4.4 Caching

Mechanisms for caching (and replication) have been widely studied at the application level, mostly in the context of web applications. It has been recently advocated that the benefits from the extensive use of caching in ICN will not be substantial [27]. Although they raise serious concerns about

the performance of the envisioned caching mechanisms, these observations are mostly based on studies performed more than a decade ago [84]. Additional research on current traffic patterns could shed additional light on the popularity characteristics of information today and thus to the possible benefits from widespread caching. For instance, a recent study has shown that web information popularity has changed during the past few years, affecting application level caching performance [85].
Another issue is that when caching takes place inside the network, as in ICN, several types of traffic will compete for the same caching space. Cache space management therefore becomes crucial for the network, and recent works, albeit based on simplified traffic models, have indicated that intelligent schemes can substantially improve performance [45], [44], [86]. Moreover, the deployment of caching and replication mechanisms inside the network opens up the possibility of jointly optimizing routing, forwarding and in-network cache management. For instance, routing decisions could be affected by cache locations, the cache-ability of information and/or indications of cache contention.

1.4.5 Mobility

Though identified as a major shortcoming of the current Internet architecture, network support for mobility has received very limited attention in ICN efforts (e.g., [46], [56]). Past research efforts on the support of mobility in the context of publish/subscribe systems [28] and on multicast-assisted mobility [87] have contributed to the understanding of the emerging issues. This work, coupled with the native ICN support for caching and multicast, has been leveraged to assist mobility in PURSUIT [54]. However, publisher (and, therefore, information) mobility remains a major challenge, since most ICN architectures use name resolution systems that are slow to update, whether they are name-based routing tables, hierarchical DHTs or hierarchical resolution handlers. The use of source routes, that may become invalid even as they are formed, is an additional complication. Even more problematic is the use of name aggregation in routing tables, as it implicitly reintroduces a location-identity binding. The most promising approach in this area is the late name binding advocated by MobilityFirst and the hybrid variant of SAIL, which simplify mobility management without losing the advantages of flat names. The performance of these schemes in practical and large scale scenarios remains to be seen, however.

1.4.6 Security, Privacy and Trust

Security in all ICN architectures is based on using encryption with keys associated with the information name. Little work exists however on how these keys will be managed, i.e., who will be responsible for creating, distributing and revoking those keys. The need for key management mech-

anisms becomes of paramount importance if we consider the fact that most ICN approaches rely on cryptographic keys and trusted entities for information-name verification [40], [47]. Moreover, most of the proposed ICN architectures envision access control mechanisms, nevertheless there is very little work on the definition of access control policies, the application of access control policies to cached information and the authentication of users (e.g., [88], [89]). ICN architectures can create severe privacy threats, as users reveal their interest in particular information and the name of the information being requested is available to all the ICN nodes processing the request [27]. A convincing solution for this threat has not been provided yet. Finally, efficient mechanisms for building trust relationships and handling privacy tussles amongst the various stakeholders are envisioned in ICN architectures (e.g., [90]), yet this still remains an open issue.

1.4.7 Transport

The information awareness in ICN architectures enables a series of new mechanisms and functionalities inside the network that make data transport a more complicated process than in the current end-to-end model. Mechanisms such as in-network caching and replication offer the opportunity for exchanging bandwidth with storage, thus radically changing the transport layer. Moreover, new delivery modes such as multicast (i.e., one-to-many) and concast (many-to-one), the ability of the network to apply anycast, as well as the support for multi-path routing in several ICN approaches, offer a rich set of mechanisms affecting the design of flow, congestion and error control functions. However, the fact that ICN architectures are still under active development, complicates research in the area. Recent efforts have started to investigate the interaction of these mechanisms (e.g., [91], [92], [93]), which is however far from being well-understood.

1.4.8 Quality of Service

Most ICN initiatives devote some thought to Quality of Service (QoS) provisioning. Nonetheless, only a few of them provide details about practical QoS mechanisms, while the rest treat the issue superficially. The most extensive treatment of QoS issues is in the COMET architecture which defines three Classes of Service (CoS) used to prioritize end-to-end information traffic. COMET maps the delivery requirements of the information as expressed by a CoS into the network paths offered by each AS via a path provisiong process [65]. Some work has also been performed on exploiting the centralized topology management and source routing of PURSUIT to implement routing algorithms that are infeasible with distributed routing, such as Steiner tree-based multicasting [94].

1.4.9 Business and Deployment Aspects

Taking a step away from technical issues, a series of questions need to be answered with regard to the business aspects of ICN. To name but a few: Who are the new actors enabled by ICN architectures? How are the roles/relationships between current actors of the Internet ecosystem going to be affected? Which are the application domains to target first? Should overlay or native ICN solutions be deployed first? For example, CDNs already provide several features of the ICN paradigm at an overlay level. It is not clear however how CDNs would possibly fit in an ICN world, as a major part of their functionality would be provided by the network itself.
A first attempt to perform a socio-economic analysis of an ICN architecture was performed in the PSIRP project [95]. According to its findings the logical order of markets to target would be government, business ICT, and information-centric applications. This is because the business opportunities in the government sector can be satisfied with the adoption of purely overlay mechanisms, which entail a smaller overall cost compared to the adoption of native mechanisms. On the other hand, native mechanisms are necessary to fully exploit the business opportunities related to the business ICT sector. Finally, the investment in information-centric applications is strongly dependent on traffic volumes, which in turn depend on the widespread access to applications, and hence requires a widespread deployment of the new architecture. According to the same analysis, the adoption of an ICN architecture should start with the adoption of overlay mechanisms in the current Internet, followed by the adoption of native mechanisms on the network backbone. The adoption of such native mechanisms should start from the business ICT sector. Issues like billing, costing and invoicing for ICN traffic however remain open.
With regard to deployment, it is clear that an incremental transition into ICN is needed, so as to maintain compatibility with TCP/IP-based applications for an extended period. Although such a transition is straightforward for overlay ICN solutions, it is not well understood how it can be achieved for the case of clean-slate ICN solutions. In addition the ICN community has not reached a consensus on several fundamental design choices (e.g., routing and forwarding in NDN vs. PURSUIT) hence there are several architectures proposed, each fitting the requirements of different networking environments and/or business scenarios. It is therefore possible to reach a state where multiple different ICN architectures will be deployed in parallel and interoperability issues may arise.

2. CACHING AND ICN

In the Internet, a large amount of content is transferred repeatedly [15]. Most of the time, the content is retransmitted from the source to serve different requests coming across the network. The efforts to reduce the amount of repeated traffic can be roughly divided into two categories: application level caches, including web caches [30] and Content Delivery Networks (CDNs, cf. e.g. [1]), and application-independent caches, which can be often found in the so-called WAN optimization products [14]. Today, these approaches together offer significant improvements to the network performance by caching some of the content.

Besides packet and chunk-level caches used inWAN optimization, in-network packet-level caches are also (re)gaining academic interest [4, 6]. It has become economically and technologically feasible for network devices to store large amounts of data [13]. Using intelligent and targeted caching at selected nodes, it is possible to reduce the network load, shorten the experienced latency, and avoid hot spots [4, 24].

Typically, in-network packet-level caches are independent of the end-to-end transport logic and therefore introduce less overhead at the caching points than the upper-layer caches. However, in spite of their simplicity and effectiveness, the packet-level caches are today used only for reducing link load, while upper level caches can also help with other inefficiencies, i.e., high latency and server load [1, 10, 30].

information-centric networks can enable fully functional packet-level caches and use them as general network components. It also manage some of the inefficiencies with packet caches, specifically from the transport layer point of view.

2.1 Universal Caching Schemes

Universal caching scheme state that the whenever any new information is found it should be cached based on the caching criteria. If there is no place for the new information in the cache then evict an old information following the replacement policy. In universal cache scheme whenever a new information arrives then some kind of caching operation should be performed. For example: if there is place for insertion then insertion operation is go-

ing to take place otherwise first eviction operation take place following the insertion operation.

2.2 Probabilistic Caching Schemes

probabilistic caching scheme, can overcome some main drawbacks of universal caching strategy in content-centric networks.The probabilistic caching scheme was used as a benchmark policy in the literature [2], [3], [4] despite its interesting performance gain. In addition, the source of determining its caching probability remains ambiguous. The insights into the behavior of the probabilistic caching scheme have never been studied. It has been roughly evaluated when it works with a particular cache replacement policy (i.e., Least-Recently- Used (LRU)). To the best of our knowledge, there has never been a criterion that suggests a decent value of the caching probability as well as its practical limitation. Even though there are other caching schemes that can efficiently utilize the network of caches [4], [7], [8], [9], [10], those sophisticated caching strategies in general add significant complexity and overheads to caching systems, which may not be applicable from the practical point of view. We, therefore, focus our study on the behavior of probabilistic caching scheme, which is simple but surprisingly effective, when it works with different cache replacement policies. The objective of this study is to provide guidelines for managing information-centric networks that are not only efficient but also practical.

3. LITERATURE REVIEW

The evaluation of caching schemes and replacement policies has been extensively studied. However, caching and replacement methods, which cooperatively manage a caching system, have been frequently evaluated as separate studies. A number of caching strategies have been recently proposed [4], [7], [8], [9], [10], [11]. Content-popularity-based caching schemes were presented in [7], [8]. These novel caching schemes have been proven to be efficient but they inevitably require synchronization among nodes, which may introduce large overheads and complexity. A few cooperative caching strategies were presented in [4], [9], [10] but those sophisticated caching schemes may be impractical, considering the scale of information-centric networks as well as the link latency. Chai et al. [11] proposed a centrality-based caching algorithm whose caching decision is based on the concept of betweenness centrality. The performance of their caching method was supported by simulation results but it was evaluated only with LRU.

Even a value based cache replacement policy was proposed in the Fadi et al.[19] proposed a caching scheme which consider multiple attribute together to cache a content. The proposal works fine in simulation but in practical case it takes much improvement in routers so that they can compute such calculation. Moreover, they did not consider the mobility of the requester.

we even consider the message passing co-operative scheme but that also create bottleneck in local routes of the network. Even though the scheme works well in small scale networks.

In previous studies of cache replacement policies, LRU and LFU were commonly considered, whereas other replacement policies were left out of the scope of interest [12], [13], [14]. Carofiglio et al. [12] explored the impact of storage management on the performance of multiple applications that concurrently share the same content-centric network. They observed that the performance of LFU is superior to that of LRU in terms of the diversity of content cached in a network. Ardelius et al. [13] provided analytical solutions for the cache hit rate and data availability of an aggregation access network. Nevertheless, only a universal caching scheme was considered in their analytical models which cannot explain the behavior of a probabilistic caching scheme. Fricker et al. [14] studied the impact of

traffic mix on the caching performance in contentcentric networks. They pointed out the inefficiency of LRU in comparison to LFU. However, their study was limited to a two-layer cache hierarchy, which may not be applicable to large networks of caches.

A more complete study that considered both caching schemes and cache replacement policies was conducted by Rossi and Rossini [3]. They evaluated the different combinations of caching schemes and cache replacement policies in various network topologies. They observed that the results of matching between randomly caching and random replacement policies were as good as that of the matching between a universal caching scheme and LRU. However, they did not clearly state the causes of their results, which is necessary for understanding the behavior of a probabilistic caching scheme. Therefore, we conduct our study with the goal to clarify the performance of the probabilistic caching scheme under various cache replacement policies.

The Information-Centric Networking (ICN) architecture was proposed in [1]. ICN uses prefix names to identify the content objects and to route packets. The prefix name can be hierarchically constructed based on the URI Representation. There are three main data structures in a CCN router, i.e., Forwarding Information Base (FIB), Content Store (CS), and Pending Interest Table (PIT). The FIB of a ICN router partly resembles the routing table of IP routers whose destination field is changed into the prefix of the content names. The CS is a cache embedded in a ICN router. The PIT is responsible for keeping the routing states of ongoing transmissions. The routing in ICN is receiver-driven, which is controlled by the forwarding of interest packets. An interest packet contains a prefix name of the content. When an interest packet arrives at a ICN router, it searches for the desired content that may be stored in the CS. If there is a matching content, a data packet is created and sent back along the reverse path of the relevant interest packet. Otherwise, the ICN router checks the PIT whether any interest packet asking for the same content has been sent. If so, the ingress face (interface) of the interest packet is added to the existing PIT entry. Otherwise, a new PIT entry is created and the FIB is consulted to determine where to forward the interest packet.When the forwarded interest packet meets its desired content, the relevant data packet follows the reverse path to the requester router based on the recoded ingress faces in the PIT. If the prefix name of the data packet matches a PIT entry, it is considered valid. Otherwise, the data packet is invalid and should be discarded. The content in a valid data packet can be cached in the CS of each CCN router it traverses depending on the caching decision and cache replacement policies.

In general, a Information-centric network consists of a number of ICN routers, which can be considered a network of caches. A traditional IP network that deploys caching systems, e.g., web caches and Content Distribution Networks (CDNs), is also a network of caches. However, the

content-centric and IP networks are different in terms of their caching granularity. An IP network may have several caches or data centers that are monitored by an administrator. In the case of CDN, the content stored in the data centers are deliberately selected in advance by content providers. On the other hand, the number of ICN routers in a content-centric network, which is seen as the number of independent caches, can vary from a few to several thousand nodes depending on where the ICN architecture is deployed. A large number of nodes in a content-centric network as well as the link latency between them could make the cooperative caching and centralized management infeasible or ineffective.

4. CACHING SYSTEM MODEL

There are two important algorithms managing a caching system: a caching scheme and a cache replacement policy. The capacity of a cache is limited by the embedded physical memory whose size is smaller than the item population. A caching scheme decides whether a caching system stores an item in its cache. An empty cache becomes full as a result of storing such items. A cache replacement policy is then needed when a new item enters the cache system and requires some space. A currently cached item is selected to be a victim for an eviction. The victim selection is governed by the cache replacement policy. The objective of a sophisticated cache replacement policy is to keep the items that are likely to be requested in the future by replacing an item that tends to be useless for future requests. A commonly used criterion for evaluating a caching system is its hit ratio, i.e., the frequency that a request finds its desired item in the cache. The caching schemes and cache replacement polices used in this study as well as their properties will be stated as follows.

4.0.1 Caching Schemes

The routers in a network should individually perform in a distributed manner and operate fast in order to fully exploit the benefit of ICN. Otherwise, the ICN architecture itself may cause a bottleneck in the network. We, therefore, focus on the two most straightforward and commonly known caching schemes, the universal and probabilistic caching schemes, which are considerably practical and scalable to a wide range of network sizes.

- Universal Caching Scheme (Always): The default ICN architecture suggests that a router should exploit a universal caching strategy, which is referred to as Always hereafter, as a caching decision policy of each ICN router [1]. A ICN router that deploys Always as its caching policy always caches the content object extracted from a valid data packet. The approach can quickly distribute content in a content-centric network. However, there are evidences pointing out that Always can put the replicas of the same content objects in multiple ICN routers and thus degrades the overall performance of innetwork caching, which is indicated by low cache hit rates at intermediate routers [2]. However, the poor performance of Always has been confirmed when it is merely used with a particular cache replacement scheme, i.e., Least Recently Used (LRU). Therefore, we

further investigate the behavior of Always when it is used with other replacement schemes in the next section.

- Probabilistic Caching Scheme (Prob(p)):The probabilistic caching scheme, which is referred to as Prob(p) from now, was used as a benchmark scheme in the literature [2], [3], [4]. The key idea is that each ICN router randomly caches a content object that traverses it at a certain caching probability, which is defined by p, where 0 ¡ p ¡ 1. Always is a special case of Prob(p), where p = 1. To the best of our knowledge, there has never been an unambiguous criterion that suggests a decent value of p and we first focus on this issue. Interestingly, the performance of Prob(p), where p ¡ 1, is better than that of Always, which can be inferred from the improvement in server hit rate and hop distance [2], [4], [7]. However, the Prob(p) has been tested only with a cache replacement policy, LRU. Note that p = 0.1 is the lowest value of p that has ever been used [4]. We propose that the value of p could be further decreased to improve the caching performance while its limit is constrained by an acceptable duration of the transit state of caching systems. The important properties of the Prob(p) can be summarized as follows: 1) Decreasing the caching probability p in Prob(p) reduces the probability that multiple ICN routers on a delivery path cache the same content object in a content delivery; and 2) Decreasing the caching probability p of Prob(p) results in a longer duration of the initial state of caching systems, given a static request pattern.

  The first property suggests that a small value of p should be assigned to Prob(p) in order to effectively distribute multiple content objects in a content-centric network and to efficiently utilize the in-network caching ability of ICN. In other words, the diversity of the content objects cached in the network can be improved by decreasing the value of p. However, setting a small value of p may result in a long duration of the initial state of caching systems according to the second property, which leads to a poor performance of capturing a high variation of access patterns.

4.0.2 Cache Replacement Policies

The capacity of a cache is generally smaller than the population of items, so all of such items cannot simultaneously reside in the cache. If the cache is full, the caching system must discard one of currently cached items before it can store a new item. A cache replacement policy determines which item is evicted. We consider three commonly known algorithms: Least Recently Used (LFU), Least Frequency used (LRU), and Randomly Replace (RR). LRU tries to keep recently active items in the cache by discarding the item

that is least-recently-used. LRU is simple to implement and operates fast since its running-time per request is O(1). However, if the capacity of cache is not large enough, LRU poorly performs when items are requested in a round robin fashion. Items will consistently enter and leave the cache without cache hit occurs. LFU replaces the least-frequently-used item with a new one. LFU is optimal when the requests received at different times are stochastically independent [15]. However, the running-time per request is logarithmic in the cache size (O(log(n))), where n is the cache size. In addition, it adapts poorly to variable access patterns by accumulating stale items with past high-frequency counts. RR is the simplest replacement policyone of currently cached items is randomly evicted whenever a replacement is invoked. It does not keep past information of access patterns and thus requires the minimal system requirements to operate. We use RR as a reference for the aforementioned policies, i.e., LRU and LFU. To the best of our knowledge, most of previous studies used LRU and RR as replacement policies of CSs in content-centric networks when Prob(p) was deployed [2], [3], [7]. We, therefore, evaluate our caching schemes of interest, i.e., Always and Prob(p), when they work with LRU, LFU, and RR by means of simulation in the next section.

5.0.3 A. Simulation Set-up:

We use ndnSIM [16], which is a NS-3 based network simulator dedicated to named data networking study, to conduct our simulations. All basic structures of ICN, FIB, PIT, and CS, are reproduced by ndnSIM. For the time being, ndnSIM models the routing mechanism of ICN which is driven by exchanging an interest packet and a data packet. However, it does not allow variable size of content object, so we ignore the content segmentation in our simulations and assume an identical size of content objects.

We assign various values of the caching probability p to Prob(p), where p 2 1.0, 0.7, 0.3, 0.01. As a result, our simulations take into account Always, Prob(0.7), Prob(0.3), and Prob(0.01). We use LRU, LFU, and RR as a replacement policy of the CS of each ICN router. We vary the CS size of each node from 1% to 10% of the content population.

The profile of content requests is modelled by using the Zipfs distribution which describes the popularity of each content object.

Each simulation run begins with all CSs being empty (i.e., cold start). Unless otherwise specified, the simulations run with the following parameters. The total simulation time is equal to 10,000 seconds with 4,000 second warm-up period. Each content requester requests content objects following the Poisson process whose mean is equal to 50 requests/s. Each content provider serves 1,000 different content objects. The uniform size of CS is varied from 1%, 2%, 5%, and 10% of the total content population. The results are reported at 95% confidence interval.

5.0.4 Evaluation Metrics:

We evaluate the performance of each caching scheme when it works with the particular cache replacement policy by observing one metric: the hit rate only.

Cache Hit ratio:

Cache hit ratio for a single cache is the fraction of time a request arrives at a node to which a the cache is attached but doesn't contain the requested data item. Average cache hit ratio is the average hitting ratio over all

caches, weighting each cache by the number of request it pass through it. It is commonly used to evaluate caching system. A high hit rate of a caching system implies its good performance.

Server load:

represents a prospective benefit of using CCN from the content provider standpoint. It directly reports the volume of traffic that a server must generate in response to its received requests. At an absence of cache in a network, the server load is equal to aggregated requests from all content requesters. An increasing server load pushes content providersto upgrade their facilities to provide an acceptable quality of service to all content requesters.

5.0.5 Network Topologies:

We conduct our simulations on the following network topologies, which are described as follows.

Cascading Network:

We use a fixed length cascading network in our simulations. A cascading network contains five ICN routers as shown in Fig. 1. We consider two study cases of using the cascading network in order to cover its practical use.

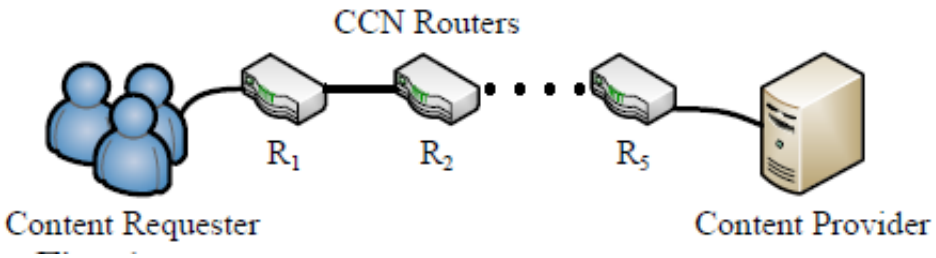

Fig. 5.1: Cascading network used in our simulations

- One content requester:The first study case is that request traffic accesses the cascading network at one ICN router. A content requester is connected to the ICN router at one end of network (R1), whereas a corresponding content provider is connected to the ICN router at the other end (R5).
- Multiple content requesters:The second study case considers that requests enter the network through multiple ICN routers. More specifically, four content requesters are connected to routers R1, R2, R3,

and R4. These content requesters request content objects from the content provider that is connected to the end of the network (R5).

5.0.6 Results and Discussions

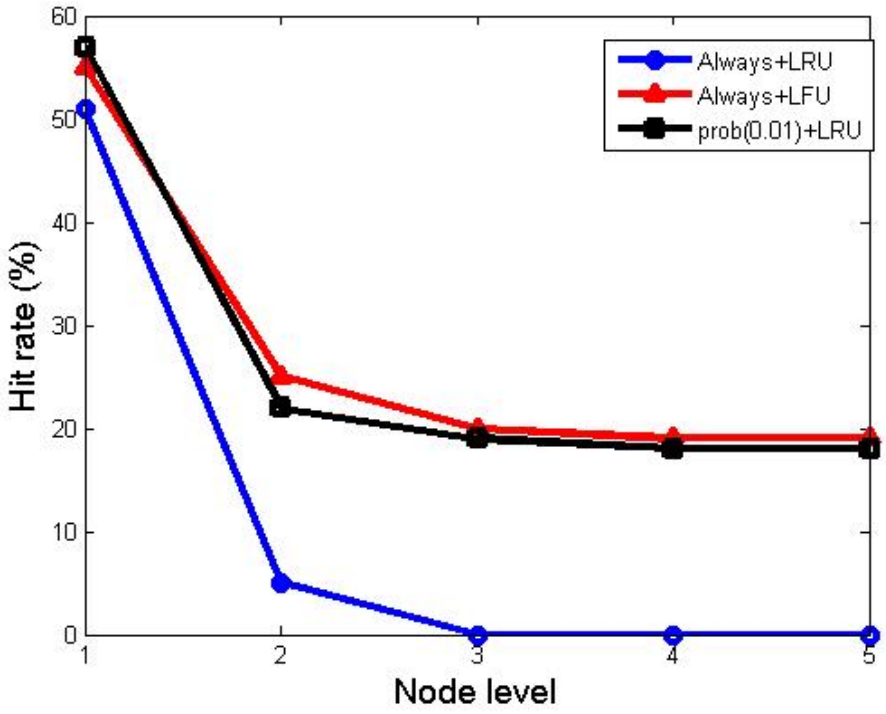

Fig. 5.2: Hit rate with respect to the node level in the cascading network (One content requester)

We measure the cache hit rate of ICN routers towards the node levels and report them in Fig 2 and Fig 3 for the first and second study cases, respectively. We show a comparison of Always + LFU, Always + LRU, Prob(0.01) + LRU, and Always + RR when the CS size of each router is equal to 10% of population due to limited space. For the first study case where the requests access the network at the node level 1, Always + LFU gives the best hit rates for all node levels in comparison to the other schemes. Interestingly, Prob(0.01) + LRU remarkably overcomes Always + LRU for all node levels. We find that Always+LRU gives a high hit rate for the node level 1 whereas the other nodes all suffer limited hit rates. In fact, Always + RR achieves higher hit rate than Always + LRU for every node level except the node level 1. For the second study case, the requests enter the network through multiple routers, so these routers become the first hop routers of some requests. Always+LRU, in essence, performs well for the first hop router, so its performance for the second case is better

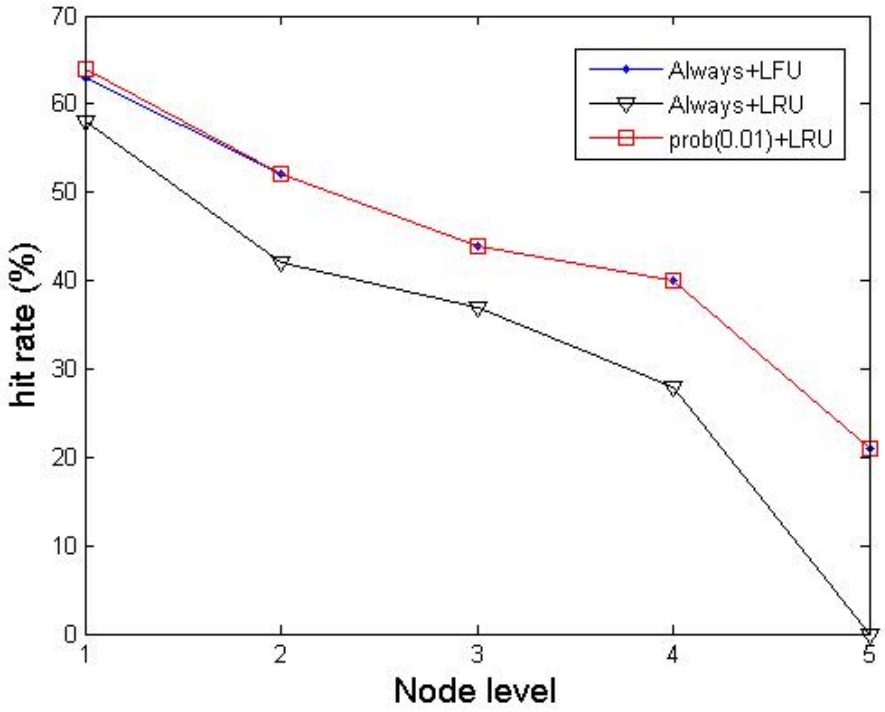

Fig. 5.3: Hit rate with respect to the node level in the cascading network (Multiple content requester)

than that for the first case. The hit rates of all node levels for Always + LFU and Prob(0.01) + LRU are almost identical. Prob(0.01) + LRU is easier to implement than Always+LFU, considering their running-time per request and caching-cost per transmission. In other words, although Prob(0.01) lets each router cache fewer content objects than Always in a transmission, it still gives the comparable performance.

6. CONCLUSION AND FUTURE WORKS

6.1 Conclusive Remarks

The behavior of a probabilistic caching scheme explicitly varies as a function of a cache replacement policy. The probabilistic caching scheme gives the improvement in the server load, round-trip hop distance, and cache hit rate compared with a universal caching scheme only when it works with LRU. The improvement increases as an inverse function of the caching probability assigned to the probabilistic caching scheme. When LFU is deployed in a content-centric network, a universal caching scheme is a policy of choice since it gives a better performance than the probabilistic caching scheme. On the contrary, the probabilistic caching scheme even magnifies the issue of LFU by letting ICN routers accumulate stale content objects with past highfrequency counts. The probabilistic and universal caching schemes have an identical behavior when RR is deployed in content-centric networks. The initial state of a network of caches is longer when the caching probability of a probabilistic caching scheme is decreased regardless of the deployed cache replacement policy.

6.2 CONCLUSION

We study the behavioral characteristics of a probabilistic caching scheme by means of computer simulation.We evaluate the probabilistic caching scheme when it works with different cache replacement policies. The evaluation metrics consist of the cache hit rate. The simulation results show that the behavior of a probabilistic caching scheme explicitly varies as a function of a cache replacement policy. The performance of probabilistic caching scheme and the duration of the initial state of a network of caches are inverse functions of a caching probability. The probabilistic caching scheme works well only in the caching system that implements Least-Recently-Used (LRU) as a cache replacement policy, whereas the limit of its performance comes from the increased duration of the initial state of the caching system.

6.3 Future Work

We are planing to implement our own probabilistic model which will give higher performance than present results does. We are hoping to explore other areas of Information Centric Networking too.

BIBLIOGRAPHY

[1] V. Jacobson, D. K. Smetters, J. D. Thornton, M. F. Plass, N. H. Briggs, and R. L. Braynard, Networking named content, in Proc. of ACM CoNEXT, pp.1-12, 2009.

[2] I. Psaras, W. K. Chai, and G. Pavlou, Probabilistic in-network caching for information-centric networks, in Proc. of the second edition of the ICN workshop on Information-centric networking, pp.55-60, 2012.

[3] D. Rossi and G. Rossini, Caching performance of content centric networks under multi-path routing (and more), Tech. Rep., Telecom ParisTech, 2011.

[4] K. Cho, M. Lee, K. Park, T. T. Kwon, Y. Choi, and S. Pack, WAVE: Popularity-based and collaborative in-network caching for contentoriented networks, in Proc. of IEEE INFOCOM WKSHPS 2012, pp.316-321, March 2012.

[5] A. Wolman, G. M. Voelker, N. Sharma, N. Cardwell, A. Karlin, and H. M. Levy, On the scale and performance of coopeartive Web procy caching, Operating Systems Review, vol.34, no.5, pp.1631, Dec. 1999.

[6] A. Ghodsi, T. Koponen, B. Raghavan, S. Shenker, A. Singla, and J. Wilcox, Information-centric networking: seeing the forest for the trees, in Proc. of ACM WKSHPS HotNets-X, pp.1-6, 2011.

[7] H. Wu, J. Li, T. Pan, and B. Liu, A novel caching scheme for the backbone of named data networking, in Proc. of IEEE ICC 2013, pp.3634-3638, June 2013.

[8] C. Bernardini, T. Silverston, and O. Festor, MPC: Popularity-based caching strategy for content centric networks, in Proc. of IEEE ICC 2013, pp.3619-3623, June 2013.

[9] S. Saha, A. Lukyanenko, and A. Yla-Jaaski, Cooperative caching through routing control in information-centric networks, in Proc. of IEEE INFOCOM 2013, pp.100-104, 2013.

[10] J. M. Wang, J. Zhang, and B. Bensaou, Intra-AS cooperative caching for content-centric networks, in Proc. of ACM SIGCOMM WKSHPS ICN 2013), pp.61-66, 2013.

[11] W. K. Chai, D. He, I. Psaras, and G. Pavlou, Cache less for more in information-centric networks, in Proc. of the 11th international IFIP TC 6 conference on Networking (IFIP12), pp.27-40, 2012.

[12] G. Carofiglio, V. Gehlen, and D. Perino, Experimental evaluation of memory management in Content-Centric Networking, in Proc. of IEEE ICC 2011, pp.1-6, June 2011.

[13] J. Ardelius, B. Gronvall, L. Westberg, and A. Arvidsson, On the effects of caching in access aggregation networks, in Proc. of ICN WKSHPS on Information-centric networking, pp.67-72, 2012.

[14] C. Fricker, P. Robert, J. Roberts, and N. Sbihi, Impact of traffic mix on caching performance in a content-centric network, in Proc. of IEEE INFOCOM WKSHPS 2012, pp.310-315, March 2012.

[15] J. E. G. Coffman and P. J. Denning, Operating Systems Theory, Prentice- Hall, 1973, pp.282.

[16] A. Afanasyev, I. Moiseenko, and L. Zhang, ndnSIM: NDN simulator for NS-3, Tech. Rep. NDN-0005, University of California, Los Angeles, 2012.

[17] A. Brodersen, S. Scellato, and M. Wattenhofer, Youtube around the world: geographic popularity of videos, in Proc. of 21st International World Wide Web Conference, pp.241-250, 2012.

[18] SINET4: Science Information NETwork 4 [Online]. Avalible: http://www.sinet.ad.jp.