Evaluating CRoS-NDN: A comparative performance analysis of the Controller-based Routing Scheme for Named-Data Networking

João Vitor Torres[a], Raouf Boutaba[b], Otto Carlos M. B. Duarte[a]

[a] *Universidade Federal do Rio de Janeiro - GTA/COPPE/UFRJ Email: {jvitor, otto}@gta.ufrj.br*
[b] *University of Waterloo Email: {rboutaba}@uwaterloo.ca*

Abstract

The huge amount of content names available in Named-Data Networking (NDN) challenges both the required routing table size and the techniques for locating and forwarding information. Content copies in different locations and content mobility worsen the scalability challenge. We present and analyze the performance of a specific Controller-based Routing Scheme, named CRoS-NDN, which preserves all NDN features using the same interest and data packets. The proposed scheme supports content mobility and provides fast content recovery from copies that do not belong to the consumer-producer path because it splits identity from localization without incurring FIB size explosion or supposing prefix aggregation. CRoS-NDN provides features similar to peer-to-peer and Content Distribution Network (CDN) in NDN, and it improves the efficiency for popular content. We compare our proposal with similar routing protocols. We derive analytical expressions for lower-bound efficiency and for upper-bound latency. In addition, we provide simulation results for data delivery efficiency and delay. The simulation results show the robust behavior of the proposed scheme that present the best performance for a wide range of scenarios. Furthermore, CRoS-NDN shows an economical use of computational resources for a growing number of prefixes.

Keywords: Named-Data, Content-Centric, Information-Centric, Networking, Software-Defined

1. Introduction

Named-Data Networking (NDN) applications refer directly to content names, avoiding host network identifiers for communication [1]. In this new paradigm, both the host mobility/multihoming and the content mobility/multihoming do not concern applications. The NDN-network layer focuses on unique network-visible names that identify content. This network layer forwards two types of packets: the interest and the data packets. The interest packet expresses consumers will for content and leaves breadcrumbs on each hop to reach the consumer back. Hence, for each interest packet, the network replies with a data packet containing the desired content. The NDN ensures efficient communication, load balance, energy efficiency, and flow control through popular content storage and data packet replies from any content cache copy [1, 2, 3, 4]. In addition, NDN is incrementally deployable because NDN packets can be transported over Internet Protocol (IP) or can replace IP. More importantly, interest and data packets one-to-one correspondence avoids link congestion due to Distributed Denial-of-Service (DDoS) attacks. Furthermore, unlike IP Multicast, NDN flow control is receiver-oriented and adapts to the link capacity of each individual consumer.

Named-data routers find and deliver content based on its name. Therefore, NDN routing schemes announce named-data prefixes diffusing their associated data location. NDN routing schemes based on Open Shortest Path First (OSPF) and Border Gateway Protocol (BGP) inherit IP characteristics due to their focus on prefix dissemination and routing. These routing schemes suffer with the amount of named-data prefixes that is intrinsically higher than the one required for IP prefixes. In addition, in order to reach content copies stored outside their original locations due to mobility, multihoming, and cache, NDN announces more routes with less-aggregated prefixes. In these scenarios, the routing schemes should store more routes and exchange more control messages to announce all the addressable contents, which results in high control overhead and possible risk of Forwarding Information Base (FIB) explosion [5]. On the other hand, announcement suppression of non-aggregated prefixes reduces the cache-hit opportunities just to copies located along the path from consumer to producer. Caching along the path supposes cache sizes big enough to accommodate popular contents that last enough time to respond repeated requests. This is a technical and economical trade off considering the amount of available content and the long tail for the content popularity distribution [6].

In a companion paper, we specified and validated the Controller-based Routing Scheme for Named-Data Networking (CRoS-NDN) that follows the Software Defined Networks (SDN) technology and preserves the same interest and data packets defined by Named-Data Networking (NDN) [7]. Thus, our proposal does not require additional packets. Therefore, packet forwarding follows default router processing through Content Store (CS), Pending Interest Table (PIT), and Forwarding Information Base (FIB) as detailed in [3]. We defined all the protocol messages, message time sequence, main features, and highlighted the main advantages. Our CRoS-NDN proposal splits the content names from the content localization and it forwards interests to the closest registered copy, irrespectively its location outside the path to producer. Unlike routing schemes based on prefix announcements, CRoS-NDN does not impose hierarchically indexed prefixes tied to location in order to summarize routing information that must fit in FIB size. In addition, unlike the location resolution approach of Domain Name System (DNS), CRoS-NDN localization is topology aware. Our scheme caches and reaches, closer to consumers, content copies that are less popular than the ones cached on routers along the path to producer. Therefore, CRoS-NDN provides low latency features like Content Distribution Networks (CDNs) and peer-to-peer applications. Furthermore, CRoS-NDN improves the mobility efficiency of content and content host because our scheme consolidates the routing information for content localization and for router adjacencies. This consolidation, provided by our controller-based approach, allows the usage of known technics to supply elastic resources for controller computation of routes and storage of content locations employing well-connected data center infrastructures. Unlike CRoS-NDN consolidation of controller functions, distributed approaches require the design of routers with processing power capacity and storage space for peak-utilization events of its local control plane functions. These events occur during network changes, but, most of the time, routers run with spare resources in distributed approaches [8].

In this paper, we analyze the CRoS-NDN data delivery efficiency and delay considering a single administrative domain and compare the results with other known distributed schemes. Our evaluation measures the communication overhead and the data delivery latency of each scheme. We derive expressions for lower bounds of the communication efficiency and upper bounds for the latency, worst-case scenario. We implement our proposal and the other distributed protocol in the ndnSIM simulator and we run a set of simulations to compare the different approaches. The obtained results demonstrate that the proposed Controller-based Routing Scheme for Named-Data Networking (CRoS-NDN) shows an efficient and robust behavior in relation the number of prefixes when compared with the distributed schemes and improves the mobility efficiency of content producers. Furthermore, the proposal decreases the messages overhead and decreases the content-delivery delay by adding peer-to-peer and CDN features.

The rest of this paper is structured as follows. Section 2 describes the main related work. Section 3 presents the CRoS-NDN proposal. Section 4 describes the comparing schemes. Section 5 presents the performance analysis of each scheme. Section 6 presents the simulation results. Finally, Section 7 concludes and presents future work.

2. Related Work

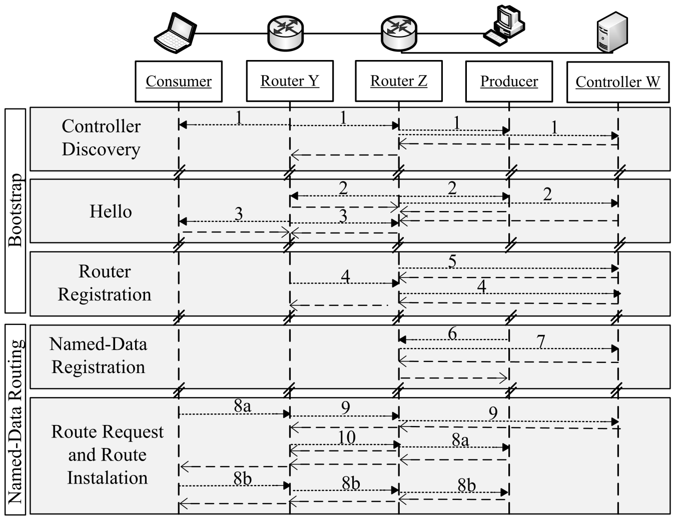
Ghodsi et al. discourage the ICN research due to the very long tail of content popularity distribution [9]. They argue that pervasive cache at all routers is worthless for an approach that cache only along the path to producer and that a single proxy cache would provide the same results. We observe that NDN mitigates server load in flash crowd events and Distributed Denial-of-Service (DDoS) attacks that are not solved by a single proxy cache. In addition, they argue that locating content copies outside the path to producer requires a localization resolution system that works at the rate given by the ratio of packet speed to mean object size. We note that the very long tail stands for aggregated measures of content popularity distribution taken for thousands of consumers employing large time windows. On the other hand, individual consumers present a much less flatter tail for popularity distribution measures of content prefixes taken for smaller time windows [10]. Thus, we argue that access routers cache the localization resolution data for local consumers. Additionally, the volume of video traffic dominates the total IP traffic today and keeps growing [11, 12]. The video traffic contributes to a lower rate of localization requests due to the large content size. Therefore, we argue that locating content copies outside the path to producer is worthy.

Various aspects of Information Centric Network (ICN) research are presented in surveys and all them point scalability as a major challenge [13, 14, 15, 16, 17, 18, 19, 20]. We argue that our proposed routing scheme reduces the routers memory requirement and the number of control messages pointed as a scalability challenge due to the vast size of the content naming space.

A number of schemes address content network, but propose a publisher-subscriber architecture [21, 22, 23]. We consider publish-subscribe approach is vulnerable to denial of service attacks, because it does not preserve the packet flow balance provided by on demand approach for individual data packets. Other schemes address the mapping problem of content identifier to location [24, 25, 26, 27, 28, 29]. For example, Baid *et al.* propose a two level indirection scheme that maps named-data prefixes to a reduced set of flat identifiers and, then, these identifiers into network addresses [27]. This Baid *et al.* scheme employs a distributed hash tables (DHT) system to provide this indirection that reduces the FIB memory requirement and the message exchange, but, like the cited mapping schemes, it does not preserve content names on forwarding decisions. We argue that our scheme can be extended to incorporate a scalable resolution scheme to execute this mapping; however, the extension should preserve the content name orientation on packet forwarding decisions to maintain the aggregation/caching opportunities and to adapt the forwarding plane to data mobility.

Afanasyev *et al.* propose a Domain Name System (DNS) to map and encapsulate data names in a reduced set of network names related to network domains [26]. The scheme reduces the FIB memory requirement, however, DNS servers have no clue of the request originator and, thus, DNS response contains multiple names and routers must execute multiple prefix-based lookups to find the shortest path choice for each content. They argue that name changes must be avoided due to complex implications on the named-based scheme. Zhang *et al.* propose a tunneling approach that changes content name and inherits the NDN benefits. We argue that both approaches should be further investigated and, more importantly, these two proposals are orthogonal to our CRoS-NDN scheme and one can be integrated to CRoS-NDN providing higher scalability on content location storage and retrieval.

A number of schemes propose Software Defined Network (SDN) technology to consolidate routing information on a centralized controller [30, 31, 32]. Fernandes *et al.* observe controller-based solutions alleviate general packet forwarding nodes from control message processing and fit well for next generation networks [33]. Rothenberg *et al.* argue the controller single point of failure is in general redundant and each controller takes charge for a limited subset of nodes overcoming the centralized criticism [34]. Shi *et al.* propose a data synchronization scheme for NDN that can replicate the controller information [35] and provide redundancy. Gao *et al.* proposes a scalable area-based hierarchical architecture (SAHA) for intra-domain communication to address the control plane scalability problem [36]. Salsamo *et al.* propose the OpenFlow-based architecture for the SDN technology applied to ICN [37]; however, the OpenFlow approach brings the well-known IP restrictions, for example, host mobility and multihoming [38]. We argue that the software-defined network approach overcomes the unnecessary control message flooding and reduces the router FIB memory requirement by storing only active consumed prefixes instead of all published prefixes, which is orders of magnitude higher than the active consumed prefixes [39], and by replacing the oldest added routing rules with new ones. We also argue that the on-demand route-request avoids the replications of routing information from controller to routers upon topology change or content mobility. In addition, the routers and the controller may sign the interests for security provenance and validity, as in VoCCN [40].



CRoS-NDN specific content names:
1 - /controller 2 - /hello/RouterZ 3 - /hello/RouterY
4 - /controllerx/ControllerW/registerrouter/RouterY/RouterZ
5 - /controllerx/ControllerW/registerrouter/RouterZ/RouterY
6 - /registerNamedData/myprefix
7 - /controllerx/ControllerW/registerNamedData/RouterZ/myprefix
8a - /myprefix/seq1 8b - /myprefix/seq2
9 - /controllerx/ControllerW/routeFrom/RouterY/myprefix/seq1
10 - /router/RouterZ/installRouteAndForward/RouterZ/prefixSize/myprefix/seq1

Figure 1: The Interest/Data packet time sequence for CRoS-NDN procedures. (1) Routers Y and Z find Controller W by sending a controller discovery message. (2) and (3) Routers Y and Z send a hello message to inform each other their presence. (4) and (5) Each router sends a router register message to register its neighbors in Controller W. (6) and (7) The Producer sends a named-data registration message to Router Z to register a named-data, in controller W. (8a) and (9) Consumer requests a content, sending a content-request message, and Router Y requests Controller W a new route for the named-data. (10) Router Y requests Router Z to install a new route to the named-data. (8b) Routers Y and Z forward further Consumer interests directly to Producer.

3. The Proposed Routing Scheme: CRoS-NDN

Our Controller-based Routing Scheme for Named-Data Networking (CRoS-NDN) is composed of two phases: Bootstrap phase and Named-Data Routing phase. The Bootstrap phase monitors the nodes and assures the knowledge of the global network topology. The Named-Data Routing phase guarantees the localization and access to the requested content [7]. Our scheme natively splits content identity from content localization, enabling content mobility. We define specific names and procedures for routers and controller efficient communication over NDN. Therefore, CRoS-NDN preserves NDN features keeping the named-data packet-forwarding scheme of NDN. In other words, unlike OpenFlow-based solutions, our proposal removes the dependency on IP for routers communication with a consolidated control plane. The control plane consolidation ensures an efficient way to register and retrieve content without flooding the entire network. CRoS-NDN automates the configuration to establish routers and controller communication and, thus, it avoids manual provisioning of network routers. Routers proactively register network information on the controller and they reactively request new routes to controller upon consumer interests to locally unknown name prefixes. It is of utmost importance to note that routers Pending Interest Table (PIT) keeps track of no-response expired interests. PIT expiration is native in NDN, but CRoS-NDN adds specific actions to remove invalid forwarding rules in the Forwarding Information Base (FIB) upon PIT entries expiration. Furthermore, CRoS-NDN routers update the controller topology view upon failure to reach neighbor routers. Unlike CRoS-NDN, NDN does not establish how to feedback the routing protocol based on PIT expiration.

CRoS-NDN executes the Controller Discovery, Hello, and Router Registration procedures in the Bootstrap phase. In Named-Data Routing phase, our scheme executes the Named-Data Registration, Route Request, and Route Installation procedures. Figure 1 presents the interest and data sequence of our scheme procedures.

Our scheme reduces routing signaling overhead by restricting network interest flooding. Routers only flood the network to initially find the controller, during the Controller Discovery procedure. Afterwards, the controller discovery only repeats upon no-response time expiration of router to controller interest. Furthermore, cache and interest aggregation reduce the discovery overhead. Therefore, CRoS-NDN wider broadcast domain does not incur additional signaling overhead for controller discovery [1]. Each router monitors its one-hop neighbors, by Hello procedure, and the router registers any topology change

[1] In order to deploy CRoS-NDN over IP, we note that IP Multicast is a solution to reach multiple IP subnets in a single domain and find the controller. Un-

in the controller, during Router Registration procedure. Routers also register in the controller the local produced content name prefixes, Named-Data Registration procedure. The controller stores the received information from network routers and it acquires knowledge of the network topology and of content location.

Unlike OpenFlow-based solutions that each router in consumer-producer path requests the controller a route, CRoS-NDN end-to-end route installation charges the controller with only one route request, during Route Request procedure. The route-requesting router informs its identifier and the requested content name in the route request sent to the controller. Upon the route request, the controller identifies the requesting router and, then, it locates the content producer router. Afterwards, the controller computes the sequence of router identifiers in the path from consumer to producer and, then, the controller answers the route request. Upon the controller answer for the route-request , the requesting router builds a specific interest that installs the new FIB entry on each router in the path from consumer to content producer, Route Installation procedure. Although the path calculation relies on router identifiers, the content-request interest forwarding relies only on content names.

The Named-Data Registration procedure provides content-copies reachability at any location. In addition, the Route Request procedure jointly resolves the content location and evaluates the best route from consumer to the content copy with the lowest cost. Therefore, CDN servers store content copies and register content location in the controller. Likewise, peer-to-peer application nodes at any location register content names to cooperate directly among themselves.

Topology changes or content mobility can invalidate FIB router entries. Therefore, unlike OpenFlow-based solutions that the controller proactively updates all routers FIB upon any change, CRoS-NDN employs data-plane feedback procedure to remove invalid FIB entries only on routers actively using that FIB entry. Interests without response cause Pending Interest Table (PIT) entry removal after the interest lifetime expiration. Then, on PIT entry removal, our scheme erases the associated FIB entries. In addition to reduce the signaling overhead, CRoS-NDN lessens the requirement for FIB memory router to the scale of simultaneous consumed prefixes. Our scheme reuses FIB memory and replaces old entries with new ones. This is in contrast to supporting all content prefixes available on the network irrespectively of consumer pattern of content requests for different prefixes.

4. Distributed Routing Schemes for Named-Data Network

In this section, we review the main distributed routing schemes for Named-Data Network that are cited in the literature. Two schemes are based on IP counterparts and the other one is a specific scheme for NDN. We use the "Like" term to denote our own implementation for each scheme due to the unavailability of the source code. It is of utmost importance to note that our implementation reflects the main limitations of each scheme concerning messages exchanges and that known optimizations that we found in the literature for the IP counterparts do not overcome these limitations.

like OpenFlow, CRoS-NDN does not require manual configuration of routers with the controller IP address that must be reachable a priori.

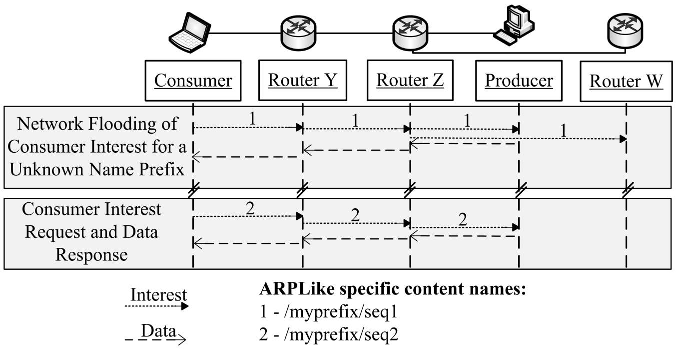


Figure 2: The Interest/Data packet sequence for ARPLike scheme procedures. (1) Consumer requests the content and Routers Y, Z, and W flood the request on all interfaces looking for content. (2) Routers directly forward further interests to the originating interface of data response for the first interest.

4.1. Address Resolution Protocol Like

The Address Resolution Protocol Like (ARPLike) routing scheme, based on proposal [41], employs a consumer-oriented approach to find content. ARPLike reacts to consumer requests flooding the network with interests for content that have unknown forwarding rules. Each router floods the network whenever the incoming interest does not match any FIB entry. Upon content response arrival, ARPLike router updates its FIB adding a new entry with the content name prefix pointing to the content incoming interface. Routers directly forward the subsequent interests with the same prefix using the new FIB entry. ARPLike employs the same CRoS-NDN procedure to remove invalid FIB entries, i.e., a PIT entry expiration timeout triggers the removal of the associated FIB entry. Figure 2 presents the interest time sequence for ARPLike procedures.

4.2. Open Shortest Path First Like

The Open Shortest Path First Like (OSPFLike) routing scheme follows CCN original routing concept [1]. OSPFLike employs a producer-oriented approach to announce content availability in a pro-active fashion. Unlike CRoS-NDN, OSPFLike content-producers periodically flood the entire network with prefix announcing interests. Each router does not monitor the connectivity to its neighbors and, therefore, routers forward the prefix announcement interest to periodically update the path to producer. Network wide recurrent flooding increases the routing signaling overhead in proportion to network size and to the number of content prefixes.

In order to flood the network, producers add a special prefix to content announcement interest messages. This prefix triggers two actions on interest-receiving routers: i) the router diffuses (replicates) the interest to all its interfaces and ii) the router adds a new FIB entry with the announced prefix pointing to the announcement-incoming interface. OSPFLike employs the

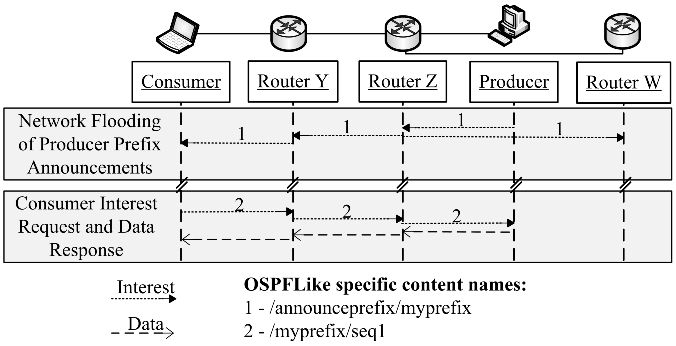


Figure 3: The Interest/Data packet sequence for OSPFLike scheme procedures. (1) Producer announces the prefix of available named-data. Afterwards, each router installs a FIB entry for the prefix and floods the prefix to its other interfaces. (2) Consumer requests and receives the content.

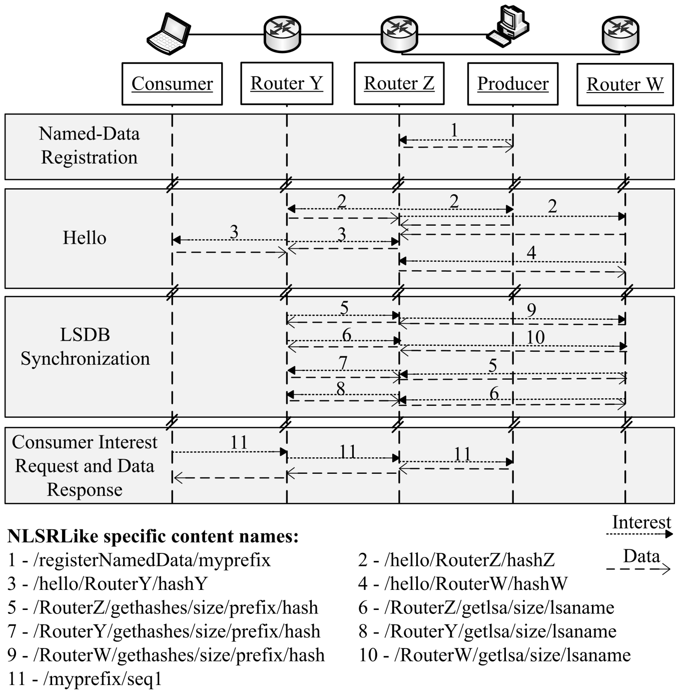


Figure 4: The Interest/Data packet sequence for NLSRLike scheme procedures. (1) Producer announces the content prefix to Router Z. (2, 3, and 4) Routers Z, Y, and W periodically announce their presence and the hash of their local database. (5, 6, 7, 8, 9, and 10) Routers Z, Y, and W synchronize theirs database. (11) Consumer requests and receives the content.

same CRoS-NDN procedure to remove the invalid FIB entries, i.e., PIT entry expiration timeout triggers the removal of the associated FIB entry.

Like our CRoS-NDN scheme, OSPFLike routers have no knowledge of network topology; however OSPFLike forwarding decisions follow the local view of the received prefix announcements. If a router receives the same announcement from multiple interfaces, then, it ranks output interfaces according to hop distance to producer. Moreover, unlike CRoS-NDN, OSPFLike router stores all available content prefixes simultaneously on its FIB memory. Figure 3 presents the interest time sequence OSPFLike procedures.

4.3. Named-Data Link State Routing Like

The Named-Data Link State Routing Like (NLSRLike) routing scheme, based on Hoque *et al.* proposal [42], avoids the OSPFLike flooding procedure. It replaces the OSPFLike periodic flooding of prefix announcements by a hop-by-hop procedure for database synchronization. Unlike the preceding schemes, each NLSRLike router maintains the full view of the network in a local database called Link State DataBase (LSDB). The LSDB stores the network topology and the content producer locations using database entries called Link State Advertisements (LSAs). The neighbor-LSA, with name `/routerid/LSAtype1/version`, stores the router adjacency list and the prefix-LSA, with name `/routerid/LSAtype2/LSAid/version`, stores the association of the content prefix with the producer identifier. Each router computes the hash for each LSA name, builds a tree with branches based on LSA name prefixes, and sums the hashes of LSA names that share equal prefix to compute the hash for each branch. NLSRLike builds a hash tree for the prefixes of LSA names and the LSDB hash is the root hash of this tree.

Producer registers the content prefix in its access router, Named-Data Registration procedure. Then, the router updates its local LSDB with a prefix-LSA. Neighbor routers exchange periodic interests to identify router adjacency, verify local connectivity, and compare their LSDB hashes (Hello procedure). Each router registers its neighbors in its local LSDB with a neighbor-LSA. If LSDB hashes of two neighbor routers differ, these routers initiates the LSDB Synchronization procedure that recursively exchange the branch hashes of LSA name prefix with hash differences until the branch leaf is reached. Then, the LSA difference is updated. Each router builds the network topology and the content prefix to producer identifier map based on its local LSDB and, then, the router evaluates locally the output interface upon consumer interest reception. Figure 4 presents the interest sequence for NLSRLike procedures.

5. Performance Analysis

In the preceding section, we have presented different routing schemes for content location and forwarding. In the ARPLike scheme, the consumer floods the network with interest packets to obtain a content. In other words, it is a consumer-oriented reactive flooding procedure. On the other hand, OSPFLike and NLSRLike proactively announce content localization, routing information, on the network, being a producer-oriented approach. OSPFLike periodically floods the network with announcements while NLSRLike employs a hop-by-hop procedure. Unlike the preceding schemes, our CRoS-NDN proposal avoids network recurrent flooding by consolidating network information on the controller. In this section, we analyze the Data Delivery Efficiency (DDE) and the Data Delivery Delay (DDD) for each routing scheme. The data delivery delay measures the delay between consumer content request and consumer content reception. The data delivery efficiency measures the ratio of

Table 1: Parameters of the routing scheme expressions.

Parameter - Description	Parameter - Description
N - Number of **N**odes	TR - **T**opology change **R**ate
L - Number of **L**inks	LD - **L**ink **D**elay
H - Network diameter **H**ops	DDE - **D**ata **D**elivery **E**fficiency
CR - **C**onsumer **R**ate	RTD - Max **R**ound **T**rip **D**elay
AP - **A**nnounced **P**refixes	CD - **C**onsumer-producer **D**elay
AR - **A**nnouncement **R**ate	AD - **A**nnouncement **D**elay
FF - **F**IB match **F**ail ratio	TD - **T**opology-update **D**elay
KR - **K**eepalive **R**ate	DDD - **D**ata **D**elivery **D**elay

Table 2: Data delivery efficiency lower bound expressions.

Scheme	Data Delivery Efficiency (DDE)
ARPLike	$1/(FF.L+(1-FF)H)$
OSPFLike	$CR(1-FF)/(AP.L.KR+CR.H)$
NLSRLike	$CR/(L(2.KR+4.TR+5.AR)+CR.H(FF+1))$
CRoS-NDN	$CR/(L(2.KR+TR)+H(N.TR+AR+CR(FF+1)))$

the consumer-received data packets to the number of interest packets sent on each network link. Therefore, local cached data on consumers yields delay zero and efficiency one. We derive mathematical expressions for DDE lower bound and DDD upper bound, worst case scenario. We employed the obtained expressions to discuss the limitation factors of each scheme. Table 1 presents the considered parameters employed for deriving the expressions for each routing scheme. The analysis considers that multiple collocated nodes ran the CRoS-NDN controller function as a single entity and that these nodes share a database that stores both the named-data location and the routers adjacency information for a single domain. This assumption does not invalidate the comparative performance analysis because it relies on data center infrastructure to host the nodes and, therefore, it eliminates processing power and storage bottlenecks of a single node.

Table 2 presents the lower bound values for data delivery efficiency of each routing scheme. The expressions consider that all network links have the same **L**ink **D**elay, LD, each **C**onsumer sends interests and receives data with a constant **R**ate, CR, consumer to producer distance equals network diameter **H**ops, H (worst case scenario), and router to controller distance equals network diameter (worst case scenario). It is worth to note that, in this scenario without cache, the lower-bound for the optimum efficiency equals $1/H$.

ARPLike efficiency depends on the fraction of interests that match an existing FIB entry, which is equal to the complimentary probability of FIB **F**ail **F**raction, $1 - FF$. ARPLike router straightly forwards to producer interests with FIB match. If an interest does not have a FIB match, the router floods the interest in its links. The higher is the fraction of directly forwarded interests, $1 - FF$, the closer ARPLike efficiency becomes to the optimum value that the lower bound is equal to $1/H$. The higher is the fraction of flooded interests, FF, the lower is the ARPLike efficiency. For large networks with restricted diameter ($L >> H$), consumer traffic with uncorrelated interest prefixes, and insufficient memory on router FIB to support all content prefixes simultaneously, then, ARPLike router recurrently floods interests and the efficiency tends to zero due to FIB entry replacement. Under router unbounded FIB memory assumption and after enough time, ARPLike routers store routes to all prefixes and FIB match failure tends to zero, and in this case, ARPLike efficiency tends to the optimum value.

In OSPFLike scheme, the number of interests on the network depends on the rate of consumer interests, CR, the rate of periodic content announcements, KR[2], and the number of **A**nnounced **P**refixes, AP. Consumer interests traverse H links to reach producer, expressed by $CR.H$ denominator factor. OSPFLike periodically floods all announced prefixes, AP, on all network links, L, with rate KR, given by $AP.L.KR$ denominator factor. The number of content data received by the Consumer is equal to the fraction of consumer interest rate that match a FIB entry and, thus $CR(1 - FF)$ is the numerator of the efficient expression. OSPFLike efficiency decreases with the number of content prefixes, AP, the rate of periodic prefix announcements, KR, and the number of networks links, L.

NLSRLike routers monitor their neighbors sending keep alive interests on all links, by Hello procedure, corresponding to $2.L.KR$ messages in efficiency denominator. Additionally, the LSDB Synchronization procedure of NLSRLike takes, respectively, five and four interests per link to synchronize new prefix-LSAs and router adjacency LSAs, given by $L(5.AR + 4.TR)$ denominator factor[3]. Producers announce new prefixes with rate AR and topology changes with rate TR. Furthermore, besides the consumer to producer interest hops given by $CR.H$, NLSR-Like FIB match failure FF takes one interest to control plane per router in the path from consumer to producer expressed by $CR.H.FF$.

The numerator of CRoS-NDN efficiency lower bound expression corresponds to consumer received content rate and it equals consumer interest request rate, CR. The denominator is composed by: $2.L.KR$ factor that corresponds to the Hello procedure of router neighbors monitoring interests; $TR.L$ factor corresponds to Controller Discovery procedure when controller discovery interest are flooded after each topology change; $H.N.TR$ corresponds to the Router Registration procedure, when all nodes registering in controller after each topology change, $H.AR$ corresponds to the Named-Data Registration procedure, when producers register available content prefixes on controller with rate AR, $H.CR.FF$ corresponds to the Route Request procedure when consumer sends to controller a route request upon consumer interest FIB match failure, and $H.CR$ corresponds to consumer to producer interests.

Next, we derive upper bounds expressions for data delivery delay DDD for all the analyzed schemes. Data delivery delay is another important performance parameter that corresponds to the delay between consumer content request and consumer content reception. DDD, see Table 3, sums three delay com-

[2]This value corresponds to the **K**eep alive **R**ate, KR

[3]Prefix-LSA and router adjacency LSA names have 4 and 3 components, respectively. Thus, the LSDB synchronization takes 4 and 3 interests to navigate from the root to the leaf of the LSDB hash tree and one additional interest to update the new LSA.

Table 3: Upper bound expressions for Data Delivery Delay (DDD) components: $DDD = CD+AD+TD$. Consumer-producer Delay (CD), Announcement Delay (AD), Topology-update Delay (TD).

Scheme	CD	AD	TD
ARPLike	RTD	0	0
OSPFLike	RTD	$RTD/2$	$1/KR$
NLSRLike	RTD	$5.RTD + H/KR$	$4.RTD + H/KR$
CRoS-NDN	$2.RTD$	$RTD/2$	$3.RTD/2 + 1/KR$

ponents: CD - delay between consumer interest dispatch and content reception; AD - delay between producer content prefix announcement and network wide reach; and TD - delay between a topology change and network forwarding rules convergence. In worst case, the routing scheme converges upon any topology change adding TD, afterwards producer can announce its content AD, and finally consumer can ask the content CD. However, not all routing schemes pass through these three phases and, then, DDD components equals zero in some cases. The maximum **R**ound **T**rip **D**elay, RTD, between any pair of routers equals the diameter delay $RTD = H.LD$. It is worth to that, in scenario without cache, the optimum DDD equals RTD.

The CD component considers the round trip delay between consumer and producer for all schemes, except CRoS-NDN. In worst case, CRoS-NDN consumer first asks the controller a new route to content producer and, then, this additional procedure adds the round trip delay between consumer and controller.

The AD component affects only the schemes that producer proactively announces content prefixes. ARPLike does not announce prefix and AD equals zero. OSPFLike and CRoS-NDN prefix announcement, respectively, adds to AD the one way producer to consumer delay and the one way producer to controller delay. NLSRLike prefix announcement employs the database synchronizing scheme. For each hop in the path from producer to consumer, NLSRLike adds to AD the LSDB hash exchange interval $1/KR$ and the prefix-LSA exchange delay. Prefix-LSA exchange employs five request and response sequential iterations and, then, it sums the delay $10.LD = 5.RTD$. Four iterations to exchange the branch hashes of the four components of LSA name and one additional iteration the exchange the LSA.

The TD component affects only schemes that routers monitor network topology changes. ARPLike does not monitor topology changes and TD equals zero. Although OSPFLike routers do not monitor topology, prefix announcement periodic interval delays new paths convergence and it adds $1/KR$ to TD. NLSRLike routers update their local LSDB with a new neighbor-LSA upon local topology change. The LSDB synchronism for neighbor-LSA is one iteration faster than for prefix-LSA, because neighbor-LSA name has three components. CRoS-NDN router periodically monitors connectivity to its neighbors at interval $1/KR$ adding this value to TD. Additionally, topology changes can incur changes in path from router to controller. In this case, CRoS-NDN router needs to search a new path to controller and to re-register in controller. Controller discovery adds the router to controller round trip delay and the register renewal adds another router to controller one way delay to TD.

Concerning the data delivery efficiency we can say that higher is the number of prefixes AP, better is CRoS-NDN and NLSRLike efficiency compared to OSPFLike. CRoS-NDN and NLSRLike only announce new prefixes with rate AR while OSPFLike periodically re-announces all prefixes AP with keep alive rate, KR. On the other hand, this OSPFLike comparative disadvantage reduces with the growth of topology change rate TR. CRoS-NDN shows a better efficiency than NLSRLike for scenarios with high number of prefix announcements. The efficiency decrease of our proposal CRoS-NDN is proportional to the prefix announcement rate and to the network diameter hops $H.AR$ while NLSRLike efficiency decrease is proportional to prefix announcement rate and to the network number of links $L.AR$. Furthermore, higher is the rate of interests for prefixes not installed in FIB $CR.FF$, better is CRoS-NDN efficiency compared to ARPLike. ARPLike floods interests without FIB match and the efficiency decreases proportionally to the number of links $L.CR.FF$. Unlike ARPLike, CRoS-NDN efficiency decreases proportionally to network diameter hops $H.AR + H.CR.FF$, $H.AR$ interest hops to producer register the content in controller and $H.CR.FF$ interest hops for consumer to request new routes from controller.

Concerning the Data Delivery Delay DDD depends directly on three parameters: network diameter in **H**ops, H, **L**ink **D**elay LD, and the keep-alive rate, KR. Lower is KR, greater is DDD for OSPFLike, NLSRLike and CRoS-NDN strategies. In special, for ($1/KR >> LD$), LD factor becomes negligible. Then, ARPLike delay tends to 0, OSPFLike delay tends to $1/KR$, NLSRLike delay tends to H/KR, and CRoS-NDN delay tends to $1/KR$. Albeit smaller KR value reduces signaling overhead, it increases DDD delay for OSPFLike, CRoS-NDN, and NLSRLike.

6. Simulation Results

In the preceding section, we analytically derived lower bounds for NDN routing schemes efficiency and upper bounds for the respective data delivery delay. In this section, we obtain performance simulation results for different scenarios that offers more detailed information of the behavior of the analyzed schemes. We implemented the schemes in the ndnSIM [43] simulator. To the best of our knowledge, ndnSIM is the closest to reality tool for NDN simulation. The ndnSIM reproduces the NDN model with a customizable forwarding strategy. Interest and data packets flow from node to node, and from/to node to/from application through faces. The strategy layer exposes customizable decisions on packet forwarding events. Thus, each routing scheme employs a specific forwarding strategy and specific applications to execute its procedures. Figure 5 shows the block diagram of the NDN node implementation on ndnSIM. We implemented two node modules to manipulate FIB and PIT entries based on data names: one executes specific forwarding strategy for each routing protocol and the other consumes/produces specific data packets related to specific routing scheme. The two modules employ internal calls to manipulate FIB, PIT, CS, and other state information.

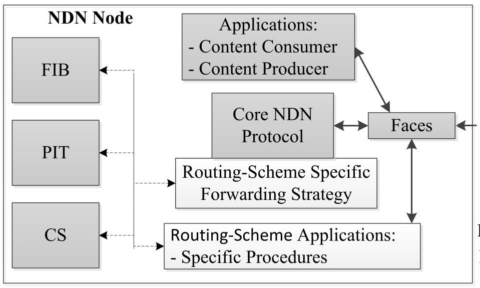
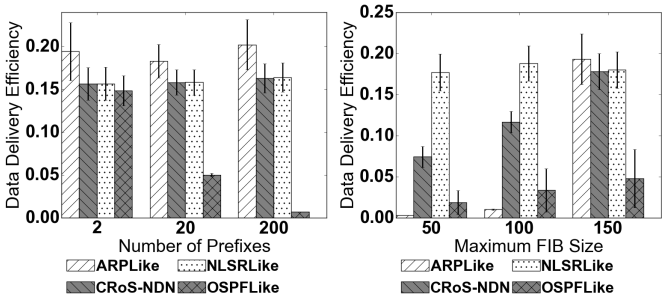


Figure 5: Customized ndnSIM node for implementing all routing schemes. The forwarding strategy module defines a specific routing scheme and interacts with specific applications module to manipulate FIB and PIT entries based on specific data names.



(a) OSPFLike efficiency decrease. (b) ARPLike efficiency decrease.

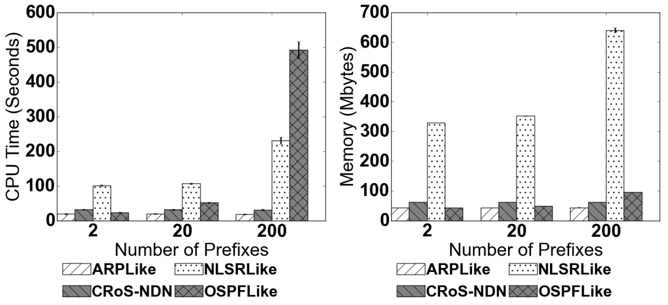
Figure 6: Data delivery efficiency for: a) unlimited FIB memory and a growing number of announced prefixes and b) different FIB sizes (50, 100, and 150) and 150 simultaneous consumed prefixes.

The simulation data employs 95 % confidence interval and we omit error bars in the temporal curves for a clean visualization. In each simulation round, consumer and producer routers are chosen randomly. The different distances from consumer to producer and from consumer to controller, in our CRoS-NDN proposal, cause the variation represented by the error bar in each plot. In addition, like other works, the simulations employ ISP-like topologies based on the largest connected component of Rocketfuel's topologies [44, 45], a mapping technique that measures real router-level ISP topologies. When not state in contrary, we employ the AS 1755 topology with 163 nodes and 366 links. It is worth to note that the network mean distance is 7.36 hops, the diameter is 22 hops, and the respective reference values for the data delivery efficiency are $DDE = 1/7.36 = 0.14$ for the mean case, and $DDE = 1/H = 0.05$ for the worst case. We choose the AS 1755 as the main topology because it has a sufficiently high number of links in comparison with diameter, $L >> H$, to reflect the flooding negative effect on efficiency. Furthermore, the keep-alive rate value KR is set to 0.1 for the OSPFLike periodic prefix announcement, like in OSPF [46], and for the NLSRLike/CRoS-NDN Hello procedure, like in NLSR [42]. We employ equal $KR = 0.1$ values in order to verify a fair comparison and we point that higher (lower) KR values decrease (increase) the efficiency and increase (decrease) the data delivery delay for these three schemes; however, different KR values do not change the comparative behavior with the increase in the number of prefixes. More importantly, we set conditioned values for simulation parameters in order to exhibit specific comparative results that would be obfuscated with real world values without any conditioning. Additionally, we emphasize the conditioning purpose is to explicit individual limitation factors of each scheme.

In the first set of simulation, we want to show the performance behavior of Data Delivery Efficiency when we increase the number[4] of prefixes by two orders of magnitude, from 2 to 200, and also when we restrict the FIB size. Figure 6a demonstrates the OSPFLike scalability weakness with the number of prefixes increase, even considering router with unlimited FIB memory. OSPFLike data delivery efficiency strongly decreases from 0.15 to 0.01 with the number of announced prefixes increase. The strong efficiency decrease of OSPFLike routing scheme is due to the periodically announcement of all available prefixes. It is worth to note that smaller KR values reduce the factor of OSPFLike efficiency decrease with the number of prefixes, but it does not change the tendency. On the other hand, ARPLike, NLSRLike, and our proposal CRoS-NDN efficiency shows very little variation with the number prefixes because these schemes avoid the periodic network flooding of available prefixes. The simulation considers two consumers and each one requests sequential data for one distinct prefix with rate of 40 interests per second.

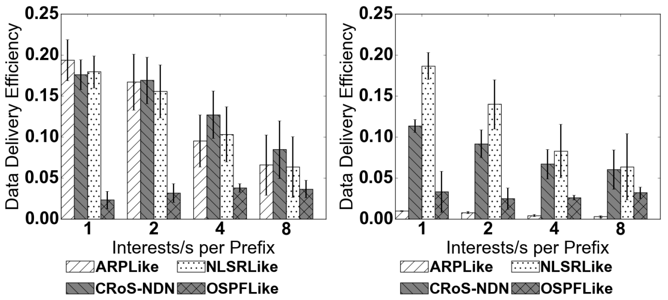
Figure 6b shows the Data Delivery Efficiency behavior for constrained FIB memory. The results demonstrate the ARPLike scalability weakness with the increase of the number consumed prefixes beyond the FIB memory capacity. The simulation employs 150 simultaneously consumed prefixes, each one with 1 interests per second, 150 announced prefixes, and a growing number of FIB entries supported per router. We choose an amount of announced prefixes that smooths the OSPFLike weakness with prefix announcements and that shows the effect of FIB memory deficiency. Under FIB memory restriction, all routing schemes replace the oldest installed entries by the new ones (first-in first-out – FIFO). ARPLike efficiency suffers a lot by each FIB entry removal because it recurrently floods the network and, thus, the efficiency decreases proportionally to the number of network links. Unlike ARPLike, all the other schemes do not flood consumer interests upon FIB match failure. NLSRLike efficiency shows very little variation with the number supported FIB entries per router because NLSRLike router employs its local control plane to reinstall the forwarding rules on FIB. OSPFLike efficiency decreases, due to the lack of memory for part of prefixes, from 0.05 to 0.02 when the FIB memory reduces from 150 to 50 entries. CRoS-NDN efficiency also reduces from 0.18 to 0.08, a reduction by a factor close to two that corresponds to the additional hop distance from consumer to controller for route requests. Consumer-producer and consumer-controller mean distances are equal to the network mean distance.

[4]We denote *consumed prefixes* the prefixes of content requested by consumers and we denote *announced prefixes* or simply *prefixes* the prefixes of content available at producers.



(a) Processing time. (b) Used memory.

Figure 7: Processing time and memory consumption for each simulation round.



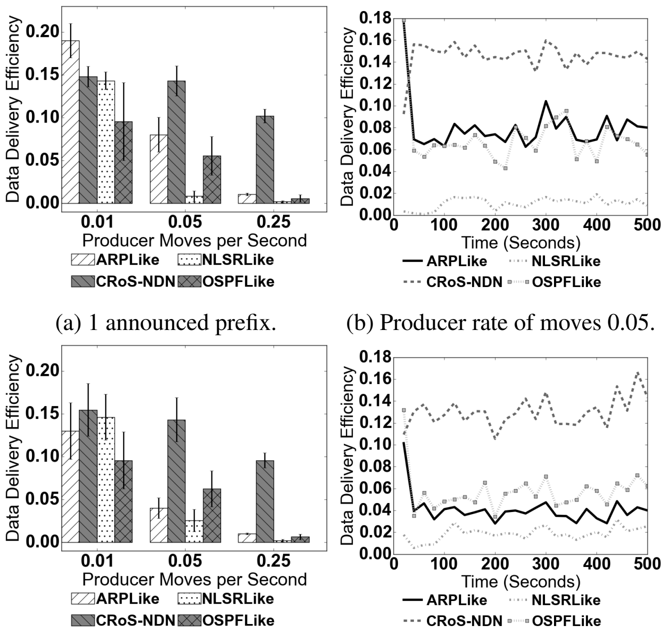
(a) Unlimited FIB memory. (b) FIB memory Size = 100.

Figure 8: Data delivery efficiency for a growing rate of consumer interests per prefix and 150 prefixes: a) unlimited FIB memory and b) limited FIB memory.

Figure 7 shows the processing time and the memory consumption of each simulation round for each scheme and for a growing number of prefixes. The results point the real consumed resources of our implementation and mirror the total network resource consumption of each scheme. NLSRLike and OSPFLike show the highest resource consumption. We note that, although the controller capacity does not scale infinitely, CRoS-NDN shows an economical use of resources for a growing number of prefixes.

Figure 8 shows the the Data Delivery Efficiency for a growing rate of consumer interests per prefix. The efficiency decreases due to congestion of excessive requests above link capacity. The results reinforce OSPFLike low efficiency with the number of prefixes, 150. Additionally, Figure 8b shows ARPLike low efficiency with FIB memory smaller than the amount of simultaneous requested prefixes.

Multihoming and mobility is a great problem in today's Internet because of the semantics overcharge of IP. Our proposal is based on plane separation and, then, natively splits localization and identification. Therefore, in this second set of simulations, we show the Data Delivery Efficiency robustness to the content-producer mobility, an important feature of our proposal CRoS-NDN as depicted in Figure 9. Furthermore, we show the robustness of CRoS-NDN efficiency when increasing by one order of magnitude the number of announced prefixes and of consumers. In order to explicit the comparative efficiency trend, the simulation considers 3 consumers per announced prefix, each consumer sending 20 interests per second, unlimited



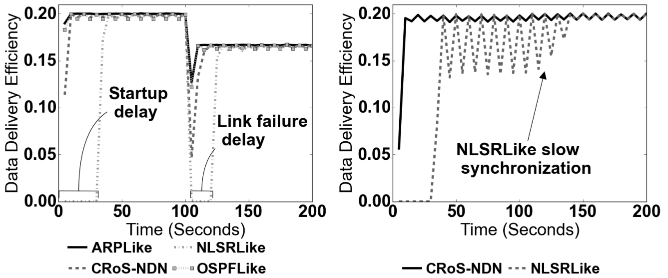
(a) 1 announced prefix. (b) Producer rate of moves 0.05.

(c) 10 announced prefixes. (d) Producer rate of moves 0.05.

Figure 9: Data delivery efficiency decrease due to the increase of the producer mobility and the number of named-data consumed prefixes: 1 prefix (figures a and b) and 10 prefixes (figures c and d).

FIB memory, and a growing rate of producer moves. Figures 9a and 9c present the data delivery efficiency for, respectively, 1 and 10 content prefixes in order to compare the combined effect of content mobility and the number of prefixes. In order to visualize the efficiency temporal evolution, Figures 9b and 9d show the curves for the rate of 0.05 producer moves per second of figures 9a and 9c, respectively. The results show that producer mobility increases both the ARPLike interest flooding for content search and the OSPFLike/NLSRLike announcements of producer prefixes and, thus, it strongly decreases the efficiency of these schemes. However, unlike for ARPLike, the growth of consumer interests rate with the number of prefixes contributes positively to OSPFLike and NLSRLike efficiency. On the other hand, CRoS-NDN presents the best results with fast convergence and low overhead for producer location update.

In the third set of simulations, we want to show the strong resiliency characteristic of our proposal that present a fast start up and link failure recovery. The data delivery efficiency *DDE* time evolution gives an indirect measure of convergence latency represented in the data delivery delay *DDD* metric. Figure 10a presents the latency for the convergence at start up and at the recovery from a link failure to a secondary longer path. CRoS-NDN presents a faster convergence delay because it only depends on routers delay to update its local information on the controller and routers delay to receive new routes from the controller. NLSRLike slower convergence is due to the hop by hop database synchronization latency. Furthermore, the set up convergence takes even longer due to the greater number of differences among routers databases. ARPLike and OSPFLike



(a) Start up and recovery to secondary path.

(b) Registration of producer new prefixes.

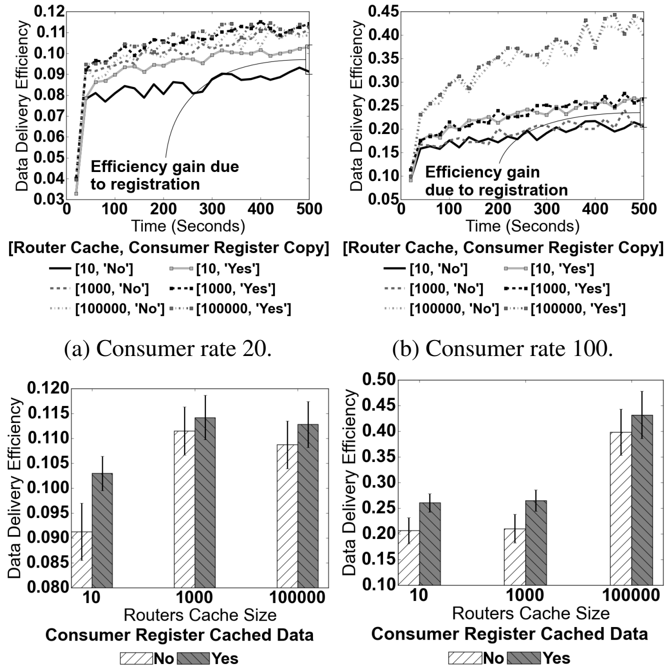
Figure 10: a) Data delivery delay (*DDD*) inference from the efficiency convergence latency at start up and after a link failure. b) CRoS-NDN and NLSRLike convergence delay for a producer registering 100 new prefixes at rate of 1 register per second.

schemes show similar and small delay values because ARPLike immediately floods interests for unknown prefixes and OSPFLike convergence depends only on the producer prefix announcement arrival to install new routes.

In Figure 10b, we demonstrate the CRoS-NDN fast propagation of new routing information in comparison with NLSRLike. The producer announces one new prefix per second in the initial 100 seconds. The prefix announcement reduces the NLSRLike efficiency due to the database synchronization packets and, additionally, NLSRLike shows a higher convergence delay.

We note that one can improve NLSRLike employing direct flooding of new LSAs on the network and, then, one can avoid the slow convergence of the LSDB Synchronization procedure for new LSAs. Moreover, unlike OSPFLike, NLSRLike avoids the need to recurrently flood content prefixes because NLSRLike routers synchronizes theirs local LSDB databases and, therefore, NLSRLike avoids the OSPFLike efficiency decrease with the number of prefixes. However, it is of utmost importance to observe that each NLSRLike router stores locally all network adjacency and all the content localization. Therefore, the number of routers and the number of contents impose serious scalability limitations on the amount of storage and processing power that each NLSRLike router must individually support. On the other hand, our proposal CRoS-NDN consolidates these resources on the controller function that can be executed by multiple nodes hosted in well-connected data center infrastructure and, then, CRoS-NDN routers focus the data plane functions in momently consumed content.

The preceding simulations showed the CRoS-NDN higher robustness with the amount of announced prefixes, simultaneously consumed prefixes, rate of consumer interests, and the producer mobility rate. In the fourth set of simulations, we further demonstrate CRoS-NDN specific features and show that CRoS-NDN enables peer-to-peer and CDN functionalities over NDN with efficiency gains. We show that, with CRoS-NDN, it is worth to the global efficiency that consumers do register the cached copies of popular content. We note that NLSRLike and OSPFLike do not reach content copies outside the path to producer without additional prefix announcements. In addition, ARPLike do not announce content location. Therefore, we restrict this evaluation to CRoS-NDN.



(a) Consumer rate 20.

(b) Consumer rate 100.

(c) Consumer rate 20.

(d) Consumer rate 100.

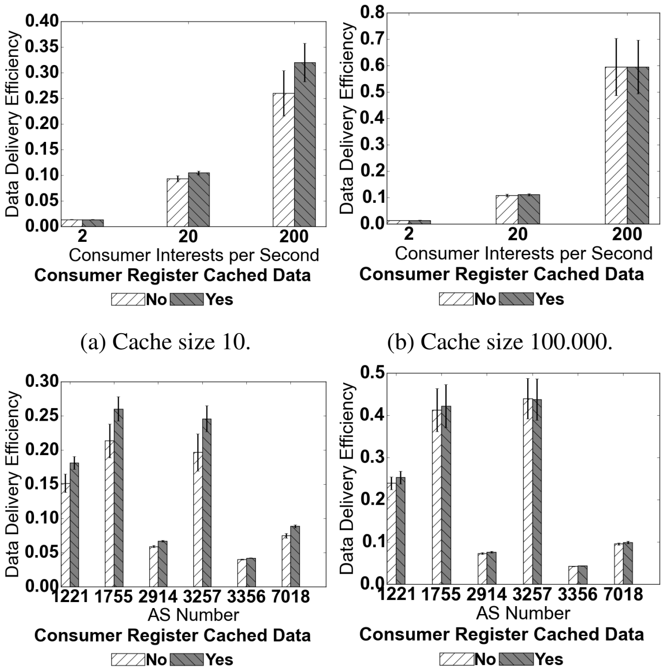
Figure 11: CRoS-NDN data delivery efficiency increase with consumer registration of data copies for consumer rates of 20 (figures a and c) and 100 (figures b and d) interests per second.

Figure 11 compares the data delivery efficiency for the CRoS-NDN scheme with and without registration of content copies stored on consumers. Consumer nodes have unlimited cache capacity and routers have a limited cache capacity. Each consumer requests the same content sequence for 20 seconds and stops. A new consumer starts at every 20 seconds. In the scenario with consumer registration of content copies, when the consumer stops, it registers the content copy location at the controller. The controller routes the interests to the closest registered copy [5].

The efficiency gain with consumer registration of content copies depends on router cache capacity and on the number of requested data. When routers have higher caching capacity than the requested data, registering content copies has no efficiency gain. Otherwise, when routers have smaller caching capacity than the requested data, registering content copies has a measurable efficiency gain. Higher is the consumer interest rate, higher is the number of requested content items and higher is efficiency gain for the same cache size. Figures 11a and 11b compare the efficiency for consumer rates of 20 and 100 interests per second respectively. Figures 11c and 11d show the efficiency with error bars for the 25^o consumer in the same simulation. Additionally, the efficiency increases with the consumer rate because the Hello rate is fixed in 0.1 interests per second.

Figure 12 reinforces the efficiency gain of CRoS-NDN with

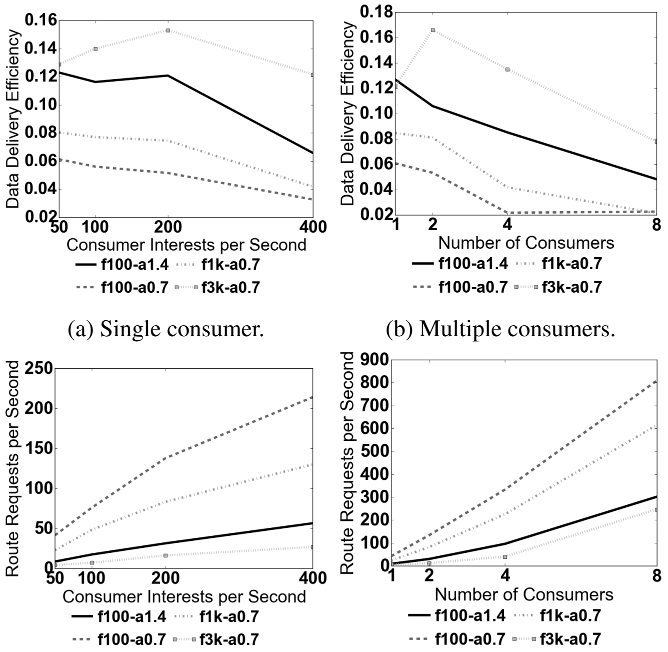
[5]Like in BitTorrent, one can modify the strategy to distribute interests among the copies instead of sending interests just to the closest one. The BitTorrent strategy speed up the content transfer when consumer access link has higher capacity than the access link of each content copy individually.



(a) Cache size 10. (b) Cache size 100.000.

(c) Cache size 10. (d) Cache size 100.000.

Figure 12: CRoS-NDN data delivery efficiency increase with consumer registration of data copies for cache sizes of 10 (figures a and c) and 100.000 (figures b and d).



(a) Single consumer. (b) Multiple consumers.

(c) Route requests received by the controller for single consumer. (d) Route requests received by the controller for multiple consumers.

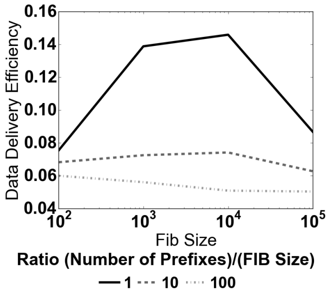
Figure 13: CRoS-NDN data delivery efficiency for consumer interests following the Zipf-Mandelbrot distribution for content prefix popularity. The simulation employs 3000 prefixes, FIB memory size of 100, 1000, 3000 entries (for f100, f1k, and f3k, respectively), and the Zipf α parameter values of 0.7 and 1.4 (for a0.7 and a1.4 respectively). Figures a and c consider a single consumer and a growing rate of consumer interests. Figures b and d consider multiple consumers and a fixed rate 50 interest per second per consumer.

registration of content copies stored on consumers over no registration of copies, when router cache size is insufficient for the requested data. Figure 12a shows that the highest efficiency gain occurs for the highest consumer rate (200) and a small cache size with 10 entries. Figure 12b shows no gain for consumer rate of 200 interests per second and a large cache size with 100,000 entries. Figures 12c and 12d show equivalent results in different topologies for cache sizes of 10 and 100,000 entries respectively.

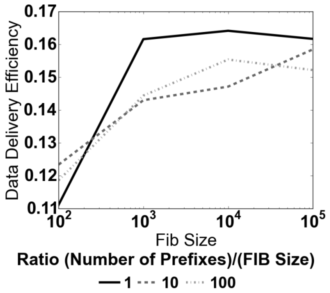
Announcing content copies location allows consumers to reach a closer copy that is outside the path to the producer. CRoS-NDN shows a low overhead for the registration of content copies location. This is in opposition to OSPFLike and NLSRLike that show poor performance when the rate of prefix announcements increases. Real traffic presents a long tail distribution for the content prefix popularity and the limited cache size of routers along the path to producer aggregates only repeated requests inside a small time window. We envision that the registration of content copies location is a potential solution for CDN over NDN. A router can proxy interest for specific prefixes and cache the respective data closer to potential consumers for longer time windows. Additionally, our scheme enables a form of peer-to-peer content distribution for NDN.

In the last set of simulations, we evaluate our proposal CRoS-NDN with consumers requesting content with a Zipf-Mandelbrot distribution for the prefix popularity. We consider constrained FIB memory, a growing rate of consumer interests, and short/long tail for the popularity distribution of content prefixes. We demonstrate that the efficiency decreases when the tail of the prefix popularity distribution increases and there is insufficient memory for the most of the available prefixes. In this case, the efficiency decreases for three reasons. The first reason is the higher rate of route request to controller that, in the worst case, halves the efficiency with one route request per consumer interest. The second reason relates to FIB and PIT association, an intrinsic characteristic of ndnSIM simulator that erases PIT entries on removal of corresponding FIB entry and, then, it leads to additional repeated interests from consumer for unanswered requests. The last reason is link congestion at higher consumer rates that, in the worst case, can congest the controller access link and causes additional interest retransmission.

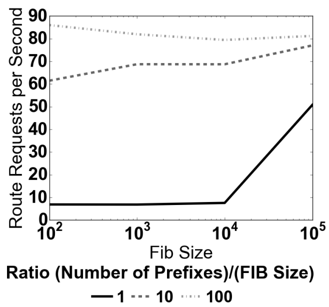
The longer is the tail of the prefix popularity distribution, *i.e.* lower α parameter of the Zipf distribution, the higher is both the rate of FIB match failures and the rate of route requests to controller when the FIB memory is insufficient for all content prefixes. Therefore, in consequence, the efficiency decreases. We choose the number of prefixes (3000), the FIB size (100, 1000, and 3000 entries), and the α values (0.7 and 1.4) in order to explicit this behavior. Figure 13a shows the efficiency with a single consumer and a growing rate of consumer interests per second. Figure 13b shows the efficiency with a growing number of consumers and each consumer with a fixed rate of 50 interest per second. Figures 13c and d show the rate of route requests received by the controller for single consumer and multi-



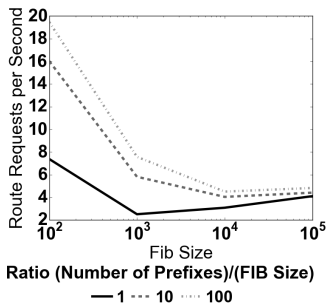
(a) Zipf α = 0.7.



(b) Zipf α = 1.4.



(c) Route requests received by the controller for Zipf α = 0.7.



(d) Route requests received by the controller for Zipf α = 1.4.

Figure 14: CRoS-NDN data delivery efficiency for the ratio of number of prefixes to FIB size. Consumer interests follow the Zipf-Mandelbrot distribution for content prefix popularity.

ple consumers cases, respectively. The higher rate of consumer interests causes higher rates of route requests. Furthermore, a higher number of consumers with the same prefix popularity distribution causes an aggregated prefix popularity distribution with longer tail, and, therefore, it decreases the efficiency due to a high rate of route requests. It is worth to observe that the aggregated rate of consumer interests is equal in Figures 13a, b, c and d. In addition, for small FIB size (100) and high rate of route requests, the FIB entry time in memory is lower than the round trip time and, thus, the early removal of a FIB entry and the associated PIT entries reduce the efficiency because of repeated route requests for the same prefix.

Figure 14 shows CRoS-NDN scalability and efficiency robustness for 3 orders of magnitude ratios of number of prefixes to FIB size. In addition, the results consider 4 orders of magnitude in the FIB size. In this scenario, a single consumer requests content with 100 interests per second. Higher is the number of prefixes to FIB size ratio and higher is the Zipf α parameter, then lower is the efficiency. It is worth to note that the higher is the number of prefixes, the lower is the ratio of requested prefixes to all prefixes considering a fixed time window and a fixed rate of consumer interests. Therefore, the efficiency decreases (stabilizes) for $\alpha = 0.7$ ($\alpha = 1.4$) with higher number of prefixes due to the limited simulation time.

Figures 13 and 14 point the CRoS-NDN potential bottleneck at the controller access link. The rate of route requests increases when there is insufficient FIB memory for the most of the solicited prefixes due to the long tail shape of prefix popularity distribution at core routers. In this scenario, the controller access link congests and causes interests retransmissions. The additional interests further reduce the efficiency. We plan to combine the Zhang tunneling approach for NDN with our CRoS-NDN scheme to overcome this bottleneck [47]. We argue, subject to further study, that the combined solution maintains our scheme features and reduces the rate of route requests to controller requiring less FIB memory at core routers for prefix popularity distributions with long tail at the network core.

7. Conclusion

We presented and analyzed the performance of the Controller-based Routing Scheme for Named-Data Networking (CRoS-NDN). Our proposal employs the same interest and data packets defined by Named-Data Networking (NDN), and, therefore, it preserves the original NDN features. CRoS-NDN is composed of two phases: Bootstrap phase, which monitors the nodes and assures the knowledge of the global network topology, and Named-Data Routing phase, which assures the localization and access to the requested content. The proposal fits well for data-center based network infrastructure that consolidates the network vision and offers the required storage and processing resources. The controller stores the contents, calculates routes from the consumer to the producer and its network global view helps to avoid unnecessary message overhead, providing an efficient data delivery with low delay. Moreover, our scheme splits content names from content localization and, thus, content can be consumed from any location. This content placement flexibility brings the known feature benefits provided by peer-to-peer and CDN networks that place content copies closer to consumers to decrease delivery latency and, additionally, it improves content mobility efficiency.

We derived lower bound analytical expressions for the efficiency and upper bound ones for the content delivery delay of our proposal and other known routing schemes. We employed the obtained expressions to discuss the limitation factors of each scheme. Furthermore, we evaluate and compare these schemes with simulations to validate and extend the analytical analysis.

The analytical analysis and the simulation results show that CRoS-NDN has the best performance for a set of scenarios and more robust performance over a wider range of scenarios, while the other schemes only show a high efficiency for limited ranges. CRoS-NDN shows a stable efficiency with the number of available prefixes while OSPFLike efficiency quickly decreases. CRoS-NDN shows a more robust efficiency with the limitation of FIB memory while ARPLike efficiency abruptly decreases. CRoS-NDN shows a better efficiency with producer mobility while the other schemes show a stronger efficiency reduction. CRoS-NDN shows a competitive delay in comparison with ARPLike and OSPFlike. Furthermore, we show that NLSRLike has a higher convergence delay besides the higher amount of required resources for each router.

The simulation results show that CRoS-NDN is resilient to link failure recovery. Our proposal presents a fast convergence because of the rapid exchange of update messages between the controller and routers signaling the fault and updating new routes, while the NLSRLike convergence is accomplished in hop-by-hop fashion. In addition, CRoS-NDN shows an economical use of computational resources for a growing number of prefixes.

The evaluation demonstrates that CRoS-NDN registration of content copies location improves the efficiency over the cache only along the path approach. The analysis shows the lower is the cache size at routers, the higher is the efficiency gain due to the registration of copies at consumers. Additionally, the evaluation validates the results in a set of topologies with different sizes and diameters.

The results points that CRoS-NDN has a potential bottleneck at the access link to controller when content requests show prefix popularity distribution with long tail and the core routers FIB memory is insufficient for the most of momently requested prefixes. We identified a potential solution combining our CRoS-NDN scheme with a tunneling approach. For future work, we will explore this solution and we also intend to test the proposal in Future Internet Testbed with Security (FITS) [32] with multi-controller taking care of domains.

Acknowledgment

This work was supported by PETROBRAS, FINEP, FUNTTEL, CNPq, CAPES, and FAPERJ.

References

[1] L. Zhang, A. Afanasyev, J. Burke, V. Jacobson, k. claffy, P. Crowley, C. Papadopoulos, L. Wang, B. Zhang, Named Data Networking, SIGCOMM Comput. Commun. Rev. 44 (3) (2014) 66–73. doi:10.1145/2656877.2656887.

[2] C. Yi, A. Afanasyev, L. Wang, B. Zhang, L. Zhang, Adaptive Forwarding in Named Data Networking, SIGCOMM Comput. Commun. Rev. 42 (3) (2012) 62–67. doi:10.1145/2317307.2317319.

[3] C. Yi, A. Afanasyev, I. Moiseenko, L. Wang, B. Zhang, L. Zhang, A Case for Stateful Forwarding Plane, Comput. Commun. 36 (7) (2013) 779–791. doi:10.1016/j.comcom.2013.01.005.

[4] G. Carofiglio, V. Gehlen, D. Perino, Experimental Evaluation of Memory Management in Content-Centric Networking, in: Communications (ICC), 2011 IEEE International Conference on, 2011, pp. 1–6. doi:10.1109/icc.2011.5962739.

[5] D. Perino, M. Varvello, A Reality Check for Content Centric Networking, in: Proceedings of the ACM SIGCOMM Workshop on Information-centric Networking, ICN '11, ACM, New York, NY, USA, 2011, pp. 44–49. doi:10.1145/2018584.2018596.

[6] S. K. Fayazbakhsh, Y. Lin, A. Tootoonchian, A. Ghodsi, T. Koponen, B. Maggs, K. Ng, V. Sekar, S. Shenker, Less Pain, Most of the Gain: Incrementally Deployable ICN, SIGCOMM Comput. Commun. Rev. 43 (4) (2013) 147–158. doi:10.1145/2534169.2486023.

[7] J. V. Torres, O. C. M. B. Duarte, An Autonomous and Efficient Controller-based Routing Scheme for Networking Named-Data Mobility, Tech. rep., Electrical Engineering Program, COPPE/UFRJ (April 2016).

[8] R. Mijumbi, J. Serrat, J. L. Gorricho, N. Bouten, F. D. Turck, R. Boutaba, Network Function Virtualization: State-of-the-Art and Research Challenges, IEEE Communications Surveys Tutorials 18 (1) (2016) 236–262. doi:10.1109/COMST.2015.2477041.

[9] 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, HotNets-X, ACM, 2011, pp. 1:1–1:6.

[10] A. Detti, M. Pomposini, N. Blefari-Melazzi, S. Salsano, Supporting the Web with an Information Centric Network That Routes by Name, Comput. Netw. 56 (17) (2012) 3705–3722. doi:10.1016/j.comnet.2012.08.006.

[11] Cisco, Cisco Visual Networking Index: Forecast and Methodology, 2014-2019. White Paper. (may 2015).

[12] Cisco, The Zettabyte Era—Trends and Analysis. White Paper. (jun 2015).

[13] M. Bari, S. Chowdhury, R. Ahmed, R. Boutaba, B. Mathieu, A survey of naming and routing in information-centric networks, Communications Magazine, IEEE 50 (12) (2012) 44–53. doi:10.1109/MCOM.2012.6384450.

[14] B. Ahlgren, C. Dannewitz, C. Imbrenda, D. Kutscher, B. Ohlman, A survey of information-centric networking, Communications Magazine, IEEE 50 (7) (2012) 26–36. doi:10.1109/MCOM.2012.6231276.

[15] G. Xylomenos, C. Ververidis, V. Siris, N. Fotiou, C. Tsilopoulos, X. Vasilakos, K. Katsaros, G. Polyzos, A Survey of Information-Centric Networking Research, Communications Surveys Tutorials, IEEE 16 (2) (2014) 1024–1049. doi:10.1109/SURV.2013.070813.00063.

[16] M. Zhang, H. Luo, H. Zhang, A Survey of Caching Mechanisms in Information-Centric Networking, IEEE Communications Surveys Tutorials 17 (3) (2015) 1473–1499. doi:10.1109/COMST.2015.2420097.

[17] C. Fang, F. R. Yu, T. Huang, J. Liu, Y. Liu, A Survey of Green Information-Centric Networking: Research Issues and Challenges, IEEE Communications Surveys Tutorials 17 (3) (2015) 1455–1472. doi:10.1109/COMST.2015.2394307.

[18] M. M. S. Soniya, K. Kumar, A survey on named data networking, in: Electronics and Communication Systems (ICECS), 2015 2nd International Conference on, 2015, pp. 1515–1519. doi:10.1109/ECS.2015.7124841.

[19] G. Carofiglio, G. Morabito, L. Muscariello, I. Solis, M. Varvello, From Content Delivery Today to Information Centric Networking, Comput. Netw. 57 (16) (2013) 3116–3127. doi:10.1016/j.comnet.2013.07.002.

[20] G. Carofiglio, M. Gallo, L. Muscariello, On the Performance of Bandwidth and Storage Sharing in Information-centric Networks, Comput. Netw. 57 (17) (2013) 3743–3758. doi:10.1016/j.comnet.2013.08.018.

[21] T. Koponen, M. Chawla, B.-G. Chun, A. Ermolinskiy, K. H. Kim, S. Shenker, I. Stoica, A Data-oriented (and Beyond) Network Architecture, SIGCOMM Comput. Commun. Rev. 37 (4) (2007) 181–192. doi:10.1145/1282427.1282402.

[22] P. Jokela, A. Zahemszky, C. Esteve Rothenberg, S. Arianfar, P. Nikander, LIPSIN: Line Speed Publish/Subscribe Inter-networking, SIGCOMM Comput. Commun. Rev. 39 (4) (2009) 195–206. doi:10.1145/1594977.1592592.

[23] A. Carzaniga, M. Papalini, A. L. Wolf, Content-based Publish/Subscribe Networking and Information-centric Networking, in: Proceedings of the ACM SIGCOMM workshop on Information-centric networking, ICN '11, ACM, 2011, pp. 56–61.

[24] R. Ahmed, F. Bari, S. R. Chowdhury, G. Rabbani, R. Boutaba, B. Mathieu, Alpha Route: Routing on Names, IEEE/ACM Transactions on Networking PP (99) (2016) 1–1. doi:10.1109/TNET.2015.2506617.

[25] W. Louati, W. Ben-Ameur, D. Zeghlache, A bottleneck-free tree-based name resolution system for Information-Centric Networking, Computer Networks 91 (2015) 341 – 355. doi:http://dx.doi.org/10.1016/j.comnet.2015.08.024.

[26] A. Afanasyev, C. Yi, L. Wang, B. Zhang, L. Zhang, SNAMP: Secure namespace mapping to scale NDN forwarding, in: Computer Communications Workshops (INFOCOM WKSHPS), 2015 IEEE Conference on, 2015, pp. 281–286. doi:10.1109/INFCOMW.2015.7179398.

[27] A. Baid, T. Vu, D. Raychaudhuri, Comparing alternative approaches for networking of named objects in the future Internet, in: Computer Communications Workshops (INFOCOM WKSHPS), 2012 IEEE Conference on, 2012, pp. 298–303. doi:10.1109/INFCOMW.2012.6193509.

[28] C. Dannewitz, M. D'Ambrosio, V. Vercellone, Hierarchical DHT-based Name Resolution for Information-centric Networks, Comput. Commun. 36 (7) (2013) 736–749. doi:10.1016/j.comcom.2013.01.014.

[29] M. Caesar, T. Condie, J. Kannan, K. Lakshminarayanan, I. Stoica, S. Shenker, ROFL: Routing on Flat Labels, SIGCOMM Comput. Commun. Rev. 36 (4) (2006) 363–374. doi:10.1145/1151659.1159955.

[30] N. McKeown, T. Anderson, H. Balakrishnan, G. Parulkar, L. Peterson, J. Rexford, S., J. Turner, OpenFlow: Enabling Innovation in Campus Networks, ACM SIGCOMM Computer Communication Review 38 (2) (2008) 69–74.

[31] D. Mattos, N. Fernandes, V. da Costa, L. Cardoso, M. Campista, L. Costa, O. Duarte, OMNI: OpenFlow MaNagement Infrastructure, in: Network of the Future (NOF), 2011 International Conference on the, 2011, pp. 52

–56.
[32] I. M. Moraes, D. M. Mattos, L. H. G. Ferraz, M. E. M. Campista, M. G. Rubinstein, L. H. M. Costa, M. D. de Amorim, P. B. Velloso, O. C. M. Duarte, G. Pujolle, FITS: A flexible virtual network testbed architecture, Computer Networks 63 (2014) 221 – 237. doi:http://dx.doi.org/10.1016/j.bjp.2014.01.002.
[33] N. Fernandes, M. Moreira, I. Moraes, L. Ferraz, R. Couto, H. Carvalho, M. Campista, L. Costa, O. Duarte, Virtual Networks: Isolation, Performance, and Trends, Annals of Telecommunications 66 (5-6) (2011) 339–355. doi:10.1007/s12243-010-0208-9.
[34] C. E. Rothenberg, M. R. Nascimento, M. R. Salvador, C. N. A. Corrêa, S. Cunha de Lucena, R. Raszuk, Revisiting routing control platforms with the eyes and muscles of software-defined networking, in: Proceedings of the first workshop on Hot topics in software defined networks, HotSDN '12, ACM, New York, NY, USA, 2012, pp. 13–18. doi:10.1145/2342441.2342445.
[35] W. Shi, A. Afanasyev, RepoSync: Combined action-based and data-based synchronization model in Named Data Network, in: Computer Communications Workshops (INFOCOM WKSHPS), 2015 IEEE Conference on, 2015, pp. 275–280. doi:10.1109/INFCOMW.2015.7179397.
[36] S. Gao, Y. Zeng, H. Luo, H. Zhang, Scalable control plane for intra-domain communication in software defined information centric networking, Future Generation Computer Systems 56 (2016) 110 – 120. doi:http://dx.doi.org/10.1016/j.future.2015.10.017.
[37] S. Salsano, N. Blefari-Melazzi, A. Detti, G. Morabito, L. Veltri, Information Centric Networking over SDN and OpenFlow: Architectural Aspects and Experiments on the OFELIA Testbed, Comput. Netw. 57 (16) (2013) 3207–3221. doi:10.1016/j.comnet.2013.07.031.
[38] R. Atkinson, S. Bhatti, Identifier-Locator Network Protocol (ILNP) Architectural Description, RFC 6740, RFC Editor (November 2012).
[39] I. Psaras, W. K. Chai, G. Pavlou, In-Network Cache Management and Resource Allocation for Information-Centric Networks, IEEE Transactions on Parallel and Distributed Systems 25 (11) (2014) 2920–2931. doi:10.1109/TPDS.2013.304.
[40] V. Jacobson, D. K. Smetters, N. H. Briggs, M. F. Plass, P. Stewart, J. D. Thornton, R. L. Braynard, VoCCN: Voice-over Content-centric Networks, in: Proceedings of the 2009 Workshop on Re-architecting the Internet, ReArch '09, ACM, New York, NY, USA, 2009, pp. 1–6. doi:10.1145/1658978.1658980.
[41] R. Chiocchetti, D. Perino, G. Carofiglio, D. Rossi, G. Rossini, INFORM: A Dynamic Interest Forwarding Mechanism for Information Centric Networking, in: Proceedings of the 3rd ACM SIGCOMM Workshop on Information-centric Networking, ICN '13, ACM, New York, NY, USA, 2013, pp. 9–14. doi:10.1145/2491224.2491227.
[42] A. K. M. M. Hoque, S. O. Amin, A. Alyyan, B. Zhang, L. Zhang, L. Wang, NLSR: Named-data Link State Routing Protocol, in: Proceedings of the 3rd ACM SIGCOMM Workshop on Information-centric Networking, ICN '13, ACM, New York, NY, USA, 2013, pp. 15–20. doi:10.1145/2491224.2491231.
[43] A. Afanasyev, I. Moiseenko, L. Zhang, ndnSIM: NDN simulator for NS-3, Tech. rep., University of California, Los Angeles (2012).
[44] A. Afanasyev, P. Mahadevan, I. Moiseenko, E. Uzun, L. Zhang, Interest flooding attack and countermeasures in Named Data Networking, in: IFIP Networking Conference, 2013, 2013, pp. 1–9.
[45] N. Spring, R. Mahajan, D. Wetherall, T. Anderson, Measuring ISP Topologies with Rocketfuel, IEEE/ACM Trans. Netw. 12 (1) (2004) 2–16. doi:10.1109/TNET.2003.822655.
[46] J. Moy, OSPF Version 2, RFC 2328, RFC Editor (April 1998).
[47] G. Zhang, X. Chen, Y. Lu, Q. Gao, Enabling Tunneling in CCN, IEEE Communications Letters 20 (1) (2016) 149–152. doi:10.1109/LCOMM.2015.2500899.