



Leveraging Replication in Content-Centric Networks

Prométhée Spathis, Salah Eddine Belouanas, Kim-Loan Thai

► **To cite this version:**

Prométhée Spathis, Salah Eddine Belouanas, Kim-Loan Thai. Leveraging Replication in Content-Centric Networks. Global Information Infrastructure and Networking Symposium, Oct 2015, Guadalajara, Mexico. hal-01206355

HAL Id: hal-01206355

<https://hal.archives-ouvertes.fr/hal-01206355>

Submitted on 1 Oct 2015

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Leveraging Replication in Content-Centric Networks

Prom  th  e Spathis, Salah-Eddine Belouanas and Kim-Loan Thai
LIP6/CNRS – UPMC Sorbonne Universities – Paris, France
{promethee.spathis, salah-eddine.belouanas, kim.thai}@lip6.fr

Abstract—This paper focuses on the communication functionalities of content-centric networks. More especially, we take advantage of the widespread availability of content as today data objects may be replicated or moved at several locations to minimize content access time or to provide resiliency. We propose a scalable interdomain routing architecture designed to meet the needs of data-intensive applications. Our protocol complements CCN’s routing framework with the ability to discover routes that allow users to retrieve replicated content from the most efficient location. The distinctive feature of our approach lies in the use of the caching capabilities of content routers to store and maintain the routes discovered towards the closest replicas. Our simulation results show an increase in content access efficiency by minimizing retrieval time and by maximizing the bandwidth demand.

I. INTRODUCTION

In the last years, there has been a revival of interest in content-centric networking that have lead to the proposal of several architectures following popular designs as [1], [2], [3]. Motivated by the ability to access content in a location-independent manner, the common underlying paradigm to these architectures calls for a shift in the way communication takes place among network entities. The emphasis is given to the content rather than the location. As such, the principles behind the design of content-centric architectures fit within the broader framework of Information-Centric Networking (ICN) [4] which is also motivated by the challenges arising from the current end-to-end host-centric design of the Internet.

The most relevant ICN architectures include the seminal papers of Gritterand and Cheriton [1] and Carzaniga et al. [5], NSF projects like NDN [6] and MobilityFirst[7], the European projects PSIRP [8], PURSUIT [9], 4WARD [10], COMET [11], and SAIL [12], as well as the French funded project ANR Connect [13] which adopts the NDN architecture. ICN architectures address research issues such as naming, in-network caching, security and mobility [14], [15], [16], [17]. Compared to ICNs, the design of content-centric architectures has a more practical purpose as they attempt to meet the data-intensive applications needs by improving content delivery performance and reduce traffic overhead.

The high point of the regain of interest in content-centric networking came with the appearance of the CCN proposal [18]. Instead of replacing IP in a clean-state approach, the guiding principles of CCN design follow a more conventional approach that aims to show that removing addresses from IP does not call for revisiting IP’s engineering principles. The key objective behind CCN design is to investigate whether

the Internet can evolve with few adjustments from a design paradigm that have been exclusively host-centric to a content-centric paradigm. As a result, the key choices that have driven the design of CCN were taken to enable the reuse of well-tried mechanisms and techniques directly borrowed from IP. Examples of the latters include DNS naming syntax, CIDR prefix aggregation, and longest prefix match forwarding.

In this paper, we focus on the communication functionalities of content-centric networks. More especially, we propose the design of a routing architecture that fully benefits from the widespread availability of content in today’s large scale networks. Replication or caching have long become common practice amongst service providers who can improve access efficiency by minimizing retrieval time while reducing bandwidth costs.

The main goal of our work is to complement the existing communication framework of CCN with a scalable interdomain routing protocol able to discover routes that allow users to retrieve replicated content from the most efficient location. We adopt an Interest-based approach for content discovery supported by the caching capabilities of content routers. To handle content replication, we borrow from the literature [19] the idea of using flat unstructured identifiers in replacement of CCN DNS-based names to avoid the overloaded semantics of URLs.

In summary, the contributions of our work are as follows:

- **Interdomain content routing** We describe a scalable discovery scheme of routes towards the most efficient replicas for a content item;
- **Route caching** We show the benefit of using the caching capabilities of content routers to store routes discovered for replicated content;
- **Flat Namespace** We introduce an unstructured location-independent naming space compliant with the replication or migration of content.

The remainder of this paper is structured as follows. In Section II, we present the main features of CCN with regards to naming, routing, and forwarding. Section III provides the detailed description of our routing architecture including the namespace, the node model as well as the types of packets, and finally our routing algorithm. Results of the simulation experiments are presented in Section IV. Finally, the paper is concluded by Section V.

II. CCN ROUTING FRAMEWORK

In this section, we first present the main CCN architectural features with regards to naming, routing, and forwarding. We

then draw on the basic principles underlying the design of our routing solution for content centric networks.

1) *CCN namespace*: CCN name syntax is inspired by the naming scheme of DNS. A CCN name is designed as a URL-like chain of characters with the aim of applying prefix aggregation to the hierarchical structure of domain names. However, the hierarchical structure of DNS names is motivated by the need to distribute the responsibility of assigning domain names. With the current lack of specific allocation policy and management for CCN namespace, the reuse of DNS-based names is expected to result in a mismatch between the name distribution and the underlying network topology which will defy the benefits of hierarchical aggregation. The use of URL-like names also has the drawback of making content replication and migration challenging. In our work, we adopt a flat namespace consisting of unstructured, location-independent, and human-unreadable identifiers to avoid broken links and the use of HTTP redirects which requires the cooperation of the originating administrative domain.

2) *Content names distribution*: As depicted in Fig. 1, CCN uses the standard OSPF link-state database flooding-based update mechanism to advertise CCN content name prefixes, throughout the routing domain. OSPF has been extended with opaque LSAs that include a field for some application specific-information. CCN uses the opaque capabilities of OSPF to distribute CCN-specific information i.e. content name prefixes. The CCN routing application attached to the routers can use the opaque LSAs to transparently advertise the content name prefixes throughout the routing domain. The distribution process of CCN-specific information is transparent to the OSPF routing process but can be used by the CCN routing application. Though link-state flooding has been proved to improve routing convergence in medium to large size enterprise or campus networks, OSPF cannot be extended so as to provide interdomain connectivity in large-scale networks. In our work, we leverage the multiprotocol extensions of BGP (i.e. MP-BGP) to carry routing information for the CCN layer.

3) *CCN forwarding*: CCN applies the concept of aggregation introduced with CIDR addressing to the context of URL-based content names. From the use of aggregation as an already well-tried technique, CCN forwarding is expected to achieve scalability. Introduced in the Internet to efficiently manage the limited addressing space and control the growth in size of the forwarding tables, CIDR address allocation scheme achieves scalability by complying with the location of nodes in the network topology. However, CIDR-like aggregation has the drawback of requiring routers to perform longest-prefix-match forwarding which makes of fast lookup schemes a challenging design issue for CCN routers. Another issue for CCN forwarding concerns how content items are selected when the requested content is replicated at several locations. Careful attention has to be given to the selection of the paths provided in the FIB to each replica. A suggested strategy consists in broadcasting the request to all replicas which poses the problem of receiving multiple copies of the same content item.

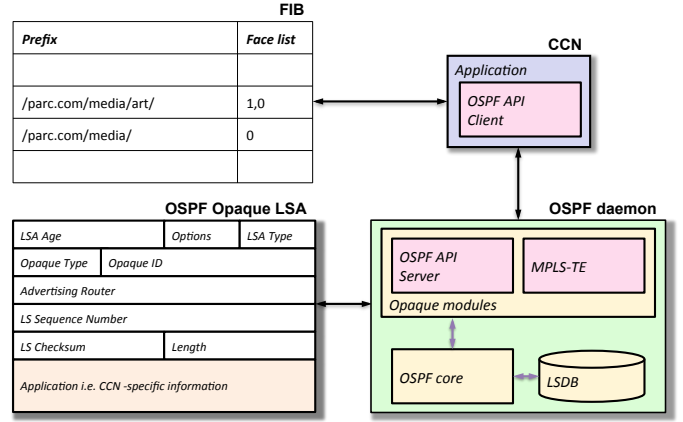


Fig. 1: CCN communication framework

III. ARCHITECTURE DETAILS

A design objective common to most content-centric architectures is to improve content delivery performance and reduce traffic overhead. To achieve this objective, we claim that content-based routing should take benefit from the widespread availability of content in large scale networks as data objects may be replicated and cached at several places. By giving emphasis on the content rather than the location, content-centric networking renders binding of content to endpoints obsolete when providing access to the content. The paradigm underlying content-centric networking thus gives a natural way to make effective use of content replication since end-to-end connectivity to the source of a replica is not required prior to access the data. As such, content-centric networks naturally provide network-level support for discovery of replicated data items.

We adopt an on-demand interdomain flat name-based routing protocol that integrates naming to routing for effective location of content replicas based on network metrics while providing resilience to network failures or content migration. Our approach overcomes the issue of scalability inherent to routing on flat names by making use of the caching capabilities of content routers for storing previously learnt routes.

The remainder of this section details how our architecture works. We first present our content identifier namespace, the node model as well as the data structures and the types of messages, and then our routing algorithm.

A. Content Identifier CID

We introduce a flat namespace consisting of unstructured, location-independent, and human-unreadable identifiers called *Content Identifiers* (CIDs). The CIDs are said to be semantic-free since they provide the ability to refer to content and services independently of the hosting endpoint regardless of its administrative domain, network location, or the network topology. In contrast to the CCN DNS-based names, content can be replicated or migrate between hosts while avoiding broken links or using HTTP redirection which requires control of former domain, increases traffic, and latency.

The value of CIDs might be a 128-bit long number chosen randomly or could be derived from the cryptographic hash of the public key of the content owner and the content itself for authentication and integrity respectively. The granularity of naming is flexible since a name can refer to a host, a content within it, or at finer granularity any content item with no reference to the hosting endpoint. At the user-level, it is assumed that a mapping service provides applications with the CIDs corresponding to human readable descriptors such as search keywords which in this case will require the mapping service to be carried out by search engines.

B. Message Types

CCN has two types of messages: Interest and Data messages. The Interest message is used to request data by name. The Data message is used to supply data. In CCN, the Interest message contains the DNS-based name prefix of the content item to be retrieved while the requested content is carried in a Data message identified by a name matching the prefix provided in the Interest. In our routing scheme, the Interest and Data messages now contain a CID field instead of the CCN DNS-based names.

We also extend both types of messages with a field borrowed to BGP update messages. This new field contains a path-vector similar to the BGP AS Path attribute which includes the list of Autonomous System Numbers that describes the sequence of ASes through which the Interest has passed. The path-vector is used to prevent Interests from looping and is returned to the querier in the Data message carrying the requested content item. All subsequent Interests will be directed towards this destination CR of the routing domain where the content was found. The path-vector is also used by content routers to aggregate the Interest messages requesting the same content. A content router stores the shortest path vector followed by previous occurrences of an Interest and only forwards a subsequent one if received along a shorter route. An Interest message is also scoped by a TTL which is initialized to the maximum number of ASes the Interest message can visit.

C. Data Structures

We complete the operation of a CCN node by introducing an additional data structure to the proposed node model. In addition to the three existing structures, namely the FIB (Forwarding Information Base), the Content Store (buffer memory), and the PIT (Pending Interest Table), we introduce a Route Store (RS). The FIB is used to forward Interests toward the sources having a copy matching the requested content. The Content Store is a cache where content objects can be stored to satisfy future Interests. The PIT keeps track of the interfaces on which Interests for a specific content are forwarded. We extend the PIT entries with information collected from the Interest header fields.

The RS is used to keep in cache the routes discovered in response to previous Interests. Each RS entry contains a path-vector representing the list of ASes that the route traverses.

When a content router (CR) starts searching the best content replica with respect to its relative location, the CR sends an Interest either along the unicast path provided by BGP or will alternatively broadcast the Interest to all its peers in the neighboring ASes depending on the popularity of the content. If a route for the requested content is present in the RS, the CR will use this route to direct the Interest towards the domain where a replica of the content is known to be found.

We extend the PIT entries in order to avoid propagating multiple instances of the same Interest. Each PIT entry is extended with the identifier of the CR who initiated the search, the sequence number of the Interest, and the shortest path vector followed by previous occurrences of this Interest sharing the same originating CR and sequence number. As in CCN, the PIT is also used to aggregate multiple Data messages so as only a single copy will be forwarded towards the content requester.

D. Routing on Content IDs

In this section, we describe how the design of our name-based interdomain routing protocol supports the efficient discovery of some piece of content by using the plurality of alternative routes to access replicated content.

In contrast to CCN, Interests are propagated between neighboring domains in hope of finding the closest content replica. Our choice to avoid designing an advertisement-based protocol is motivated by the need of resiliency to content migration and network failures. Relying on periodic advertisements of routing information is expected to increase the impact on the network when transposed to highly replicated content.

1) *Domain*: We will refer to a *domain* as the administrative entity corresponding to a routing domain or an Autonomous System. The boundary of a domain may also be defined by reflecting other organizational considerations related to the ownership of content available within the domain boundary. As such a domain may refer to a campus, a company, or a content provider.

A domain distinguishes content depending whether it is the originator of the content or a purveyor of a copy of the content. A domain containing a local copy of a content considers this content as an internal content. A *home domain* is a specific case since it contains the original copy of a content item provided by the content owner himself/herself. We assume that an intradomain routing protocol creates the routes needed by local content routers to forward the Interests along the shortest path towards an internal content.

A content item is considered as external to a domain if a copy cannot be found locally. In the absence of a local copy, a domain can still have a route previously learnt for this content. If not, a border content router (BCR) will initiate a search by sending the Interest to its neighboring domains in the same way BGP propagates its update messages. Border content routers are configured to be BGP peers and as so, will establish a BGP session running on top of TCP prior to the exchange of the Interest messages throughout the search process. Inside

each domain, a BCR is in charge of propagating the Interests requesting an external content.

2) *Requesting an external content:* As depicted in Fig. 2, when an Interest for a piece of content replicated in other domains is received, the BCR first checks if the path-vector included in the Interest contains its own domain ID in which case the Interest will be discarded. In the absence of loops, the BCR appends the local domain ID (or alternatively, the AS Number or ASN) to the Interest path-vector and checks the PIT for an entry matching the incoming Interest, i.e. containing the same original requester and sequence number. A PIT-match indicates that a previous Interest has already been sent for the same content. The Interest is discarded if the path-vector included in the PIT entry is shorter than the one carried by the Interest.

Otherwise the PIT entry is updated with the Interest path-vector and arrival interface. If no PIT entry is found, a new PIT entry is created with the identifier of the BCR who initiated the search, the sequence number of the Interest, the path-vector it carries, and the face on which the Interest arrived. The BCR then looks in its Content Store for a matching content. If found, a Data message carrying the content will be returned to the requester by following the PIT entries left behind the Interest.

In case of no Content Store match, the BCR checks in its Route Store (RS) for a route recently learnt through a previous search. If such a route exists, the Interest is forwarded along this route towards the content. If no matching route can be found in the RS, the BCR decrements the TTL of the Interest message. If the Interest TTL did not reach zero, the BCR sends out the Interest to all its neighboring peers or along the unicast BGP route.

A timer is associated to each PIT entry to limit the rate of Interests and to handle unsuccessful searches. If no matching Data is received, the original requester is responsible for generating a new Interest distinguishable from previous occurrences thanks to an incremented sequence number.

3) *Responding with an internal content:* On receiving an Interest for an internal content, a BCR starts performing loop-detection on the Interest path vector in a similar way to BGP. The BCR will append the ID of its domain to the path-vector if loop-free. The resulting sequence of domains corresponds to the complete route between the originating BCR that initiated the search and the local replica of the content. This route is sent back to the originating BCR along with the copy of the local replica in a Data message. The Data message also inherits the Interest sequence number and includes the identifier of the replying BCR.

Upon receiving the Data message, a content router extracts the path vector contained in its header and stores the corresponding route in its Route Store along with the identifier of the BCR who originated the Data message. This path vector will be used to direct subsequent Interests requesting the same content. The CR also stores the content object carried by the Data message in its Content Store before forwarding the Data message on every interfaces listed in the PIT entry left by previous Interests. The PIT entry is erased once the Data

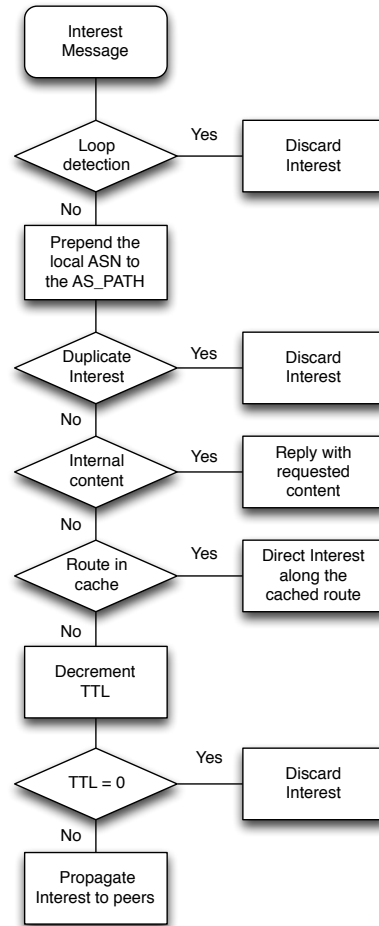


Fig. 2: Handling Interests

message forwarded as a way to prevent subsequent copies of the content from being propagated later.

The route contained in the Data message will be advertised within the domain of the original requester so that internal content routers may also learn how to reach the external content in case of subsequent Interests. This route will also be used so as to handle Interests received from external peer domains.

4) *Default routing for sparse content:* Since some content can exhibit a low degree of replication, we introduce an alternative using the unicast route towards the home domain of the requested content. In order to enable the use of CIDs for unicasting Interests, we use the multiprotocol extensions of BGP (i.e. MP-BGP) [20] to carry routing information for the CCN layer using CIDs.

We propose to define a new address family called CCN AF used to identify CCN as the network layer protocol carried by BGP. To make BGP advertise a set of CIDs as reachable destinations together with the next hop identifier to be used for forwarding along the feasible routes to these destinations, we use the CCN AF in combination with the two BGP attributes Multiprotocol Reachable NLRI (MP_REACH_NLRI) and Multiprotocol Unreachable NLRI

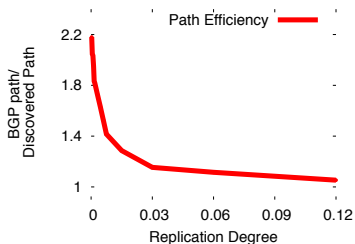


Fig. 3: Discovered route evaluation

(MP_UNREACH_NLRI) introduced to enable in BGP the multiprotocol extensions capability. Both attribute contains one or more triples that consists of the Address Family information, the Next Hop Information, and the Network Layer Reachability Information (NLRI).

In our protocol, the Address Family information identifies CCN as the network layer protocol that is being carried within the update. The Next-Hop Information is the identifier of the next content border router on the path to the destination. In our protocol, the NLRI is encoded as one or more 2-tuples with the following format: Length (1 octet): Total length of the CID advertised; CID: The content identifier.

IV. EVALUATION

A. Experimental Setup

In order to evaluate our proposal, we developed *ccnSim++*, an extension to the *ccnSim* simulator [21]. *ccnSim* is a chunk-level simulator built in C++ and developed on top of the *Omnet++*. Its ability to scale well in dense topologies allowed us to conduct our simulations on a large dataset made publicly available by the Cooperative Association for Internet Data Analysis (CAIDA). This dataset consists of a snapshot of more than 36,000 ASes and around six million announced BGP routes. In our simulations, we assume that all links have unlimited bandwidth and are not affected by congestion. All our tests were performed on a machine equipped with 144GB of RAM and a 24 core Intel Xeon processor.

Inside each Autonomous System, we suppose all content objects are forwarded efficiently according to an intradomain routing protocol. Border routers are considered to be content repositories and are responsible for handling Interest and Data messages. To simulate content placement and replication over the Internet, we distribute content replicas randomly across all repositories. The catalog we consider consists of 10^4 unique content objects.

ccnSim++ generates Interests messages according to a Zipf distribution. By keeping track of the number of Interests, every router is able to decide locally which content is popular. As better routes to closer content are only searched for popular content, we simulate the routes lifetime by implementing a LRU Route Store. When the cache fills up, the routes to repositories containing less popular content are removed and replaced with new ones.

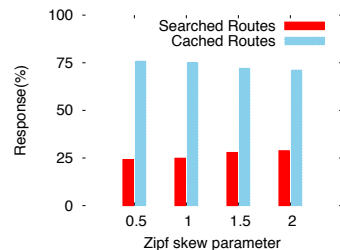


Fig. 4: Discovery process benefit

B. Route Discovery and Replication

Fig. 3 shows the length ratio of the BGP route to the original content over the routes discovered by our solution to a replica as a function of the replication degree of the content considered. We can see that the routes we discover outperform the routes to the original content as provided by BGP. This result indicates that our discovery scheme is able to always find the shortest path to the closest content replica. BGP has no way of detecting the repositories holding a replica of the requested content. Moreover, BGP policies may prevent the selection of the shortest path among the routes advertised toward the original content. The length ratio converges to 1 as the number of replicas increases. Due to the random distribution of all content replicas, the placement of replicas is not guaranteed as optimal with regard to the queriers. For large number of replicas, the distant replicas outbalance the replicas close to each querier. The BGP route provided for the original content shares similarities with the routes discovered to the replicas as their number increases and more replicas are well scattered across the network. In all other cases, our query-based content discovery scheme outperforms the BGP path.

C. The Influence of Popularity

In figure Fig. 4, we show the percentage of routes that are actually discovered when searching for a content or reused from a previous successful search as a function of the popularity of the content which follows a Zipf distribution. We can see that over than 75% of the routes used to retrieve a content are provided from the Route Store of the content routers. This shows the benefit of caching the routes discovered toward a replica as a result of requesting a content. The overhead of the discovery process is outweighed by the benefit of caching the routes to the replicas we discover. The benefit of caching routes toward previously discovered replicas can also be found in the number of Interest messages.

Fig. 5 shows the total number of Interest messages forwarded in the network as a function of the popularity of the content. Though the number of Interest messages issued for a content increases with the popularity of the content, we can see an opposite behavior when we increase the Zipf skew parameter. This indicates that Interest messages travel less distance in the network to discover a content as its popularity increases. Interest messages can draw on previous searches as content routers intercept them and reply with cached routes

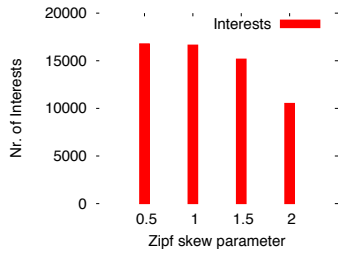


Fig. 5: Search messages overhead

learnt as a result of previous searches. It is also expected that the Data messages sent on a route fetched from a Route Store will travel along a shorter path, reducing this way the downstream bandwidth utilization and improving the latency for users retrieving the content.

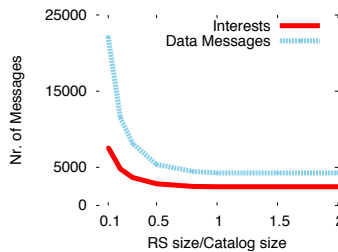


Fig. 6: Route caching influence

D. Route Store and Catalog Size

With a large catalog size, the route store is unable to hold all the best routes to the popular content. In Fig. 6, we evaluate the impact of the size of the content catalogue as we expect the limited storage of Route Stores to impact of the performance of our route discovery scheme. We vary the size of the Route Stores for different sizes of the content catalogue. We measure the resulting number of Interest and Data messages as a function of the ratio of the RS size over the catalogue. We can observe that for small increases in the size of the Route Stores, the number of messages generated by our architecture decreases significantly. For size ratios greater than 0.8%, this number of Interest and Data messages remains stable. This is due to the fact that the size of the route stores becomes large enough in comparison to the catalogue size to hold the routes discovered for all popular contents.

V. CONCLUSION

In this paper, we have presented a scalable on-demand routing protocol that takes advantage of the paradigm of content-centric networking to meet the needs of content related services and data-intensive applications. As part of our design process, we take advantage of the high availability of content in large-scale networks. The novelty of our approach lies in the use of the caching capabilities of content routers to store routes discovered for content replicas. Our protocol complements the CCN routing framework with the ability to discover efficient routes that allow users to retrieve replicas in a scalable fashion.

Our simulation results show an increase in access efficiency by minimizing retrieval time and by maximizing the bandwidth demand. As a future work, we aim to study the resilience of our protocol to the episodic availability of hosting nodes due to network failures or content migration.

ACKNOWLEDGMENT

This work was partially funded by the French National Research Agency (ANR) under project ANR DataTweet (ANR-13-INFR-0008).

REFERENCES

- [1] D. Cheriton and M. Gritter, "An architecture for content routing support in the internet," in *Proc. USENIX Symposium on Internet Technologies and Systems (USITS'01)*, San Francisco, CA, USA, Mar. 2001, pp. 4–4.
- [2] M. Caesar, T. Condie, J. Kannan, K. Lakshminarayanan, and I. Stoica, "Roff: routing on flat labels," in *Proc. Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications (SIGCOMM'06)*, Pisa, Italy, Oct. 2006, pp. 363–374.
- [3] T. Koponen, M. Chawla, B.-G. Chun, A. Ermolinskiy, K. H. Kim, S. Shenker, and I. Stoica, "A data-oriented (and beyond) network architecture," in *Proc. Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications (SIGCOMM'07)*, Kyoto, Japan, Sep. 2007, pp. 181–192.
- [4] G. Xylomenos, C. N. Ververidis, V. A. Siris, N. Fotiou, C. Tsilopoulos, X. Vasilakos, K. V. Katsaros, and G. C. Polyzos, "A survey of information-centric networking research," *IEEE Communications Surveys and Tutorials*, vol. 16, no. 2, pp. 1024–1049, 2014.
- [5] A. Carzaniga and A. L. Wolf, "Forwarding in a content-based network," in *Proc. Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications (SIGCOMM'03)*. Karlsruhe, Germany: ACM, Aug. 2004, pp. 163–174.
- [6] L. Zhang, D. Estrin, J. Burke, V. Jacobson, J. D. Thornton, D. K. Smetters, B. Zhang, G. Tsudik, K. Claffy, D. Krioukov, D. Massey, C. Papadopoulos, T. Abdelzaher, L. Wang, P. Crowley, and E. Yeh, "Named data networking (ndn) project," Tech. Rep. NDN-0001, Oct. 2010.
- [7] "Nsf mobility first project," <http://mobilityfirst.winlab.rutgers.edu/>.
- [8] "Fp7 psirp project," <http://www.psirp.org/>.
- [9] "Fp7 pursuit project," <http://www.fp7-pursuit.eu/PursuitWeb/>.
- [10] "Fp7 4ward project," <http://www.4ward-project.eu/>.
- [11] "Fp7 comet project," <http://www.comet-project.org/>.
- [12] "Fp7 sail project," <http://www.sail-project.eu/>.
- [13] "Anr connect project," <http://anr-connect.org>.
- [14] G. Carofiglio, G. Morabito, L. Muscariello, I. Solis, and M. Varvello, "From content delivery today to information centric networking," *Computer Networks*, vol. 57, no. 16, pp. 3116–3127, Nov. 2013.
- [15] M. Diallo, V. Sourlas, P. Flegkas, S. Fdida, and L. Tassioulas, "A content-based publish/subscribe framework for large-scale content delivery," *Computer Networks*, vol. 57, no. 4, pp. 924–943, 2013.
- [16] S. K. Fayazbakhsh, Y. Lin, A. Tootoonchian, A. Ghodsi, T. Koponen, B. Maggs, K. Ng, V. Sekar, and S. Shenker, "Less pain, most of the gain: Incrementally deployable icn," in *Proc. Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications (SIGCOMM'13)*. Hong Kong, China: ACM, 2013, pp. 147–158.
- [17] J. Kurose, "Information-centric networking: The evolution from circuits to packets to content," *Computer Networks*, vol. 66, pp. 112–120, 2014.
- [18] V. Jacobson, D. K. Smetters, J. D. Thornton, M. F. Plass, N. H. Briggs, and R. L. Braynard, "Networking named content," in *Proc. 5th International Conference on Emerging Networking Experiments and Technologies (CoNEXT'09)*, Rome, Italy, 2009, pp. 1–12.
- [19] M. Walfish, H. Balakrishnan, and S. Shenker, "Untangling the web from dns," in *Proc. USENIX Symposium on Networked Systems Design and Implementation (NSDI'04)*, San Francisco, CA, USA, Mar. 2004.
- [20] T. Bates, R. Chandra, D. Katz, and Y. Rekhter, "Multiprotocol extensions for bgp-4," Tech. Rep. 4760, Jan. 2007.
- [21] R. Chiochetti, D. Rossi, and G. Rossini, "ccnsim: An highly scalable ccn simulator," in *Proc. IEEE International Conference on Communications (ICC'13)*. IEEE, Jun. 2013, pp. 2309–2314.