Opportunistic Off-Path Content Discovery in Information-Centric Networks

Onur Ascigil*, Vasilis Sourlas*, Ioannis Psaras*, and George Pavlou*
*Department of Electronic and Electrical Engineering, University College London, UK.
Email: {o.ascigil, v.sourlas, i.psaras, g.pavlou}@ucl.ac.uk

Abstract—Recent research in Information-Centric Networks has considered various approaches for discovering content in the cache-enabled nodes of the network. Such approaches include scoped flooding and deploying a control plane protocol to disseminate the cache contents in the network, to name a few. In this work, we consider an opportunistic approach that uses trails left behind by data packets from the content origin to the sources in order to discover off-path cached content. We evaluate our approach using an ISP topology for various system parameters. We propose two new forwarding strategies built on top of our approach. Our results indicate that the opportunistic discovery mechanism can significantly increase cache hit rate compared to NDN's default forwarding strategy, while limiting the overhead at acceptable levels.

Index Terms—Information-Centric Networks, Off-path Caching, Crowded Networks, ndnSIM.

I. INTRODUCTION

The proliferation of content-centric applications in the Internet motivated a research trend towards content-oriented networking, which has been realized through the Information-Centric Networking (ICN) paradigm. In ICN, routing and forwarding is based on uniquely identified and authenticated content identifiers/names, rather than end-host addresses [1]. Thus, in ICN, content can be transparently delivered from any cache-enabled in-network device (*e.g.,* router), as long as this device holds a valid copy of the requested content and the request traverses it. Based on this principle, a new research field has emerged, named "in-network caching". In this paper, we use the abbreviations NDN to refer to the main architecture proposed in [2], whereas the term ICN is used to describe the generic information-centric architectural paradigm.

Optimization of in-network packet-level caching in ICN has been extensively studied from different aspects, such as content popularity estimation (*e.g.,* [3]), criteria for determining the probability of performing local caching (*e.g.,* [4], [5]), techniques to reduce caching redundancy (*e.g.,* [6]), caching prioritization by popularity assessment (*e.g.,* [7]) and content locality (*e.g.,* [8]). Also, in-network caching has been examined as an information resilience scheme for the retrieval of content in disruptive, fragmented networks (*e.g.,* [9]). Apart from the optimization of the caching strategy, another important part of the in-network caching research in ICN is the actual content discovery mechanism, namely the mechanism to direct content requests to the right cache mainly to increase cache hit ratio and minimize content delivery latency.

Content discovery, or else, request to cache routing can follow one of two approaches, either *oppostunistic on-path*

routing, where content is searched on-path as the request is travelling towards the content source, or *co-ordinated off-path resolution-based*, where requests are forwarded off the shortest path to some designated cache that is likely to hold this content. The off-path resolution-based routing is a deterministic solution which maps requests to content items cached in nearby (or not) nodes, usually at predetermined rendezvous points *e.g.,* [10]-[13].

Though off-path resolution-based content discovery mechanisms produce relatively small extra traffic in the network, they introduce considerable amounts of coordination/communication overhead and delay and as such, result in inherently less scalable solutions. On the other hand, on-path content discovery mechanisms provide a probabilistic solution, but with inherently limited gain that can be increased by decreasing the correlation of the cached items in the routers of the network (*e.g.,* [5]). Since naive network wide flooding is rarely permitted due to its traffic overhead, recently, the authors in [14] have proposed scoped flooding-based content discovery. The proposal includes a ring model, which limits the spread of the flood to the neighbourhood. The results show that although there is some overhead, the scoped-flooding approach is far from prohibitive and can in fact scale and achieve considerable gains.

Both on-path and off-path caching and content discovery present trade-offs. On-path mechanisms require less co-ordination and management, but may provide limited gains. Conversely, off-path techniques can attain higher hit rates at the cost of extra co-ordination and communication overhead. In this paper, we try to combine the merits of both worlds by using traditional on-path mechanisms enhanced with a lightweight off-path content discovery approach, that requires from each router to keep track of only a minimal amount of information in order to locate content.

Particularly, we enhance the NDN [2] router architecture with a new component called *"Downstream Forwarding Information Table (D-FIB)"*, that keeps track of the direction (*i.e.,* next hop) in which the Data packets were sent in the past. The newly introduced D-FIB is combined with scoped interest forwarding techniques in an attempt to decrease the delivery latency of requested items and increase the cache hit rate of the system, by exploring the caching capabilities of nearby routers. We extend ndnSIM [15] and evaluate the proposed content discovery mechanism through extensive simulations in realistic network topologies for various system parameters. We show that the usage of a secondary table such as D-FIB

reduces the delivery latency by fetching items from caches located closer to the users and significantly improves cache hit rate performance, but at the same time maintains the traffic overhead at reasonable levels.

The proposed discovery mechanism is useful in crowded network scenarios, where despite the fact that connectivity to the network and the content origin exists, the communication is challenged, due to network congestion (*i.e.*, high contention for bandwidth resources). For instance, in the 5G research area the usage of wireless networks with high access rates and increased densification of the infrastructure (*i.e.*, a large number of access points deployed in a small area) is expected to severely stress (congest) the backhaul of such networks [16]. This motivates the exploration of nearby caches at the edge of the network for the retrieval of matching cached content. The proposed discovery mechanism might slightly increase the overall traffic overhead, but this traffic is better balanced/distributed across the network. Therefore, as clearly stated in [17] caching and opportunistic content discovery mechanisms have the potential to become the third key technology for the sustainability of the networks operating in future crowded scenarios (*i.e.*, together with increased access rate and increased infrastructure densification).

Also, in the vast majority of crowded network scenarios (*e.g.*, football matches, festivals *etc.*) communication is all about the exchange of content (*e.g.*, news, video-on-demand, emergency announcements *etc.*), which in many cases can be of interest to multiple participants; *e.g.*, updates on concurrent sports matches or departure times for public transportation after a football match or assembly points for emergency scenarios or public demonstrations. Those content-centric communication needs are perfectly aligned with the named-data routing and caching characteristics of the ICN paradigm.

The rest of the paper is organized as follows. In Section II, we present the functionality of the proposed content discovery mechanism and the necessary modifications to the original NDN router design in order to support it. In Section III, we evaluate through simulations the performance of the proposed mechanism, while we conclude the paper in Section IV where we also give pointers for future investigation.

II. CONTENT DISCOVERY MECHANISM

In this section, we present the functionality of the proposed content discovery mechanism. The rationale behind our design is to increase a router's awareness regarding the cached content of its neighbouring routers and benefit from the name-oriented routing to retrieve content from a closer off-path cache.

A. Downstream FIB

In order to support the proposed content discovery mechanism, we add the *Downstream Forwarding Information Base (D-FIB)* to the original NDN content router design presented in [2] and shown in Fig. 1. The functionality of the remaining NDN router components, namely the Content Store (CS), the Pending Interest Table (PIT) and the Forwarding Information Base (FIB) remain the same.
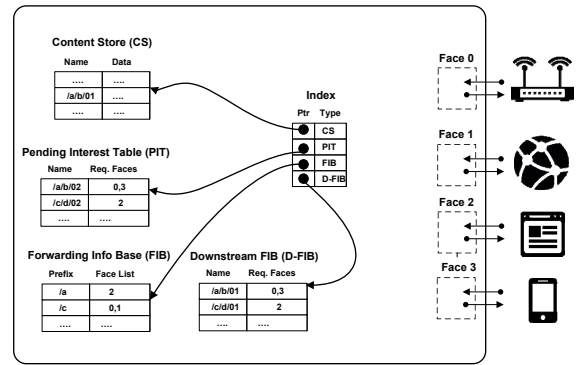

Fig. 1. Content Router design with the new Downstream Forwarding Information Table (D-FIB).

The D-FIB keeps track of the direction of the Data packets that are heading towards the users of the network. In that case, an Interest packet at a router with matching D-FIB entries can be forwarded both upstream towards the content origin (*i.e.*, following FIB entries), but also downstream towards the direction of users that successfully issued similar interests in the past (*i.e.*, following D-FIB entries). This is done in an attempt to decrease retrieval latency and increase the cache hit rate of the system by retrieving content from local caches. D-FIB entries, like PIT and FIB entries, also allow for a list of outgoing faces, pointing to multiple potential sources for the requested data. The number of outgoing faces (and therefore potential sources) that is eventually queried is subject to the chosen forwarding mechanism.

An entry in the D-FIB is triggered by a returning Data packet and comprises a trail for a matching Interest packet to follow downstream towards the direction of users with similar satisfied interests. Following the hierarchical naming in [2], where content items are segmented and named into chunks (*e.g.*, BBC/News/Elections/chunkID), a D-FIB entry is also built in that granularity. D-FIB entries, unlike PIT entries (but similarly to FIB entries), are not consumed by the first matching interest, and they remain in the D-FIB so that more Interest packets can be forwarded downstream to caches with matching cached content. The introduction of the D-FIB component increases the amount of state that should be maintained in the routers and as such we assume that the D-FIB component has limited size, *i.e.*, effectively D-FIB acts like a cache, where entries follow an LRU replacement policy. This means that entries are removed when other entries for more popular content push them at the end of the information base. Additionally, and in order to avoid following obsolete entries, D-FIB entries are also removed according to an expiration methodology (*e.g.*, a time-to-live parameter).

The D-FIB entries are similar in rationale to the *"Breadcrumbs"* proposed in [18]. Here, instead of redirecting a request towards the direction indicated by a breadcrumb, we take advantage of the named data routing scheme of ICN architectures and opportunistically *"multicast"* the interest both towards the origin server and the direction pointed by the D-FIB entries. This way, we are introducing a lightweight scheme as regards the information (*i.e.*, state) that should

be kept in a D-FIB entry (*e.g.,* there is no need to keep a timestamp of when content or interest passed through a router last as in [18]). Actually, routers act autonomously and add/remove D-FIB entries without any synchronisation with the rest of the routers.

B. Interest and Data Packet Processing

In order to support the proposed operation, we introduce an *Off-path Forwarding Flag (OFF)* bit to the Interest packet to distinguish whether it is following the FIB entries towards the content origin (OFF is set to zero by the corresponding forwarding router), or is heading towards the direction of users with similar satisfied interests. In the second case (OFF set to one by the forwarding router), the Interest packet follows matching entries in the D-FIB of each passing-by router. We also introduce a *Total Forwarding Counter (TFC)* to control the total number of Interests that will be generated from a single data request as explained below.

1) Total Forwarding Counter (TFC): An Interest packet arriving at a router with a number of matched D-FIB entries (*i.e.,* one entry for the requested chunk with multiple faces pointing in multiple directions) can be duplicated and multicasted towards all possible directions in an attempt to retrieve the requested content. This, however, might have a huge impact in the corresponding network traffic due to the possible duplicate Data packet responses. In this paper, we allow only Interest packets that head towards the origin (OFF equal to zero) to initiate downstream interests following D-FIB entries, which means that downstream interests are not further branched/multicasted upon arrival at a router. Particularly, we allow each Interest to initiate up to a specific number of downstream interests. This number of interests is a predetermined value that is included in the TFC of the Interest packet. In more details we assume that each router (*e.g.,* based on the FIB entries) is aware of the distance (*i.e.,* in hops, following the shortest path) to the content origin and the TFC is set to a value equal or larger than this distance (*i.e.,* in special cases could also be set to smaller values). The TFC value can be either consumed to reach the content origin (by decreasing it at every hop), or it can be consumed to initiate downstream interests (or both) as we describe later and show in the example of Fig. 2. This mechanism allows routers independently to control the total number of interests and accordingly the extra traffic overhead.

2) Interest packet processing: Whenever a user issues an Interest packet the OFF bit is by default set to zero and the TFC is later set to a predetermined value. This means that when an Interest packet arrives on some face of a router, the router processes the packet in the exact same way as in NDN (the packet heads towards the content origin). Particularly, if a matching chunk is found in the CS, the router sends back the Data packet. The router sends the Data packet to the face the interest arrived on and discards the Interest packet (since it was satisfied) and does not create a new PIT entry for it. Instead of a PIT entry, the router creates a new D-FIB entry from the Interest packet and its arrival face if the interest was

received from a neighbouring router (not from a user) and this face is not the same with the corresponding FIB entry for the same Data packet (*i.e.,* no D-FIB entries pointing towards end users and no downstream interests are initiated towards the same direction with the content origin). Note that the D-FIB is an exact-match lookup table similar to the PIT and CS.

If the router does not find matching content in CS and there is an exact-match PIT entry, the interest's arrival face is added to the PIT entry's Requesting Faces list. The above procedure is exactly the same to the functionality of NDN [2]. In the proposed discovery mechanism, if the Interest's TFC value is not equal to zero (we use the $|t|$ variable for this value), the router searches for a matching D-FIB entry. If a match is found the arrival face is removed from the face list of the matching entry and if the resulting list is not empty (*i.e.,* we use the l_v symbol for this set and the $|l_v|$ for the number of faces in the list), a new downstream Interest packet (*i.e.,* OFF set to one and TFC to zero) is sent out to b faces. The number of the new downstream interests b is bounded by the following expression:

$$0 \leq b \leq \max\{|l_v|, |t|\}. \tag{1}$$

Also a new PIT entry is created for each downstream interest and its arrival face. Equation (1) implies that a router with matching D-FIB entries can initialise any number of downstream interests towards the direction where matching data have been forwarded in the past, as long as this number does not exceed the current value of the Total Forwarding Counter (TFC). The exact number of interests is a management decision of the router and in the evaluation section we will examine various schemes. After initializing b downstream interests and if there is a matching FIB entry and $b+1 \leq |t|$, the router forwards the Interest packet upstream towards the content origin and creates another PIT entry from the interest and its arrival face. The packet's new TFC value is decreased by $b+1$.

In the proposed discovery mechanism a router can choose to either search all or part of the neighbouring routers' caches, or search less aggressively in neighbouring caches and let the Interest reach the origin server. Furthermore, a router might choose to check more neighbouring routers when D-FIB entries are found closer to the requesting user and less as the Interest moves deeper inside the network. Alternatively, approaches similar to the one used in [18] can be applied, where an upstream interest "waits" for the responses (either Data packet or NACK) of the downstream interests before forwarding towards the origin server. However, this approach requires significant overhead in updating the D-FIB entries and also increases significantly the delivery latency of the requested content. Here we adopt an opportunistic scheme, where each Interest packet is associated with some *quotas* and each router along the path to the server decides whether or not to spend them searching in nearby neighbours or request the server itself.

When an Interest packet arrives on some face of a router and its OFF bit is set to one (*i.e.,* Downstream interest), the router checks for a matching content in its CS. If a matching
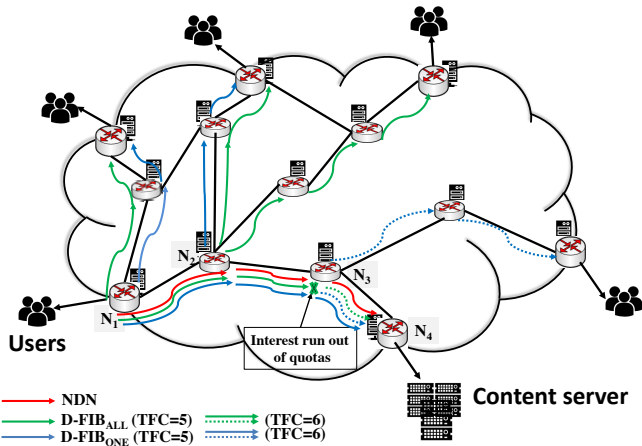
Fig. 2. Different interest forwarding strategies.

content is found, the router sends back the corresponding Data Packet, creates a new D-FIB entry and does not forward the interest further. If the router does not find a matching content in CS, it searches the PIT. If a matching entry is found the interest's arrival face is added to the PIT entry's Requesting Faces list and the Interest packet is discarded. Otherwise, it skips searching the FIB (the OFF is set to one) and checks for an exact-match D-FIB entry. If no such entry is found in D-FIB the router discards the Interest packet. This means that either the router is the last on the path of routers set by the corresponding D-FIB entries, or a matching entry has been replaced before the arrival of the interest. Note that the downstream interest is forwarded only towards one face of the matching entry, in order to minimize the corresponding overhead and control the number of issued Interest packets.

In Figure 2 we present an example of different interest forwarding strategies. The strategy in red is the default NDN forwarding strategy where the Interest packet is forwarded along the shortest path to the content origin. The strategies in blue and green are two variants of the newly proposed discovery mechanism. Here, router N_1, which received an Interest by an attached user knows a priori the length of the path to the origin and sets the TFC to a larger value (TFC = 5 and TFC = 6 in the example assuming the path to the origin is 3 hops). A value of TFC = 5 means that the routers along the path can either initiate up to 5 downstream interests or use the 3 out of the 5 quotas to reach the origin node and the remaining for opportunistic off-path search. The green strategy, which in the evaluation section we call *D-FIB$_{all}$*, initiates at each router downstream interests for every matching D-FIB entry (*i.e., $b = \max\{|l_v|, |t|\}$*), whereas the blue one, which we call *D-FIB$_{one}$*, initiates at each router only up to one downstream interest for each matching D-FIB entry (*i.e.,* sends an interest only at one of the faces in the D-FIB entry faces list). In the example of Figure 2 the D-FIB$_{all}$ strategy fails to reach the content origin when TFC = 5, since it has consumed all the quotas to reach node N_3 and initiate three downstream interests. On the other hand, the D-FIB$_{one}$ strategy managed to

reach origin both when TFC = 5 and TFC = 6 but by initiating less downstream interests in the first case. This means that the proposed discovery mechanisms might retrieve content from a closer off-path router minimizing traffic and latency, or merely fail to even reach the origin based on the following forwarding strategy. In this paper we assume that when an interest fails to retrieve the requested content following one of the proposed discovery mechanisms, the router where the user issued the interest is attached to resends the interest enabling the basic NDN strategy (the interest is forwarded to the origin server).

3) Data packet processing: The Data packet processing is exactly the same to NDN [2]. Particularly, a Data packet is not routed but simply follows the chain of PIT entries back to the requesting user(s). An exact-match lookup of a Data packet's Content Name takes place upon the arrival of the packet at each router. A CS match means that the Data packet is a duplicate so it is discarded. A FIB match means that there are no matching PIT entries so the Data packet is unsolicited and it is discarded. A PIT match (there may be more than one) means that the Data packet was solicited by interest(s) sent by this node. A list is created, that is the union of the Requesting Faces list of each PIT match minus the arrival face of the Data packet. The Data packet is sent out each face on this list, the PIT entries are removed, and for each face a new D-FIB entry is created. The new D-FIB entries are exactly the same to the PIT entries matching the Data packet. Finally, the Data packet is (optionally) cached to the Content Store of the router.

During the data packet processing, the list of next hops of all D-FIB entries are kept ordered according to their usage. The most recently inserted (or re-inserted if it was already in the list) next-hop is found at the beginning of the list, whereas the least recently inserted face is found at the end. Taking advantage of the ordering of faces, the *D-FIB$_{one}$* and *D-FIB$_{all}$* strategies give higher preference to the most recently added faces when selecting next hops from the available set of next hops in the D-FIB entry.

III. PERFORMANCE EVALUATION

In this section we evaluate the performance of the two proposed forwarding strategies described above (*i.e.,* D-FIB$_{all}$ and D-FIB$_{one}$). We compare them against the performance of the NDN's default Interest forwarding strategy [2] based on a wide range of parameters and for various values of the TFC. The objective is to evaluate their performance in terms of content retrieval efficiency and incurred traffic overhead.

A. Evaluation Setup and Metrics

For the evaluation of the proposed discovery scheme we used, after adequately extending, ndnSIM [15] (version 2.1) —an ns-3 based, packet-level simulator. The extensions to the ndnSIM simulator required to support the proposed content discovery mechanism involved enhancements to the ndn-cxx library and the NDN Forwarding Daemon (NFD) components. Enhancements to the ndn-cxx library allow the Interest packets to carry an OFF bit, which is set when an interest is forwarded downstream and a Total Forwarding Counter (TFC) field, described in Section II-B1. The NFD library was upgraded
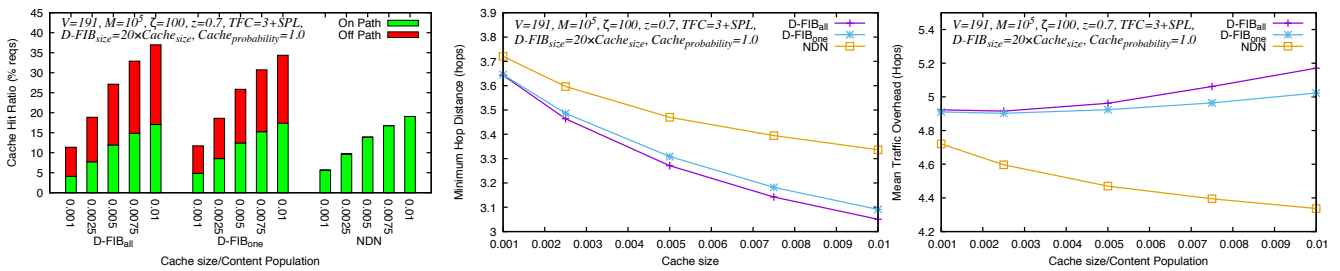
Fig. 3. The impact of the cache capacity of each router in the performance of the examined forwarding strategies.

to include the D-FIB interest forwarding mechanisms. Each of the new Interest forwarding mechanisms is implemented as a new strategy in the NDN forwarding strategy layer. Finally, the data processing portion of the NFD was slightly modified to add new entries to the D-FIB table as data packets are processed.

We use the 4755.pop.cch network topology as provided through the Rocketfuel dataset [19]. Spanning over the Asian continent, the 4755.pop.ch topology has $V = 191$ routers and 242 bidirectional links. Out of the 191 routers, 148 are edge routers (with degree less than 3), 39 are gateway routers (i.e., connected to an edge router and has degree larger than 2), and 4 backbone routers. We consider a scenario where the content item population is $M = |\mathcal{M}| = 10^5$ and requests are generated to the network with a rate equal to $\zeta = 100$reqs/sec.

Requests are generated in the network with rate $\boldsymbol{r} = \{r^1, \ldots, r^M\}$, where r^m denotes the aggregate incoming request rate (in requests per second) for content item $m \in \mathcal{M}$. The request rate for each item is determined by its *popularity*. Here we approximate the popularity of the items by a Zipf law of exponents z [20]. In that way the aggregate incoming request rate (in requests per second) for an information item $m \in \mathcal{M}$ is given by:

$$r^m = \zeta \cdot \frac{1/k^z}{\sum_{i=1}^{M} 1/i^z}, \tag{2}$$

assuming that the particular item is ranked k-th out of the M information items within the Zipf distribution. Our evaluation is based on the following metrics:

- *Cache Hit Ratio* (in % of issued interests): It is the percentage of the interests that have been satisfied (found the requested item) from a cache/router of the network (off-path and on-path).
- The *Minimum Hop Distance* (in *hops*): It is the number of hops traveled by the (first) data arriving at the user from a responding router or the content origin for each successful request. This metric is indicative of the transfer delay as a function of hops in the network.
- The *Mean Traffic Overhead* (in *hops*): It is the mean number of hops that the initiated Data packets travel in the network, until they are discarded or reach the consumer, for each satisfied interest.

Our simulation setup consists of a set of consumers (i.e., hosts) each connected to exactly one edge router, and for each request, one of the consumers is selected uniformly random as the originator of the request. We use a simplified scenario

with a single producer (e.g., large content cache) acting as the content origin for all the requests, and it is conveniently located near the core of the network[1] with direct links to two randomly chosen backbone routers. In a real deployment scenario, the distance between the content origins (e.g., CDNs, content providers) and consumers can be more than just a few (router-level) hops. In fact, the content origins can be one or more *domain-level* hops away from the consumers, which could translate to significant latency in the end-to-end communication. Therefore, our results comparing the hop distances when fetching content from the origin with the hop distances when fetching the content from the routers' caches can be thought of as the best-case scenario; in reality the additional distance that needs to be covered to reach a content origin is expected to be more than just a few router-level hops.

In all the experiments presented below, we observe the network over a duration of one hour, following an hour of "warm-up" phase during which the requests are forwarded using NDN's default strategy and the D-FIB table and content caches are populated during data processing. Finally, we assume without loss of generality that the routers have the same storage capacity and that each responded packet is cached in all routers along the return path. This caching policy is also known as *Leave Copy Everywhere (LCE)* and is the default policy in NDN [2] ($Cache_{probability} = 1$).

B. Impact of the router's cache size

In Fig. 3 we depict the impact of the cache capacity, expressed as the fraction of the items population that can be stored in the cache (i.e., CS component) of a router. A first and straightforward observation is that the usage of the proposed forwarding strategies leads to $\approx 100\%$ increase in the cache hit ratio compared to the traditional NDN strategy. Particularly, we observe that when the cache capacity of each router is only 0.5% of the total item population, almost 30% of the total issued requests can be satisfied from the caches within the network, whereas the NDN's default strategy only manages to satisfy less than 14% of the requests from the on-path caches.

Additionally, the newly proposed strategies also manage to decrease the hop count (i.e., latency) for the retrieval of content items. For both of the proposed strategies, the minimum hop distance is reduced by $\approx 10\%$ when the cache size of each router is only 1% of the content population. In a real deployment scenario, we expect the difference to be larger

---

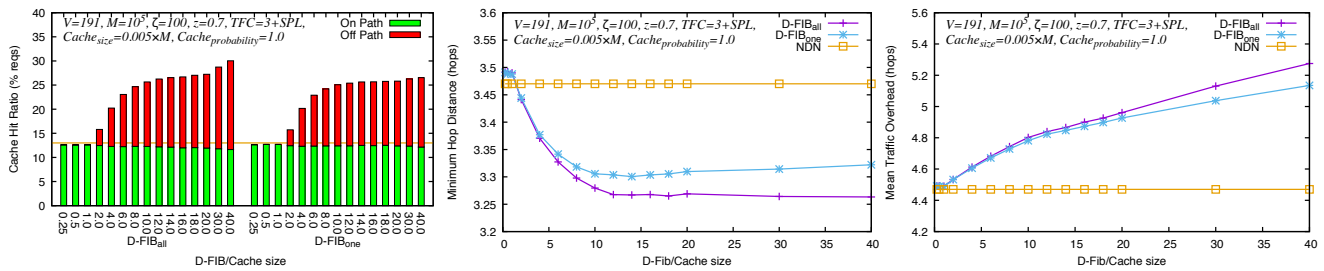[1]The average distance between edge routers and the producer is ≈ 3.5 hops.

Fig. 4. The impact of the D-FIB size in the performance of the examined forwarding strategies.

as the requests will travel outside the network to fetch content. Therefore, the consumers would potentially accrue more benefit from the increase in cache hit rates as this translates to better QoS for users, namely the latency in obtaining data. From the comparison of the two proposed downstream interest forwarding strategies we observe that the *all* strategy performs slightly better regarding hit ratio compared to the *one* strategy, but at the cost of slightly increased network overhead. Note here that we assumed only three extra quotas (*i.e.,* TFC is equal to three plus the shortest path length (SPL) in hops to the origin). This means that both strategies manage to double the cache hit ratio of the system initializing only a very small number of downstream interests. In the rest of the experiments presented below, we set the capacity of the routers to 0.5% of the total item population.

C. Impact of the router's D-FIB size

As mentioned in Section II-A, the D-FIB table on each router has a limited size and entries follow an LRU replacement policy. As such, the size of D-FIB also affects the performance of the proposed forwarding strategies, since small D-FIB sizes means that even if there are cached copies at nearby routers, the lack of entries in the D-FIB of routers cannot exploit them. In Fig. 4, we depict the impact of the D-FIB size, expressed as the ratio of the router's cache size. From the cache hit rate plot we observe that when the D-FIB size is less than the cache size of a router, the proposed strategies cannot retrieve any content exploiting off-path caches. This is due to the rate at which D-FIB entries are updated, which does not allow an interest for a piece of content to establish a trail for a similar future interest. Both strategies start to perform significantly well in terms of cache hit rates for sizes of D-FIB that are at least 10 to 16 times larger than the cache size (*i.e.,* in number of entries and not actual disk capacity). This finding is in line with the work in [10] where it has been shown that, depending on the network topology, such a table should be approximately 20 times larger than the size of the cache in terms of entries to provide an adequate performance enhancement.

D. Impact of the total forwarding counter (TFC) value

In Section II-B1 we mentioned that the TFC of each Interest packet is set to a predetermined value, by the first-hop router where the user who issued the request is attached to. In the above experiments this value is set to $TFC = 3 + SPL$, where SPL is the shortest path length in hops from each router to

the origin server and *extra quota* is set to 3. In this section we examine the performance of the proposed forwarding scheme for different values of extra quota (note that the values in the x-axis of Fig. 5 are additional to the SPL, which is obtained by the first-hop router from its FIB table).

From Fig. 5 we observe that increasing the TFC value does not increase the hit ratio. On the contrary, a large TFC value slightly decreases the hit ratio of the system. This is mainly due to the LCE policy, since a large TFC value means a larger number of downstream interests. This, in turn, results in more duplicate responses and merely in a higher rate of cache replacements. In other words, a single interest that is multicasted towards many different directions results in duplicate responses of the same data packets that will be cached along the paths towards the user and replace already cached content. This increases the correlation and redundancy of cached data in the network and minimizes the probability of future interests for different content to find matching cached content. For cache probabilities set to $\approx 50\%$, the hit ratios slightly increase with increasing TFC value, but we do not present these results due to space limits.

Extra Quota	Sat. Rate_all	Sat. Rate_one
-2	26.3%	26.3%
-1	31.3%	31.3%
0	91.3%	91.3%
1	98.5%	98.5%
2	99.2%	99.7%
3	99.7%	99.9%
...	...	...
11	100%	100%

TABLE I
REQUEST SATISFACTION RATE OF THE FIRST REQUESTS FOR DIFFERENT
EXTRA QUOTA VALUES

In Fig. 5, we also consider negative and zero extra quota values where TFC in each packet is set to *extra quota + SPL*. A request packet carrying a negative quota can only retrieve the content from a router's cache since the TFC allowance will be spent before reaching the content origin even when the request follows the shortest path. As mentioned before, the first hop router is responsible for setting the TFC value in the packets and also for retransmitting interests when the previous attempt is not successful. The TFC value can be set according to different goals. If the primary goal is to reduce the percentage of requests that reach the content origin and minimize the overhead of duplicate data packets, then the first-
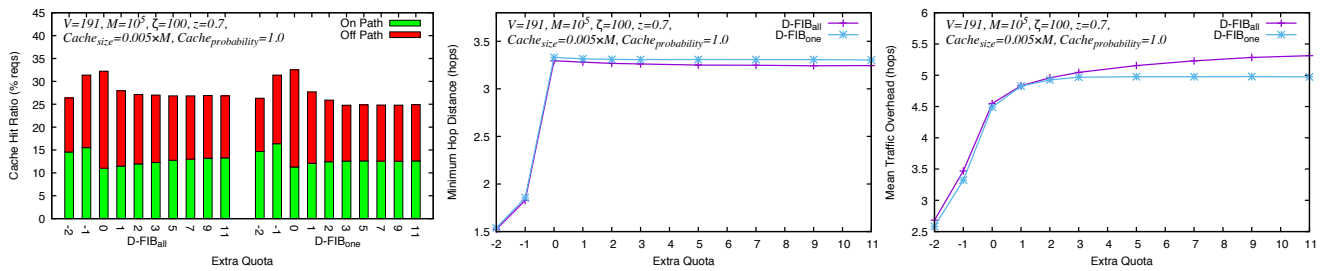
Fig. 5. The impact of the TFC value in the performance of the examined forwarding strategies.

hop router can initially try a negative quota value (*e.g.,* -1). When the request is not successful in returning data, larger quota values can be used in the retransmitted requests. On the other hand, if the primary goal is to obtain the data at first attempt and thus increase QoS, the TFC value in the request can be set to a positive value to guarantee retrieval of data.

Table I shows the satisfaction rate of first requests (*i.e.,* excluding retransmissions) for various extra quota values for the *"one"* and *"all"* forwarding strategy. For negative extra quota values, the satisfaction rate of the first requests is below 32%. However, for zero extra quota value, the satisfaction rate reaches 91% and only 59% of the queries reach the content origin, whereas 32% of the queries are satisfied from the caches.

The overall performance and overhead of the proposed forwarding mechanisms do not differ significantly, due to the sparseness of the ISP topology that we used in our experiments. However, their performance differs more significantly for denser topologies that we have also examined, but not depicted due to space limitations.

IV. CONCLUSIONS AND FUTURE WORK

We have proposed an opportunistic off-path content discovery mechanism in ICN, which utilizes a D-FIB table to trail the direction of data packets that head towards users. We have presented two simple yet efficient strategies that exploit the D-FIB table to multicast interests towards both the content origin and towards users who had retrieved the same content in the near past. In order to limit the number of replicas of a request sent to the network, our scheme uses a total forwarding counter (TFC) state in the packets. Our results have shown that the off-path content discovery mechanism *doubles the cache hit rate* compared to the NDN's default (shortest-path) routing strategy while keeping the resulting overhead at acceptable levels. Such an increase in the cache hit rate translates to significant reduction in content retrieval latency in a real deployment scenario.

As our initial results are encouraging, we plan to extend our work with more sophisticated strategies and complement it together with user-assisted caching similarly to [9]. Such sophisticated approaches can use adaptive mechanisms when making forwarding decisions and also when picking the initial TFC value at the edge routers.

ACKNOWLEDGMENTS

REFERENCES

[1] G. Xylomenos, C. Ververidis, V. Siris, N. Fotiou, C. Tsilopoulos, X. Vasilakos, K. Katsaros, and G. Polyzos, "A survey of information-centric networking research," *IEEE Communications Surveys Tutorials*, 2014.

[2] V. Jacobson, D. K. Smetters, J. D. Thornton, M. F. Plass, N. H. Briggs, and R. L. Braynard, "Networking named content," *ACM CoNEXT*, 2009.

[3] G. Carofiglio, M. Gallo, L. Muscariello, and D. Perino, "Modeling data transfer in content-centric networking," *ITC*, 2011.

[4] A. Ioannou and S. Weber, "Towards on-path caching alternatives in information-centric networks," *IEEE Local Computer Networks (LCN)*, 2014.

[5] W. K. Chai, D. He, I. Psaras, and G. Pavlou, "Cache "less for more" in information-centric networks," *Computer Communications*, 2013.

[6] I. Psaras, W. K. Chai, and G. Pavlou, "In-network cache management and resource allocation for information-centric networks," *IEEE Transactions on Parallel and Distributed Systems*, 2014.

[7] K. Cho, M. Lee, K. Park, T. Kwon, Y. Choi, and S. Pack, "Wave: Popularity-based and collaborative in-network caching for content-oriented networks," *Computer Communications Workshop*, 2012.

[8] G. Tyson, S. Kaune, S. Miles, Y. El-khatib, A. Mauthe, and A. Taweel, "A trace-driven analysis of caching in content-centric networks," *ICCCN*, 2012.

[9] V. Sourlas, L. Tassiulas, I. Psaras, and G. Pavlou, "Information resilience through user-assisted caching in disruptive content-centric networks," *IFIP Networking*, 2015.

[10] I. Psaras, K. V. Katsaros, L. Saino, and G. Pavlou, "Lira: A location independent routing layer based on source-provided ephemeral names," *arXiv preprint arXiv:1509.05589*, 2015.

[11] V. Sourlas, P. Flegkas, and L. Tassiulas, "A novel cache aware routing scheme for information-centric networks," *Computer Networks*, 2014.

[12] D. Trossen and G. Parisis, "Designing and realizing an information-centric internet," *IEEE Communications Magazine*, 2012.

[13] L. Saino, I. Psaras, and G. Pavlou, "Hash-routing schemes for information centric networking," *ACM ICN*, 2013.

[14] L. Wang, S. Bayhan, J. Ott, J. Kangasharju, A. Sathiaseelan, and J. Crowcroft, "Pro-diluvian: Understanding scoped-flooding for content discovery in information-centric networking," in *ACM ICN*, 2015.

[15] S. Mastorakis, A. Afanasyev, I. Moiseenko, and L. Zhang, "ndnSIM 2.0: A new version of the NDN simulator for NS-3," NDN, Technical Report NDN-0028, 2015.

[16] J. Andrews, S. Buzzi, W. Choi, S. Hanly, A. Lozano, A. Soong, and J. Zhang, "What will 5g be?" *IEEE JSAC*, 2014.

[17] G. Paschos, E. Baştuğ, I. Land, G. Caire, and M. Debbah, "Wireless caching: Technical misconceptions and business barriers," *arXiv preprint arXiv:1602.00173*, 2016.

[18] E. Rosensweig and J. Kurose, "Breadcrumbs: Efficient, best-effort content location in cache networks," *IEEE INFOCOM*, 2009.

[19] N. Spring, R. Mahajan, D. Wetherall and T. Anderson "Measuring isp topologies with rocketfuel," *IEEE/ACM Transactions on Networking (TON)*, 2004.

[20] G. Dán and N. Carlsson, "Power-law revisited: Large scale measurement study of p2p content popularity," *IPTPS*, 2010.