A Popularity-aware Centrality Metric for Content Placement in Information Centric Networks

Junaid A. Khan, Cedric Westphal, Yacine Ghamri-Doudane

▶ To cite this version:

A Popularity-aware Centrality Metric for Content Placement in Information Centric Networks

Junaid Ahmed Khan‡, Cedric Westphal* and Yacine Ghamri-Doudane†
*University of California, Santa Cruz, CA, USA
& Huawei Technology, Santa Clara, CA, USA
‡Univ Lyon, INSA Lyon, INRIA, CITI Lab, Villeurbanne, France
†L3i Lab, University of La Rochelle, France
junaid.khan@inria.fr, cedric@soe.ucsc.edu, yacine-ghamri@univ-lr.fr

Abstract—Information-centric networks enables a multitude of nodes, in particular near the end-users, to provide storage and communication. At the edge, nodes can connect with each other directly to get content locally whenever possible. As the topology of the network directly influences the nodes' connectivity, there has been some work to compute the graph centrality of each node within the topology of the edge network. The centrality is then used to distinguish nodes at the edge of the network. We argue that, for a network with caches, graph centrality is not an appropriate metric. Indeed, a node with low connectivity (and thereby low centrality) that caches a lot of content may provide a very valuable role in the network.

To capture this, we introduce a *popularity-weighted content-based centrality (P-CBC) metric* which takes into account how well a node is connected to the *content* the network is delivering, rather than to the other nodes in the network. To illustrate the validity of considering content-based centrality, we use this new metric for a collaborative caching algorithm. We compare the performance of the proposed collaborative caching with typical centrality based, non-centrality based, and non-collaborative caching mechanisms. Our simulation implements P-CBC on three random instances of large scale realistic network topology comprising $2,896$ nodes with three content replication levels. Results shows that P-CBC outperforms benchmark caching schemes and yields a roughly $3x$ improvement for the average cache hit rate.

Index Terms—Information/Content Centric Networking, Content Caching, Fog Networking, Content Offload.

I. INTRODUCTION

Under the pressure of increasing bandwidth demand, service providers have been trying to offload traffic away from their networks unto local caches, either co-located with access points from WiFi networks or small cells or base stations, or provided by other users in an ad hoc fashion.

Peers, small cell base stations, and access points in a local area near a user may offer storage or caching capability for the users of the network. They may also have a wide range of connectivity varying from very poor in sparse areas to very rich in dense environments. To better offload content away from the wide area networks, these nodes can cache content. Which content should depend on how connected the node is, and how popular the content.

Centrality [1], a concept from graph theory typically applied to social networks, is used to find important nodes in a graph. A high centrality score reflects a high topological connectivity

for a node in the network. Typical centrality measures are: degree (the number of directly connected nodes), closeness (the average length of the shortest paths between the node and all other nodes in the graph), betweenness (the number of shortest paths between all pairs of nodes in the graph going through a specific node), and eigenvector centrality (a measure of node influence in the network).

Computing centrality on the topological graph yields interesting insights. Yet it fails to capture that, in network of caches, the consumer is interested in connecting to the *content*, not to a specific *node*. If a node is well connected, but holds content of little value, it is not useful to the user. It has low centrality, in the context of *content* delivery, despite having a high centrality within the topological graph.

To address this, we utilize the concept of *Popularity-Weighted Content-Based Centrality* (P-CBC). We introduce here a weighted content-based centrality metric. Namely, for a cache v, its P-CBC is calculated as a function of the number of shortest paths which go through v for content x, the popularity of x and the number of shortest paths from the users to x. It is defined more formally in Section III.

To show the usefulness of the P-CBC, we propose to place content in caches depending on the cache's P-CBC metric. Namely, we suggest to place content in a way that ensures the P-CBC metric is high for most caches, so that all users have connectivity to the most popular content, and not only those in densely covered areas.

We evaluate our proposed P-CBC-based content placement policy by simulations with three different realistic typologies and three content replication ratios. The results show that the proposed P-CBC-based content placement outperforms typical centrality schemes, or schemes without coordination. Our contributions can be summarized as follows:

- We introduce the concept of *Popularity-Weighted Content-Based Centrality* (P-CBC), which we believe is better suited for network of caches (such as Information-Centric Networks) than traditional graph centrality;
- We introduce a *scalable method* to compute P-CBC without a priori knowledge of the content placement, based only upon replication rules and some basic assumptions about content placement;

- We present a P-CBC-based *content placement* algorithm which takes into account the number of paths to content passing through a node to decide where to cache in the network;
- We *evaluate our algorithm* through simulations which show a significant improvement.

The remainder of the paper is organized as follows. Section II discusses the related work. In Section III we define our network model and introduce our content-based centrality metrics. Section IV shows our content placement algorithm. In Section V, we discuss the performance evaluation and results. Section VI concludes the paper along with insights into future directions.

II. RELATED WORK

Content Caching has been studied for some time by the research community, spanning a wide spectrum, including Small Cell Networks (SCNs) [2],[3], Content Distribution Networks (CDNs) [4] and Information/Content Centric Networking (ICNs/CCNs). For example in [5], distributed cache management decisions are made in order to efficiently place replicas of information in dedicated storage devices attached to nodes in the network using ICN. Similarly [6] addresses the distribution of the cache capacity across routers under a constrained total storage budget for the network. The authors found that network topology and content popularity are two important factors that affect where exactly should cache capacity be placed. [7] looked at pushing content to the edge to anticipate network congestion, while [8] computed the capacity of an ad-hoc network of caches. [9] looked at how content caches are connected in relation to the users and the servers, but did not offer a systematic approach of content-based centrality.

In a recent work [10], game theory is exploited for caching popular videos at small cell base stations (SBSs). Another work, [11] proposed a game theoretic approach in ICN to stimulate wireless access point owners to jointly lease their unused bandwidth and storage space to a content provider under partial coverage constraints. Both papers targeted a pricing model instead of providing an efficient content placement solution.

[12] defines a "conditional betweenness centrality" and uses this metric to chooses which nodes will cache the content. Socially-Aware Caching Strategy (SACS) [13] for Content Centric Networks (CCNs) uses social information in order to privilege influential users in the network by pro-actively caching the content they produce. The authors detect the influence of users within a social network by using the Eigenvector and PageRank centrality measures.

Another centrality based caching approach in CCN is presented in [14] where the sizing the content store is based upon centrality. The authors exploit different centralities (betweenness, closeness, stress, graph, eccentricity and degree) to heterogeneously allocate content store at nodes instead of homogeneous allocation. It is proposed that a simple degree centrality based allocation is sufficient to allocate content

store. Similarly, [15] shows that a higher cache hit rate can be achieved if content is cached at high betweenness centrality nodes. Similar to this work, [16] considers content-based centrality, but does not take into account the popularity of the content.

We argue that the topological connectivity only partially relates to the content, and there is a need to consider the content reachability in the network not addressed in the prior art. A well connected node in the topological graph is not necessarily closer to end-users requesting the content. To the best of our knowledge, we are the first to propose to compute a content-based centrality.

III. POPULARITY-WEIGHTED CONTENT-BASED CENTRALITY

A. System Model

1) Connectivity Model: We assume that nodes are connected to a proximity network (also denoted as fog network in some places) through local, ad hoc connections. The connectivity between nodes is modeled by a graph $G(V, E^v)$, where $V = \{v\}$ is the set of nodes and $E^v(t) = \{e_{jk}(t) \mid v_j, v_k \in V, j \neq k\}$ is the set of edges $e_{jk}(t)$ modeling the existence of a communication link between nodes j and k at time t. We assume the time is slotted: $T = (\overline{t_1}, \overline{t_2}, ...)$ as a sequence of regular time-slots, where the kth time-slot is represented as $\overline{t_k} = [t_k, t_{k+1})$. We assume that during a particular time-slot the content placement and the connectivity are stable. In practice, the nodes will stay connected for a period of minutes or hours, depending on the application.

2) Caching Model: We define the set of known content as $X = \{x_1, \ldots, x_N\}$ for a catalog of N pieces of content, where x_j is an indivisible content chunk in the network. In the remaining of the paper, we will deal with individual content chunks $x_i \in X$, however a larger size content can be composed of several such content chunks. The nodes can fetch the content either from the service provider using the operator's infrastructure link or locally from peers in the fog, using a low-cost connectivity.

Definition: (Empirical Content Popularity) The content popularity is represented by the probability $p_x = \frac{\lambda(l,\overline{t})}{\Lambda(l,\overline{t})}$ as a measure of the user interests for content x at location l and time $\overline{t}$. Here, $\lambda(l,\overline{t})$ represents the number of interest for the content x at location l and time $\overline{t}$ and $\Lambda = \sum_{X_v} \lambda(l,\overline{t})$ is the total interests for all contents items. The content popularity can be shared with the nodes using three approaches, (i) Offline method by the content operator as a control message. (ii) Local monitoring by the nodes taking into account the number of user interests for the content, and (iii) part of content header shared by a service provider.

B. A Popularity-Weighted Content-Based Centrality Metric

Figure 1 shows an illustration to explain the concept of content-based centrality. Here the node v_1 is on the path from the users to the content in the server, but only for the cache misses for v_2, v_3 or v_4, and for the users directly connected to

v_1. Therefore, for users connecting to content that is cached at v_4 or v_3, v_1 is not on the path and therefore has very low centrality.

However, for cache misses at the other nodes v_2, v_3, v_4, *all* requests will go through v_1 and therefore v_1 will have very high centrality as all the traffic will go towards the origin server.

It is easy to see that the usual topological definition of centrality should therefore be modified as it varies on the type of content and where this content is cached. If all the nodes v_1, v_2, v_3 and v_4 keep a copy of the most popular content, then v_1 will have low centrality for the most popular content. Therefore the rate of request going through v_1 will be low as well. Therefore, both the content and its popularity needs to be taken into account to compute the centrality of v_1 in an information-centric network.

Other typical node-based centrality measures such as degree, closeness, betweenness and eigenvector consider the node v_2 as the highest centrality node as well, independently of where the content is located and how it is being accessed.

Content that is popular will generate more traffic, therefore the nodes that offer paths for this content should also have a higher content-based centrality. Therefore we offer the following definition of popularity-weighted content-based centrality:

Definition: the *Popularity-Weighted Content-Based Centrality (P-CBC)* of a node v is the weighted sum of the ratio of the number of shortest paths from all users to all content that passes through the node v to the total number of shortest paths between all the (user,content) pairs, where the sum is weighted by the popularity of the content.

Formally, for a node v, and for a probability distribution p_x for the content $x \in X$ and if $\sigma_v(u, x)$ is the number of shortest paths from user u to content x going through node v and $\sigma(u, x)$ the total number of shortest paths between u and x, then:

$$\text{P-CBC}(v) = \sum_{u,x} \frac{\sigma_v(u, x)}{\sigma(u, x)} \times p_x \tag{1}$$

The popularity-weighted CBC can then normalized by the number of possible paths to all content in the network such as:

$$(\text{P-CBC}(v))_{normal} = \frac{P-CBC(v) - \min_{w \in V}(P-CBC(w))}{\max_{w \in V}(P-CBC(w)) - \min_{w \in V}(P-CBC(w))} \tag{2}$$

We have used a modified version of Betweenness centrality in our definition of P-CBC, but we could use other forms of centrality as well, such as Closeness or Eccentricity for instance. Closeness P-CBC would measure the inverse of the weighted sum of the path length from users to content for instance.

C. Practical and Scalable Computation of P-CBC

Computing the proposed centrality appears to be extremely difficult, since the catalog X may be very large, the topology complex, and it requires knowing the content placement a priori, whereas we would like to compute the content-based
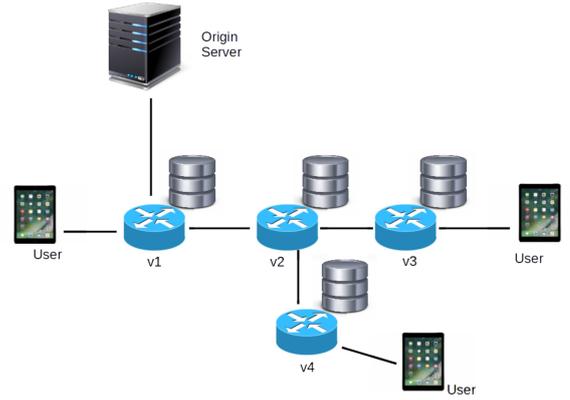

Figure 1: An example of Content-based Centrality

centrality so as to guide the content placement. Namely we would like an idea of how well a node connects users to the content *before* we assign the content to it. However, there is a practical method to compute P-CBC at each node.

Our solution is to note that the centrality of a piece of content does not depend on the specific chunk x_j, but rather, on the relative placement of the multiple copies of a chunk x_j. For instance, if N_m objects are not cached anywhere in the network (and only at the origin server), their contribution to P-CBC(v) is equal to N_m times the number of shortest paths to the origin server going through v divided by the number of shortest paths to the origin server, weighted by the aggregated popularity of all the N_m objects. We do not need to know which N_m objects are cache misses, only that there are N_m of them and what is their expected popularity. And for these N_m objects, only one computation is required using a traditional computation of centrality.

In the case of these N_m objects, their popularity will be that of cache misses, namely the tail of the popularity distribution. For content that is cached in the network, we need a different method.

In practice, content will be replicated in the network following some simple placement rules and heurisitcs. Namely, the most popular content will be placed at the edges but the rest of the cached content will be placed only at a handful of nodes so as to ensure that more content is cached within the proximity network.

More formally, a simple content placement rule is that a fraction of the cache in each node v will be allocated to the most popular content and this content will be at all nodes. The rest of the cached content will be placed in at most one node in the proximity (or fog) network so as to cover more content. Other rules are possible, but we will demonstrate how to compute P-CBC under this one rule for illustration purpose.

We define by α the fraction of the cache allocated to that most popular content that is common to all node. α corresponds to our *replication policy*. For a total cache size of b_v at node v (assumed here to be uniform for all nodes, but this

assumption can be relaxed), $C = \alpha b_v$ contains content that is common at all the caching nodes (i.e. most popular content), while $D_v = (1 - \alpha)b_v$ is the fraction of node caches with unique (different) content compared to other nodes. With the knowledge of α, we can compute the P-CBC *without* knowing which content is placed at which node. The only knowledge is the rule that C content is common and each node v contains D_v content that is specific to it.

However, in the content placement phase, we need to ensure that this replication rule is satisfied, by attempting to place content such that a fraction α is replicated at all nodes, and the rest is unique to each node.

Since the most popular content is the one that will be at all nodes (in C) and that the next most popular content will be spread over the caches (in each D_v for cache v), then we can compute the centrality as:

$$\text{P-CBC}(v) = \sum_{u,C} \frac{\sigma_v(u,C)}{\sigma(u,C)} \times p_C + \sum_{u,D_v} \frac{\sigma_v(u,D_v)}{\sigma(u,D_v)} \times p_{D_v},$$

$$p_C = \sum_{i=1}^{C} p_i, \quad p_{D_v} = \frac{1}{M} \sum_{i=C+1}^{MS} p_i,$$

where we have M caches with D_v content unique to the node v, p_C and p_{D_v} are sum of the probability for content in C and D_v respectively. This way, the popularity-based centrality is computed *per cache* and not *per content*. This is the same complexity as traditional centrality. Yet it gives us a metric of the connectivity of the users to the content.

This specific replication policy of splitting the cache into C and D_v is a simple first step, to demonstrate the feasibility of a content-based centrality metric. We can define more sophisticated rules that allow to compute P-CBC without knowing the specific content placement. It nonetheless gives us a simple tuning parameter α to know which content makes sense to be duplicated everywhere to be accessible.

IV. CONTENT PLACEMENT ALGORITHM

The problem of content placement can be formulated as follows:

$$\begin{array}{ll} \underset{s}{\text{maximize}} & \sum_v \text{P-CBC}(v) \\ \text{subject to} & \sum_{x \in X} b_v^x \leq b_v, \forall v \end{array},$$

where s is content placement for the content cached in the local network. Thus, the objective function maximizes the content from a particular service provider available at the nodes in the fog near the users. The constraint considers that the node buffer where a content cached at an individual node buffer should not exceed the maximum available threshold space. We also consider a defined replication factor α where for a node cache size b_v, $C = \alpha b_v$ contains content common

---

Algorithm 1 Content Placement Algorithm
1: **INPUT:** $V, X, b_v^t, \forall v \in V, p_x, \forall x \in X, \alpha$
2: **OUTPUT:** $s \subseteq V, X_s, X_v \in X$
3: Initialize $s = \phi, b_{v^*} = \phi, X_s = \phi$
4: **for** each location $l \in L$, time-slot $\bar{t} \in T$ **do**
5: $\quad v^* = \underset{v \in V}{\arg \max}(\text{P-CBC}(v)), v^* \notin s$
6: $\quad$ **while** $(b_{v^*} \leq C)$ **do**
7: $\quad\quad b_{v^*} \leftarrow \underset{x \in X}{\arg \max}(x, p_x),$
8: $\quad\quad$ Update b_{v^*}, X_{v^*}
9: $\quad\quad X_s = X_s \cup X_{v^*}$
10: $\quad$ **end while**
11: $\quad$ **while** $(b_{v^*} \leq b_v^t)$ **do**
12: $\quad\quad b_{v^*} \leftarrow \underset{x \in X, x \notin X_s}{\arg \max}(x, p_x),$
13: $\quad\quad$ Update b_{v^*}, X_{v^*}
14: $\quad\quad X_s = X_s \cup X_{v^*}$
15: $\quad$ **end while**
16: $\quad s = s \cup v^*$
17: **end for**
18: **return** s

---

at all nodes, while $D_v = (1 - \alpha)b_v$ contains content unique to this node.

The P-CBC is initially computed without any content placement, using the replication rule from the previous section. Then, the distributed content placement is optimized at co-located nodes using Algorithm 1. For a location l and time-slot $\bar{t}$, the node $v^* \in V$ with the highest P-CBC becomes the delegate to locally cache content in the fog as shown in Line 5. It initializes the fog formation by creating the subset s and caches the most popular content while respecting the replication factor α for its respective storage buffer (Line 6). It continues caching content with decreasing popularity till the storage conditions based on the replication factor α are met. Once the storage buffer is at C or αb_v, it updates the cached content and start filling the remaining content with decreasing popularity. The condition in Line 11 continues adding content until the node buffer b_v attains its maximum threshold to cache content as indicated in our optimization constraint. The cached content is updated in the fog s (Line 14) and the node adds itself to the fog (Line 16). Similarly, in a decreasing order of node P-CBC, the remaining nodes cache content in the fog x with same content at αb_v and different at $(1 - \alpha)b_v$.

Algorithm 1 ensures that a maximum amount of popular content X_s are cached in the fog which in turn maximizes our objective function. At the same time, it yields low overhead as it requires a node to exchange information regarding its cached content with its one hop neighbors where the frequency of such exchanges as well as the P-CBC computation depends on the application requirements. Therefore, the complexity is proportional to the amount of node neighbors or simply its one hop degree.
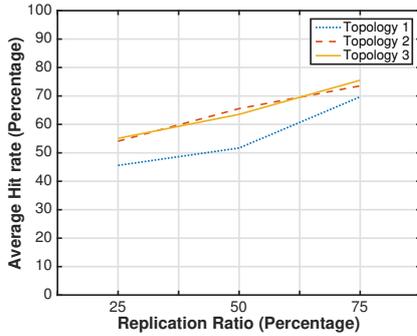
Figure 2: Average cache hit rate achieved by P-CBC on three different topologies with three content replication ratios.

V. NUMERICAL EVALUATION

We evaluate the proposed P-CBC and our associated caching algorithm using NS-3 where the named-data networking model of the ICN architecture is implemented. Three different topologies are extracted from a realistic large scale trace of $2,986$ nodes (vehicles) in Köln, Germany in order to validate the scalability of our caching approach. This allows us to evaluate the concept of fogs on realistic topologies reflecting connectivity in an urban environment. The topologies are the time snap-shots of the network connectivity at the initial, at 30th minutes and at 60th minutes respectively.

A. Simulation Scenario

The simulation scenario implements consumer nodes which generate interests for a pre-known content sequence of 100 unique items following a Zipf distribution (coefficient=1), i.e. more frequent interests for more popular content. Any provider node already caching the content responds to the consumer interest where interests can be routed using any p2p routing schemes, though we use [17] as the social aware interest forwarding scheme. We allow intermediate nodes with uniform buffer size to perform in-network caching. We consider 30% of the nodes as consumers, 30% as providers (up to 900 caching nodes) nodes and the remaining nodes with disabled caching in order to accurately evaluate the performance of caching nodes.

We also define α as a content replication factor where for a node buffer b_v, αb_v contains content which is same at all nodes in the fog, while $(1-\alpha)b_v$ is the buffer space for the cached content unique to each node. The Köln $6 \times 6 km^2$ city center is divided into 36 neighborhoods, i.e. all nodes are divided in to 36 fogs of 25 nodes in each fog.

As a first step, we randomly populate the caches of nodes in the fog with content in order to compute each node P-CBC. Random interests are generated. Then, we implement the collaborative placement Algorithm 1 where the highest P-CBC node caches the most popular content. Similarly content are populated at the remaining nodes' cache with decreasing content popularity and node P-CBC score. We implement our proposed P-CBC as well as a variant with no popularity con-

sideration in computing centrality denoted as CBC, however content are placed using the Algorithm 1. We compare both P-CBC and CBC with the following three different caching approaches:

- **Centrality-based** Caching popular content at fog of high Degree, Closeness, Betweenness and Eigenvector centrality nodes. Algorithm 1 is implemented using each scheme where the most popular content is placed at the highest centrality node, then remaining content are placed at the nodes with decreasing node centrality and content popularity.
- **Non-centrality based** Social-unaware approach by caching greedily popular content at all nodes in the fog along a Least Recently Used (LRU) based content replacement policy.
- **Non-collaborative based** approach where no fog is formed between nodes and each individual node cache indifferently. P-CBC is used to identify nodes.

The following performance metrics are used to evaluate P-CBC and the proposed collaborative caching algorithm:

- Cache Hits: It is the average number of content responds from the node cache, calculated as the ratio of the number of content responses to the number of received interests by the nodes.
- Success Rate: this is the ratio of the number of interests responded with content or forwarded by the nodes to the number of generated interests.

B. Simulation Results

1) Cache Hits: We computed the cache hit rate for each scheme; (i) Fog with P-CBC, CBC without considering popularity, Degree, Closeness, Betweenness and Eigenvector centrality, (ii) non-centrality based fog, and (iii) no fog where individual nodes cache indifferently. Figure 2 shows the average cache hit rate of the nodes in fog classified by P-CBC. We compare the hit rate of our approach for three topologies from the Cologne trace and three replication ratios ($\alpha = 25\%, 50\%$ and 75%) to find the impact of varying underlying topologies. It is shown that the average cache hit rate achieved differs with respect to topology. Furthermore, we observe for all topologies that there is an increase in the cache hit rate with increasing content replication ratio. Thus, increasing popular content placement results in high hit rate with no impact of the underlying network topological connectivity.

Similarly, we implement benchmark centrality schemes by first placing popular content at the highest centrality node identified by each scheme. Then, following a decreasing node centrality order, we place the content at the set of nodes in the fog with decreasing content popularity order. Figure 3 shows the hit rate comparison where Figures 3a, 3b and 3c show result for each topology. Thus, irrespective of the network topology, P-CBC achieves a high cache hit rate when compared to all other approaches. It resulted in around 55% hit rate for Topology 1, around $55-73\%$ for Topology 2 and around $57-73\%$ for the Topology 3.

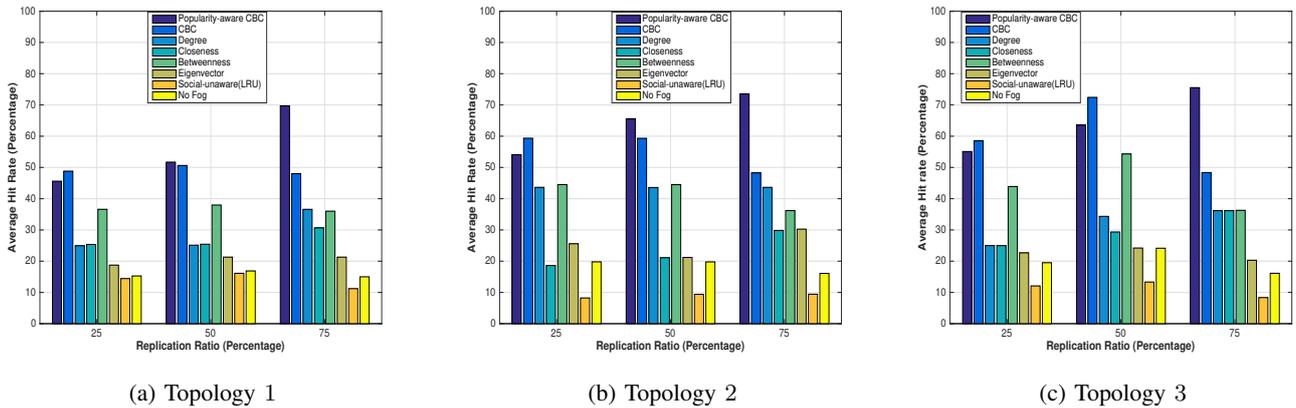(a) Topology 1   (b) Topology 2   (c) Topology 3

Figure 3: Hit rate comparison for centrality-based (P-CBC, CBC (no popularity), Degree, Closeness, Betweenness, Eigenvector), non-centrality based or social-unaware (LRU) and non-collaborative (no fog) based caching.
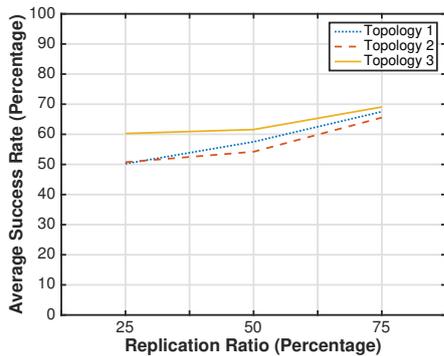

Figure 4: Average success rate achieved by P-CBC on three different topologies with three content replication ratios.

P-CBC is followed by simple CBC where it resulted in similar performance for 25% and 50% replication ratio where the similar performance is due to relatively less amount of popular content commonly cached at nodes. However, P-CBC outperforms it for a cache replication ration of 75% since the amount of popular content in the cache are increased. All the centrality schemes resulted in substantially higher hit rate than the case without non centrality (social-unaware). For all three topologies, this resulted in around 10% hit rate with a maximum of 18% cache hit rate for Topology 1. This validates the fact that P-CBC not only works better for the case when more popular content are common between caches but also outperforms other schemes in the case of less amount of popular contents commonly cached at nodes in a fog.

Figure 3 also depicts the result by implementing a case where there is no collaboration between the caching nodes, namely no fog scenario. It resulted in slightly better performance than the case of non-centrality based caching, though it achieves a lower average hit rate where it merely achieves around 20% cache hit rate. Thus, the overall comparative analysis of cache hit rate on three different topologies and three content replication ratios reveals that P-CBC achieves a better hit rate. It is because it consider jointly the number of paths to content and its popularity instead of considering nodes as a content-centric approach.

2) Success Rate: We evaluate the success rate as the content responded or forwarded by the nodes in the Fog. In Figure 4, we show the average success rate achieved by the nodes in the Fog classified using P-CBC. Similar to the hit rate, we analyze three topologies and three replication ratios. Each topology resulted in a different success where Topology 3 shows the best performance with more than 60% success rate for all replication ratios, followed by Topology 1 and Topology 2, thus suggesting that topology impacts performance. Another observation is that the success rate results from all topologies follow similar trend, thus validates the robustness of P-CBC when applied to different topologies. We see in the case of Topology 1 and 2 that by varying the replication ratio, a slight increase in the success rate is achieved from 75% to 50% replication ratio. This suggest that for a node, it is optimal to constantly keep more amount of popular content.

Figure 5 shows a comparative analysis of the success rate achieved by different caching approaches. We observe that overall, P-CBC outperforms all other approaches, yielding an average success rate of up to 68, 67 and 70% for Topology 1, 2 and 3 respectively.

An interesting behavior is seen in the case of Topology 1 (Figure 5a) where fogs based on P-CBC, degree and betweenness centrality have similar success rate, though their score differs between replication ratio. For 25% replication ratio, both degree and betweenness centrality show a higher success rate (60%) than P-CBC, though for 75% replication ratio, P-CBC dominates both of them. We investigated this behavior and found that since both betweenness and degree are node-centric metrics, and therefore better connected nodes at the relative center of the topology and are acting as bridge between large number of nodes, thus resulting in a higher success rate. This underlies the fact that content-based centrality is still related to other centrality metrics, especially in situations with poor caching performance.

Nevertheless, nodes classified by our approach outperform them when more popular content is placed in the fog, despite the possibility that such nodes might not be well placed in the network.

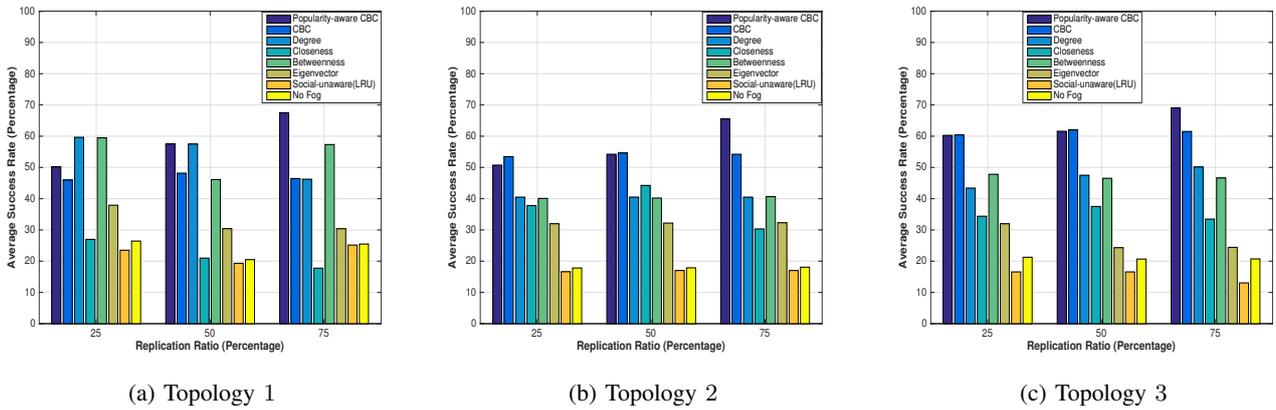| (a) Topology 1 | (b) Topology 2 | (c) Topology 3 |

Figure 5: Success rate comparison for centrality-based (P-CBC, CBC (no popularity), Degree, Closeness, Betweenness, Eigenvector), non-centrality based or social-unaware (LRU) and non-collaborative (no fog) based caching.

On the other hand, the non-centrality based socially-unaware resulted in the poorest performance in all topologies. Similarly, in the case without fog formation, i.e. non-collaboration resulted in poor performance with an overall success rate of less than 20%.

VI. CONCLUSIONS AND FUTURE DIRECTIONS

Our results show that content placement in ICN can be efficiently managed by forming fogs allowing nodes to collaboratively cache at high content-based centrality nodes. "Content" based centrality outperforms the node-centric approach to classify nodes for distributed caching.

In this paper, we targeted the content placement problem at caches near the network edge as in a fog network. To do so, we suggest to exploit graph properties to identify the suitable nodes for caching, however, unlike typical node-centric centrality scheme, we first presented a new popularity-aware content-based centrality scheme, P-CBC, where the content popularity and the number of paths to all content, instead of nodes, counts towards the node centrality. Then, we proposed an algorithm for collaborative content placement in the fog, where the nodes place popular content at high centrality nodes, followed by placing the remaining content with decreasing popularity at nodes with decreasing centrality score, according to a replication rule which allows to compute our proposed centrality without knowing the actual content placement. The benefits of P-CBC are evaluated via simulations on 2,986 nodes, three different topologies along varying three content replication levels. Results show that nodes in the fog based on our centrality outperforms, in terms of cache hit rate and success rate, existing centralities, non-centrality based and non-collaborative caching approaches.

Future work includes moving to dynamic topologies and study the impact of mobility on P-CBC. We plan to use it for content retrieval where the user interests are forwarded to high centrality nodes.

REFERENCES

[1] S. P. Borgatti, "Centrality and network flow," *Social networks*, vol. 27, no. 1, pp. 55–71, 2005.

[2] E. Baştug, M. Bennis, M. Kountouris, and M. Debbah, "Cache-enabled small cell networks: Modeling and tradeoffs," *EURASIP Journal on Wireless Communications and Networking*, vol. 2015, no. 1, 2015.

[3] B. Bharath, K. Nagananda, and H. V. Poor, "A learning-based approach to caching in heterogenous small cell networks," *IEEE Transactions on Communications*, vol. 64, no. 4, pp. 1674–1686, 2016.

[4] K. Hamidouche, W. Saad, and M. Debbah, "Many-to-many matching games for proactive social-caching in wireless small cell networks," in *Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks (WiOpt), 2014 12th International Symposium on*. IEEE, 2014.

[5] V. Sourlas, L. Gkatzikis, P. Flegkas, and L. Tassiulas, "Distributed cache management in information-centric networks," *IEEE Transactions on Network and Service Management*, vol. 10, no. 3, pp. 286–299, 2013.

[6] Y. Wang, Z. Li, G. Tyson, S. Uhlig, and G. Xie, "Design and evaluation of the optimal cache allocation for content-centric networking," *IEEE Transactions on Computers*, vol. 65, no. 1, pp. 95–107, 2016.

[7] Y.-T. Yu, F. Bronzino, R. Fan, C. Westphal, and M. Gerla, "Congestion-aware edge caching for adaptive video streaming in information-centric networks," in *IEEE CCNC Conference*, 2015.

[8] B. Azimdoost, C. Westphal, and H. Sadjadpour, "Fundamental limits on throughput capacity in information-centric networks," *IEEE Transactions on Communications*, 2016.

[9] J. Ren, W. Qi, C. Westphal, K. Lu, J. Wang, S. Liu, and S. Wang, "Magic: a distributed max-gain in-network caching strategy in information-centric networks," in *Proceedings of the IEEE INFOCOM workshop on Emerging Name-Oriented Mobile Networking Design-Architecture, Algorithms, and Applications*, Apr. 2014.

[10] J. Li, H. Chen, Y. Chen, Z. Lin, B. Vucetic, and L. Hanzo, "Pricing and resource allocation via game theory for a small-cell video caching system," *IEEE Journal on Selected Areas in Communications*, 2016.

[11] M. Mangili, F. Martignon, S. Paris, and A. Capone, "Bandwidth and cache leasing in wireless information centric networks: a game theoretic study," *IEEE Transactions on Vehicular Technology*, 2016.

[12] P. Pantazopoulos, I. Stavrakakis, A. Passarella, and M. Conti, "Efficient social-aware content placement in opportunistic networks," *IFIP/IEEE WONS*, pp. 3–5, 2010.

[13] C. Bernardini, T. Silverston, and O. Festor, "Socially-aware caching strategy for content centric networking," in *Networking Conference, 2014 IFIP*. IEEE, 2014, pp. 1–9.

[14] D. Rossi, G. Rossini et al., "On sizing ccn content stores by exploiting topological information." in *INFOCOM Workshops*, 2012, pp. 280–285.

[15] W. K. Chai, D. He, I. Psaras, and G. Pavlou, "Cache "less for more" in information-centric networks," in *International Conference on Research in Networking*. Springer, 2012, pp. 27–40.

[16] J. A. Khan, C. Westphal, and Y. Ghamri-Doudane, "A content-based centrality metric for collaborative caching in information-centric fogs," in *IFIP Networking-ICFC*, 2017.

[17] J. A. Khan and Y. Ghamri-Doudane, "Strive: Socially-aware three-tier routing in information-centric vehicular environment," in *IEEE Globecom 2016*. IEEE, 2016.