Accepted for publication at IEEE JSAC 2018



# Understanding Scoped-Flooding for Content Discovery and Caching in Content Networks

Liang Wang, Suzan Bayhan, Jörg Ott, Jussi Kangasharju, and Jon Crowcroft

***Abstract*—Scoped-flooding is used for content discovery in a broad networking context and it has significant impact on the design of caching algorithms in a communication network. Despite its wide usage, a thorough analysis on how scoped-flooding affects a network's performance, e.g., caching and content discovery efficiency, is missing. To develop a better understanding, we first model the behaviour of scoped-flooding by the help of a theoretical model on network growth and utility. Next, we investigate the effects of scoped-flooding on various topologies in information-centric networks (ICN). Using the proposed ring model, we show that flooding can be constrained within a small neighbourhood to achieve most of the gains which come from areas with relatively low growth rate, i.e., the network edge. We also study two flooding strategies and compare their behaviours. Given that caching schemes favour more popular items in competition for cache space, popular items are expected to be stored in diverse parts of the network compared to the less popular items. We propose to exploit the resulting divergence in availability along with the routers' topological properties to fine tune the flooding radius. Our results shed light on designing both efficient content discovery mechanism and effective caching algorithms for future ICN.**

***Index Terms*—Scoped-flooding, information-centric network, content discovery, graph analysis, optimisation, network protocol, network growth, caching.**

## I. Introduction

One of the biggest challenges mobile network operators have recently been tackling with is the constantly increasing demand for capacity: monthly mobile data traffic is now around a few GBs and expected to reach tens of GBs depending on the region by 2023 [1]. Hence, operators are looking for cost-effective solutions that can increase their network capacity to provide high user satisfaction. While the spectra of proposed solutions are broad from physical layer approaches such as large-scale multi-antenna systems [2] to expansion in the spectrum resources like LTE in the unlicensed bands [3], one idea is to focus on how the Internet is used today and exploit the network entities for storing the most-frequently-requested content. More specifically, this line of proposals, also known as *Information-Centric Networks* (ICN), are based on the fact that users are interested in content more than who is the source of the content, and hence by exploiting the repetitions in the content requests as well as each network element as a content storage unit, we can decrease the burden on the mobile operator's network.

L.Wang (corresponding author: liang.wang@cl.cam.ac.uk) and Jon Crowcroft are with University of Cambridge, UK. S.Bayhan is with TU Berlin and J.Ott is with TU München, Germany. J.Kangasharju is with University of Helsinki, Finland.

Manuscript received December 10, 2017; revised April 8, 2018; accepted April 18, 2018.

An ICN [4]–[7] promises bandwidth efficiency to the network operator and shorter latency to the clients as it supports ubiquitous caching and content delivery from the nearest cache in the neighbourhood of a client. However, content, especially popular content, in an ICN may reside "anywhere", therefore the distribution efficiency heavily relies on the effectiveness of content discovery mechanisms. Considering the gap between large content objects and scarce router resources, designing intelligent content discovery to balance protocol simplicity, computational complexity, and traffic overhead is crucial in every ICN architecture.

Content discovery in ICN is generally achieved by resolution-based [5]–[11] or routing-based [4], [12], [13] solutions, or a hybrid of two approaches [14]. Resolution-based discovery which utilizes a lookup-by-name approach is a deterministic solution which maps requesters with content providers at rendezvous points. The rendezvous point can be either statically configured or referred by proper content addressing [6], [10], [11]. Though resolution-based discovery has relatively small traffic footprint, its performance may degrade quickly in face of large and dynamic content demands. Moreover, fast change in the content location may result in a huge overhead to keep the knowledge on the content location up-to-date [14], [15]. On the other hand, the routing-based discovery usually provides a probabilistic solution. The chances of finding the content can be improved by exploring a larger area of the network, i.e., via collaboration or flooding. In practise, naive network-wide flooding is rarely used due to its significant traffic overhead. A flooding operation is usually constrained within a well-defined neighbourhood (or scope) which is often referred to as *scoped-flooding*. Technically, such constraint on the neighbourhood size is achieved by setting a hop limit for each flooding (e.g., TTL limit).

The use of flooding is based on the following considerations. First, flooding can significantly reduce the protocol complexity and simplify the design, which is very desirable in an unstable environment [16]. Second, in addition to the well-known temporal locality, user requests also possess strong spatial locality [17]. The two localities together indicate that it is highly likely to discover a popular content among nearby nodes. Third, flooding can reduce the state maintained in the network for a routing-based discovery [18]. Fourth, the communication between close neighbours is relatively cheap (regarding delay, transmission cost and etc.) compared to using backhauls in many cases. Therefore, flooding remains as the default fallback strategy for content discovery if normal forwarding fails in CCNx [19], and also used in various routing and caching designs [13], [18], [20]–[23].

Despite its wide application (e.g., in ICN [4], [23], MANET [24], [25]), a thorough understanding of how scoped-flooding affects content discovery is still lacking. More precisely, the following key questions are awaiting answers: (1) what is the optimal radius of scoped-flooding? (2) where do most of the gains come from in a network? (3) how do topological properties of a network impact scoped-flooding? The answers will shed light on designing more intelligent strategies by *flooding for the proper content at the right place with the optimal radius*.

In this paper, to address the aforementioned problems, we propose a node-centric, ring-based model to analyse scoped-flooding. Based on the ring model, we first investigate neighbourhood growth model on general network topologies. The results show that average growth rate increases at least exponentially and can be well estimated using the information within 2-hop neighbourhood. Along with Bayesian techniques, we solve the optimal radius problem and further compare two flooding strategies (static and dynamic) on specific network models.

Specifically, our contributions are threefold:

1) Different than the literature using flooding as a content delivery protocol without a clear understanding of its operation, we provide a theoretical analysis of scoped flooding using a node-centric ring-based model. Our model is independent of the specific ICN architecture which makes our findings relevant for a broad networking context. Later, via simulations, we provide an ICN-specific analysis, e.g., byte hit rate.
2) Modelling the utility of scoped flooding as a function of content availability, node neighbourhood, and cost of flooding, we formulate an optimization problem to derive the optimal scope for each node and content. The analytical results along with the evaluations on realistic network topologies show optimal flooding radius is very small, e.g., mostly no more than 3 hops.
3) Most of the gains of scoped-flooding are from very small neighbourhoods located at network edges, indicating flooding is more proper at the network edge instead of the network core.

The rest of the paper is organized as follows. Section II overviews the related literature and discusses how the current paper differs from the existing studies. Section III introduces the considered setting where scoped flooding is applied for content discovery, while Section IV presents our approach to model the cost incurred by scoped flooding, namely network growth model. Next, Section V formalizes the optimal flooding scope considering content availability both as a priori information and a posteriori. Using this formalization, we derive the optimal radius from a closed-form equation. Section VI explains two strategies for flooding, namely static and dynamic flooding which differ in how they set the flooding radius. Finally, Section VII evaluates the performance of the proposed strategies both using synthetic network models and real network topologies, while Section VIII concludes the paper.

## II. Related Work

We can categorize the literature into two as *resolution-based discovery* and *routing-based discovery*.

Resolution-based discovery [5]–[11] provide deterministic solutions, i.e., at least one copy will be found as long as the content is stored within the network. Therefore, such solutions either require complete knowledge on content distribution and network topology [5], [7], [8] or utilize hash-based content addressing [6], [9]–[11]. Essentially, demands and supplies meet at *rendezvous points* (the actual name differs depending on specific architecture). The rendezvous point either returns a locator or copy of the requested content [5], [6], or redirects the request to one content provider [8], [10], or constructs a distribution topology [7] depending on an actual design. A resolution-based scheme is sometimes referred to as *Name Resolution Service* (NRS) similar to DNS in Internet.

Resolution-based solutions can reach high success rate but have to maintain the states of content distribution in a network hence are confronted with scalability issue when dealing with large and highly dynamic content demands. Therefore, resolution-based discovery schemes consider the scalability of NRS as one of the key goals to achieve in content discovery. A content's location needs to be updated in the NRS to fetch the content from the nearest location so that the overall network traffic is decreased. However, the maintenance requires networked caches to update the NRS about their contents, e.g., addition of a new content in the cache or eviction from the cache. Hence, for an NRS to meet its goal, the entailed control message overhead must not overwhelm the network. To understand under which cases the NRS benefits outweigh the overhead, Azimdoost et al. [15] analyze the overhead due to maintaining the location of content in an ICN as well as the bandwidth savings facilitated by nearest replica routing. Analysis in [15] using rate-distortion theory shows that for large data items resolution based discovery provides bandwidth efficiency as the cost of updating the NRS is markedly lower than the cost of data download. In a similar spirit, Bayhan et. al. [14] design a partial NRS which maintains location information for some items while for the rest, content discovery relies on scoped flooding. To decide on which items the NRS should track, authors consider the cost of content delivery both in terms of network traffic and monetary costs as retrieving content from an external network, e.g., another AS, is more costly than retrieving from within the AS.

Routing-based discovery [4], [12], [13], [26] usually only provides opportunistic solutions, i.e., content will be found with certain probability. The chances of discovery can be improved by either collaborating with nearby nodes [13], [27] or exploring a network via flooding [12], [20], [21], [26], [28]. Both introduce extra traffic overhead. [27] propose using Bloomfilters to exchange information on content availability to improve caching performance. [20], [21], [29] empirically showed that opportunistic flooding can improve content discovery and delivery, also reduce the states maintained in a network. [12], [23], [28], [30] showed that flooding is especially preferred in an unreliable environment to compensate for potential message loss. Empirically or analytically, all [12],

[21], [26], [30] attested that naive flooding is hardly viable in practise, the scope needs to be regulated carefully to reduce the cost. Our work complements these works by providing first a theoretical analysis on the optimal scope and then designing a dynamic-scope setting scheme for flooding based on the node's and requested content's properties.

Scoped-flooding is also applied in opportunistic networks for content search as resolution-based solutions are difficult to realize in such ad hoc networks due to the lack of a reliable infrastructure. As in mobile opportunistic networks, nodes are mostly energy-limited small devices, ensuring efficiency of scoped-flooding without being energy-wasteful is crucial. Motivated by this concern, [31] models the spread of content discovery messages in an opportunistic network under a given hop-limit (similar to the scope in scoped-flooding). Authors [31] show that search scope can be limited to a few hops as after several hops the extra gain diminishes as a function of content availability and tolerated delay. While our conclusions in the current paper agrees with the results of [31], our paper presents a theoretical ground for the observed small scope, as opposed to simulation-based analysis of [31].

Regarding the neighbourhood growth model, besides [32]–[34], another important line of research is *expander graph* [35]. In general, the advances in graph theory has improved our understanding on network graphs and laid the foundation of this work. Nonetheless, none carefully analysed the scoped-flooding for content discovery from network topology perspective, not to mention examining the distribution of gains and improvements within a network. Our analysis attributes the cause of small radius to the exponential growth rate of neighbourhoods, and the evaluation explains why most of the gains are from the network edge by studying the distributions of utility and improvement. To our knowledge, ours is the first study analytically and empirically examining the effects of scoped-flooding on well-defined network models.

## III. System Model

We assume an information-centric network whose topology is represented with a graph $G = (V, \rho)$, where $V$ is a set of nodes characterized with degree distribution $\rho$. $\rho_k$ denotes the probability that a node has exactly degree $k$. The distribution can be arbitrary. For a node $v_i$, we organize its neighbourhood into $r$ concentric rings according to the lengths of shortest paths between $v_i$ and its neighbours. We denote $n_r$ as average number of $r$-hop neighbours on the $r^{th}$ ring. We refer to this model as node-centric ring-based model, or simply a *ring model*. In reality, nodes may break down resulting in lost messages. We denote the stability with $\gamma \in (0, 1]$ to represent the probability that a router is up and working properly, i.e., the reliability rate. Equivalently, $(1-\gamma)$ denotes the failure rate.

Designing a fully-fledged protocol is out of the scope of this paper. Instead, we briefly describe general flooding behaviours in the following. Nodes in a network receive requests from either directly connected clients or neighbours. We exclude clients from the model and focus only on core network. Whenever a request arrives, a node first looks for a match in its local cache. If the node cannot find the requested content

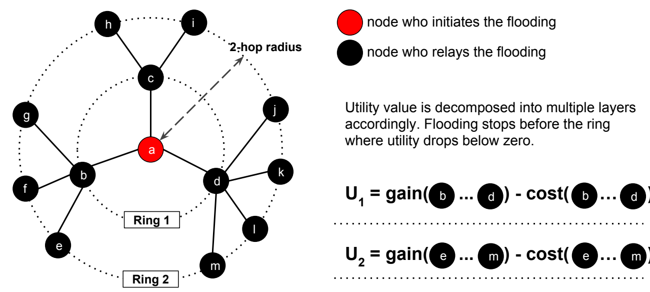



Fig. 1: In a ring model, to estimate whether the utility on the $(r+1)^{th}$ ring drops below zero, a node on the $r^{th}$ ring needs to know three pieces of information (1) which ring it is; (2) how its neighbourhood grows; (3) content availability.

locally, it decides whether to initiate a scoped-flooding before simply forwarding the request to the next hop along the path to original content providers. The flooding is constrained within an $r$-hop neighbourhood by maintaining a hop counter in packet header. The counter is decreased by one every time when passing a node. The node forwards the message on all its network interfaces except the one where the message arrives and terminates flooding if the counter reaches zero. After the content is discovered, we assume it is routed to the initial flooding node in a reverse route similar to CCNx. To prevent loops, nodes do not flood the messages they have seen before.

For simplicity, we do not consider the case of partially matched content. A node $i$ either has the exact requested content or not. The value of response $R$ is described as an i.i.d random variable which follows a Bernoulli distribution $R \sim$ Bernoulli($p$), with 1 representing a successful discovery and 0 otherwise. $p$ is often referred to as *content availability*. Note that $p$ itself can follow different distributions to model content availability (e.g., Zipf or Weibull). By definition, we let $q \triangleq 1 - p$ denote the probability of failing to find the matched content on a node. Fig. 1 shows our ring model.

There are many resource constraints in a network such as energy, bandwidth, and storage. In our model, we use $c$ to represent the cost induced by receiving and processing flooding requests. Besides, consecutive requests may affect content delivery in terms of added queueing and processing delay which we assume to be roughly proportional to the number of messages. However, the cost can increase faster and requests may be dropped in a busy network. To generalise, we model the cost as a linear function of number of nodes involved in flooding.

Intuitively, an efficient scoped-flooding scheme discovers the content in the neighbourhood of the searching node with high probability without resulting in high overhead. To find the optimal scope, we need to model the following three components: cost of scoped flooding, gain of scoped flooding, and utility representing the net benefit under the given cost and gain model. We provide our cost model in Section IV and describe the gain and utility models in Section V.

## IV. Neighbourhood Growth Model

The first step for discovering the optimal radius is to understand how neighbourhood grows as a function of flooding radius. Newman derived this functional relation in [33] using a general graph model $G = (V, \rho)$. We recap briefly the major steps of the derivation in Section IV-A, based on which we investigate two specific types of networks, i.e., random networks and scale-free networks. Then, we examine the accuracy of estimates on both synthetic and realistic networks.

### A. Average Number of $r$-Hop Neighbours

The effective topology due to a flooding can be viewed as a distribution tree. On non-trivial topologies, such a tree cannot be easily decomposed into multiple linear models (from root to leaves). We apply ring model to organize the neighbourhood of node $v$ into $r$ concentric rings according to the neighbour's distance to $v$ so that we can study the growth ring by ring. Calculating $n_1$, namely the average number of directly connected neighbours, is trivial. Let $\langle k \rangle$ denote the mean of a given degree variable $k$. Average number of 1-hop neighbours equals the node's average degree as follows:

$$n_1 = \langle k \rangle = \sum_{k=0}^{\infty} k\rho_k. \tag{1}$$

However, calculating $n_r$ $(r \geq 2)$ is not as straightforward as $n_1$ since the degree distribution of a node's neighbour is not the same as the general degree distribution of the whole network [32]. Let $v_j$ be one of $v_i$'s next-hop neighbours and $\tau_k$ be the probability of $v_j$ having $k$ emerging edges which lead to $k$ new next-hop neighbours. Note that we exclude the edge leading back to $v_i$ from $v_j$ since it does not contribute to new nodes. The results in [34] show $\tau_k$ is proportional to both $v_i$'s degree and general degree distribution of the network. The reason is that the edges of a high-degree node have a higher chance to connect to any given edge in the network. The probability of $v_j$ having $k$ new next-hop neighbours is:

$$\tau_k = \mathbf{Pr}[deg(v_j) = k|\rho] = \frac{(k+1)\rho_{k+1}}{\sum_m m\rho_m}. \tag{2}$$

Therefore, the average number of new nodes from $v_j$ is:

$$\begin{aligned}\sum_{k=0}^{\infty} k\tau_k &= \frac{\sum_{k=0}^{\infty} k(k+1)\rho_{k+1}}{\sum_m m\rho_m} \\ &= \frac{\sum_{k=0}^{\infty} k(k-1)\rho_k}{\sum_m m\rho_m} = \frac{\langle k^2 \rangle - \langle k \rangle}{\langle k \rangle}. \end{aligned} \tag{3}$$

Because we did not assume $v_i$ is on any specific concentric ring except $r \geq 2$, we can use the same $\tau_k$ and the same logic above to calculate arbitrary $r$-hop neighbours. Namely, $n_r$ equals the average number of nodes on the $(r-1)^{th}$ ring multiplied by their average out-degree to the $r^{th}$ ring.

$$\begin{aligned} n_r = n_{r-1} \sum_{k=0}^{\infty} k\tau_k &= \frac{\langle k^2 \rangle - \langle k \rangle}{\langle k \rangle} n_{r-1} \\ &= \left[\frac{\langle k^2 \rangle - \langle k \rangle}{\langle k \rangle}\right]^{r-1} \cdot \langle k \rangle \end{aligned} \tag{4}$$

Eq.(4) shows that the number of $r$-hop neighbours is a function of the degree variable. Using Eq. (4), we can calculate $n_2 = \langle k^2 \rangle - \langle k \rangle$. As we know $n_1 = \langle k \rangle$, by applying the replacement recursively, we can rewrite Eq. (4) as below, which eventually leads us to the same function found in [33].

$$n_r = \left[\frac{n_2}{n_1}\right]^{r-1} \cdot n_1. \tag{5}$$

Eq.(5) shows that $n_r$ can also be expressed as a function of the ratio between average number of 2-hop and 1-hop neighbours. The neighbourhood size only converges if there are fewer 2-hop neighbours than 1-hop ones, i.e., $\frac{n_2}{n_1} < 1$, which implies that the network has multiple components with high probability. We define *neighbourhood growth rate* $\beta$ as:

$$\beta \triangleq \frac{n_2}{n_1} \triangleq \frac{\langle k^2 \rangle - \langle k \rangle}{\langle k \rangle}. \tag{6}$$

Note that the derivation above applies to any general network of arbitrary degree distributions. The result gives an interesting implication on the topological inference, which says that a node can approximate $\beta$ by utilizing the local knowledge within its 2-hop neighbourhood. In the following, we focus on the growth rate $\beta$ and study two specific network models: random network and scale-free network. Both are prominent in networking research due to their representativeness of many realistic networks [36], [37]. Specifically, empirical evidence shows mobile and opportunistic networks can be either random [38] or scale-free [39], whereas fixed and wired networks are mostly scale-free [36].

### B. Case 1: Random Networks

Random networks have a binomial degree distribution $B(|V|, \rho)$ which is given by the following formula [40]:

$$\rho_k = \binom{|V|-1}{k} \rho^k (1-\rho)^{|V|-k-1}.$$

For very big $|V|$ and small $\rho$, the binomial distribution above converges to the Poisson distribution in its limit. Then, the degree distribution $\rho_k$ becomes:

$$\lim_{|V| \to \infty} \rho_k = \frac{\langle k \rangle^k e^{-\langle k \rangle}}{k!}.$$

For calculating $\beta$ in Eq.(6), we need to derive the second moment of random variable $k$, i.e., $\langle k^2 \rangle$. Using Touchard polynomials [1], the $r^{th}$ moment of a variable with Poisson distribution can be calculated as eq. (7) shows. $\left\{ {r \atop k} \right\}$ denotes *Stirling numbers of the second kind* [40] which represents the number of ways to partition a set of $r$ objects into $k$ non-empty subsets, and is known for calculating $\langle k^r \rangle$.

$$\langle k^r \rangle = e^{-\langle k \rangle} \sum_{k=0}^{\infty} \frac{\langle k \rangle^k \cdot k^r}{k!} = \sum_{k=1}^{r} \left\{ {r \atop k} \right\} \langle k \rangle^k \tag{7}$$

Combining Eq. (7) and Eq. (4) yields:

$$n_2 = \left\{ {2 \atop 2} \right\} \langle k \rangle^2 + \left\{ {2 \atop 1} \right\} \langle k \rangle - \langle k \rangle = \langle k \rangle^2. \tag{8}$$

[1]We can also use moment generating functions for a Poisson random variable with parameter $\lambda$, i.e., $M_X(t) = e^{\lambda(e^t-1)}$, and we derive $\langle k^2 \rangle$ by calculating $M''_X(t=0)$. This gives us: $\langle k^2 \rangle = \langle k \rangle^2 + \langle k \rangle$. $\langle k^r \rangle$ can be calculated using higher order moments similarly.

Similarly, by applying the replacement recursively, we get:

$$n_r = \langle k \rangle^r \Longrightarrow \beta = \langle k \rangle \tag{9}$$

Eq. (9) shows that $n_1, n_2, n_3$ ... form a geometric series. The growth rate is $\beta = \langle k \rangle$. It is worth noting that many topological properties (e.g., average degree, density etc.) are homogeneous on random networks. In other words, a randomly chosen sub-network possesses similar characteristics as the whole network which is also known as self-similarity.

### C. Case 2: Scale-free Networks

Although random networks give a very neat form of growth rate, many realistic networks are scale-free and the node degree follows a power-law distribution, i.e., $\rho \propto k^{-\alpha}$ with $\alpha > 2$ [36], [37], [39]. For a power-law distribution, the $r^{th}$ moment of random variable $k$ equals:

$$\langle k^r \rangle = k_{min}^r \cdot \frac{\alpha - 1}{\alpha - 1 - r} \qquad \forall \alpha > r + 1 \tag{10}$$

Note that a power-law distribution is extremely right-skewed and has a heavy tail. Only the first $\lfloor \alpha - 1 \rfloor$ moments exist, the other moments are infinite. If we plug Eq.(10) into Eq.(4) and let $k_{min} = 1$, the growth rate equals:

$$\beta = \frac{1}{\alpha - 3} \qquad \forall \alpha > 3. \tag{11}$$

Eq. (11) means that though most real-life networks have a well-defined average node degree, their variance is infinite, which further indicates the growth rate $\beta$ is unbounded.

For $3<\alpha<4$, the growth rate is bounded but the neighbourhood size never converges. It is also interesting to notice when $\alpha > 4$, $n_r$ converges to zero at its limit $r \to \infty$. The reason is the existence of super hubs with extremely high degrees which strengthens the small-world effect and makes the network diameter extremely short. We refer to [40] for more thorough and interesting discussions on graph topological properties. For both random network and scale-free network, we can see neighbourhood growth is at least exponential which sheds light on the flooding strategy design.

### D. Accuracy on Estimating Neighbourhood Growth Rate $\beta$

In the previous sections, we have provided closed-form equations to calculate $\beta$. However, while modelling network growth, we have not identified multiple counting of the same node to keep our model simple. As one node might be connected to multiple nodes in the previous ring, we expect one node to be counted multiple times. To identify how much inaccuracy this simplification may lead to, we compare the number of neighbours derived from our model with that in the actual network. First, we generate random and scale-free topologies for which we calculate the actual average neighbourhood at each hop distance, i.e, $\bar{n}_r$. To derive the $n_r$ estimated by the model, we first find the parameter of a corresponding degree distribution, i.e., Poisson for Erdős-Rényi random graph and power-law for scale-free graph, by maximum likelihood estimation.[2] After finding the distribution parameter, we calculate $n_r$ using Eq.(9) or Eq.(11) and compute the deviation at the $r^{\text{th}}$ hop as follows:

$$\sigma_r = \frac{(n_r - \bar{n}_r)}{\bar{n}_r}. \tag{12}$$

For both topologies, we set the number of nodes to $N$=10000. If a generated network is not connected, we use the largest component hence $V$ can be smaller than $N$. The link probability parameter $\rho$ determines the number of edges in an Erdős-Rényi graph, similar to $\alpha$ in a scale-free network.

Table I summarizes the network properties along with the deviation, i.e., overestimation ratio. We exclude $r$=1 in the table as all nodes are aware of their one hop neighbours and hence the inaccuracy converges to 0 for all settings. For almost every setting, the model overestimates the reality only slightly for $r = 2$ and $r = 3$. For $V = 339$, we attribute the deviation to both the finite size effect as well as the absence of random graph property, i.e., the network does not exhibit Poisson degree distribution as the model assumes. Increasing hop count makes the model deviate significantly from the reality, especially when $r \geq l$, which is expected as a result of finite network size. For $r = 4$, the model captures the reality quite well for large $V$ and moderate $\langle k \rangle$ – the region where the random graph property exists but the network is not so densely connected. The deviation is higher for the settings with higher $\langle k \rangle$ due to higher clustering and smaller network diameter.

For scale-free networks, Eq.(11) may either underestimate or overestimate depending on the power-law exponent $\alpha$. For $\alpha \approx 3$, the expected growth rate is very large resulting in overestimation in neighbourhood (e.g., topology-7 in Table I). For $\alpha > 3$, the estimated growth is more stable which leads to underestimation of the real growth, e.g, topology-8 and topology-9. We attribute this dispersion to the diversity of the degree distribution in a scale-free network and limitations of our model to represent this diversity accurately.

The ISP networks are smaller, ranging from a couple of hundreds to thousands of nodes [36], which results in a slower growth after certain hops. To understand this effect, we derive the growth rate at $r^{th}$ hop as $\beta_r = \frac{n_{r+1}}{n_r}$ and plot them in Fig.2 for eight ISP networks [36]. Recall that in the analysis we have a single $\beta$ value for the whole networks with $N \to \infty$. As the figure shows, the growth rate decreases with increasing hop due to the finite size of the network. Although the growth rate is a decreasing function of $r$, we can observe in Fig. 2 that the neighbourhood keeps growing for several hops, e.g., $r \approx 5$. $\beta_r$ takes values below 1 for $r$ greater than average path length that varies between 3.36 hops to 5.51 hops. In general, the neighbourhood growth model performs very well within a moderate scope on both synthetic and realistic networks.

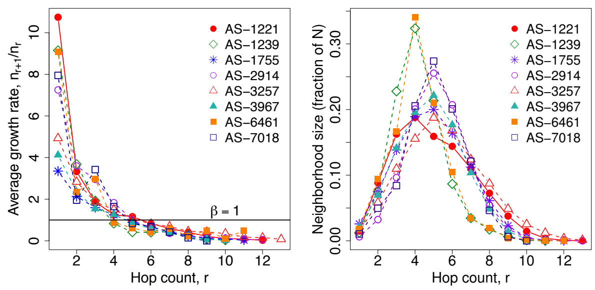
## V. Optimal flooding radius for content discovery

In the previous section, we have introduced the first component, i.e., the cost model, for scoped flooding. Now, we move on to modelling the remaining two components: gain and utility of scoped flooding under a particular scope. Next,

[2]For scale-free networks, we use the method described in [41].

TABLE I: Overestimation of the model at each hop for various network graphs. $V$: Number of nodes and $E$: Number of nodes in the generated instance of the graph, $l$: average path length. Shaded cells represent the cases where the error is below 0.20.

| Id | Topology | V | E | $\langle k \rangle$ | $l$ | Clustering | Overestimation of the model | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | | | $r = 2$ | $r = 3$ | $r = 4$ | $r = 5$ | $r = 6$ |
| 1 | Random | 339 | 338 | 1.994 | 23.07 | 0 | 0.327 | 1.046 | 2.359 | 4.692 | 9.092 |
| 2 | Random | 8030 | 9761 | 2.431 | 12.03 | 0 | 0.152 | 0.371 | 0.642 | 0.972 | 1.399 |
| 3 | Random | 9426 | 15068 | 3.197 | 8.30 | 0.00040 | 0.060 | 0.130 | 0.212 | 0.332 | 0.565 |
| 4 | Random | 9811 | 20073 | 4.091 | 6.75 | 0.00049 | 0.023 | 0.053 | 0.106 | 0.259 | 0.873 |
| 5 | Random | 9928 | 25060 | 5.048 | 5.88 | 0.00048 | 0.004 | 0.017 | 0.079 | 0.419 | 2.79 |
| 6 | Random | 9989 | 35020 | 7.011 | 4.95 | 0.00066 | 0.003 | 0.030 | 0.229 | 2.139 | 54.124 |
| 7 | Scale-free, $\alpha$ =3.24 | 7141 | 9648 | 2.70 | 7.88 | 0.00057 | 0.093 | 0.271 | 0.529 | 1.069 | 2.599 |
| 8 | Scale-free, $\alpha$ =3.35 | 5869 | 7347 | 2.50 | 8.66 | 0.00076 | -0.115 | -0.174 | -0.194 | -0.16 | 0.013 |
| 9 | Scale-free, $\alpha$ =3.50 | 5960 | 7357 | 2.47 | 8.99 | 0.00013 | -0.356 | -0.555 | -0.68 | -0.757 | -0.794 |




(a) Growth rate at each hop. (b) Neighbourhood size growth.

Fig. 2: Change in neighbourhood in real ISP networks. Neighbourhood growth is constrained by the finite size of real networks. The number of neighbours decreases after 4 hops.

we calculate the optimal flooding radius in two cases: with and without prior knowledge on content availability.

### A. *Effective Nodes*

For a scope $r$, we are interested in the number of nodes that will receive the content discovery message to calculate the expected cost of scoped flooding as well as the chances of discovering the content. We use the neighbourhood growth model to derive total number of nodes the flooding messages will reach. However, given that some nodes might fail and become unreachable, we need to tune our model in the previous section according to the failure of each node.

Let $\gamma$ denote the probability that a node is up, namely a node's reliability rate. We define the *effective nodes* $\hat{n}_r$ as the nodes that are working and also reachable on the $r^{th}$ ring. Since only the effective nodes contribute to flooding, i.e., improving content discovery, it is crucial to know the growth of effective nodes for a specific $\gamma$ to derive the optimal radius.

Given a node has $n_1$ 1-hop neighbours, its effective 1-hop neighbours equals $\hat{n}_1 = \gamma n_1$ by assuming a node's state (up or down) is independent of each other. Given growth rate $\beta$, the effective 2-hop neighbours equals $\hat{n}_2 = \beta\gamma^2 n_1$. Similarly, we can calculate $\hat{n}_3$ using $\hat{n}_2$. Applying iteratively, we calculate the effective nodes on the $r^{th}$ ring as follows:

$$\hat{n}_r = (\beta\gamma)^{r-1}\gamma n_1 = \gamma^r n_r. \tag{13}$$

It is easy to see the similarity between Eq.(5) and Eq.(13). In fact, $\hat{n}_1 = \gamma n_1$ is the effective 1-hop neighbours and $\beta\gamma$ can be viewed as effective growth rate given nodes may fail with certain probability $(1 - \gamma)$. For low reliability rates, the gap between the number of effective nodes and the $r$-hop neighbourhood will quickly increase with an increasing $r$.

### B. *Content Availability as A Priori*

Now, let us discuss our gain model. The purpose of flooding is to increase the chance of discovery by visiting enough nodes. Hence, we represent the gain from a flooding as the probability of finding the content. However, the chance of finding the content depends on the likelihood of the content being in the network, which is mostly a function of content popularity and caching scheme. For example, a highly popular content item is more likely to be present in the cache of a node compared to a less-popular item. We represent this fact by content availability metric $p$ which simply represents the probability that this particular content is stored in the cache of a node. Note that content availability as well as content popularity might differ from one region to another as people might have different interests in different regions. Similarly, popularity and availability observed by each network node might differ depending on the node's location in the network, e.g., filtering effect due to cache hits before the request reaches to a node. While our model is valid for all such cases, we will denote availability by $p$ for the sake of simplicity. Given $n$ visited nodes, the probability of finding the content of availability $p$ equals $(1-q^n)$ where $q$=$(1-p)$ is the probability that the content is not available at a node. On the other hand, the resulting cost equals to $n \cdot c$. Then, a bigger $n$ also introduces larger cost which limits the utility $U$ as below:

$$U = (1 - q^n) - n \cdot c. \tag{14}$$

$-U$ in Eq.(14) is convex as an exponential function is convex and the linear combination of convex functions preserves convexity. Our aim is to maximize the utility by tuning the number of visited nodes $n$. The optimal number of nodes $n^*$ can be calculated simply by finding the value of $n$ which maximizes (14) as follows:

$$U'(n) = 0 \Longrightarrow -q^n \cdot \ln q - c = 0 \Longrightarrow n^* = \frac{\ln c - \ln \ln q^{-1}}{\ln q}.$$

After deriving $n^*$, we can calculate the optimal radius by summing up the effective nodes from ring 1 to $r$ and then solving the equation below:

$$\sum_r \hat{n}_r = \sum_r (\beta\gamma)^{r-1}\gamma n_1 = n^*.$$

As the above equality might not always hold, we can relax the equation as:

$$\sum_r (\beta\gamma)^{r-1}\gamma n_1 \leqslant n^*,$$

to find the smallest value of $r$ that ensures the desired number of nodes are reached by the content discovery message.

### C. Inferring the Content Availability

We previously assumed that the content availability $p$ is known *a priori*. Technically, we can set up monitoring nodes to sample request streams. However, monitoring can be expensive and sometimes may not even be feasible. Nevertheless, the probability of finding a specific content in a neighbourhood is a good indicator for its actual availability, since the more popular a content is, the more probable it is to find it among nearby neighbours. The common approach for modelling content availability is to use Che's approximation [42] under the assumption that caching logic is Least-Recently-Used (LRU) and incoming requests follow Independent Reference Model (IRM). However, this model requires the knowledge of popularity of the content items and relies on the assumptions on the IRM request model and LRU caching scheme. For a more generic approach which does not require much knowledge on the node, we use the Bayesian technique proposed in [26] to estimate content availability. Main intuition is that if a content discovery message has followed a path of $i$ hops, then it shows the fact that none of the visited $i$ nodes could satisfy this request. Higher values of $i$ signals that the content's availability is low. With this insight in mind, a node estimates the content availability based on the observed hop count $i$ which can be extracted from the header of the content discovery message. More formally, Eq.(15) is the probability density function of $p$ conditioned on previous $i$ negative (i.e., unsuccessful) queries.

$$f(p|i) = \frac{\mathbf{Pr}(i|p)\cdot f(p)}{\int_0^1 \mathbf{Pr}(i|p)\cdot f(p)dp}, \tag{15}$$

where $\mathbf{Pr}(i|p)$ is the probability that a content discovery message for a content with availability $p$ travels $i$ hops long. Assuming that nodes act independently in managing their cache storage, we calculate $\mathbf{Pr}(i|p)$ as $\mathbf{Pr}(i|p) = q^i$ where $q = 1 - p$. Next, letting $f(p) = 1$, then we have:

$$f(p|i) = \frac{q^i}{\int_0^1 q^i dp} = (i+1)q^i.$$

After getting the posterior of $p$, we can calculate the expected $p$ after $i$ negative queries as below:

$$\langle p\rangle = \int_0^1 p(i+1)q^i dp = \int_0^1 (i+1)(1-q)q^i dq = \frac{1}{i+2}. \tag{16}$$

Note that neither $p$ nor $q$ appears in Eq.(16). Moreover, note that this approach is not the most accurate one to estimate the content availability as discussed in [26]. However, it is practical for realistic networks especially when monitoring is not possible or the content has never been observed before.

### D. Content Availability as Posteriori

Without prior knowledge on content availability, we cannot apply the conventional optimization as that in Section V-B. Even with the Bayesian inference introduced in Section V-C, deciding the optimal radius can be difficult, especially when the request comes from directly connected clients or does not carry any information about the number of nodes it has traversed. To get around this challenge, we let a node flood its 1-hop neighbours by default to bootstrap the inference on $p$. Then, we consider the utility of each ring separately and adaptively adjust the estimate of $p$ on every ring. The general mechanism can be summarized as follows:

**1.** If a request does not contain useful information for estimating the availability (e.g., number of nodes queried), a node initiates a flooding to its directly connected neighbours. A flood message carries three pieces of information: (i) the node's local growth rate $\beta = \frac{n_2}{n_1}$; (ii) number of 1-hop neighbours $n_1$; and (iii) a counter $r$ to record the number of hops it has travelled. [3]

**2.** When a node receives a flood message, it first estimates the availability $p$ using $\beta$, $n_1$ and $r$ embedded in the message by assuming the requested content cannot be found so far (within $r$-hop neighbours). More particularly, as follows:

$$\langle p\rangle = \frac{1}{\beta^{r-1}\gamma^r \cdot n_1}$$

Using this estimated $p$, the node then estimates the potential utility of the next ring. Based on the estimated utility, the node decides whether to continue the flooding or terminate.

More specifically, the overall utility of scoped-flooding is decomposed according to our ring model. Let $R_r$ and $C_r$ represent the aggregated gain and cost on the $r^{th}$ ring respectively, the net utility of flooding is as follows:

$$U = \sum_r U_r = \sum_r (R_r - C_r).$$

According to Eq.(13), the average cost on the $r^{th}$ ring is:

$$E(C_r) = \hat{n}_r c = \gamma^r n_r c$$

and the average value of gross gain $R_r$ is:

$$E(R_r) = 1\cdot(1 - q^{\gamma^r n_r}) + 0\cdot q^{\gamma^r n_r} = 1 - q^{\gamma^r n_r}.$$

The net utility value from the $r^{th}$ ring therefore can be expressed as the difference between $E(R_r)$ and $E(C_r)$, namely:

$$U_r = E(R_r) - E(C_r) = (1 - q^{\gamma^r n_r}) - \gamma^r n_r c. \tag{17}$$

An intermediate node forwards the flooding message to its next-hop neighbours only if the next ring can bring positive net utility, which can be easily tested with Eq.(17). The flooding radius should stop increasing whenever the expected utility of the next ring falls below zero. Technically, this is solved by calculating the root of Eq.(17) which is the maximum number of effective nodes on the $r^{th}$ ring. Note that $U_r \geq 0$ indicates

[3] Note that $\beta$, $n_1$, and $n_2$ here refer to the **local** properties of a specific node instead of the global average. We avoid new notations because the following derivation on optimal radius applies to both local and global cases which is independent on the parameters plugged in. As we will show in Section VI, Dynamic flooding uses local parameters while Static uses global ones.

$c \leq \frac{1-q^{\gamma^r n_r}}{\gamma^r n_r}$ which provides a clear decision boundary on whether to continue a flooding operation. Given $\gamma = 1$, which indicates a stable network of no failures, the root of Eq.(17) above reduces to: $1 - q^{n_r} = n_r c$. The mixture of exponential and polynomial functions can be solved with the Lambert W function, which gives:

$$n_r^* = -\frac{1}{\ln q} W_k\left(\frac{\ln q}{c} e^{\frac{\ln q}{c}}\right) + c^{-1}. \quad (18)$$

$n_r^*$ represents the maximum number of nodes that the $r^{th}$ ring can have in order to keep the cost smaller than the gain. By plugging Eq.(18) into Eq.(4), we can easily derive the optimal flooding radius $r^*$ as a function of cost, content availability, and neighbourhood growth rate.

$$n_r = \beta^{r-1} n_1 = n_r^* \Longrightarrow (r-1)\ln\beta + \ln\langle k\rangle = \ln n_r^* \quad (19)$$

$$\Longrightarrow r^* = \frac{\ln n_r^* + \ln\beta - \ln\langle k\rangle}{\ln\beta} \quad (20)$$

Given $0 < \gamma < 1$, we have the same derivation except $\hat{n}_r$ replaces $n_r$ in Eq.(19). After some manipulations, we have:

$$\hat{n}_r = \gamma^r \beta^{r-1} n_1 = n_r^* \Longrightarrow r^* = \frac{\ln n_r^* + \ln\beta - \ln\langle k\rangle}{\ln\gamma + \ln\beta}. \quad (21)$$

Obviously, $\gamma = 1$ indicates $\ln\gamma = 0$, then Eq.(21) reduces to Eq.(20) as expected. Since $\ln\gamma$ is a monotonically increasing function and only appears in the denominator of Eq.(21), $r^*$ is hence a decreasing function of $\gamma$. In practise, Eq.(21) means the flooding radius tends to be bigger in an unstable network to achieve the same gain. Moreover, for a given reliability $\gamma$, the optimal radius is a decreasing function of growth rate $\beta$.

## VI. Two Flooding Strategies: Static vs. Dynamic Flooding

We first discuss the design rationale for a scoped-flooding scheme, then introduce two strategies for later comparison.

**Design Guidelines:** A good flooding strategy requires that: (1) a node is aware of its neighbourhood with an accurate topological inference; (2) a node is aware of content availability with an accurate statistical inference on user request streams. These two awareness (solved in Section IV and V respectively) together enable a node to decide its optimal flooding radius based on the estimated utility. Additionally, the flooding radius should be adjusted in different parts of the network according to local topological properties as the network structure may not be homogeneous, i.e., some parts are denser and some parts are sparser (regarding degree distribution). Hence, a predetermined radius may lead to suboptimal performance.

**Static Flooding:** Static flooding uses a predetermined and fixed flooding radius for all nodes. The flooding radius is optimized over the whole network topology by e.g., the network operator for each availability value. The average growth rate is calculated using the average number of 1-hop neighbours and 2-hop neighbours of the whole network then plugged into either Eq.(20) or Eq.(21) to derive the optimal radius. Therefore, static flooding ignores the heterogeneity of the topological properties in different areas of the network. Static flooding is simple and popular but it is only suitable for random networks wherein the network structure is homogeneous and nodes have similar growth rates. We include static flooding in our evaluation as a baseline for comparison.

**Dynamic Flooding:** Compared to static flooding, dynamic flooding is more attendant to the differences among the nodes and it assigns a specific radius for each node individually. Considering that the degree distribution in a scale-free network is not homogeneous, dynamic flooding lets each node use its own 1-hop and 2-hop neighbours to calculate the local growth rate. Then, each node optimises locally within its neighbourhood, hence each has its own optimal flooding radius. Such a strategy considers a node's position in a network. Nodes in denser areas tend to have smaller radius while nodes in sparser areas tend to have bigger radius.

For content with lower availability, a node may prefer routing toward the original content provider rather than initiating flooding. To decide on whether apply scoped-flooding or route towards the original content provider, the node needs to calculate the expected gain and loss. For this purpose, by letting $r = 1$, Eq.(17) calculates the availability threshold of whether initiating a flooding as below:

$$U_1 > 0 \Longrightarrow q^{\gamma n_1} < 1 - \gamma n_1 c$$
$$p > 1 - \sqrt[\gamma n_1]{1 - \gamma n_1 c}. \quad (22)$$

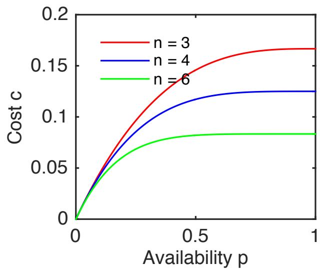
If the availability falls far below the threshold, a node will not flood the request. If content availability is unknown, dynamic strategy floods its 1-hop neighbours by default to bootstrap the inference as described in Section V-D. As we can expect, without content availability information, dynamic flooding is supposed to introduce more overhead due to its aggressive 1-hop flooding. However, the evaluation in Section VII shows that such overhead is almost negligible.

## VII. Performance Evaluation

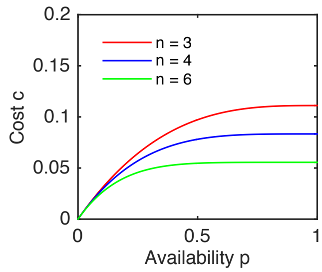
We evaluate the two flooding strategies on various topologies to gain a comprehensive understanding of their pros and cons. Our evaluations focus on two network models: random networks and scale-free networks. Both models have a network of 10,000 nodes and 60,000 edges but their degree distributions are different, namely one is Poisson and the other is power-law. We experimented with a large number of network parameters and various availability and cost values to guarantee the robustness and consistency of our claims. In our evaluations, we used both Matlab and LiteLab [43] for calculating results and simulating scoped-flooding. Nodes apply LRU cache replacement policy.

### A. Impact of Availability and Cost

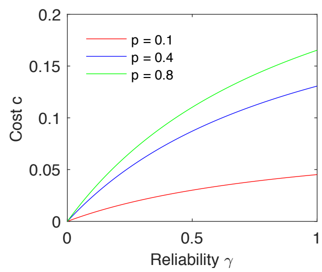
Fig. 3 depicts the model behaviours with different cost and availability parameters. In Fig. 3a and 3b, the curves are the decision boundaries below which a node will initiate flooding for given cost and availability values. The lower cost for higher $n_r$ shows that a node with a large neighbourhood is more parsimonious in flooding compared to a node with a smaller neighbourhood, and only initiates flooding for lower costs. Fig. 3b is a condensed version of Fig. 3a due to setting



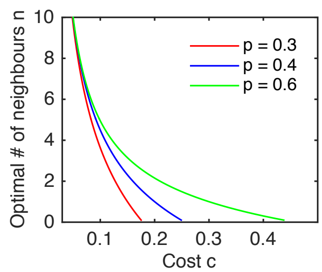

(a) $c$ vs. $p$, $\gamma = 1$.




(b) $c$ vs. $p$, $\gamma = 0.5$.

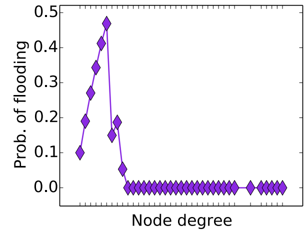



(c) $c$ vs. $\gamma$, $n = 3$.

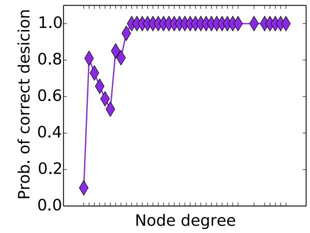



(d) $n$ vs. $c$, $\gamma = 1$.

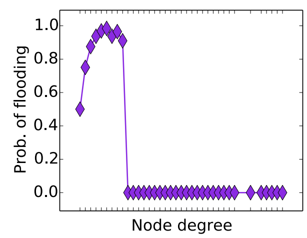
Fig. 3: Ring model behaviours with different $p$, $c$, and $\gamma$.



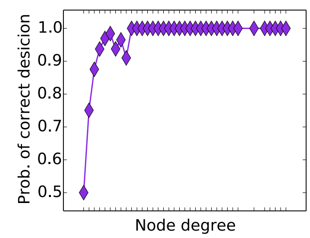

(a) Flooding decision for low availability.




(b) Correctness of scoped flooding decisions for low availability.




(c) Flooding decision for high availability.




(d) Correctness of scoped flooding decisions for high availability.

Fig. 4: In a real topology, the probability that a node decides to initiate scoped flooding and the probability that the given decision, i.e., flood or terminate, agrees with the decision under exact knowledge on the content availability.

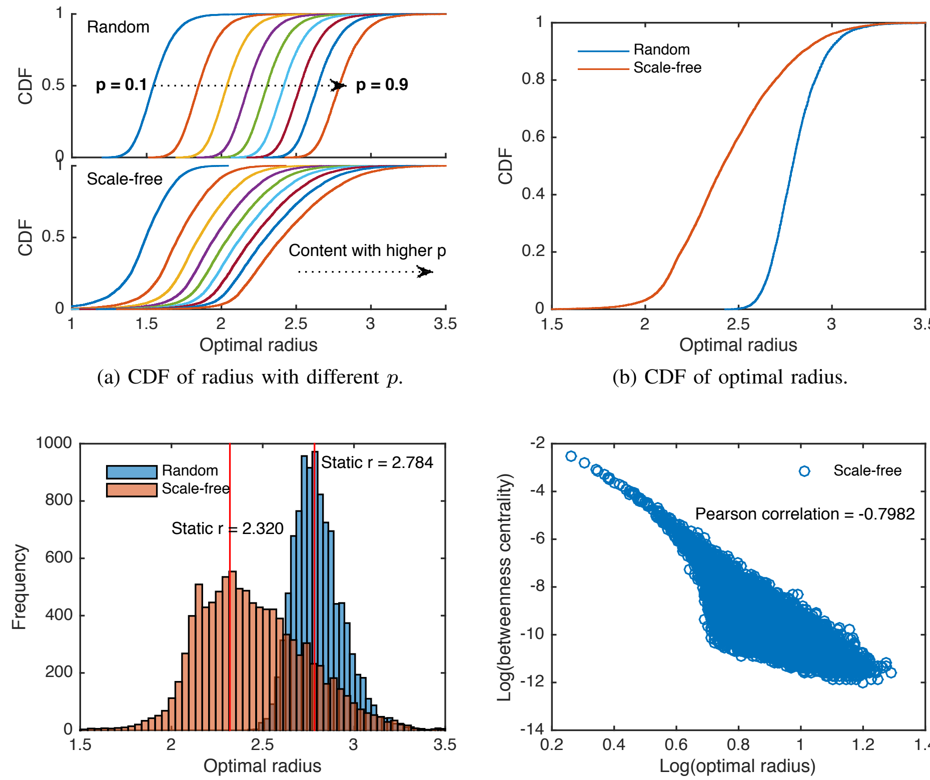
$\gamma = 0.5$, indicating that an unstable network cannot tolerate high cost values. For both figures, the steep increase in the interval $[0, 0.4]$ indicates the strong preference on high available content in scoped-flooding. Fig. 3c indicates that a reliable network has high tolerance on cost especially for popular content, which also explains the difference between Fig. 3a and Fig. 3b. Fig. 3d shows when the cost slightly increases from its minimum (i.e., zero), the optimal number of neighbours drops drastically regardless of content availability. After a certain point, e.g., 5 or 6 nodes, the figure shows a slower decrease for content with high availability. Fig. 3d indicates that for higher availability content, it is worth flooding to more neighbours since the content will be discovered with high probability therefore the gain is guaranteed. For low availability content, even with relatively low cost, the flooding is rather conservative.

Next, we analyze how nodes in a real network topology decide to start flooding or fall backs to default algorithm, which is on-path routing toward the content publisher. For this purpose, we use network topologies from Rocketfuel ISP networks and consider three content availabilities—low, medium, and high with availability values $p$=0.1, $p$=0.3, $p$=0.5 as an example. We set the cost as $c = 0.1$. Using Monte Carlo simulations of $10^4$ runs, we derive the fraction of times each node initiates flooding for the incoming content discovery messages. For this decision, we use Eq.(22). Each node estimates the availability of the content by collecting reports periodically from its neighbours. Simply, content availability equals to the ratio of nodes storing the requested content over total number of neighbours. At each Monte Carlo run, we change the content placement and put the content to each node with probability $p$. We want to investigate: (i) whether scoped-flooding is preferred or not, (ii) how the availability affects it, (iii) which nodes prefer flooding which not, and (iv) how does the inaccuracy in availability estimation affect the decision. For (iv), we take the decision as correct that is based on real availability and compare it with the decision under estimated availability.

We report the median of the runs for AS-2914 as other topologies exhibit similar behaviour. Fig.4 depicts the probability that scoped-flooding is initiated as well as the probability of the decision being correct. As observed in Fig.4a, high degree nodes refrain them from flooding as it would be very wasteful. Low degree nodes are also reluctant to flooding. We attribute this behaviour to the estimated low availability of the content from their neighbourhood. If we check the accuracy of such decisions, we see in Fig.4b that the given decisions are correct with high probability. For a high-availability content, high-degree nodes are again conservative and terminate flooding whereas the lower degree nodes are more willing to flood as there is a higher chance that the content will be in the neighbourhood. As Fig.4d, the scoped flooding decisions have higher likelihood of being correct compared to the content with low availability. For low-degree nodes, the accuracy of availability estimations is expected to be lower due to lower number of samples from the node neighbourhood. Consequently, we observe lower correctness in scoped-flooding decisions.

### B. Flooding Radius Distribution

Fig. 5a shows the CDF of nodes' optimal radii using different $p$ on both random (upper figure) and scale-fee (lower figure) networks. By increasing $p$ from 0.1 to 0.9, the CDF curves shift toward right indicating high available content is worth large radius. In random network, the shapes of the




(a) CDF of radius with different $p$.

(b) CDF of optimal radius.

(c) Histogram of optimal radius.

(d) Radius vs. betw. centrality.

Fig. 5: Optimal radius distribution in dynamic flooding, the radius negatively correlates to nodes' betweenness centrality.

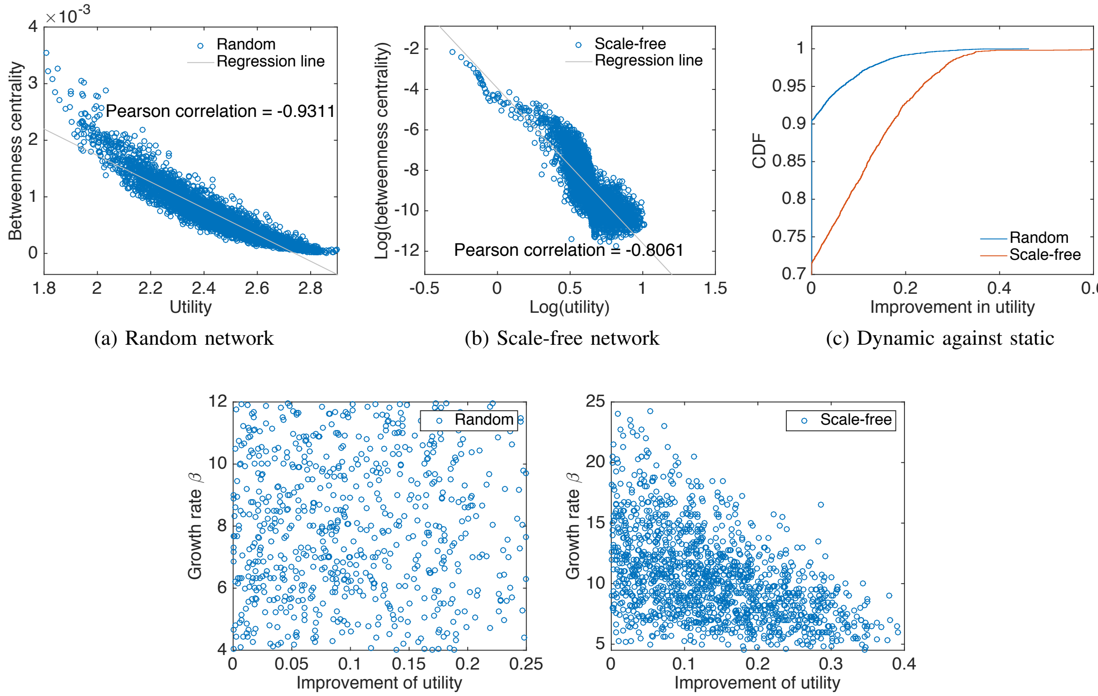
curves are mostly identical, whereas in scale-free network, the curves are more stretched for higher $p$ values, which indicates more heterogeneity in scale-free networks. Fig. 5b and 5c specifically plot the radius distributions for $p = 0.8$. The scale-free network has smaller flooding radii than the random network in both dynamic and static flooding. The two red vertical lines in Fig. 5c represent the optimal radii of static flooding, i.e., $2.320$ for scale-free and $2.783$ for random network. For dynamic flooding, the mean and variance of the radii are $2.447$ and $0.094$ on the scale-free (red area), and $2.805$ and $0.014$ on the random (blue area). Static flooding ignores the difference of topological characteristics between two nodes. Fig. 5b shows that over $60\%$ of the nodes in scale-free network have a radius less than $2.5$ whereas almost all the nodes' radii in random network are bigger than $2.5$. In both Fig. 5b and 5c, the left tail of scale-free network is heavier than that of random network due to the existence of high-degree nodes. The radius distribution of random network is more condensed in a smaller range (reflected as a small variance $0.014$) because of its homogeneous structure.

There is a relatively strong negative correlation between optimal radius and node's degree as well as optimal radius and node's betweenness centrality. We report the results for betweenness centrality in Fig. 5d as a function of radius for the scale-free network. Note the logarithmic scale in the axis. We attribute the negative correlation to the nodes with high betweenness centrality that are located in the well-connected parts of the network wherein the link density is very high and therefore the radius is small due to the high growth rate.

### C. Utility and Its Improvement Distribution

Inspired by [44], Fig.6a and 6b plot betweenness centrality as a function of a node's utility. Please note that Fig.6b is log-log plot. We observe a strong negative correlation between the two variables, utility and betweenness centrality of the node. The corresponding Pearson correlation coefficient can reach $-0.93$ and $-0.80$ for random and scale-free network, respectively. The reason for the negative correlation is that, in the dense area where a node has a high betweenness centrality value, its neighbourhood size is usually large. Although the radius is also small, the node may still include more neighbours than necessary (the optimum) which renders a higher cost and drags down the net utility. Sometimes, even 1-hop neighbours include too many nodes, which in fact highlights the need for more conservative flooding schemes such as selective flooding to only some of the node's neighbours based on some criteria. As the growth rate in the sparser area is much lower, nodes have a better control over the neighbourhood size by fine-tuning their radius leading to lower cost and better net utility.

Dynamic flooding aims at providing better system performance than static flooding in terms of higher utility value. To compare dynamic flooding against static one, we let $U_{dy}$ denote the optimal utility achieved by dynamic flooding and $U_{st}$ by static flooding. Then, we calculate the utility improvement as: $\frac{U_{dy}-U_{st}}{U_{st}}$. Fig. 6c plots the CDF of the utility improvement. We notice that dynamic flooding is less




(a) Random network (b) Scale-free network (c) Dynamic against static

(d) Random network (e) Scale-free network

Fig. 6: In dynamic flooding, the utility distribution strongly correlates to nodes' positions. However, regarding improvements, the strength of correlation between the significance of improvement and the position depends on the network model.

effective on random networks, only $10\%$ of the nodes actually improve their performance and over half have less than $10\%$ improvement. Such lower effectiveness of dynamic flooding is due to the homogeneous structure of the random network. As we showed in the previous section, the static optimal radius deviates from the dynamic optimal radius only slightly for a random network. Hence, the improvement in utility is marginal. On the other hand, nodes in a scale-free network have much more significant utility improvement, namely about $30\%$ of the nodes are improved, among which over $60\%$ have larger than $10\%$ improvement.

Specifically, we take a closer look at those nodes with improved utility, i.e., the $10\%$ in the random and $30\%$ in the scale-free network. Fig. 6d and 6e plot local growth rate $\beta$ as a function of improvement. Note the difference in both X-range and Y-range of the two figures. As for X-range, the utility shows a wider range of improvement in scale-free networks due to the diverse growth rate of the nodes shown on the Y-axis. Scale-free network has a larger $\beta$ due to hub nodes compared to the random network with more homogeneous node characteristics. Fig.6d shows that the correlation between $\beta$ and the utility improvement on random network is close to zero, more precisely $-0.0031$, indicating that the significance of improvement is irrelevant of a node's growth rate and its position in the network. Meanwhile, such correlation on scale-free network is much stronger, with Pearson correlation of $-0.5273$. The results show that nodes with high growth rate are less likely to have significant benefit by using dynamic flooding. The reason is that the optimal radii of the nodes with high $\beta$ values in both static and dynamic flooding are small and close to each other. However, dynamic flooding usually significantly increases the radius of the nodes with low $\beta$.

TABLE II: Results for each metric are in the format of ***n**etwork-wide*—***s**tatic*—***dy**namic* flooding. The cost is measured by the # of flood messages and normalized with the maximum value.

| AS | Byte hit rate | | | Cost | | | Avg. hops | | |
|---|---|---|---|---|---|---|---|---|---|
| | nw | st | dy | nw | st | dy | nw | st | dy |
| 1239 | 0.44 | 0.40 | 0.43 | 1.0 | 0.27 | 0.28 | 1.90 | 1.60 | 1.62 |
| 2914 | 0.49 | 0.42 | 0.47 | 1.0 | 0.31 | 0.32 | 1.75 | 1.55 | 1.58 |
| 3356 | 0.42 | 0.39 | 0.42 | 1.0 | 0.25 | 0.27 | 2.02 | 1.69 | 1.74 |
| 7018 | 0.47 | 0.41 | 0.45 | 1.0 | 0.26 | 0.28 | 1.87 | 1.54 | 1.63 |
| Guifi | 0.51 | 0.44 | 0.49 | 1.0 | 0.22 | 0.23 | 1.71 | 1.32 | 1.38 |

## *D. Flooding in the Wild*

To confirm our analysis on realistic networks, we choose four realistic ISPs along with one community network (i.e., Guifi Catalunya region) [36], [37] to compare network-wide, static, and dynamic flooding. Note that dynamic flooding is not aware of content availability but use the inference technique in the evaluations. In case there are multiple components in a network, we use the biggest one. The smallest network (AS 3356) has 3,107 nodes and 6,097 edges while the biggest one (AS 7018) has 9,732 nodes and 10,047 edges. Each node is equipped with a $4$ GB cache using LRU for cache replacement, and assigned a pair of geographical coordinates according to the topology traces. The content set is based on the Youtube Entertainment Category trace [45] which contains 1,687,506 objects (average size is $8.0$ MB and aggregated size is 12.87 TB). The trace contains video id, length, views, rating and etc. All the nodes of degree 1 are considered as content requesters while 10 to 20 content providers (proportional to the network size) are randomly selected among the nodes in the network. A node cannot be a requester and a provider at the same time. To take into account both temporal and spatial locality, we use Hawkes process-based algorithm [46] with different *spatial locality factors* $[0, 1)$ to generate user request streams. Locality factor 0 indicates the request pattern reduces to *Independent Reference Model*, 1 indicates high spatial localisation. The "warm-up" period for pre-filling the caches is excluded from an evaluation and the result is averaged over at least 50 runs.

Table II reports the results with spatial locality factor $0.5$. Although network-wide flooding always achieves the best byte hit rate, the improvement is rather marginal over dynamic flooding (less than $5\%$). Such a small gain in cache hits is at the price of $2{\sim}3$ times increase in the number of control messages as the second column shows. Intuitively, without prior knowledge on content availability, the performance of dynamic flooding is mostly affected by network topology and should be worse than static flooding which can exploit such knowledge. The results however show that dynamic flooding consistently outperforms static one, which further attests the effectiveness of Bayesian inference and justifies the design of dynamic flooding. Compared to static one, dynamic flooding has slightly higher cost because it tends to explore more nodes (recall the default flooding to 1-hop neighbours in dynamic flooding), which explains its gain in byte hit rate. The third column shows the average hops between a requester and the first discovered content. Network-wide flooding has the worst values and static flooding is slightly higher than dynamic one. In all cases, most content are discovered within 2 hops.

We investigate the effects of spatial locality by varying the factor between 0 and 1. For network-wide flooding, spatial locality does not appear to have any noticeable impacts on the byte hit rate and the cost except that high factor values lead to shorter average hops. Higher spatial locality improves byte hit rate and average hop count in both static and dynamic flooding. For dynamic flooding, by increasing the locality factor from $0.1$ to $0.9$, the byte hit rate improves $9\%{\sim}22\%$ and essentially reaches the performance of network-wide flooding. The average hops metric has $7\%{\sim}19\%$ improvement. In terms of byte hit rate, the difference of all three strategies becomes smaller as the locality factor increases but dynamic flooding consistently outperforms static one by at least $7.5\%$. Meanwhile, the cost of all three strategies almost remain unchanged. The reason is that content availability and local topological property are the determinant factors of the cost (due to being a function of $p$ and $r$) in static and dynamic flooding respectively, neither will be affected by changing the locality factor.

### *E. Summary and Discussion*

Our results indicate that dynamic flooding is more effective on the networks of heterogeneous topological structure, and most of the gains come from sparse areas wherein local growth rate is low, namely at network edges. The optimal flooding radius in a dense area is small and nodes suffer from high flooding cost. On realistic networks for which the evaluations further take the spatial locality of content into account, dynamic flooding is consistently superior to static one even without prior knowledge on content availability, and quickly approaches the byte hit rate of network-wide flooding but with much smaller cost in control messages.

The small cost of dynamic flooding compared to the static one can be further eliminated by keeping track of the success rate of flooding to enhance the inference on availability, so that dynamic flooding is able to avoid unnecessary flooding on unpopular content. Developing better inference techniques of content availability is reserved as our future work.

The neighbourhood growth model and results presented in this paper have profound implications on the design of caching algorithms in a communication network. Especially for the collaborative caching algorithms such as [13], [47], the nodes collaboratively exchange information to improve their local knowledge on the global content distribution in the network, which can further help them in making better caching decisions to increase hit rate and reduce the traffic footprint. Even though the scoped-flooding can accelerate the dissemination of information on content distribution, it also introduces the communication cost.

If a collaborative caching algorithm utilizes the global optimization, the induced communication cost from network-wide flooding can be very high. But, our results indicate that network-wide global optimisation is not needed in practise and small collaboration radius can already lead to nearly optimal solutions as shown in [47]. With small collaboration radius, we can significantly reduce the communication overhead.

## VIII. CONCLUSION

This paper aims to comprehend scoped-flooding for content discovery in an information-centric network. We first proposed a model to mimic how a content discovery message spreads in a network. Next, using the proposed ring model, we studied the functional relation between the neighbourhood growth and flooding radius, based on which we derived the optimal search radius. With the acquired insights from the theoretical analysis, we proposed two strategies, i.e., static and dynamic flooding, for setting the scope of the content discovery messages. Both our theoretical analysis and empirical evaluations suggest that due to the exponential growth of neighbourhood size, the optimal flooding radius is usually very small (i.e., a couple of hops). Most of the gains of flooding come from the sparse area at the network edge where the neighbourhood growth rate is low. To certain extent, our results justify the rationale of deploying caches at network edge from content discovery perspective. Dynamic flooding is consistently superior to static one, especially on scale-free networks. With strong spatial locality, the performance of dynamic flooding quickly converges to network-wide flooding but with much smaller cost.

As future work, we acknowledge that the following aspects need further investigation: (1) Our neighbourhood growth model does not take clustering coefficient into account which leads to overestimation in small networks. (2) Our utility model assumes sub-linear gain and linear cost which requires further reality checks. (3) Other in-network caching schemes can be more effective than simple LRU and a thorough comparison is needed to gain a deeper understanding.

## REFERENCES


[1] "Ericsson mobility report," Tech. Rep., November 2017, available at https://www.ericsson.com/en/mobility-report/reports/november-2017. , last accessed Dec.8, 2017.

[2] K. Zheng, L. Zhao, J. Mei, B. Shao, W. Xiang, and L. Hanzo, "Survey of large-scale MIMO systems," *IEEE Communications Surveys & Tutorials*, vol. 17, no. 3, pp. 1738–1760, 2015.

[3] H. Cui, V. C. Leung, S. Li, and X. Wang, "LTE in the unlicensed band: Overview, challenges, and opportunities," *IEEE Wireless Communications*, 2017.

[4] V. Jacobson, D. K. Smetters, J. D. Thornton, M. F. Plass, N. H. Briggs, and R. L. Braynard, "Networking named content," in *Proceedings of the 5th ACM Conext*. New York, NY, USA: ACM, 2009, pp. 1–12.

[5] C. Dannewitz, D. Kutscher, B. Ohlman, S. Farrell, B. Ahlgren, and H. Karl, "Network of information (netinf) - an information-centric networking architecture," *Comput. Commun.*, vol. 36, no. 7, pp. 721–735, Apr. 2013. [Online]. Available: http://dx.doi.org/10.1016/j.comcom.2013.01.009

[6] M. D'Ambrosio, C. Dannewitz, H. Karl, and V. Vercellone, "MDHT: a hierarchical name resolution service for information-centric networks," in *Proceedings of the ACM SIGCOMM workshop on Information-centric networking*, ser. ICN '11, 2011, pp. 7–12.

[7] D. Trossen, G. Parisis, K. Visala, B. Gajic, J. Riihijarvi, P. Flegkas, P. Sarolahti, P. Jokela, X. Vasilakos, C. Tsilopoulos, and S. Arianfar, "PURSUIT Conceptual Architecture: Principles, Patterns and sub-Components Descriptions," PURSUIT, Tech. Rep., May 2011.

[8] T. Koponen, M. Chawla, B.-G. Chun, A. Ermolinskiy, K. H. Kim, S. Shenker, and I. Stoica, "A data-oriented (and beyond) network architecture," *SIGCOMM Comput. Commun. Rev.*, vol. 37, no. 4, pp. 181–192, 2007.

[9] D. G. Thaler and C. V. Ravishankar, "Using name-based mappings to increase hit rates," *IEEE/ACM Trans. Netw.*, vol. 6, no. 1, pp. 1–14, Feb. 1998.

[10] L. Wang, O. Waltari, and J. Kangasharju, "Mobiccn: Mobility support with greedy routing in content-centric networks," in *2013 IEEE Global Communications Conference (GLOBECOM)*, Dec 2013, pp. 2069–2075.

[11] S. Roos, L. Wang, T. Strufe, and J. Kangasharju, "Enhancing compact routing in CCN with prefix embedding and topology-aware hashing," in *Proceedings of the 9th ACM workshop on Mobility in the evolving internet architecture*. ACM, 2014, pp. 49–54.

[12] M. Varvello, I. Rimac, U. Lee, L. Greenwald, and V. Hilt, "On the design of content-centric manets," in *International Conference on Wireless On-Demand Network Systems and Services (WONS)*, Jan 2011, pp. 1–8.

[13] W. Wong, L. Wang, and J. Kangasharju, "Neighborhood search and admission control in cooperative caching networks," in *IEEE Global Communications Conference (GLOBECOM)*, 2012, pp. 2852–2858.

[14] S. Bayhan, L. Wang, J. Ott, J. Kangasharju, A. Sathiaseelan, and J. Crowcroft, "On content indexing for off-path caching in information-centric networks." in *ACM ICN*, 2016, pp. 102–111.

[15] B. Azimdoost, C. Westphal, and H. R. Sadjadpour, "Resolution-based content discovery in network of caches: Is the control traffic an issue?" *IEEE Trans. on Communications*, vol. 65, no. 7, pp. 2943–55, July 2017.

[16] P. Eugster, R. Guerraoui, A.-M. Kermarrec, and L. Massoulie, "Epidemic information dissemination in distributed systems," *Computer*, vol. 37, no. 5, pp. 60–67, May 2004.

[17] S. Traverso, M. Ahmed, M. Garetto, P. Giaccone, E. Leonardi, and S. Niccolini, "Temporal locality in today's content caching: why it matters and how to model it," *ACM SIGCOMM CCR*, vol. 43, no. 5, pp. 5–12, 2013.

[18] E. Baccelli, C. Mehlis, O. Hahm, T. C. Schmidt, and M. Wählisch, "Information centric networking in the iot: Experiments with ndn in the wild," in *Proceedings of the 1st International Conference on Information-centric Networking*, ser. ICN '14, 2014, pp. 77–86.

[19] PARC, "Ccnx 1.0 protocol specifications," PARC, Tech. Rep., 2014.

[20] M. Badov, A. Seetharam, J. Kurose, V. Firoiu, and S. Nanda, "Congestion-aware caching and search in information-centric networks," in *Proceedings of the 1st International Conference on Information-centric Networking*, ser. ICN '14, 2014, pp. 37–46.

[21] R. Chiocchetti, D. Rossi, G. Rossini, G. Carofiglio, and D. Perino, "Exploit the known or explore the unknown?: Hamlet-like doubts in ICN," in *Proceedings of the ACM Workshop on Information-centric Networking*, 2012.

[22] J. Garcia-Luna-Aceves, "Name-based content routing in information centric networks using distance information," in *Proceedings of the 1st International Conference on Information-centric Networking*, ser. ICN '14. New York, NY, USA: ACM, 2014, pp. 7–16. [Online]. Available: http://doi.acm.org/10.1145/2660129.2660141

[23] A. Sathiaseelan, L. Wang, A. Aucinas, G. Tyson, and J. Crowcroft, "Scandex: Service centric networking for challenged decentralised networks," in *Proceedings of the ACM DIY Networking*, 2015, pp. 15–20.

[24] J. Hassan and S. Jha, "Optimising expanding ring search for multi-hop wireless networks," in *Global Telecommunications Conference, 2004. GLOBECOM '04. IEEE*, vol. 2, Nov 2004, pp. 1061–1065 Vol.2.

[25] J. Deng and S. Zuyev, "On search sets of expanding ring search in wireless networks," *Ad Hoc Netw.*, vol. 6, no. 7, pp. 1168–1181, Sep. 2008.

[26] E. Hyytiä, S. Bayhan, J. Ott, and J. Kangasharju, "Searching a needle in (linear) opportunistic networks," in *ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems*, ser. MSWiM '14, 2014, pp. 187–196.

[27] M. Lee, J. Song, K. Cho, S. Pack, J. Kangasharju, Y. Choi *et al.*, "Content discovery for information-centric networking," *Computer Networks*, 2014.

[28] N. Chang and M. Liu, "Controlled flooding search in a large network," *Networking, IEEE/ACM Transactions on*, vol. 15, no. 2, pp. 436–449, April 2007.

[29] G. Rossini and D. Rossi, "Coupling caching and forwarding: Benefits, analysis, and implementation," in *Proceedings of ACM ICN*, ser. ICN '14. New York, NY, USA: ACM, 2014, pp. 127–136.

[30] C. Liu and J. Wu, "An optimal probabilistic forwarding protocol in delay tolerant networks," in *ACM International Symposium on Mobile Ad Hoc Networking and Computing*, ser. MobiHoc '09, 2009, pp. 105–114.

[31] S. Bayhan, E. Hyytiä, J. Kangasharju, and J. Ott, "Two hops or more: On hop-limited search in opportunistic networks," in *ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWiM)*, 2015, pp. 115–124.

[32] M. Molloy and B. Reed, "A critical point for random graphs with a given degree sequence," *Random Struct. Algorithms*, vol. 6, no. 2-3, pp. 161–180, Mar. 1995.

[33] M. E. J. Newman, *Random graphs as models of networks*, 2005, pp. 35–68.

[34] S. L. Feld, "Why your friends have more friends than you do," *The American Journal of Sociology*, vol. 96, no. 6, pp. 1464–1477, May 1991.

[35] S. Hoory, N. Linial, and A. Wigderson, "Expander graphs and their applications," *Bulletin of the American Mathematical Society*, vol. 43, no. 4, pp. 439–561, 2006.

[36] N. Spring, R. Mahajan, and D. Wetherall, "Measuring ISP topologies with rocketfuel," in *Proceedings of ACM SIGCOMM*, 2002.

[37] D. Vega, L. Cerda-Alabern, L. Navarro, and R. Meseguer, "Topology patterns of a community network: Guifi.net," in *IEEE WiMob*, Oct 2012, pp. 612–619.

[38] R. Hekmat and P. Van Mieghem, "Connectivity in wireless ad-hoc networks with a log-normal radio model," *Mob. Netw. Appl.*, vol. 11, no. 3, pp. 351–360, Jun. 2006.

[39] E. Yoneki, P. Hui, and J. Crowcroft, "Distinct types of hubs in human dynamic networks," in *Proceedings of the 1st Workshop on social Network Systems*. ACM, 2008, pp. 7–12.

[40] S. Bornholdt, H. G. Schuster, and J. Wiley, *Handbook of graphs and networks*. Wiley Online Library, 2003, vol. 2.

[41] A. Clauset, C. R. Shalizi, and M. E. Newman, "Power-law distributions in empirical data," *SIAM review*, vol. 51, no. 4, pp. 661–703, 2009.

[42] H. Che, Y. Tung, and Z. Wang, "Hierarchical web caching systems: Modeling, design and experimental results," *IEEE Journal on Selected Areas in Communications*, vol. 20, no. 7, pp. 1305–1314, 2002.

[43] L. Wang, A. Sathiaseelan, J. Crowcroft, and J. Kangasharju, "Litelab: Efficient large-scale network experiments," in *IEEE Annual Consumer Communications Networking Conference (CCNC)*, Jan 2016, pp. 60–66.

[44] L. Wang, S. Bayhan, and J. Kangasharju, "Effects of cooperation policy and network topology on performance of in-network caching," *IEEE Communications Letters*, vol. 18, no. 4, pp. 680–683, 2014.

[45] M. Cha, H. Kwak, P. Rodriguez, Y.-Y. Ahn, and S. Moon, "I tube, you tube, everybody tubes: analyzing the world's largest user generated content video system," in *Proceedings of the 7th ACM SIGCOMM conference on Internet measurement*, ser. IMC '07, 2007, pp. 1–14.

[46] A. Dabirmoghaddam, M. M. Barijough, and J. Garcia-Luna-Aceves, “Understanding optimal caching and opportunistic caching at ”the edge” of information-centric networks,” in *International Conference on Information-centric Networking*, ser. ICN ’14, 2014, pp. 47–56.

[47] L. Wang, G. Tyson, J. Kangasharju, and J. Crowcroft, “Milking the cache cow with fairness in mind,” *IEEE/ACM Transactions on Networking*, vol. 25, no. 5, pp. 2686–2700, Oct 2017.

**Liang Wang** is a senior research associate in the Computer Laboratory at University of Cambridge, United Kingdom. In 2003, he received his BEng in Computer Science and Mathematics from Tongji University, Shanghai, China. Later, he received both his MSc and PhD degrees in Computer Science from University of Helsinki, Finland in 2011 and 2015 respectively. Liang’s research interests include system and network optimisation, modelling and analysis of complex networks, information-centric networks, and distributed data processing, and so on.

**Suzan Bayhan** received her PhD in computer engineering in 2012 from Bogazici University, Istanbul. Between 2012-2016, she was a postdoctoral researcher at University of Helsinki. Currently, she is a senior researcher at TKN, TU Berlin and also a docent in computer science at University of Helsinki. She received the Google Anita Borg EMEA scholarship in 2009 and co-authored the best paper at ACM ICN 2015. Suzan is on N2Women Board as one of the mentoring co-chairs. Her current research interests include resource allocation in wireless networks, coexistence in the unlicensed spectrum, information-centric networks, and edge/fog communications.

**Jörg Ott** holds the Chair of Connected Mobility in the Department of Informatics at the Technical University of Munich. He is also Adjunct Professor for Networking Technology at Aalto University. He received his diploma and doctoral (Dr.-Ing.) degree in computer science from TU Berlin (1991 and 1997, respectively), and his diploma in industrial engineering from TFH Berlin (1995). His research interests are in network architecture, (Internet) protocol design, and networked systems, with a focus on (mobile) decentralized applications and services.

**Jussi Kangasharju** received his MSc from Helsinki University of Technology in 1998. He received his Diplome d’Etudes Approfondies from the Ecole Superieure des Sciences Informatiques in Sophia Antipolis. In 2002 he received his PhD from University of Nice Sophia Antipolis/Institut Eurecom. In 2002 he joined Darmstadt University of Technology, first as post-doctoral researcher, and from 2004 onwards as assistant professor. Since June 2007 Jussi is a Professor at University of Helsinki.

**Jon Crowcroft** (SM95F04) received the B.S. degree in physics from University of Cambridge in 1979, and the MSc degree in computing and PhD degree from UCL in 1981 and 1993, respectively. He has been a Professor at the University of Cambridge since 2001, working in Internet support for multimedia communications. Prof. Crowcroft is a Fellow of the Royal Society, the Association for Computing Machinery (ACM), the British Computer Society, the IET, and the Royal Academy of Engineering.