Contents

Acknowledgements 3

Abstract 4

1 Introduction and Related Work 5

2 Network Model 11

3 Communication Scheme and Analysis 13

4 Optimization Problem 26
   4.1 $m_2 = \Theta(M)$ .............................. 32
   4.2 $m_2 = o(M)$ .............................. 34

5 Serving Locally 36
   5.1 Optimization Problem .......................... 38

6 Contents with different sizes 40

7 Hybrid networks over a shared channel 42

8 Simulations and Experiments 50
   8.1 Simulation Setup .............................. 50

9 Conclusions 55

Appendix A 56

Appendix B 58

References 59
Acknowledgements

I would like to thank my advisor Prof. Edmund Yeh for his consistent support and patient guidance throughout this project. I have learned a lot from Prof. Yeh’s keen insight of complicated problems and his useful suggestions.
Abstract

We study the throughput and delay characteristics of wireless networks based on the Named Data Networking architecture, where users are mainly interested in retrieving content stored in the network, rather than in maintaining source-destination communication. Nodes are assumed to be uniformly distributed in the network area. Each node has a limited-capacity Content Store, which it uses to cache contents according to the proposed caching scheme. We consider a content-centric traffic model with a general content popularity distribution, where users use multihop communication to retrieve the requested content from the closest cache.

Our study begins with precisely establishing the definitions of throughput and delay suitable for content-centric networks. Then, we derive the throughput-delay tradeoff of the proposed network paradigm for a general content popularity distribution, and formulate the problem of joint optimization of caching and forwarding strategies. We, then, evaluate the network performance for a Zipf content popularity distribution, letting the number of content types and the network size both go to infinity. In addition, we extend our analysis in various directions, considering contents with different sizes and hybrid network scenarios. Finally, through extensive simulations, we verify our theoretical results.
1 Introduction and Related Work

Today’s Internet has an hourglass architecture centering on a global network layer, IP, which implements the minimal functionality needed for universal interconnectivity [1]. The communication network design of IP was aimed to solve point-to-point communication problems, while recent growth in e-commerce, digital media, and smartphone applications has resulted in the Internet mostly being used as a distribution network. The main issues arising from this incompatibility between these two models are [2]

- Availability: Reliable and fast content delivery needs pre-planned and application-specific mechanisms such as CDNs and P2P networks, and/or requires excessive bandwidth capacity.
- Location-dependence: Mapping content to host locations requires complicated configuration and implementation of network services.

The direct and unified approach to solve these issues is by replacing where with what. i.e., we need to replace the traditional client-server communication model with one based on the identity of data. Named data networking (NDN) [1], and content-centric networking (CCN) [2], are two proposed network architectures for the Internet that replaces the traditional host-to-host communication paradigm with one in which the main networking functionalities are directly driven by object identifiers, rather than host addresses. This change is derived from the primary use of today’s Internet. Today’s Internet end users are mostly interested in obtaining the data they want, rather than where it is located. Aiming at redesigning the entire Internet architecture, including core routers, with named data as the central element of the communication, NDN has gained a lot of attention over the past few years. An essential element of NDN, or in general content delivery networking (CDN), is the caching strategy. That is, how many copies of the available contents to store in the network and where. The advantages of in-network caching
have been demonstrated in different networking models, such as P2P, and Publish-Subscribe Networks.

In order for NDN to accomplish content delivery, two types of packets are being used. Nodes requesting data objects generate *Interest Packets* (IPs), which contain the data names and a random nonce value. Then, the IPs are forwarded along the routes determined locally by the *Forwarding Information Base* (FIB) at each node. The FIB determines to which neighboring node(s) one should transmit the IP. Received IPs are recorded in the Pending Interest Table (PIT) at each node, thus allowing repeated requests for the same object to be suppressed. Also, by using nonce value contained in each IP, duplicate IPs received over different paths may be discarded. Nodes can have the ability to store some of the data objects they receive in their *Content Stores* (CSs) according to a caching strategy. Upon reception of an IP, if the node has the requested content in its CS, it creates a Data Packet (DP) containing the requested data object. The DP is, then, transmitted back along the path taken by the corresponding IP, as recorded by the PIT at each node. Intermediate nodes along the path as well as the requesting nodes may optionally cache the content contained in the received DPs, so that they can fulfill the received IPs requesting the same content in the future.

Another major technology in networking is wireless communications. As number of mobile users are growing exponentially, there has been a great interest in the field of wireless networks and how they perform in various scenarios. Gupta and Kumar [3] first introduced a random network model for studying throughput scaling in a static wireless network. In their random network model, \( n \) nodes are distributed independently and uniformly on a unit disk. Each node has a randomly chosen destination node and can transmit at \( W \) bits per second provided that the interference is sufficiently small, using a Protocol model. Thus, each node can be simultaneously a source, a destination, and a relay for other source-destination pairs. They showed that
the throughput of the proposed network model scales as $\Theta(1/\sqrt{n \log n})$ per source-destination pair. Their result also showed that cooperation among users is necessary to conquering the limiting effects of interference. In [11], A. El Gamal, et al. showed that the optimal throughput-delay trade-off is given by $D(n) = \Theta(nT(n))$, where $T(n)$ and $D(n)$ are the throughput and delay scaling, respectively.

As shown in [3], the maximum common rate sustainable for all flows in the network scales inversely proportional to the number of hops. This issue causes wireless networks not sustain long multihop communication schemes. Therefore, it is inevitable that content-centric networking will also plays an important role in the wireless domain. There has been several recent application ideas of using device-to-device, opportunistic communications among mobile users to reduce the downlink traffic in cellular networks [4], [5].

Also, there has been some efforts to evaluate the performance and scalability of CDN-based networks in the literature [6],[7]. In [6], the asymptotic properties of the joint delivery and replication problem in static grid-based wireless networks with multihop communications and caching has been investigated. They studied a problem formulation in terms of minimizing the average link capacity subject to the content replication constraints and derived the scaling laws of link capacity when content popularity follows the Zipf distribution. In their network model, wireless nodes are located statically on a grid, while in our model, nodes are randomly distributed in the network area. In addition, we have managed to calculate the more practical performance parameters, delay and throughput as well as their tradeoff of the proposed network and verify our theoretical results through experimental simulations.

In [7], the performance of a mobile ad-hoc network operating in a content-centric scenario for the Zipf content popularity distribution has been analyzed based on a priori known average delay and idle time intervals. Then, using a transmission power control scheme, the asymptotic behavior of the network
in terms of throughput and delay is given. In their network paradigm, nodes are moving, and they have assumed that the average delay of the network, $\bar{D}$, is known a priori and more importantly, used it along with an average idle time interval, $\bar{I}$, to calculate the node throughput, $\lambda$, given by

$$\lambda = \frac{1}{\bar{I} + \bar{D}} \quad (1)$$

However, using the Jensen’s inequality we know that this equation provides us only with the lowerbound of the average throughput. Then, the optimization problem is solved subject to this lowerbound.

In [8], the content placement problem in a wireless femto-cellular network with using helper nodes was studied. Shanmugam, et al, formulated the problem as the minimization of the average of the total downloading delay for a given content popularity distribution and network topology, reflected by the connectivity graph and by the link average rates. They show that the uncoded optimum file assignment is NP-hard, and develop a greedy strategy that is provably within a factor 2 of the optimum.

Ji, et al, in [9], analyzed base station-assisted D2D wireless networks with caching capability. They proposed a cellular grid network model in which communication between wireless nodes are only single-hop, and derived the asymptotic throughput-outage tradeoff of the network model.

In this paper, we are proposing an NDN-based wireless random network, in which we make the following contributions:

1. Since the previous definitions of the throughput and delay in the source-destination communication paradigms are not suitable for the content-delivery network models, We establish precise definitions for the throughput and delay of the proposed network.

2. In previous wireless networks analysis, it was assumed that the request for content has been received by the source. Hence, they only considered the downloading phase, in which data packet is transmitted toward
the requesting node. However, since the Interest Packet in NDN (or CCN) is containing the data name which can be a long name, we cannot simply ignore the first phase where the IP is transmitted toward the closest cache. Therefore, in this paper we are considering a two-way communication, rather than a simplistic one-way transmission.

3. We first establish the throughput-delay tradeoff of the proposed network for a general content popularity distribution and then formulate the optimization of joint caching and forwarding strategies for minimizing the average delay of the network.

4. Deriving the optimal caching and forwarding strategies obtained from the optimization problem, we evaluate the network performance assuming Zipf content popularity distribution in Section 4.

5. Then, we extend the achieved results in various directions. In Section 5, we consider studying the improvements of network performance when we allow nodes to satisfy their requests for particular contents from their local cache, if they have cached it already. Even though, this capacity seems to be obviously implied when analyzing in-network caching models, however, in previous works this subject has been avoided due to the complications of its analysis.

6. We also consider the case where contents can have different sizes in Section 6. As in real world, different contents have different sizes, and in order to be transmitted to the requesting node, they need to be chunked up into some standard sized packets and then, be forwarded toward the intended destination. In this paper, we study the effects of different-sized contents on the average delay and throughput of the network.

7. Moreover in Section 7, we consider the hybrid wireless networks in which, there are multiple Base Stations in the network area, in addition
to the mobile nodes. In this case, we will give answer to these questions:

- How many Base Stations are required to improve the asymptotic performance of the network paradigm?
- How good does the hybrid network perform under different content popularity distributions?

8. Finally in Section 8, through extensive simulations, we verify our theoretical results, and demonstrate the credibility of information theoretic results in practical network scenarios.
2 Network Model

We assume a random wireless network with \( n \) nodes distributed uniformly in a unit-sized torus. Our analysis is based on the protocol model introduced by Gupta and Kumar [3]. Each node transmits at rate \( W \) bits per second, which is a constant, independent of \( n \). Each node has a local cache, named Content Store (CS), of size \( K \) units of content. There are \( M \) contents, each with the same size (one unit). We assume that the number of contents, \( M \), scales as \( n^\beta \), where \( 0 < \beta < 1 \).

In order to avoid scheduling packets in the network, we let the content sizes, or equivalently the sizes of IPs and the DPs, to be arbitrary small. We refer to this as the fluid model, similarly defined in [11]. In this model, the data sent in a time-slot could correspond to multiple packets. Hence, the time spent for a packet transmission may only be a small fraction of the time-slot itself. Furthermore, a packet received by a node in some time-slot cannot be transmitted by the node until the next time-slot. Therefore, by allowing the size of packets scales proportionally with the throughput, all the packets arrived waiting for transmission will be transmitted in one time-slot.

We are considering the case where \( K < M \), in order for the problem of replication not to be trivial. That is, each node has to select which files to cache. Also, for the network to have sufficient memory to store each file at least once, we need to have

\[
nK \geq M \tag{2}
\]

We are assuming that nodes, independent from one another, generate requests for data. Each request is directed to a particular content \( m \in M \), depending on the content \( m \)'s popularity, \( p_m \). That is, the probability of content \( m \) be the requested content for a given node is \( p_m \) for \( m \in \{1, 2, \ldots, M\} \). Also, the content popularity distribution \( [p_m] \), is assumed to be time-invariant and independent of the requesting nodes. Due to the time-invariant assumption, we can consider static caching allocations. That is, the caching alloca-
tions are performed at the initial stage. Let $\chi_m$ be the set of nodes which have cached content $m$ in their CS. We call these nodes the *holders* of content $m$ and let $X_m = |\chi_m|$ be the number of holders of content $m$. 
3 Communication Scheme and Analysis

Content delivery in NDN is accomplished using two types of packets [10]. Requests for data objects lead to the creation of Interest Packets (IPs), which are forwarded along routes determined by the Forwarding Information Base (FIB) at each node. The FIB tells the node where to forward the IP. Received IPs are recorded in the Pending Interest Table (PIT) at each node so that when they receive the Data Packet (DP) from the upstream node, they can send the DP along the path that the corresponding IP has come from. When a node receives an IP for a content which it is cached in the local CS, it creates a DP containing the requested data object and transmits it back to the downstream node from which has received the corresponding IP.

In order for a transmission to be successful, certain requirements need to be met. Here, we consider the well-known Protocol Model [3], in which the conditions for successful transmissions are postulated geometrically.

Definition 1. The Protocol Model: All nodes use a common transmission range $r(n)$ for all their communications. Suppose node $i$ transmits an information packet at time $t$. Then, a node $j$ can receive this packet successfully if and only if for any other transmitting node $k$ in the network, $|U_i(t) - U_j(t)| \leq r(n)$ and $|U_k(t) - U_j(t)| > (1 + \Delta) r(n)$, where $U_i(t)$ is the 2D location of node $i$ at time $t$ and $\Delta$ is a positive constant.

Lemma 1. In order to ensure that there is no isolated node in the network [3], the transmission range $r(n)$ must satisfy

$$r(n) = \Omega \left( \sqrt{\frac{\log n}{n}} \right)$$

In this paper, we assume that all $n$ nodes in the network act as requesters. Also, time is assumed to be slotted and $t = 0, 1, 2, \ldots$. As explained before, there exist $X_m$ holders which have cached content $m$ in their CS. Since we do not assume any geographical dependency on the content popularities, then
without loss of optimality, these $X_m$ holders are uniformly distributed in the network area. Assume that routing (topology discovery and data reachability) has already been accomplished in the network, so that the FIBs have been populated for the contents. Hence, we can assume that the IP will be directed toward the closest holder to the requester, in terms of Euclidean distance. Since the nodes are static in the network, the DP will be downloaded from the closest holder to the requester, using the reverse path taken by the IP.

To study the asymptotical behavior of the network, we will let the number of nodes and content types both go to infinity. We say an event holds with high probability (w.h.p.) if the event occurs with probability 1 as $n$ goes to infinity.

**Definition 2.** Scheme: A scheme $\pi$ for a random network is a sequence of caching and communication policies, $\{\pi_n\}$, where policy $\pi_n$ determines how caching allocation and communication takes place in a network of $n$ nodes.

**Definition 3.** Throughput: For a given scheduling and caching scheme $\pi$ and content $m \in \{1, 2, \ldots, M\}$, let $B_{\pi,m}(i, t)$ be the total number of bits of Data Packets of content $m$ received by the requesting node $i$ from the closest holder of content $m$ up to time $t$ and let $B_{\pi}(i, t)$ be the total number of bits of all the Data Packets received by the requesting node $i$ up to time $t$. Thus

$$B_{\pi}(i, t) = \sum_{m=1}^{M} B_{\pi,m}(i, t)$$

(4)

The long-term throughput of content $m$ for node $i$ is

$$\liminf_{t \to \infty} \frac{1}{t} B_{\pi,m}(i, t).$$

Similarly, the total long-term throughput of node $i$ is

$$\liminf_{t \to \infty} \frac{1}{t} B_{\pi}(i, t).$$
The average throughput of content \( m \) and the average per-node throughput for a given network realization and content request sequence, over all nodes of the network are given, respectively

\[
\lambda'_{\pi,m}(n) = \frac{1}{n} \sum_{i=1}^{n} \lim_{t \to \infty} \frac{1}{t} B_{\pi,m}(i,t).
\]

(5)

\[
\lambda'_\pi(n) = \frac{1}{n} \sum_{i=1}^{n} \lim_{t \to \infty} \frac{1}{t} B_{\pi}(i,t).
\]

The throughput of content \( m \), \( \lambda_{\pi,m}(n) \), and the per-node throughput, \( \lambda_\pi(n) \), are defined as the expectation over all network realizations (initial locations of the nodes), and content request sequences of the average throughputs over all nodes of the network. That is,

\[
\lambda_{\pi,m}(n) \triangleq E[\lambda'_{\pi,m}(n)]
\]

\[
\lambda_\pi(n) \triangleq E[\lambda'_\pi(n)]
\]

Using (4), we have this relationship between the per-node throughput and the throughput of content \( m \):

\[
\lambda_\pi(n) = \sum_{m=1}^{M} \lambda_{\pi,m}(n)
\]

(6)

**Definition 4.** Delay: For a given scheduling and caching scheme \( \pi \) and content \( m \in \{1, 2, \ldots, M\} \), let \( D_{\pi,m}(i,k) \) be the delay for the \( k \)-th request of content \( m \) by node \( i \), and let \( D_\pi(i,k) \) be the delay of the \( k \)-th requested content by node \( i \) (measured from the moment IP leaves the requester for the closest holder until the corresponding DP(s) arrives at the requester from
the holder). The delay of content \( m \) for node \( i \) is

\[
\limsup_{r \to \infty} \frac{1}{r} \sum_{k=1}^{r} D_{\pi,m}(i,k).
\]

Similarly, the average delay of node \( i \) is given by

\[
\limsup_{r \to \infty} \frac{1}{r} \sum_{k=1}^{r} D_{\pi}(i,k).
\]

By averaging the delay for content \( m \) and the delay for all the contents for a given network realization and content request sequence over the nodes of the network, we obtain respectively

\[
D'_{\pi,m}(n) = \frac{1}{n} \sum_{i=1}^{n} \limsup_{r \to \infty} \frac{1}{r} \sum_{k=1}^{r} D_{\pi,m}(i,k).
\]

\[
D'_\pi(n) = \frac{1}{n} \sum_{i=1}^{n} \limsup_{r \to \infty} \frac{1}{r} \sum_{k=1}^{r} D_{\pi}(i,k).
\]

The packet delay of content \( m \), \( D_{\pi,m}(n) \), and the packet delay of the network, \( D_\pi(n) \), are defined as the expectation over all network realizations (initial locations of the nodes), and content request sequences of the average delays over all nodes of the network. That is,

\[
D_{\pi,m}(n) \triangleq E[D'_{\pi,m}(n)] \quad \text{for all } m \in \{1, 2, \ldots, M\}
\]

And,

\[
D_\pi(n) \triangleq E[D'_\pi(n)]
\]

In addition, it is easy to show that the packet delay of the network is equal
to the expectation of the delay of content $m$ over all the $M$ contents. i.e.,

$$D_\pi(n) = \sum_{m=1}^{M} p_m D_{\pi,m}(n)$$ (7)

Note that $B_{\pi,m}(i,t)$ and $D_{\pi,m}(i,k)$ are both random variables and depend on the node locations in the network area as well as the content request sequences. However, the throughputs $\lambda_{\pi,m}$, $\lambda_\pi$ and delays $D_{\pi,m}$, $D_\pi$ are ensemble averages.

For a given content popularity distribution, $[p_m]$ and given sets of holders $\chi_m$, we want to have $X_m$ nodes cache content $m$ for $m \in \{1, 2, \ldots, M\}$, in order to serve the requests of other nodes, as well as their own requests for this content. The summation of $X_m$’s for all $m \in \{1, 2, \ldots, M\}$ should be less than or equal to the total caching capacity of the network, $nK$. This constraint is a relaxed version of the per-node caching capacity constraint, in which each node can cache up to $K$ different contents. Moreover, each node can cache at most one copy per content in its CS.

If a node is requesting a content which is cached in its local CS, the request will be satisfied immediately and there will be no need to generate an IP. Since the CS is a limited cache space, this is not usually the case. We are addressing this case in Section 5. For now, we are assuming that if the requested content is in the local cache, the node still generates an IP for it, transmits it to the nearest holder excluding itself, and uses the network to retrieve the content. Later in Section 5, we consider the nodes’ ability to serve the requests locally and shows that the delay and throughput of both cases scales similarly.

Before presenting the tradeoff scheme, first, we iterate two lemmas proved in [11] about the geometry of $n$ nodes uniformly distributed on a unit-sized torus divided into square cells of area $a(n)$. Then, we present Lemma 4, in which we show the average length to the closest holder. Finally in Lemma 5, we prove a generalized version of Lemma 3 in [11], where we calculate
the upperbound of the total communication lines passing through each cell w.h.p..

**Lemma 2.** If $a(n) \geq 2 \log n/n$, then each cell has at least one node w.h.p..

**Lemma 3.** Under the Protocol model, the number of cells that interfere with any given cell is bounded above by a constant $N = 16(1 + \Delta)^2$, independent of $n$.

**Lemma 4.** Let $\chi_m$ be the set of holders of content $m$, uniformly distributed in the unit-sized network area and $X_m = |\chi_m|$. For any node requesting content $m$, the average Euclidean distance to its closest holder is $\Theta\left(\frac{1}{\sqrt{X_m}}\right)$.

**Proof.** Since the holders are uniformly distributed, the probability that no holder is in the distance of less than or equal to $\tau$ of the requester is $\Pr(d \geq \tau) = (1 - \pi \tau^2)^{X_m}$ for $0 \leq \tau \leq 1/\sqrt{\pi}$. Therefore, the average distance of the closest holder to the requester is

$$
E[d] = \int_0^\infty \Pr(d \geq \tau) d\tau
$$

$$
= \int_0^{\frac{1}{\sqrt{\pi}}} (1 - \pi \tau^2)^{X_m} d\tau
$$

Using variable exchange $\sqrt{\pi}\tau = \cos \theta$ and applying integration by parts, we

---

$^1$whether the requester is one of the holders or not, this Lemma still holds true, noting that in the case where the requester is one of the holders, the distance to the closest holder excluding the requester itself is of order $\Theta\left(\frac{1}{\sqrt{X_m-1}}\right) = \Theta\left(\frac{1}{\sqrt{X_m}}\right)$. 

---
have

\[
E[d] = \frac{1}{\sqrt{\pi}} \int_0^{\pi/2} \sin^{2X_m+1} \theta d\theta
\]

\[
= \frac{1}{\sqrt{\pi}} \frac{2X_m}{2X_m+1} \cdot \frac{2X_m-2}{2X_m-1} \cdot \cdots \cdot \frac{2}{3} \cdot \int_0^{\pi/2} \sin \theta d\theta
\]

(8)

\[
= \frac{1}{\sqrt{\pi}} \frac{2X_m}{2X_m+1} \cdot \frac{2X_m-2}{2X_m-1} \cdot \cdots \cdot \frac{2}{3}
\]

(9)

\[
= \Theta\left(\frac{1}{\sqrt{X_m}}\right)
\]

(10)

where (8) has derived from

\[
\int \sin^n x \, dx = -\frac{1}{n} \sin^{n-1} x \cos x + \frac{n-1}{n} \int \sin^{n-2} x \, dx
\]

(11)

Also, (10) has been proved in Appendix Appendix A.

In this paper, we are assuming that each node can transmit to nodes within its cell and to the 8 neighboring cells. To provide the necessary condition, each cell of the network must have an area of \(a(n) = r^2(n)/8\). Assume a fixed node \(i\), randomly chosen out of the \(n\) nodes of the network, is requesting content \(m\). Let \(L_{H,R}(i,m)\) be the straight line connecting \(i\) to the closest holder of content \(m\). From Lemma 4, we know that the average length of \(L_{H,R}(i,m)\) is of order \(\Theta\left(\frac{1}{\sqrt{X_m}}\right)\). i.e., for \(1 \leq i \leq n\) and \(1 \leq m \leq M\) we have

\[
E[|L_{H,R}(i,m)|] = \Theta\left(\frac{1}{\sqrt{X_m}}\right).
\]

(12)

where \(|L|\) denotes the length of line \(L\). Let \(H_{i,m}\) be the number of hops along a path (sequence of nodes) which belongs to the set of cells intersecting the \(L_{H,R}(i,m)\) line between the requester \(i\), and the closest holder of content \(m\), where exactly one node per cell relays a particular packet along the path. To compute \(E[H_{i,m}]\), we need to separate the two cases where the holder is
within one hop of the requester of it is farther than one hop. We have

\[
E[H_{i,m}] = E[H_{i,m}|L_{H,R}(i,m)| \leq \sqrt{a(n)}] \Pr(|L_{H,R}(i,m)| \leq \sqrt{a(n)}) \\
+ E[H_{i,m}|L_{H,R}(i,m)| > \sqrt{a(n)}] \Pr(|L_{H,R}(i,m)| > \sqrt{a(n)})
\]

(13)

Clearly, \(E[H_{i,m}|L_{H,R}(i,m)| \leq \sqrt{a(n)}] = 1\). Also, since the side-length of each cell is \(\sqrt{a(n)}\), it can be shown that \(E[H_{i,m}|L_{H,R}(i,m)| > \sqrt{a(n)}] = \Theta(E[|L_{H,R}(i,m)|]/\sqrt{a(n)}) = \Theta(1/\sqrt{a(n)X_m})\). Denoting \(\Pr(|L_{H,R}(i,m)| > \sqrt{a(n)}) = \alpha(n)\), from (13) we obtain

\[
E[H_{i,m}] = 1 + \left[ \frac{1}{\sqrt{a(n)X_m}} - 1 \right] \alpha(n)
\]

(14)

Now, using a similar argument of the one in the proof of Lemma 4, \(\alpha(n) = \Pr(d > \sqrt{a(n)}) = (1 - \pi a(n))^X_m\). Note that as \(n \to \infty\), \(a(n) \to 0\) and \(X_m \to \infty\). Expanding \(\alpha(n)\) using the binomial form, and noting that \(\binom{n}{k} \leq \frac{n}{k!}\), for \(n \geq k \geq 1\), we have

\[
1 + \sum_{i=1}^{X_m} (-1)^i \frac{(a(n)X_m)^i}{i^i} \leq \alpha(n) \leq e^{-a(n)X_m}
\]

(15)

Now, taking \(n \to \infty\), for \(X_m = \Omega(1/a(n))\), \(e^{-a(n)X_m} \to 0\), and hence, \(\alpha(n) \to 0\). Applying this result in (14) it follows that \(E[H_{i,m}] = 1\). On the other hand, for \(X_m = o(1/a(n))\), \(a(n)X_m \to 0\), resulting in both bounds given in (15) to converge to 1, as \(n \to \infty\). Substituting \(\alpha(n) = 1\) in (14) gives us

\[
E[H_{i,m}] = \frac{1}{\sqrt{a(n)X_m}}
\]

Therefore, the average number of hops \(^2\) can be written

\(^2\)Note that \(H_{i,m}\) is not necessarily the minimum number of hops along the path, as we are restricting the scheme to have exactly one node per cell relaying a particular packet. While in the minimum hops scheme, at most one node per cell relays a packet. Although, it can be shown that the minimum number of hops along the path, denoted as \(H'_{i,m}\), is, in fact, of the same order as given in (17) for any \(i\) and \(m\).
as
\[
E[H_{i,m}] = \begin{cases} 
\Theta(1) & X_m = \Omega(a^{-1}(n)), \\
\Theta \left( \frac{1}{\sqrt{a(n)X_m}} \right) & X_m = o(a^{-1}(n)), 
\end{cases}
\]

We can, then, merge the two expressions and obtain
\[
E[H_{i,m}] = \Theta \left( \max \left\{ \frac{1}{\sqrt{a(n)X_m}}, 1 \right\} \right), \forall m \in \{1, 2, \ldots, M\} \quad (16)
\]

**Lemma 5.** For \( a(n) \geq 2 \log n/n \), the number of \( L_{H,R} \) lines passing through each cell is \( O(n \sum_{m=1}^{M} p_m \max \{ \sqrt{a(n)/X_m}, a(n) \} ) \) w.h.p.

**Proof.** For a given content request sequence \((m_1, m_2, \ldots, m_n)\) at time \( t \) and a given node \( i \), we know that \( H_{i,m_i} = H_{i,m} \), w.p. \( p_m \), for \( m = 1, 2, \ldots, M \). Therefore, we have
\[
E[H_{i,m_i}] = \sum_{m=1}^{M} p_m E[H_{i,m}]
\]
\[
= \Theta \left( \sum_{m=1}^{M} p_m \max \left\{ \frac{1}{\sqrt{a(n)X_m}}, 1 \right\} \right) \quad (17)
\]

There are \( 1/a(n) \) cells. Fix a cell \( j \) and define \( Y_{i,m_i}^j \) to be the indicator of the event that the \( L_{H,R}(i, m_i) \) line passes through cell \( j \). That is,
\[
Y_{i,m_i}^j = \begin{cases} 
1 & \text{if } L_{H,R}(i, m_i) \text{ passes through cell } j, \\
0 & \text{otherwise,}
\end{cases}
\]

for \( 1 \leq i \leq n, 1 \leq j \leq 1/a(n) \) and \( 1 \leq m_i \leq M \). We know that \( Y_{i,m_i}^j = Y_{i,m}^j \), w.p. \( p_m \), for \( m = 1, 2, \ldots, M \). Hence, we obtain
\[
E[Y_{i,m_i}^j] = \sum_{m=1}^{M} p_mE[Y_{i,m}^j].
\]

Summing up the total number of hops in the cell for any \( m \) in two different
ways gives us:

\[
\sum_{i=1}^{n} \sum_{j=1}^{1/a(n)} Y_{i,m}^j = \sum_{i=1}^{n} H_{i,m}
\]  

(19)

Taking the expectation on both sides of (19), and noting that \( E[H_{i,m}] \) are the same for each node \( i \) and \( E[Y_{i,m}^j] \) is equal for every \( i \) and \( j \) due to symmetry on the torus, we have

\[
\sum_{i=1}^{n} \sum_{j=1}^{1/a(n)} E[Y_{i,m}^j] = \sum_{i=1}^{n} E[H_{i,m}]
\]

\[
nE[Y_{i,m}^j]/a(n) = n[H_{i,m}]
\]

Therefore,

\[
E[Y_{i,m}^j] = a(n) \cdot E[H_{i,m}]
\]

\[= \Theta \left( \max \{ \sqrt{a(n)/X_m}, a(n) \} \right) \]

(20)

Also,

\[
E[Y_{i,m}^j] = \sum_{m=1}^{M} p_m E[Y_{i,m}^j]
\]

\[= \Theta \left( \sum_{m=1}^{M} p_m \max \{ \sqrt{a(n)/X_m}, a(n) \} \right) \]

(21)

The average total number of \( L_{H,R} \) lines passing through a fixed cell \( j \), is obtained by \( Y = \sum_{i=1}^{n} Y_{i,m}^j \). Hence, \( E[Y] = \Theta(n \sum_{m=1}^{M} p_m \max \{ \sqrt{a(n)/X_m}, a(n) \}) \).

Note that nodes are uniformly and independently distributed in the network and \([p_m] \) is invariant for all the nodes. Therefore, \( Y \) is the sum of i.i.d. random variables and also \( 0 \leq Y_{i,m}^j \leq 1 \) for each \( i \) and \( j \). Applying the Chernoff
bound for the sum of i.i.d. random variables yields [12]

\[ P\{Y > (1 + \delta)E[Y]\} \leq \exp\left(-\frac{\delta^2 E[Y]}{3}\right) \]  

(22)

Choosing \( \delta = \sqrt{6 \log n / E[Y]} \), we are guaranteed that \( \delta = o(1) \). This is true since we are assuming that \( a(n) = \Omega (\log n / n) \). Also, as explained later, we do not allow any content to be cached in more than \( \Theta(1/a(n)) \) holders. Due to the total caching capacity constraint, \( \sum_{m=1}^{M} X_m \leq nK \), and the fact that \( M = \Theta(n^\beta) \), where \( 0 < \beta < 1 \), we are assured that \( E[Y] = \omega(na(n)) \), or equivalently, \( E[Y] = \omega(\log n) \), resulting in \( \delta = o(1) \). Substituting \( \delta \) in (22), we are given

\[ P\{Y > (1 + \delta)E[Y]\} \leq \frac{1}{n^2} \]  

(23)

Therefore, \( Y = O(E[Y]) \) with probability \( \geq 1 - 1/n^2 \). Applying union bound over all the \( 8/r^2(n) \) cells provide us with the proof that the number of \( L_{H,R} \) lines passing through each cell of the network is \( O(E[Y]) = O(n \sum_{m=1}^{M} p_m \max \{\sqrt{a(n)/X_m}, a(n)\}) \) with probability \( \geq 1 - 1/n \).

Now we are ready to present the achievable scheme, similar to Scheme 1 in [11], which is parameterized by the cell area \( a(n) \), where \( a(n) = \Omega (\log n / n) \) and \( a(n) \leq 1 \).

1. Divide the unit torus using a square grid into square cells, each of area \( a(n) \).

2. For the given realization of the random network, check if the following conditions are satisfied

   - Condition 1: No cell is empty.
   - Condition 2: The number of \( L_{H,R} \) lines through each cell is of the order \( O(n \sum_{m=1}^{M} p_m \max \{\sqrt{a(n)/X_m}, a(n)\}) \).
3. If either of the above conditions is not satisfied, then use a time-division policy, where each of the \( n \) requesters communicate directly to the closest holder in a round-robin fashion.

4. If both conditions are satisfied, use the following policy \( \pi_n \):

   (a) Each cell becomes active regularly once every \( 1 + N \) time-slots (Lemma 3). Several cells which are sufficiently far apart become active simultaneously. Hence, the scheme uses TDM between neighboring cells.

   (b) Requesting nodes transmit IPs to the closest holders by hops along the adjacent cells intersecting \( L_{H,R} \) lines. Similarly, the holders transmit DPs to the requesting nodes on the same path taken by their corresponding IPs, in the reverse direction.

   (c) Each time slot is split into two sub-slots. In the first sub-slot, each active cell will transmit a single IP for each of the \( L_{H,R} \) lines passing through the cell toward the closest holder. In the second sub-slot, similarly, the active cell will transmit a single DP for each of the \( L_{H,R} \) lines passing through the cell toward the requesting node.

To present the tradeoff, we first derive the achievable throughput and then we calculate the average packet delay of the network. Note that if the time-division policy with direct communication is used, then the throughput is \( 2W/n \) with a delay of 1. But since it happens with a vanishingly low probability, as shown by Lemmas 2 and 5, the throughput and delay for the achievable scheme are determined by that of policy \( \pi_n \).

**Theorem 1.** The achievable the per-node throughput is w.h.p. of the order

\[
\lambda(n) = \Theta \left( \frac{1}{n \sum_{m=1}^{M} p_m \max \{ \sqrt{a(n)/X_m}, a(n) \}} \right)
\]  

(24)
Proof. When policy $\pi_n$ is used, since Condition 1 is satisfied, each cell has at least one node. This assures us that requester-holder pairs can communicate with each other by hops along adjacent cells on their $L_{H,R}$ lines. From Lemma 3, each cell gets to transmit packets every $1 + N$ time-slots. Hence, the cell throughput is $\Theta(1)$. The total traffic through each cell is that due to all the $L_{H,R}$ lines passing through the cell, which is $O(n \sum_{m=1}^{M} p_m \max \{\sqrt{a(n)/X_m}, a(n)\})$ since Condition 2 is also satisfied. This shows that

$$\lambda(n) = \Omega\left(\frac{1}{n \sum_{m=1}^{M} p_m \max \{\sqrt{a(n)/X_m}, a(n)\}}\right)$$

(25)

And the proof is complete. \qed

By Lemma 3, each cell can be active once every $N + 1$ time-slots, where $N$ is constant and independent of $n$. Also, as we are assuming that packets scales in proportion to the throughput $\lambda(n)$, each packet arriving at a node in the cell departs in the next active time-slot of the cell. Hence, the packet delay is at most $N + 1$ times the number of hops from the requester to the holder and from the holder to the requester. For a given realization of the random network, where node $i$ is requesting $m_i$ for $i = 1, 2, \ldots, n$, and $m_i \in 1, 2, \ldots, M$, let $h_{i,m_i}$ be the number of hops from the requester $i$ to its closest holder of content $m_i$ in the given realization. Furthermore, since the DP is taking the reverse path taken by its corresponding IP, then, the average delay of the network realization is given by the mean sample of the $h_{i,m_i}$’s times two. That is, $\frac{2}{n} \sum_{i=1}^{n} h_{i,m_i}$. As $n \to \infty$, by Law of Large Numbers, we know that

$$\frac{2}{n} \sum_{i=1}^{n} h_{i,m_i} \simeq 2E[H_{i,m_i}]$$

(26)

Using equation (18), we obtain the following theorem.

**Theorem 2.** The average delay of the network for the throughput given in
Theorem 1 is given by

\[ D = \Theta \left( \sum_{m=1}^{M} p_m \max \left\{ \frac{1}{\sqrt{a(n)X_m}}, 1 \right\} \right) \]  \hspace{1cm} (27) 

As you can notice, for the case \( X_m = \Omega(a^{-1}(n)) \), \( \lambda(n) \) and \( D(n) \) are independent of the number of holders. This means that for larger \( X_m \)'s, the average throughput and delay do not change in the order sense. However, for smaller \( X_m \)'s, the larger \( X_m \) becomes, the greater the throughput gets and the smaller delay becomes, which is obviously intuitive. As the number of holders grows, there exist more nodes caching the requested content, hence, the requester will receive the content in fewer hops. This makes the delay smaller and the throughput larger.

Combining (24) and (27), we have the following theorem stating the delay-throughput tradeoff.

**Theorem 3.** The tradeoff between the average delay and per-node throughput of the proposed network scenario is given by

\[ D(n)\lambda(n) = \Theta((na(n))^{-1}) \]  \hspace{1cm} (28) 

Note that Theorem 3 is true for any feasible set of \( (X_m) \). That is, we can randomly choose a caching policy and this tradeoff still holds. In next Section, we are going to find the optimized set of \( (X_m) \) by which it will minimizes the delay and maximizes the throughput.

### 4 Optimization Problem

Now we are going to optimize the delay-throughput tradeoff within the achievable solutions subject to the caching capacity. Let \( K \) be the caching capacity of each node. That is, each node can cache up to \( K \) different contents in its CS. We relax the integer constraint on \( X_m \) and thereby, \( X_m \) is assumed
to be a real number. Also, similar to [6], we relax the per-node caching capacity constraint into the total caching capacity constraint, \( \sum_{m=1}^{M} X_m \leq nK \).

As it has been mentioned previously, for \( X_m = \Omega(a^{-1}(n)) \), the delay and throughput do not change with \( X_m \) in the order sense. Intuitively, this is due to the restriction we are enforcing that nodes cannot serve their requests locally. Hence, there will be no need to cache more than one copy of the file per cell. Having said that, in order to derive the optimized tradeoff and make the optimization problem into a standard form, the number of holders will be assigned \( a^{-1}(n) \) for contents with \( X_m \geq a^{-1}(n) \). Specifically, for a decreasing ordered \( [p_m] \), \( p_1 \geq p_2 \geq \ldots \geq p_M \), we can see intuitively that \( X_1 \geq X_2 \geq \ldots \geq X_M \) must hold, due to the fact that the more popular a content is, the more copies of it should be cached in the network.

Let \( m_1 \) be the smallest index of the content with \( X_{m_1} < a^{-1}(n) \), and \( m_2 \) be the smallest index of the content with \( X_{m_2} \leq 1 \). Accordingly, we portion the set of all the contents into three subsets \( \mathcal{M}_1 = \{1, 2, \ldots, m_1 - 1\} \), \( \mathcal{M}_2 = \{m_1, m_1 + \ldots, m_2 - 1\} \), and \( \mathcal{M}_3 = \{m_2, m_2 + \ldots, M\} \). Considering the case in which \( r(n) = \Theta(\sqrt{\frac{\log n}{n}}) \), then, we assign \( X_m = \frac{n}{\log n} \) for every \( m \in \mathcal{M}_1 \), and \( X_m = 1 \ \forall m \in \mathcal{M}_3 \). Therefore, the contents in the sets \( \mathcal{M}_1 \), and \( \mathcal{M}_3 \) will consume \( \frac{(m_1-1)n}{\log n} \) and \( M - m_2 + 1 \) aggregate cache space, respectively, and the remaining cache space will be assigned to the contents in set \( \mathcal{M}_2 \). Applying these changes to the optimization problem, the objective function will be simplified from \( D = \sum_{m \in \mathcal{M}_1} p_m + \sum_{m \in \mathcal{M}_2} \frac{p_m}{\sqrt{\alpha(n)X_m}} + \sum_{m \in \mathcal{M}_3} \frac{p_m}{\sqrt{\alpha(n)}} \) to \( \sum_{m \in \mathcal{M}_2} \frac{p_m}{\sqrt{\alpha(n)X_m}} \), as the other parts are constant for a given content popularity distribution and transmission range. Also, note that by (28), minimizing the average delay is equivalent to maximizing the average throughput. We
obtain the minimum delay by solving the following optimization problem:

\[
\begin{aligned}
\min_{\{X_m\}} & \sum_{m \in M_2} \frac{p_m}{\sigma(n)X_m} \\
\text{subject to:} & \sum_{m \in M_2} X_m \leq n(K - \frac{(m_1 - 1)}{\log n} - \frac{(M - m_2 + 1)}{n}) = nK' \\
& 1 \leq X_m \leq a^{-1}(n) \quad \text{for } m \in M_2
\end{aligned}
\]

Solving the above problem using the Lagrange multiplier, we have the following result

\[
X_m^* = \begin{cases} 
\frac{n}{\log n} & \text{for } m = 1, 2, \ldots, m_1 - 1 \\
\frac{p_m^{2/3}}{\sum_{j=m_1}^{m_2-1} p_j^{3/2}} nK' & \text{for } m = m_1, \ldots, m_2 - 1 \\
1 & \text{for } m = m_2, \ldots, M
\end{cases}
\]

\[
D^*(n) = \Theta\left(\sum_{j=1}^{m_1-1} p_j + \left(\sum_{j=m_1}^{m_2-1} p_j^{2/3}\right)^{3/2} \frac{1}{\sqrt{K' \log n}} + \sqrt{\frac{n}{\log n} \sum_{j=m_2}^{M} p_j}\right)
\]

So far, we have been considering a general content popularity distribution. In the literature ([6],[7]), Zipf distribution has been used commonly for the content popularity distribution. We also are going to use Zipf distribution as an example for \([p_m]\). Let \(p_m = \frac{1}{H_\alpha(M)} \cdot m^{-\alpha}\) be the Zipf popularity distribution, where \(\alpha\) is the Zipf’s law exponent, and \(H_\alpha(M) = \sum_{i=1}^{M} i^{-\alpha}\) is a normalization constant. We have [6]

\[
H_\alpha(n) - H_\alpha(m) = \begin{cases} 
\Theta\left(\frac{(n+1)^{1-\alpha} - (m+1)^{1-\alpha}}{1-\alpha}\right) & \alpha \neq 1 \\
\Theta\left(\log \frac{n+1}{m+1}\right) & \alpha = 1
\end{cases}
\]

28
Or equivalently,

\[
H_\alpha(M) = \begin{cases} 
\Theta(1) & \alpha > 1 \\
\Theta(\log M) & \alpha = 1 \\
\Theta(M^{1-\alpha}) & \alpha < 1 
\end{cases} \tag{33}
\]

We can, then, compute the delay and throughput for \( M = \Theta(n^\beta) \) where \( 0 < \beta < 1 \), for Zipf popularity distribution.

**Theorem 4.** The throughput-delay performance of the proposed scheme using Zipf distribution is given by:

\[
D^*(n) = \begin{cases} 
\Theta(1) & \alpha > 3/2 \\
\Theta(\log M) & \alpha = 3/2 \\
\Theta(\frac{M^{3/2-\alpha}}{\sqrt{\log M}}) & 1 < \alpha < 3/2 \\
\Theta(\frac{\sqrt{M}}{\log M}) & \alpha = 1 \\
\Theta(\frac{1}{\sqrt{M \log M}}) & \alpha < 1 
\end{cases} \tag{34}
\]

\[
\lambda^*(n) = \begin{cases} 
\Theta(\frac{1}{\log M}) & \alpha > 3/2 \\
\Theta(\frac{1}{(\log M)^2}) & \alpha = 3/2 \\
\Theta(\frac{M^{3/2-\alpha}}{\sqrt{\log M}}) & 1 < \alpha < 3/2 \\
\Theta(\frac{\log M}{M}) & \alpha = 1 \\
\Theta(\frac{1}{\sqrt{M \log M}}) & \alpha < 1 
\end{cases} \tag{35}
\]

The proof of this theorem is given in Appendix B. Here we introduce a content placement algorithm for the random network model, similar to Algorithm 1 given in [6] for a grid-based network, in order to satisfy the per-node caching capacity constraint and also, study the effects of integer-numbered \( X_m \)'s in the implementation. Let \( \hat{X}_m = \lfloor X_m \rfloor, \forall m \). Now for a fixed \( m \), dividing the network area into \( \hat{X}_m \) regions, we can put a copy of content \( m \) in the CS of a unique node within each region. In this way,
content \( m \) will have \( \hat{X}_m \) holders in the network area. In each region, we choose the node with the minimum number of contents in its CS, and place a copy of content \( m \) in it. If there are multiple nodes with the minimum number of contents in their cache, we choose one randomly. The algorithm starts from the most popular content \( m = 1 \), and after placing the file in all the \( \hat{X}_1 \) regions, it does the same for content \( m = 2, 3, \ldots, M \). It can be shown that the network performance for this algorithm is of the same order given in Theorem 4.

Note that this algorithm starts with the most popular content, and ends with the least popular one. For each content \( m \), the algorithm places the copies of content \( m \) in the CSs of the nodes with the least contents in them. As the algorithm proceeds and \( m \) increments, the number of nodes within each region increases. This assures us that the algorithm distributes the contents evenly among the nodes. To show that this algorithm satisfies the per-node caching capacity constraint, suppose that the proposed algorithm violates the per-node caching capacity constraint for an arbitrary node \( j \). That is, assume that node \( j \) has cached \( K + 1 \) or more contents in its CS. As the algorithm distributes the contents evenly among the nodes, all other nodes must have cached at least \( K \) contents in their CSs. Hence, the total number of copies of all the contents in the network is at least equal to \( nK + 1 \) which contradicts with the total caching capacity constraint. Therefore, the per-node caching capacity constraint is met for every node.

Now, to further examine the proposed network and also provide the tools needed for the proof of Theorem 4, we need to estimate the indices \( m_1 \) and \( m_2 \). By definition, \( m_1 \) is the smallest index of which its number of holders is less than \( \frac{n}{\log n} \). That is, \( X_{m_1} < \frac{n}{\log n} \) and \( X_{m_1-1} \geq \frac{n}{\log n} \) for \( m_1 > 1 \), which leads to:
\[ \log n \left( K - \frac{(m_1 - 1)}{\log n} - \frac{(M - m_2 + 1)}{n} \right) < m_1^{\frac{2\alpha}{3}} [H_{\frac{2\alpha}{3}}(m_2 - 1) - H_{\frac{2\alpha}{3}}(m_1 - 1)] \]  

(36)

and,

\[ \log n \left( K - \frac{(m_1 - 2)}{\log n} - \frac{(M - m_2 + 1)}{n} \right) \geq (m_1 - 1)^{\frac{2\alpha}{3}} [H_{\frac{2\alpha}{3}}(m_2 - 1) - H_{\frac{2\alpha}{3}}(m_1 - 2)] \]  

(37)

Hence, for \( m_1 > 1 \), an approximation of \( m_1 \) can be obtained from:

\[ \log n \left( K - \frac{(m_1 - 1)}{\log n} - \frac{(M - m_2 + 1)}{n} \right) \simeq m_1^{\frac{2\alpha}{3}} [H_{\frac{2\alpha}{3}}(m_2 - 1) - H_{\frac{2\alpha}{3}}(m_1 - 1)] \]  

(38)

Similarly, by the definition of \( m_2 \), we know \( X_{m_2 - 1} > 1 \) and \( X_{m_2} \leq 1 \) for \( m_2 \leq M \) and thus, we have:

\[ n(K - \frac{(m_1 - 1)}{\log n} - \frac{(M - m_2 + 1)}{n}) > (m_2 - 1)^{\frac{2\alpha}{3}} [H_{\frac{2\alpha}{3}}(m_2 - 1) - H_{\frac{2\alpha}{3}}(m_1 - 1)] \]  

(39)

and,

\[ n(K - \frac{(m_1 - 1)}{\log n} - \frac{(M - m_2)}{n}) \leq m_2^{\frac{2\alpha}{3}} [H_{\frac{2\alpha}{3}}(m_2) - H_{\frac{2\alpha}{3}}(m_1 - 1)] \]  

(40)

Therefore, for \( m_2 \leq M \), \( m_2 \) can be computed approximately by:

\[ n(K - \frac{(m_1 - 1)}{\log n} - \frac{(M - m_2 + 1)}{n}) \simeq (m_2 - 1)^{\frac{2\alpha}{3}} [H_{\frac{2\alpha}{3}}(m_2 - 1) - H_{\frac{2\alpha}{3}}(m_1 - 1)] \]  

(41)
Now, dividing inequality (36) by inequality (39), we get:

$$\frac{\log n}{n} < \left(\frac{m_1}{m_2 - 1}\right)^{2\alpha}$$ \hspace{1cm} (42)

This inequality will change to asymptotic equality if $m_1 > 1$ and $m_2 \leq M$. That is, if $\mathcal{M}_1$ and $\mathcal{M}_3$ are not empty we have

$$\frac{m_1}{m_2} = \Theta\left(\left(\frac{\log n}{n}\right)^{\frac{3}{2\alpha}}\right)$$ \hspace{1cm} (43)

Similar to [6], to better analyze the network performance, we are going to consider two different cases, in terms of magnitude of the $m_2$: the case in which $m_2 = \Theta(M)$ or equivalently, $M - m_2 = o(M)$ and the other case where $m_2 = o(M)$, or equivalently $M - m_2 = \Theta(M)$.

4.1 $m_2 = \Theta(M)$

**Lemma 6.** For the case $m_2 = \Theta(M)$, $m_1$ is given by:

$$m_1 = \begin{cases} 
\Theta(\log n) & \alpha > 3/2 \\
\Theta(1) & \alpha = 3/2 \\
\text{converging to 1} & \alpha < 3/2 
\end{cases}$$ \hspace{1cm} (44)

**Proof.** $\alpha > 3/2$: As $M = o(n)$, then $M - m_2 = o(n)$. Therefore, $K' \rightarrow K - \frac{(m_1 - 1)}{\log n}$. Using (37), we have

$$\log n(K - \frac{(m_1 - 1)}{\log n}) \simeq (m_1 - 1)^{\frac{2\alpha}{3}} [H_{\frac{2\alpha}{3}} (m_2 - 1) - H_{\frac{2\alpha}{3}} (m_1 - 1)]$$

$$\simeq (m_1 - 1)^{\frac{2\alpha}{3}} \frac{[-(m_1 - 1)^{1 - \frac{2\alpha}{3}}]}{1 - \frac{2\alpha}{3}}$$ \hspace{1cm} (45)

which leads to

$$m_1 \simeq 1 + \frac{2\alpha - 3}{2\alpha} K \log n$$ \hspace{1cm} (46)
\( \alpha = 3/2 \): similar to the previous case, by using (37), we have
\[
(m_1 - 1)^{\frac{2\alpha}{3}} \simeq \frac{K \log n}{\log m_2} = \Theta(1)
\] (47)
where the last equation holds by the fact \( \log M = \Theta(\log n) \) for any \( 0 < \beta < 1 \).

\( \alpha < 3/2 \): using (37), we are given
\[
(m_1 - 1)^{\frac{2\alpha}{3}} \leq \frac{K \log n}{[H_{\frac{2\alpha}{3}}(m_2 - 1) - H_{\frac{2\alpha}{3}}(m_1 - 1)]}
\] (48)
applying (32), the RHS is converging to zero. Therefore, \( m_1 \rightarrow 1 \) as \( n \) grows.

**Lemma 7.** As \( n \rightarrow \infty \), for the case \( \alpha \leq 3/2 \), \( m_2 = M + 1 \) or equivalently, \( M_3 = \emptyset \). Also, for the case \( \alpha > 3/2 \), if \( 0 < \beta \leq \frac{3}{2\alpha} \), then, \( M_3 = \emptyset \).

**Proof.** First we show the conditions on which \( m_2 = \Theta(M) \) holds. For \( \alpha < 3/2 \): \( m_1 \rightarrow 1 \), then, using (39) and (33), we are given:
\[
nK \geq \frac{m_2 - 1}{1 - \frac{2\alpha}{3}}
\] (49)
Therefore, \( m_2 = \Theta(M) \) iff \( M \leq (K - \frac{2\alpha}{3})n \).

For \( \alpha = 3/2 \): \( m_1 = \Theta(1) \). Using (39) and (33), we obtain
\[
nK \geq (m_2 - 1)[\ln(m_2 - 1)]
\] (50)
Hence, \( m_2 = \Theta(M) \) iff \( M \ln M \leq Kn \).

For \( \alpha > 3/2 \): \( m_1 = \Theta(\log n) \). Using (39) and (33), we have
\[
n(K - \frac{(m_1 - 1)}{\log n}) = nK' \geq \]
\[
\geq (m_2 - 1)^{\frac{2\alpha}{3}} \left[ (m_2 - 1)^{1 - \frac{2\alpha}{3}} - m_1^{1 - \frac{2\alpha}{3}} \right]
\]
which leads to
\[(m_2 - 1)^{2\alpha \frac{3}{2}} \leq \left\lfloor \frac{n(\frac{2\alpha}{3} - 1)K'}{m_1^{1 - \frac{2\alpha}{3}}} \right\rfloor \] (51)

Therefore, using straight calculation, it is obtained that \(m_2 = \Theta(M)\) iff
\[M = \Theta(n^{\frac{3\alpha}{2\alpha}} (\log n)^{1 - \frac{3}{2\alpha}})\] (52)

Now, since \(M_3 = \emptyset\) iff \(X_M > 1\), then starting the above derivations with \(X_M > 1\) - instead of \(X_{m_2} > 1\) - will provide us with the conditions of \(M_3\) being empty. As we are considering the case \(M = o(n)\), hence, for \(\alpha \leq 3/2\), \(m_2 = M + 1\) or equivalently, \(M_3 = \emptyset\). Moreover, for \(\alpha > 3/2\) if \(\beta \leq \frac{3}{2\alpha}\), then, \(M_3 = \emptyset\).

\[\square\]

4.2 \(m_2 = o(M)\)

As shown in Lemma 7, for \(0 < \beta < 1\) and \(0 < \alpha \leq 3/2\), \(m_2 = \Theta(M)\). Therefore, for the case \(m_2 = o(M)\), we only need to consider content popularity distribution with exponent \(\alpha > 3/2\). First, we estimate the index \(m_1\) and then, using (43), we give an estimation of \(m_2\)

**Lemma 8.** For the case \(m_2 = o(M)\) where \(\alpha > 3/2\), the scaling of indices \(m_1\) and \(m_2\) is given by
\[m_1 = \Theta(\log n)\] (53)
\[m_2 = \Theta\left(n^{\frac{3\alpha}{2\alpha}} (\log n)^{1 - \frac{3}{2\alpha}}\right)\] (54)

**Proof.** Using (37), and (32), we have
\[\log n(K - \frac{(m_1 - 2)}{\log n} - \frac{(M - m_2 + 1)}{n}) \approx m_1^{\frac{2\alpha}{3}} m_1^{1 - \frac{2\alpha}{3}} = \frac{3m_1}{2\alpha - 3}\] (55)
leading to
\[ m_1 \simeq K \left( \frac{2\alpha - 3}{2\alpha} \right) \log n \] (56)

Also, using (43), we have
\[ m_2 \simeq m_1 \left( \frac{n}{\log n} \right)^{\frac{3}{2\alpha}} \simeq \frac{2\alpha - 3}{2\alpha} K n^{\frac{1}{2\alpha}} (\log n)^{1 - \frac{3}{2\alpha}} \] (57)
5 Serving Locally

In the previous sections, similar to [6] and [7] we have assumed that the requesting node does not satisfy the requested content from its CS, if it has cached it. This assumption made it easier to analyze the delay-throughput tradeoff. Intuitively, the ability to retrieve the requested data from the local cache will improve the performance of the network, as it does not “consume” the network resources, while it will satisfy requests for contents almost immediately.

In order to derive the delay-throughput tradeoff of this network, consider a node, say node 1. Node 1 has a capacity of caching $K$ contents in its CS. Also, as there are $X_m$ nodes caching content $m$ chosen randomly in the network, the probability of content $m$ being in the CS of node 1 is $X_m/n$ for $m \in \{1, 2, \ldots, M\}$. Let the vector $(r_1, r_2, \ldots, r_s)$ be the content request sequence of node 1 up to time $t$. That is, node 1 will first request content $r_1$, then content $r_2$, and so on. Using the above argument, the probability of the request for $r_j$ being served from the local cache of node 1, that is, $r_j$ is in the CS of node 1, is equal to $X_{r_j}/n$, where $1 \leq j \leq s$.

Due to the homogeneity of the nodes and the symmetry of the torus, we can argue that the probability of request for content $m$ being satisfied by the local cache of node $i$, given that content $m$ has been requested by node $i$ at any given time $t > 0$, is $X_m/n$.

Note that when a fixed node $i$ is requesting content $m$, whether $i \in \chi_m$ or not, Lemma 4 still holds. That is, whether node $i$ is a holder of content $m$ or not, $\Theta(1/\sqrt{X_m-1}) = \Theta(1/\sqrt{X_m})$. Therefore, the average length of $L_{H,R}(i, m)$ line is still of order $\Theta(1/\sqrt{X_m})$. However, we need to modify the definition of the “closest holder” to be excluding the requesting node itself. In order to derive the delay-throughput tradeoff for the proposed network, first we need to generalize the result of Lemma 5 for the “self-serving” scenario.

As explained previously, when node $i$ is requesting content $m$, with probability $X_m/n$ the request will be served from the local cache and no Interest
Packet will be sent out to the network. Hence, to calculate the number of \( L_{H,R} \) lines passing through each cell, we only need to consider the case when the request packet is sent out to the network. This event happens with prob. \( 1 - \frac{X_m}{n} \) given that the requested content is content \( m \), where \( m \in \{1,2,\ldots,M\} \). Therefore, the event that content \( m \) is requested by a fixed node \( i \) and it is not cached in the node’s CS, happens with prob. \( p_m (1 - \frac{X_m}{n}) \). Thus, the probability of node \( i \) at time \( t \) sends out an Interest Packet requesting content \( m \) is \( p_m (1 - \frac{X_m}{n}) \), while this probability was equal to \( p_m \) for the former case when no request could be satisfied from the local cache. The rest of the analysis for the derivation of a generalized form of Lemma 5 is similar and therefore, the number of \( L_{H,R} \) lines passing through each cell is at most \( O(n \sum_{m=1}^{M} p_m (1 - \frac{X_m}{n}) \max \{ \sqrt{a(n)/X_m}, a(n) \}) \) w.h.p.

Also, the approach to derive the average throughput and the associated average delay for this network scenario is very similar to the analysis of Theorems 1 and 2, respectively. Hence, we have the average per-node throughput and the average delay for the network scenario with self-serving given by:

\[
\lambda(n) = \Theta \left( \frac{1}{n \sum_{m=1}^{M} p_m (1 - \frac{X_m}{n}) \max \{ \sqrt{a(n)/X_m}, a(n) \}} \right) \tag{58}
\]

\[
D = \Theta \left( \sum_{m=1}^{M} p_m (1 - \frac{X_m}{n}) \max \left\{ \frac{1}{\sqrt{a(n)/X_m}, 1} \right\} \right) \tag{59}
\]

As you can notice the above equations are similar to (24) and (27), respectively, with only difference of \( p_m (1 - \frac{X_m}{n}) \) substituting \( p_m \).

Since \( 1 - \frac{X_m}{n} < 1 \), we clearly obtain smaller average delay and greater throughput, comparing with the case without self-serving. Interestingly, the trade-off is exactly the same as in (28).
5.1 Optimization Problem

Now that we have derived the throughput and the delay of the proposed network, we can optimize the performance of the network subject to the number of holders for each content. Similar to the optimization problem given in (29), we can minimize the objective function $\sum_{m \in \mathcal{M}} p_m (1 - \frac{X_m}{n})$ for the Zipf content popularity distribution subject to the constraints of the optimization problem in (29).

Using a similar partitioning on contents, let $m^*$ be the smallest index of the content with $X_{m^*} \leq 1$. Then, for contents with indices $m \geq m^*$, the number of holders will be 1. However, we do not restrict $X_m$ to be less than $a^{-1}(n)$ as we allow serving locally in this case. Therefore, in this case $1 \leq X_m \leq n$. Solving this optimization problem for $m = 1, 2, \ldots, m^* - 1$ by using Lagrange multiplier, we obtain that the optimized $X_m$'s should hold in this equation

$$p_m = \frac{c \cdot X_m^2 \Gamma X_m / n}{\sum_{j=m_1}^{m^*-1} X_j^\alpha / \Gamma X_j / n}$$

where $c = \sum_{m=1}^{m^*-1} p_m$. Assuming $Y_m = \frac{X_m^3}{\Gamma X_m / n}$, we can transform this set of equations into a matrix form. We obtain

$$
\begin{bmatrix}
p_1 - c & p_1 & \cdots & p_1 
p_2 & p_2 - c & \cdots & p_2 
\vdots & \vdots & \ddots & \vdots 
p_{m^* - 1} & p_{m^* - 1} & \cdots & p_{m^* - 1} - c
\end{bmatrix}
\begin{bmatrix}
Y_1 
Y_2 
\vdots 
Y_{m^* - 1}
\end{bmatrix}
= 
\begin{bmatrix}
0 
0 
\vdots 
0
\end{bmatrix}
$$

(61)

since the $\sum_{m=1}^{m^*-1} p_m - c = 0$, the rank of the above matrix is $m^* - 2$. Therefore, there is one and only one non-zero vector in the nullity space of this system. After solving for this vector, we need to solve $m^* - 1$ inverse cubic equations of the form $x^3 + a^2x = b$ where $x = X_m^{-1/2}$, $a = 1/\sqrt{n}$, and $b = 1/Y_m$. 38
Using Omar Khayym’s solution for cubic equations [13], we are assured that there is a unique positive and real solution to each of these equations and therefore, $X_m$ can be obtained successfully.

Now, solving this set of equations for $X_m$’s in an analytical form is quite difficult. However, we can solve them numerically for different content popularity distributions. In section 8, it is demonstrated that the asymptotic performance of the network model when the serving locally is enabled is only a constant factor better than the case where it is not enabled.
6 Contents with different sizes

In this section, we are considering the case where contents can have different sizes. Let \( S_m \) be the number of chunks per each content \( m \). As we are considering the fluid model, the size of each chunk scales proportionally with the throughput. However, the number of chunks for each content is assumed to be fixed. It is also worth noting that in previous sections, we assumed \( S_m = 1 \) for all \( m \in \{1, 2, \ldots, M\} \).

Similar to previous sections, nodes request contents with distribution \([p_m]\), and send out an IP toward the closest holder to retrieve the requested DP. Although, since each content consists of multiple DPs, the requester needs to send an IP for each one sequentially. Upon the reception of the IP for a particular chunk of the data object, the holder will fetch the requested chunk, create the DP and send it back to the requesting node. We are assuming an uncoded content placement, and hence, each holder caches the whole content, rather than a portion of it. In order to calculate the throughput delay tradeoff of this scenario, again, we modify Lemma 5 slightly to match the recent assumptions.

It is easy to show that for \( a(n) \geq 2 \log n/n \), the number of \( L_{H,R} \) lines passing through each cell is \( O(n \sum_{m=1}^{M} S_m p_m \max \{ \sqrt{a(n)/X_m}, a(n) \} ) \) w.h.p. This result is similar to the result of Lemma 5, with the only modification of inserting a weight \( S_m \) for content \( m \) in the above formulation. This is due to the fact that each content \( m \) has \( S_m \) DPs and therefore, it takes \( S_m L_{H,R}(i, m) \) lines connecting from the closest holder of content \( m \) to the requesting node \( i \) in order to transfer the chunks.

Following the same approach for the proofs of Theorems 1 and 2 we are given the expressions of the throughput and delay of the proposed network by

\[
\lambda(n) = \Theta \left( \frac{1}{n \sum_{m=1}^{M} S_m p_m \max \{ \sqrt{a(n)/X_m}, a(n) \} } \right)
\]
\[
D = \Theta \left( \sum_{m=1}^{M} S_m p_m \max \left\{ \frac{1}{\sqrt{a(n) X_m}}, 1 \right\} \right) \tag{63}
\]

Now, to formulate the optimization problem of minimizing the average delay with respect to the content placement, it is important to take into account the caching capacity constraint when contents have different sizes. The per-node caching constraint requires the network to put at most \( K \) chunks \(^3\) of contents in the CS of each node. However, we relax this constraint to a total caching constraint similar to the optimization problem (29). Given that each content \( m \) is portioned into \( S_m \) chunks, the total caching constraint will be \( \sum_{m=1}^{M} S_m X_m \leq nK \).

Solving the stated optimization problem using the Lagrange multiplier, we have the following results.

\[
X^*_m = \begin{cases} 
\frac{n}{\log n} & \text{for } m = 1, 2, \ldots, m_1 - 1 \\
\frac{p_{m_1}^{2/3}}{\sum_{j=m_1}^{m_2-1} S_j p_j^{2/3}} nK' & \text{for } m = m_1, \ldots, m_2 - 1 \\
1 & \text{for } m = m_2, \ldots, M 
\end{cases} \tag{64}
\]

\[
K' = K - \frac{n}{\log n} \sum_{m=1}^{m_1} S_m + \sum_{m=m_2}^{M} S_m 
\]

\[
D^*(n) = \Theta \left( \sum_{j=1}^{m_1-1} S_j p_j + \frac{\left( \sum_{j=m_1}^{m_2-1} S_j p_j^{2/3} \right)^{3/2}}{\sqrt{K' \log n}} + \sqrt{\frac{n}{\log n} \sum_{j=m_2}^{M} S_j p_j} \right) \tag{66}
\]

\(^3\)We are using the term “chunk” loosely here, as we are ignoring the header bits added to the packets in the lower-layers.
7 Hybrid networks over a shared channel

So far, we have considered a pure ad hoc network in which all the wireless nodes are homogeneous and there is no base station in the network. Now, we consider the scenario where there are multiple base stations in the network, all connected to each other and to the Internet using high-speed wired networks. Let \( f(n) \) be the number of BSs, where \( f(n) \) is a non-decreasing function of \( n \). For analysis, we assume \( f(n) = \Theta(n^\mu) \), where \( 0 \leq \mu < 1 \).

We divide the unit-sized network area into \( f(n) \) cells and place the BSs at the center of each cell. Nodes can communicate with other wireless nodes, whether they are within the same cell or not, and their co-cell BS. As the area of each cell is \( \frac{1}{f(n)} \), hence, the average number of nodes per cell is \( \frac{n}{f(n)} \). Similar to previous sections, we assume that \( X_m \) holders are uniformly distributed in the whole area. The important questions we would like to give answer to are:

- How many base stations are required to improve the asymptotic performance of the network paradigm?
- How good does the hybrid network perform under different content popularity distributions?

Since we are evaluating the wireless part of this network model, we assume that all the requested contents, upon reception at the base stations, will be satisfied immediately. In other words, we are not considering the wired network delay in our analysis.

Each node with probability \( p_m \) requests content \( m \). If the closest holder of content \( m \) is closer to the requesting node than the co-cell BS, then the content will be downloaded from the closest holder. Otherwise, the content will be downloaded from the base station.

It is easy to see that the average distance from the closest base station is of order \( \Theta\left(\frac{1}{\sqrt{f(n)}}\right) \). Therefore, the average length of the \( L_{H,R}(i,m) \) line
connecting the requesting node $i$ to the closest cache of content $m$ (either a wireless node or a base station) is given by:

$$E[|L_{H,R}(i,m)|] = \Theta\left(\min\left\{1 \sqrt{X_m}, \frac{1}{\sqrt{f(n)}}\right\}\right). \quad (67)$$

Then, the average of the minimum number of hops along the $L_{H,R}$ line is followed by:

$$E[H_{i,m}] = \Theta\left(\max\left\{1, \min\left\{1 \sqrt{a(n)X_m}, \frac{1}{\sqrt{a(n)f(n)}}\right\}\right\}\right) \quad (68)$$

Using a similar approach to the one in the proof of Lemma 5, it is obtained that for $a(n) \geq 2\log n/n$, the number of $L_{H,R}$ lines passing through each cell\footnote{Note that here by cell, it is meant the scheduling cell with area $a(n)$ that is used to avoid interference.} is $O(n \sum_{m=1}^{M} p_m \max\{a(n), \min\{\sqrt{a(n)X_m}, \sqrt{a(n)f(n)}\}\})$ w.h.p.

Therefore, the throughput and the delay of hybrid network model is given by:

$$\lambda(n) = \Theta\left(\frac{1}{n \sum_{m=1}^{M} p_m \max\{a(n), \min\{\sqrt{a(n)X_m}, \sqrt{a(n)f(n)}\}\}}\right) \quad (69)$$

$$D = \Theta\left(\sum_{m=1}^{M} p_m \max\left\{1, \min\left\{1 \sqrt{a(n)X_m}, \frac{1}{\sqrt{a(n)f(n)}}\right\}\right\}\right) \quad (70)$$

Now, in order to establish the optimization problem for the minimization of the average delay subject to the total caching caching constraints and given content popularity distribution $[p_m]$ which is assumed to be non-increasing with $m$, we take into consideration that for contents with low enough popularity that the average distance between the requesting node and its closest holder is greater than the one between the requester and the base station, we discard the copies of them in the CSs of the wireless networks and will
rely solely on base stations to satisfy the requests for these contents. Let $m_1$ be the smallest index of the content with $X_{m_1} < a^{-1}(n)$, and $m_2$ be the smallest index of the content with $X_{m_2} \leq f(n)$. Accordingly, we portion the set of all the contents into three subsets $\mathcal{M}_1 = \{1, 2, \ldots, m_1 - 1\}$, $\mathcal{M}_2 = \{m_1, m_1 + 1, \ldots, m_2 - 1\}$, and $\mathcal{M}_3 = \{m_2, m_2 + 1, \ldots, M\}$. Considering the case in which $r(n) = \Theta(\sqrt{\log n})$, then, we assign $X_m = \frac{n}{\log n}$ for every $m \in \mathcal{M}_1$, and $X_m = 0$ for $m \in \mathcal{M}_3$.

Using a similar method for relaxing the optimization problem of (29), we are given:

$$\begin{align*}
\text{min}_{\{X_m\}} & \sum_{m \in \mathcal{M}_1} p_m + \sum_{m \in \mathcal{M}_2} \frac{p_m}{\sqrt{a(n)X_m}} + \sum_{m \in \mathcal{M}_3} p_m \\
\text{subject to:} & \\
\sum_{m \in \mathcal{M}_2} X_m & \leq n(K - \frac{(m_1 - 1)}{\log n}) = nK' \\
0 & \leq X_m \leq a^{-1}(n) \quad \text{for} \ m \in \mathcal{M}_2
\end{align*}$$

Solving the above problem using the Lagrange multiplier, we have the following result

$$X_m^* = \begin{cases}
\frac{n}{\log n} & \text{for } m = 1, 2, \ldots, m_1 - 1 \\
\frac{p_m^{2/3}}{\sum_{j=m_1}^{m_2-1} p_j^{2/3}} nK' & \text{for } m = m_1, \ldots, m_2 - 1 \\
0 & \text{for } m = m_2, \ldots, M
\end{cases}$$

$$D^*(n) = \Theta(\sum_{j=1}^{m_1-1} p_j + \left(\sum_{j=m_1}^{m_2-1} p_j^{2/3}\right)^{3/2} \frac{n}{\sqrt{K'\log n}} + \sqrt{f(n)\log n} \sum_{j=m_2}^{M} p_j)$$

Similar to the pure ad hoc case, we also are going to use Zipf distribution as an example for $[p_m]$ for the hybrid network paradigm. Similarly, to examine the proposed network, we need to estimate the indices $m_1$ and $m_2$. 

44
By definition, $m_1$ is the smallest index of which its number of holders is less than $\frac{n}{\log n}$. That is, $X_{m_1} < \frac{n}{\log n}$ and $X_{m_1-1} \geq \frac{n}{\log n}$ for $m_1 > 1$, which leads to the inequalities (36), (37), and finally the estimation of (38).

In addition, by the definition of $m_2$, we know $X_{m_2} - 1 > f(n)$ and $X_{m_2} \leq f(n)$ for $m_2 \leq M$ and thus, we have:

$$n(K - \frac{(m_1 - 1)}{\log n}) > (m_2 - 1) \frac{2\alpha}{\pi} [H_{\frac{2\alpha}{\pi}}(m_2 - 1) - H_{\frac{2\alpha}{\pi}}(m_1 - 1)] f(n) \quad (74)$$

and,

$$n(K - \frac{(m_1 - 1)}{\log n}) \leq m_2^2 [H_{\frac{2\alpha}{\pi}}(m_2) - H_{\frac{2\alpha}{\pi}}(m_1 - 1)] f(n) \quad (75)$$

Therefore, for $m_2 \leq M$, $m_2$ can be computed approximately by:

$$n(K - \frac{(m_1 - 1)}{\log n}) \simeq (m_2 - 1) \frac{2\alpha}{\pi} [H_{\frac{2\alpha}{\pi}}(m_2 - 1) - H_{\frac{2\alpha}{\pi}}(m_1 - 1)] f(n) \quad (76)$$

Now, dividing inequality (36) by inequality (74), we get:

$$\frac{f(n) \log n}{n} < (\frac{m_1}{m_2 - 1}) \frac{2\alpha}{\pi} \quad (77)$$

This inequality will change to asymptotic equality if $m_1 > 1$ and $m_2 \leq M$.

That is, if $\mathcal{M}_1$ and $\mathcal{M}_3$ are not empty we have

$$\frac{m_1}{m_2} = \Theta((\frac{f(n) \log n}{n})^{\frac{3}{2\alpha}}) \quad (78)$$

It is worth noting that in order to have non-trivial solution, we assume that $f(n) = o(\frac{1}{\log(n)}) = o(\frac{n}{\log n})$. Hence, $m_2 > m_1$ always holds.
Lemma 9. Taking $n \to \infty$, $m_1$ and $m_2$ scales as:

$$m_1 = \begin{cases} 
\Theta(\log n) & \alpha > 3/2 \\
\Theta(1) & \alpha = 3/2 \\
\text{converging to 1} & \alpha < 3/2 
\end{cases} \tag{79}$$

$$m_2 = \begin{cases} 
\Theta \left( \frac{n}{f(n)} \right)^{2\alpha \frac{3-\alpha}{2}} (\log n)^{1-\frac{\alpha}{2}} & \alpha > 3/2 \\
\Theta \left( \frac{n}{f(n) \log n} \right) & \alpha = 3/2 \\
\Theta \left( \frac{n}{f(n)} \right) & \alpha < 3/2 
\end{cases} \tag{80}$$

Proof. $\alpha > 3/2$: similarly to the proof of Lemma 6, we can obtain the result in equation (46). Also, by using equation (78), the expression for $m_2$ for the case $\alpha > 3/2$ is achieved.

$\alpha = 3/2$: Again, by using similar approach in Lemma 6, and using (37), we have

$$(m_1 - 1)^{2\alpha} \simeq \frac{K \log n}{\log m_2} \tag{81}$$

Now, if $m_2 = \Theta(n^\gamma), \forall \gamma > 0$, we have $m_1 = \Theta(1)$. As $f(n) = \Theta(n^\mu)$, for some $\mu > 0$, then $m_2 = \Theta\left( \frac{n}{f(n) \log n} \right)$ guarantees the necessary condition, $m_2 = \Theta(n^\gamma), \forall \gamma > 0$.

$\alpha < 3/2$: using (76), we are given

$$nK \simeq (m_2 - 1)^{2\alpha} \frac{(m_2 - 1)^{1-2\alpha} - 1}{1 - \frac{2\alpha}{3}} \tag{82}$$

which leads to

$$m_2 \simeq 1 + \frac{3 - 2\alpha}{3} \frac{n}{f(n)} K \tag{83}$$

Now, using similar approach to Lemma 6, we obtain $m_1 \to 1$ as $n$ grows. □

Theorem 5. The throughput-delay performance of the proposed hybrid network using Zipf distribution and transmission range $r(n) = \Theta(\sqrt{\log n/n})$, is
given by:

\[
D^*(n) = \begin{cases} 
\Theta(1) & \alpha > 3/2 \\
\Theta\left(\frac{\log\left(\frac{n}{f(n)}\right)3/2}{\sqrt{\log n}}\right) & \alpha = 3/2 \\
\Theta\left(\frac{\left(\frac{n}{f(n)}\right)^{3/2-\alpha}}{\sqrt{\log n}}\right) & 1 < \alpha < 3/2 \\
\Theta\left(\frac{n}{f(n)\log n}\right) & \alpha \leq 1 
\end{cases}
\]  
(84)

\[
\lambda^*(n) = \begin{cases} 
\Theta\left(\frac{1}{\log n}\right) & \alpha > 3/2 \\
\Theta\left(\frac{1}{\sqrt{\log n}\log\left(\frac{n}{f(n)}\right)3/2}\right) & \alpha = 3/2 \\
\Theta\left(\frac{1}{\sqrt{\log n}\left(\frac{n}{f(n)}\right)^{3/2-\alpha}}\right) & 1 < \alpha < 3/2 \\
\Theta\left(\frac{n}{f(n)\log n}\right) & \alpha \leq 1 
\end{cases}
\]  
(85)

**Proof.** We are going to compute the average delay, and the average throughput can be calculated easily by equation (28). Substituting \(p_j\)'s equation (73) with the Zipf distribution, we are given

\[
D = \frac{H_\alpha(m_1)}{H_\alpha(M)} + \frac{[H_{2\alpha}(m_2 - 1) - H_{2\alpha}(m_1 - 1)]^{3/2}}{\sqrt{K'\log n}H_\alpha(M)} \\
+ \sqrt{\frac{n}{f(n)\log n}} \cdot \frac{H_\alpha(M) - H_\alpha(m_2 - 1)}{H_\alpha(M)}
\]  
(86)

where \(K' = K - \frac{(m_1 - 1)}{\log n}\). Similar to the proof of Theorem 4, let the three expressions of the RHS of the equation be denoted as \(D_1, D_2, \) and \(D_3\), respectively.

By Lemma 9, we know for \(\alpha \leq 3/2, K' \to K\). Also, due to the assumptions on the \(f(n)\), we have

\[
\log n < \frac{n}{f(n)} < M, \text{ for } \alpha \leq 3/2
\]  
(87)
and,

$$\log n < \frac{n}{f(n)} < \frac{M^{2\alpha/3}}{(\log n)^{2\alpha/3-1}}, \text{ for } \alpha > 3/2$$  \hspace{1cm} (88)

For $\alpha < 1$: For this case, $D_1 = o(1)$. Since, $m_2 = \Theta\left(\frac{n}{f(n)}\right)$, we have

$$D = D_2 + D_3 = \frac{(n/f(n))^{3/2-\alpha}}{\sqrt{\log n} M^{1-\alpha}} + \sqrt{\frac{n}{f(n) \log n}}$$

$$= \sqrt{\frac{n}{f(n) \log n}} \left[1 + \left(\frac{n}{f(n) M}\right)^{1-\alpha}\right]$$

$$= \sqrt{\frac{n}{f(n) \log n}}$$  \hspace{1cm} (89)

where the last part comes from (87).

For $\alpha = 1$: Similarly, for this case, $D_1 = o(1)$ and we have

$$D = D_2 + D_3 = \frac{(n/f(n))^{3/2-\alpha}}{\sqrt{\log n} M^{1-\alpha}} + \sqrt{\frac{n}{f(n) \log n}}$$

$$= \sqrt{\frac{n}{f(n) \log n}} \left[1 + \frac{1}{\log M}\right]$$

$$= \sqrt{\frac{n}{f(n) \log n}}$$  \hspace{1cm} (90)

For $1 < \alpha < 3/2$: For $\alpha > 1$, $D_1 = \Theta(1)$. We can see in the following that $D_2 = \Theta(D_3)$ and therefore, $D = \Theta(D_2)$. we have

$$D = 1 + D_2 + D_3 = 1 + \frac{(n/f(n))^{3/2-\alpha}}{\sqrt{\log n}} + m_2^{1-\alpha} \cdot \sqrt{\frac{n}{f(n) \log n}}$$

$$= \frac{(n/f(n))^{3/2-\alpha}}{\sqrt{\log n}}$$  \hspace{1cm} (91)
For $\alpha = 3/2$: using Lemma 9 and (87), we have $D_3 = \Theta(1)$.

$$D_3 = m_2^{-1/2} \cdot \sqrt[n]{\frac{n}{f(n) \log n}}$$

$$= 1$$  \hspace{1cm} (92)

Therefore, $D = \Theta(D_2)$, and we have:

$$D_2 = \Theta(\frac{[\log(n)]^{3/2}}{\sqrt{\log n}})$$  \hspace{1cm} (93)

For $\alpha > 3/2$: using similar calculation for the equation (92), it can be shown that for $\alpha > 3/2$, $D_3 \rightarrow 0$. Also, $D_2 = \Theta(\frac{1}{\sqrt{\log n}}) = o(1)$. Therefore, $D = \Theta(1)$.

Comparing the results of the proposed hybrid network given in Theorem 5 and the pure ad hoc network given in Theorem 4, we conclude that the number of base stations in the network needs to be greater than $\frac{n}{M}$ to improve the order of the performance of the hybrid network, in terms of the average throughput and delay.
8 Simulations and Experiments

In order to evaluate our theoretical results, we turn to simulations to check the credibility of the proposed schemes in practical scenarios. To characterize the performance of the system, we are going to measure the average delay of downloading different contents by different nodes, according to the described communication and scheduling schemes in previous sections.

8.1 Simulation Setup

As specified in previous sections, nodes are randomly distributed in a 2D network area with uniform distribution. The network area is a square with area of 1. Each node has a content store with the capacity of storing up to $K = 10$ contents. Here, we are assuming contents have the same size of 1. The transmission range is common to all nodes and is equal to $r = \sqrt{\frac{16 \log n}{n}}$. The number of contents is considered to be $M = n^\beta$, where $\beta = 0.8$. Contents are cached in the nodes’ Content Stores in the initial mode of the system, according to the optimized solution given in (30). As specified, the holders of each content are chosen uniformly at random. Also, the routing tables are populated with shortest paths toward the closest holders of each content.

To overcome the transmission interference we are considering a cell-based network and using TDMA scheme to activate far-enough cells simultaneously in each time slot. The area of each cell is set to $r^2(n)/8$. Also, the number of cells per cluster is considered to be $N = 16(1 + \Delta)^2$, where $\Delta$ is the Protocol model parameter and is set to be 0.2 in the simulations. Using a round-robin scheduling, each cell is activated once every $N$ time slots. When a cell is activated, in the first subslot, it will randomly choose one IP to be transmitted through the cell toward the intended holder, if there is any. Similarly, in the second subslot, it will randomly choose one DP to be passed through toward the requesting node, if there is any.

Each node is requesting contents with the content popularity distribu-
tion, \([p_m]\) which is set to be Zipf distribution with exponent \(\alpha\). As shown in the analysis, \(\alpha\) is an essential parameter when evaluating the network performance.

Each node generates up to \(T = 100\) requests for contents with respect to \([p_m]\). Then, using the path obtained from routing tables, the Interest Packet will be forwarded to the closest holder, and then, the holder generates a Data Packet containing the requested content. The DP will be forwarded toward the requester using the reverse path taken by the corresponding IP. All these packet transmissions are done by following the described scheduling scheme.

In Figure 1, the performance of the primarily proposed network paradigm is plotted. As you can see the average network delay in terms of number of time slots vs. the number of nodes, \(n\), is demonstrated for different Zip’s law exponent, \(\alpha\). Applying fitting techniques, it can be shown that the results given in (34) are the plotted curves shown in Figure 1. For instance, the curve corresponding to \(\alpha = 0.8\) can be fitted to \(\frac{68n^{0.4}}{\sqrt{\log n}}\) and the one corresponding to \(\alpha = 1.2\) can be fitted to \(\frac{215n^{0.24}}{\sqrt{\log n}}\). It is also worth noting that we have normalized the network delay by number of content requests, \(T\), which nodes request during the simulation.

Using the serving locally capability in the proposed network improves the performance by a constant factor. Figures 2 and 3 shows the performance of average network delay associated with both proposed methods vs. number of nodes for Zipf’s law exponent \(\alpha = 1.2\) and 1.6, respectively. In both figures, number of contents is assumed to be \(M = n^\beta\), where \(\beta = 0.8\). As it is demonstrated, the delay associated with serving locally enabled is of the same order of the delay associated with the case where it is not. In addition, as \(\alpha\) increases, the outperformance of the serving-locally method is more significant. The reason is that as \(\alpha\) grows, the content popularity decreases with a steeper slope and hence, the number of “popular contents” are less. Therefore, serving these popular contents locally can result in smaller delay.

Finally, the performance of the Hybrid network described in Section 7
Figure 1: The normalized average network delay in terms of number of time slots vs. the number of nodes

Figure 2: Delay improvement of the proposed network model by serving requests locally vs. number of nodes, when $\alpha = 1.2$. 
Figure 3: Delay improvement of the proposed network model by serving requests locally vs. number of nodes, when $\alpha = 1.6$.

Figure 4: Delay comparison between the proposed Hybrid and Ad Hoc network models vs. number of nodes.
is experimented through simulations. In Figure 4, the performance of the Hybrid network is compared with the pure Ad Hoc model. As the theoretical results suggest, the outperformance of the Hybrid network is significant when $\alpha < 1$. Here, we are considering $\alpha$ to be 0.8. Considering two cases where $\mu = 0.25$ (equivalently, $f(n) = 9n^{0.25}$), and where $\mu = 0.5$ (equivalently, $f(n) = 2n^{0.5}$), shows that increasing the number of Base Stations decreases the network delay, as expected. Using theoretical results given in (84), we obtain the scaling of the delay for the case $\alpha = 0.8$, and $\mu = 0.25$ to be $D = \Theta(\frac{n^{0.375}}{\sqrt{\log n}})$ and for $\mu = 0.5$ to be $D = \Theta(\frac{n^{0.25}}{\sqrt{\log n}})$. Comparing with the pure Ad Hoc model, for $\alpha = 0.8$, we get $D = \Theta(\frac{n^{0.24}}{\sqrt{\log n}})$ and for $\alpha = 1.2$, we have $D = \Theta(\frac{n^{0.24}}{\sqrt{\log n}})$. These results are verified in Figure 4.
9 Conclusions

We have investigated the asymptotic behavior of wireless networks with a Named Data networking architecture. Our study begins with precisely establishing the definitions of throughput and delay suitable for Named Data networking. After deriving the throughput-delay tradeoff of the proposed network paradigm for a general content popularity distribution, and formulating the problem of joint optimization of caching and forwarding strategies, we evaluated the network asymptotic performance for a Zipf content popularity distribution. We also studied the case where contents can have different sizes. Furthermore, we designed and analyzed a hybrid network model in which, there are multiple Base Stations in the network area, in addition to the mobile nodes. We gave answer to the questions: 1) How many Base Stations are required to improve the asymptotic performance of the network paradigm? 2) How good does the hybrid network perform under different content popularity distributions? Finally we verified our theoretical results through simulations.
Appendix A

In order to prove (10), we define \( g(n) \) as

\[
g(n) = \frac{n-1}{n} \cdot \frac{n-3}{n-2} \cdot \ldots \cdot \frac{2}{3}
\]  

(94)

We want to show that \( g(2X_m + 1) = \Theta(1/\sqrt{X_m}) \), or equivalently, \( g(n) \) scales as \( 1/\sqrt{n} \) where \( n \) is an odd number. Let \( n_1 \) and \( n_2 \) be two odd numbers and \( n_1 > n_2 \). Then, we can write

\[
g(n_1) = \frac{n_1-1}{n_1} \cdot \frac{n_1-3}{n_1-2} \cdot \ldots \cdot \frac{n_2+1}{n_2+2} \cdot g(n_2)
\]

Reordering the sequence in RHS and dividing both sides by \( g(n_2) \)

\[
\frac{g(n_1)}{g(n_2)} = \frac{n_2+1}{n_1} \left( \frac{n_1-1}{n_1-2} \cdot \frac{n_1-3}{n_1-4} \cdot \ldots \cdot \frac{n_2+3}{n_2+4} \right)
\]

\[
= \frac{n_2+1}{n_1} \cdot \frac{g(n_2+1)}{g(n_1-1)}
\]

Hence, we have

\[
\frac{g(n_1)}{g(n_2)} \cdot \frac{g(n_1-1)}{g(n_2+1)} = \frac{n_2+1}{n_1}
\]

(95)

Also, it can easily be shown that \( g(n) \) is a non-increasing function of \( n \). Thus,

\[
\frac{g(n_1-1)}{g(n_2+1)} \geq \frac{g(n_1)}{g(n_2)}
\]

(96)

Therefore,

\[
\left( \frac{g(n_1)}{g(n_2)} \right)^2 \leq \frac{g(n_1)}{g(n_2)} \cdot \frac{g(n_1-1)}{g(n_2+1)} = \frac{n_2+1}{n_1}
\]

(97)

Also,

\[
\left( \frac{g(n_1-1)}{g(n_2+1)} \right)^2 \geq \frac{g(n_1)}{g(n_2)} \cdot \frac{g(n_1-1)}{g(n_2+1)} = \frac{n_2+1}{n_1}
\]

(98)
Combining (97) and (98) we have the required bound for the proof:

\[
\frac{n_2}{n_1 + 1} \leq \left( \frac{g(n_1)}{g(n_2)} \right)^2 \leq \frac{n_2 + 1}{n_1}
\]  

(99)
Appendix B

We show the results for the average delay, and the average throughput can be calculated easily by equation (28). Substituting $p_j$’s equation (31) with the Zipf distribution, we obtain

$$D = \frac{H_\alpha(m_1)}{H_\alpha(M)} + \frac{[H_{\frac{3}{2}}(m_2 - 1) - H_{\frac{3}{2}}(m_1 - 1)]^{3/2}}{\sqrt{K' \log n H_\alpha(M)}}$$

$$+ \sqrt{\frac{n}{\log n}} \cdot \frac{H_\alpha(M) - H_\alpha(m_2 - 1)}{H_\alpha(M)}$$

where $K' = K - \frac{(m_1 - 1)}{\log n} - \frac{(M - m_2 + 1)}{n}$. Also, let the three expressions of the RHS of the equation be denoted as $D_1$, $D_2$, and $D_3$, respectively.

By Lemma 7, we know that for $\alpha \leq 3/2$, $m_2 = M + 1$. Therefore, $D_3$ is zero. Also, by Lemma 6, we know for $\alpha < 3/2$, $m_1 \to 1$ and for $\alpha = 3/2$, $m_1 = \Theta(1)$. Hence, $K' \to K$.

For $\alpha < 1$: For this case, $D_1 = o(1)$ and $D = \Theta(D_2)$

$$D = \Theta\left(\frac{(m_2 - 1)^{3/2 - \alpha}}{\sqrt{K \log n M^{1-\alpha}}}\right) = \Theta\left(\sqrt{\frac{M}{\log M}}\right)$$

where the last part comes from the fact that $\log M = \Theta(\log n)$.

For $\alpha = 1$: Similarly, for this case, $D_1 = o(1)$ and $D = \Theta(D_2)$

$$D = \Theta\left(\frac{(m_2 - 1)^{1/2}}{\sqrt{K \log n \log M}}\right) = \Theta\left(\frac{\sqrt{M}}{(\log M)^{3/2}}\right)$$

For $1 < \alpha < 3/2$: For $\alpha > 1$, $D_1 = \Theta(1)$. As shown in the following, $D_2 = \omega(1)$ and therefore, $D = \Theta(D_2)$. we have

$$D_2 = \Theta\left(\frac{M^{3/2-\alpha}}{\sqrt{K \log n}}\right) = \Theta\left(\frac{M^{3/2-\alpha}}{\sqrt{\log M}}\right)$$

58
For $\alpha = 3/2$: Similar to the $1 < \alpha < 3/2$ case, $D = \Theta(D_2)$, and we have:

$$D_2 = \Theta\left(\frac{(\log M)^{3/2}}{\sqrt{K \log n}}\right) = \Theta(\log M) \quad (104)$$

For $\alpha > 3/2$: $D_2 = \Theta\left(\frac{1}{\sqrt{\log n}}\right) = o(1)$. Also, as shown in the following $D_3 = o(1)$. Therefore, $D = \Theta(D_1) = \Theta(1)$.

$$D_3 = \Theta\left(\sqrt{\frac{n}{\log n} m_2^{1-\alpha}}\right) \quad (105)$$

Whether $m_2 = \Theta(M)$ or $m_2 = o(M)$, $m_2 = \Theta(n^{\frac{3}{2} \alpha}(\log n)^{1-\frac{3}{2} \alpha})$. If $m_2 = \Theta(M)$ then this result comes from (52). Otherwise, it comes from (54). In either way, $D_3 = o(1)$.

References


