Modeling data transfer in content-centric networking (extended version)

Giovanna Carofiglio[1], Massimo Gallo[2], Luca Muscariello[2], Diego Perino[1]
[1] Bell Labs, Alcatel-Lucent, France, [2] Orange Labs, France Telecom, France

[1]first.last@alcatel-lucent.com, [2]first.last@orange-ftgroup.com

Version: 31/05/2011

Keywords: Content-centric networking; Performance evaluation; Stochastic model.

Contents

Abstract

Content-centric networking proposals, as Parc's CCN, have recently emerged to define new network architectures where content, and not its location, becomes the core of the communication model. These new paradigms push data storage and delivery at network layer and are designed to better deal with current Internet usage, mainly centered around content dissemination and retrieval.

In this paper, we develop an analytical model of CCN in-network storage and receiver-driven transport, that more generally applies to a class of content oriented networks identified by chunk-based communication. We derive a closed-form expression for the mean stationary throughput as a function of hit/miss probabilities at the caches along the path, of content popularity and of rcontent/cache size. Our analytical results, supported by chunk level simulations, can be used to analyze fundamental trade-offs in current CCN architecture, and provide an essential building block for the design and evaluation of enhanced CCN protocols.

1 Introduction

Internet usage has significantly evolved in the last years, and today is mostly centered around content dissemination and retrieval. We assist to an exponential growth of digital information diffused over the Internet, eased by cheaper storage and bandwidth supports, and driven by the increasing popularity of highly demanding services, such as cloud computing or video delivery. On the other hand, Internet architecture is still based on the end-to-end model and appears to be unsuited to deal with the aforementioned trends. A large range of over-the-top (OTT) solutions, like Content Delivery Networks (CDNs), have been designed and widely deployed to overcome this mismatch at application layer, and today carry a large fraction of Internet traffic.

In parallel, significant research projetcs have been funded in the last years focusing on the definition of novel architecture designs of a future Internet (e.g. US NSF GENI [1], or EU FIA [2]). In this research arena, *content-centric* proposals, as Parc's CCN [3], PSIRP [4], or DONA [5], aim at redesigning the Internet architecture with named data as the central element of the communication paradigm, instead of its physical location. These proposals radically change data transfer by pushing content storage and delivery at network layer itself. Content-centric networks are in fact characterized by receiver-driven transport protocols (query or pull based), where data is only sent in response to users' request, and packet-level caching which is transparently performed at every node. Specifically, a content is split into a sequence of chunks[1] uniquely identified and separately requested by the receiver through explicit per-chunk requests. Data packets flow down to the receiver following the reverse path of requests and are cached by intermediate network nodes. These are distinctive features of content-centric architectures and play a fundamental role on data delivery.

A challenging problem is to analytically characterize data transfer in CCN given the complex interplay between receiver-driven transport and per-chunk caching. Even for a single storage node employing LRU (Least Recently Used) replacement policy, the analysis of cache dynamics under such request process is not straightforward. When the scenario under study is a network of caches operating under CCN principles, different modeling issues arise. The output process of the first node, that is the process of missing chunk

[1] Chunks are intended to be packet-size entities in CCN.

requests, needs to be precisely modeled in order to define the modified content request process feeding upstream caches.
Transport performance is affected by caching dynamics and a unified modeling framework is necessary to evaluate data transfer performance as well as to guide the design of optimized CCN protocols. To the best of our knowledge this is the first attempt to analytically study chunk-based data transfer in a network of caches as CCN.
Our results apply more generally to chunk-based content delivery networks based on similar transport principles. For instance, in chunk-based CDNs, as Coblitz [6] or Anycast CDN [7], the transport principles appear similar, however designed for different contexts. These are application-layer solutions making use of HTTP range requests to retrieve chunks of content that are stored in network web caches or different servers.
The main contributions of the paper are the following: (i) an analytical model of the single cache miss probability and, hence, miss rate under a two levels Markov Modulated Rate Process (MMRP) of requests with Zipf-distributed content popularity; (ii) a model for network of caches with and without request aggregation; (iii) a characterization of the stationary throughput and hence of content delivery time as a result of previous analysis; (iv) an assessment of model results via chunk-level simulations. The rest of the paper is organized as follows. Sec. 2 provides system description. Sec.3 is devoted to related work, while Sec. 4 introduces notation and main modeling assumptions. In Sec. 5 we report the main analytical results about single cache dynamics, while in Sec. 6 we extend the model to network case and provide explicit formulae for stationary throughput. Sec. 8 presents an example of model application while Sec. 9 concludes the paper.

2 System description

In this paper we primarily focus on the content-centric networking (CCN) proposal by Parc [3], though the modeling framework has broader applicability. Let us briefly describe how such systems work.

Content items are split into chunks uniquely identified by a name, and permanently stored in one (or more) repository. Users can retrieve them using a receiver-driven transport protocol based on per-chunk queries triggering data chunk delivery.

A name-based routing protocol guarantees that queries are properly routed towards data repository. Every intermediate node keeps track of pending queries, in order to deliver the requested chunks back to the receiver and temporarily cache data chunks in a LRU managed cache. In addition, intermediate nodes perform request aggregation (also denoted as filtering throughout the paper), i.e. avoid forwarding multiple requests for the same chunk while the first one is pending.

Data may come from the repository or from any hitting cache, that is a cache with a temporary copy of the data chunk, along the path. Chunks of the same content can therefore be retrieved from multiple locations with different round trip times (RTTs), affecting the delivery performance.

The resulting throughput and hence content delivery time are strongly affected by this notion of average distance between the user and the chunks of the requested content, which we will explicitly define by the variable *virtual round trip time* (VRTT) in analogy with connection-based transport protocols like TCP.

3 Related Work

Previous work on content-centric networks has mainly focused on global architecture design ([3] [4], [5]) while less effort has been devoted to analyze caching and transport mechanisms in such architectures. More recently, Somaya *et al.* [8] analyze the feasibility of caching in routers at line-speed, while Lee *et al.* [9] consider the benefits of CCN in-network storage in terms of energy efficiency with respect to traditional distribution architectures. Also, Carofiglio *et al.* show the role played by storage management in CCN by means of experimental evaluation [10]. However, none of the aforementioned works provides an analytical characterization of transport performance, and its interaction with chunk-level caching dynamics. In the context of Web caching there have been previous attempts to modeling content-level cache dynamics, most of them related to a single cache scenario under LRU replacement policy. The majority of analytical models of LRU caches start from the relation between the LRU miss probability, and he tail of the search cost distribution for the Move-To-Front (MTF) searching algorithm ([11]). In [12] an integral expression for the Laplace transform of the search cost distribution function is derived, that needs to be numerically integrated with complexity proportional to the cache size and the number of contents. Alternative combinatorial approaches are developed in [13]- [14]. In [15], authors give an asymptotic characterization, for a large number of contents, of the MTF search cost distribution and hence of the LRU miss probabilities both in the light-tailed and in the heavy-tailed case. A recent work in [16] provides an analytical characterization of the miss probability and thus miss rate under Poisson assumptions of content requests' arrivals.

It is worth to remark that almost all of these prior studies are devoted to the analysis of LRU-based rules for a single cache and with unit-sized objects. In [17] authors apply a single LRU cache model to study a cache coordination technique, whereas recently, the approximated single cache model of [18] has been applied in [19] to a network of caches, under the Independence Request Model (IRM) assumption.

To the best of our knowledge no attempts have been done to model LRU cache dynamics (i) at network level, (ii) in chunk-based systems where content retrieval is receiver-driven, imposing a certain correlation structure either in the request process and in cache dynamics.

4 Model Description

4.1 Assumptions and notation

In our setting, we make the following assumptions.

- We consider a set of M different content items equally partitioned in K classes of popularity, i.e. content items of class k are requested with probability $q_k, \;\; k = 1, \dots, N$. In the rest of the paper, we assume a Zipf popularity distribution, hence $q_k = c/k^{\alpha}$, with parameter $\alpha > 1$.
- Content items are segmented into chunks and have different sizes: σ denotes the average content size in terms of number of chunks (chunks are assumed to be of the same size);

N	Number of network nodes
M	Number of different content items ($m = M/K$ in each class k)
x	Content Store (Cache) size
$\lambda, \lambda(i)$	Total content request rate at first node, at node $i > 1$
λ_k	Content request rate for class k w/o filtering
λ_k^f	Content request rate for class k with filtering
σ	Average content size in number of chunks
$q_k, q_k(i)$	Content popularity distribution of requests at node 1,i
$p_k(i)$	Miss probability for class k at node i
$p_k^f(i)$	Miss probability with filtering for class k at node i
R_i	Round trip delay between client and node i
VRTT_k	Virtual round trip delay of class k w/o filtering
VRTT_k^f	Virtual round trip delay of class k with filtering
$\text{RVRTT}_k(i)$	Residual VRTT_k at node i w/o filtering
$\text{RVRTT}_k^f(i)$	Residual VRTT_k at node i with filtering
Δ, Δ_k	Filtering time windows imposed and effective for class k
X_k	Chunk delivery rate or throughput of class k

Table 1: Notation

- Each node in the network has a cache (also referred to as content store in Parc's CCN) of size x chunks.
- We focus on different topologies (fig.3), meant to represent (segments of) the aggregation network which gathers a large number of users requests over time.
- We define virtual round trip time of class k, VRTT_k, the average time that elapses between the dispatch of a chunk request and the chunk reception in steady state. This variable plays a similar role to what the round trip time is for TCP connections in IP networks.

$$\text{VRTT}_k = \sum_{i=1}^{N} R_i(1 - p_k(i)) \prod_{j=1}^{i-1} p_k(j) \qquad (1)$$

with $k = 1, \dots, K$. $VRTT_k$ is defined as a weighted sum of the round trip delays R_i associated to node i, where weights correspond to the stationary hit probabilities $(1-p_k(i))$ to find a chunk of class k at node i given that a miss was originated by all previous nodes. More details are provided in Sec.6.

4.2 Content requests process

The content request process is structured in two levels, *content* and *chunk*, which is important to characterize.

In this section we build a fluid model of such two level request process that captures the first-order dynamics of the system in steady state. The main assumption behind is that lowest level nodes in Fig.3 aggregate requests issued by a large number of users.

The request arrival process is modeled through a Markov Modulated Rate Process (MMRP) [20] as depicted in fig.1: requests for contents in class k are generated according to a Poisson process of intensity

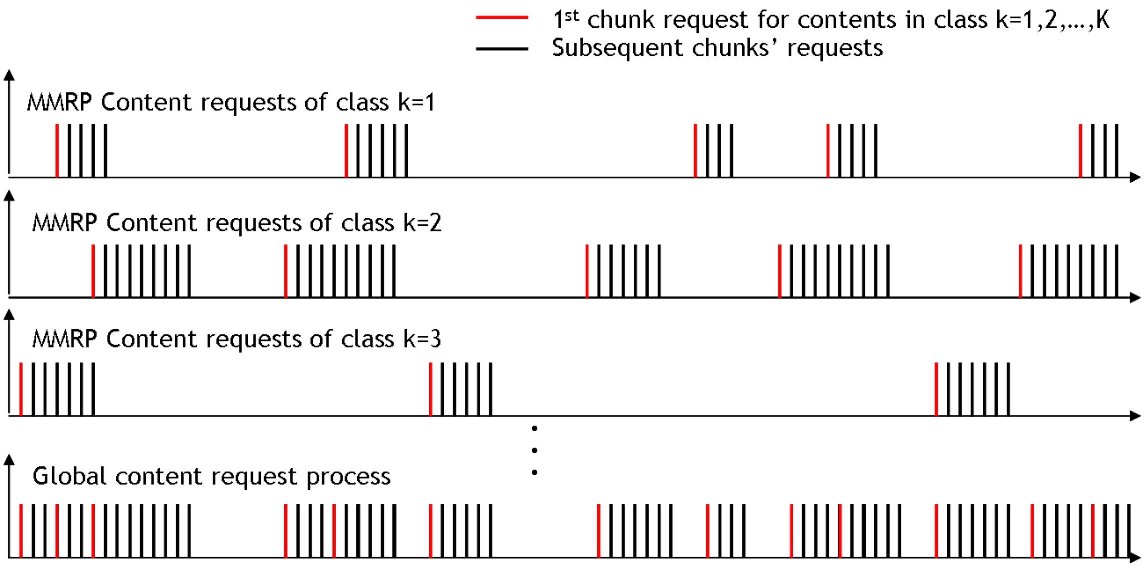


Figure 1: Single class content request process and global content request process.

$\lambda_k = \lambda q_k$, and the content to be requested is uniformly chosen among the m different contents in the given popularity class. A content request coincides with the request of the first chunk of the content. Once a chunk is received, a new chunk request is emitted and so on until the reception of the last chunk of the content. The model applies to the more general case of a window of $W > 1$ chunks requested in parallel.

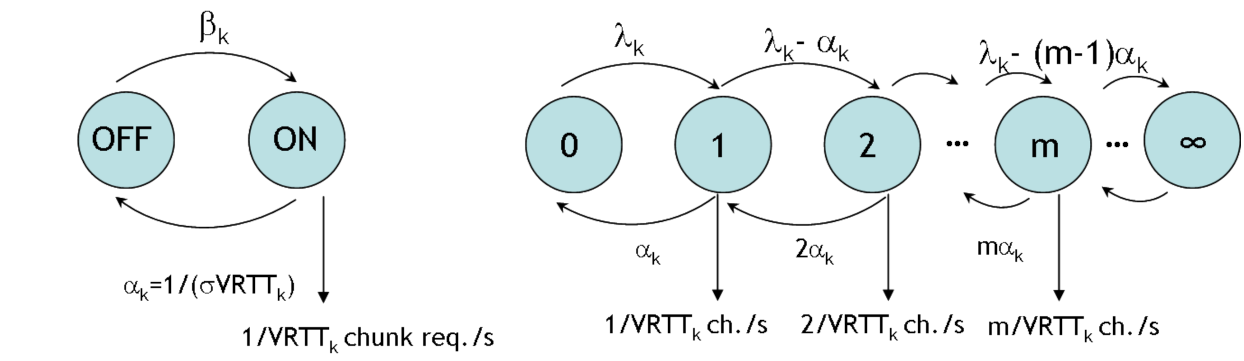
Observation 4.1. *Note that the interest rate control proposed for CCN [3] is initially a simple pipelining and in CCN it remains an open issue how to let W vary over time through a sliding window control. Our work aims at characterizing analytically the average content delivery time as a function of network parameters (content popularity, content size, cache size, topology, content request rate) in order to later design an ad-hoc receiver-based flow control.*

The number of chunks requested, N_{ch}, is assumed to be geometrically distributed with mean σ, i.e.

$$\mathbb{P}(N_{ch} = s) = \frac{1}{\sigma}\left(1 - \frac{1}{\sigma}\right)^{s-1}, \quad s = 1, 2, 3, ...$$

. This means that we can never have zero chunks in a content request. The inter-arrival between two subsequent chunk requests is deterministic and equal to the average Virtual Round Trip Time of class k, VRTT_k.

Observation 4.2. *Virtual Round Trip Times are not constant in practice as the hit/miss probabilities may vary over time. However, we look at the system in steady state under a stationary content request process where the average hit/miss probabilities and hence $VRTT_k$ have converged to a constant value.*



(a) Markov chain of a single content request. Deterministic inter-arrivals

(b) Superposition of content requests of class k. MMRP underlying Markov chain

Figure 2: Single content request (a) and superposition of class k content requests (b) graph model.

After the ON phase, where all chunks of a given content are requested, a residual exponential time represents the OFF phase until the next request for that given content. The underlying Markov chain is reported in fig.2(a): $\alpha_k = 1/(\sigma VRTT_k)$ and $\beta_k = (1/\lambda_k - 1/\alpha_k)^{-1}$ represent respectively the transition rate from ON to OFF and viceversa. The superposition of different content requests defines the MMRP process (fig.1), whose underlying Markov chain is represented in fig.2(b). Note that besides the limited number of different content per popularity class (m) there can be multiple ongoing requests for the same content as the request process aggregates a large number of users' requests.

The choice of a MMRP model is natural within the context of chunk-based content centric networks: in fact, *at content level* the Poisson assumption is motivated by previous work on Internet traffic modeling at session level, whereas it results to fail at packet/flow level [21]. *At chunk level* we suppose to look at system dynamics in *steady state* where the content store dynamics and hence the *chunk level rate* X_k, $k = 1, ..., K$, have converged to a stationary value. The large number of content requests served by the aggregation network justifies the fluid assumption behind the MMRP [20]. It is worth to remark that we do not assume any other temporal correlation in the input process as in the IRM (Independence Reference Model), but we model the temporal correlation induced by content request aggregation (filtering). Such feature allows to keep trace of the ongoing requests at each node and avoid forwarding chunk requests whether a request for the same chunk has been already sent. A more detailed description of the filtering operation performed by nodes is given in Sec.5.1.

5 Single cache model

Let us now characterize the miss probability in steady state of class $k = 1, ..., K$ at the first cache, under the content request process above described. In [16], authors give an analytical characterization of the miss

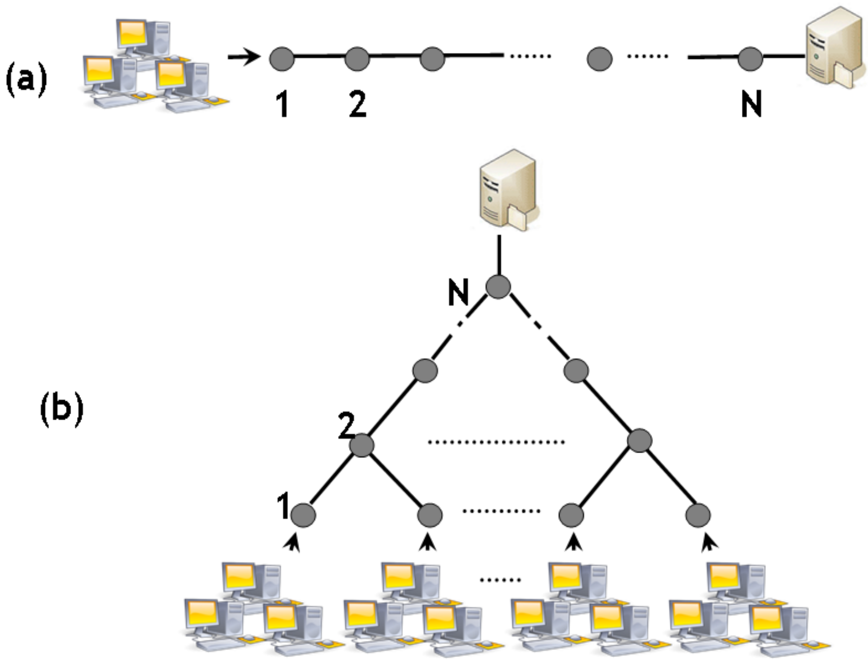


Figure 3: Network topologies: cascade (a), binary tree (b).

probability and hence of the miss rate for the case of a Poisson content request process when the cardinality of the set of different contents M tends to infinity. In this section we extend their result to the case of

- a MMRP content request process (content/chunk levels) with content request rate $\lambda_k = \lambda q_k$, where $q_k = c/k^\alpha$,
- K popularity classes partitioning with the same number of contents $m = M/K > 1$ in each one,
- non-unitary content size (σ is the average content size in chunks),
- pending request aggregation (filtering) over a time interval Δ (sec.5.1).

Proposition 5.1. *Given a MMRP content request arrival process as described in Sec.4.2 with intensity λ, popularity distribution $q_k = \frac{c}{k^\alpha}$, $\alpha > 1$, $c > 0$, and average content size σ, be $x > 0$ the cache size in number of chunks, then the stationary miss probability for chunks of class k, p_k, is given by*

$$p_k \equiv p_k(1) \sim e^{-\frac{\lambda}{m} q_k g x^\alpha} \tag{2}$$

for large x, where $1/g = \lambda c \sigma^\alpha m^{\alpha-1} \Gamma\left(1 - \frac{1}{\alpha}\right)^\alpha$.

Before evaluating the miss process of the cache, let us first give the intuition behind the proof. A request for a given chunk generates a cache miss when more than x different chunks are requested after its

previous request (it implies that the given chunk has been removed from the cache before the arrival of the new request).
In addition, thanks to the memory-less property of the Poisson process (content level) and of the geometric content size distribution (chunk level), for a given class k the number of chunk requests in the open interval (τ_{n-1}, τ_n) is independent from the number of chunk requests in (τ_n, τ_{n+1}), where $\{\tau_1, \tau_2, ...\}$ is the sequence of points at which a miss in class k occurs. In addition, the distribution of the number of requests in between two consecutive requests for the same chunk can be computed from that of the number of requests among two subsequent requests for any two chunks of class k.

Proof. We define $R^{i(c_k)}(u,t)$ as the number of requests in the open interval (u,t) of a chunk i of content c_k in class k $R^{i(c_k)}(u,t) \sim \textit{Poisson}(\lambda q_k/m)$ and $B^{i(c_k)}(u,t) = \mathbb{1}_{\{R_{i(c_k)}(u,t)>0\}}$ the Bernoulli variable associated to the event that at least one chunk i in class k is requested in the open interval (u,t) with $\mathbb{P}[B^{i(c_k)}(u,t) = 1] = 1 - e^{-\frac{\lambda q_k}{m}(t-u)}$. $S(u,t)$ denotes the number of different chunks requested over all popularity classes in (u,t),

$$S(u,t) = \sum_{k=1}^{K} \sum_{c_k=1}^{m} \sum_{i(c_k)=1}^{N_{ch}} B^{i(c_k)}(u,t) \tag{3}$$

In following lemmas we will also use $S^{(i,k)}(u,t)$ defined as the sum of different requested chunks between two subsequent requests for chunk i of content c_k, in class k in (u,t),

$$S^{(i(c_k))}(u,t) = \sum_{k'=1}^{K} \left(\sum_{c'_k=1}^{m} \sum_{j(c'_k)=1}^{N_{ch}} B^{j(c'_k)}(u,t) \right) \tag{4}$$

We start by proving the following auxiliary results, Lemmas 5.2 and 5.3.

Lemma 5.2. *Under above assumptions, as $K \to \infty$, $t \to \infty$*

$$1/g \equiv \lim_{t\to\infty} \frac{\mathbb{E}[S(0,t)]^\alpha}{t} = \lambda c \sigma^\alpha m^{\alpha-1} \Gamma\left(1 - \frac{1}{\alpha}\right)^\alpha . \tag{5}$$

Proof. First notice that for a given chunk in class k, $i(c_k)$,

$$\mathbb{E}[B^{i(c_k)}(u,t)] = \mathbb{P}(B^{i(c_k)}(u,t) = 1) = \mathbb{P}(R^{i(c_k)}(u,t) > 0) = 1 - e^{-\frac{\lambda q_k}{m}(t-u)}.$$

This is due to the fact the contents with the same popularity are requested uniformly with probability $1/m$, that $\{R^{i(c_k)}(0,t), t \geq 0\}$ is a Poisson process of rate $\lambda q_k/m$ by MMRP properties and chunks are indistinguishable.

In order to compute $\lim_{t\to\infty} \mathbb{E}[S(0,t)]$, let us first evaluate the following lower bound as $K \to \infty$,

$$
\begin{aligned}
\mathbb{E}[S(0,t)] &= \sum_{k=1}^{\infty} \sum_{c_k=1}^{m} \mathbb{E}\left[\sum_{i(c_k)=1}^{N_{ch}} B^{i(c_k)}(0,t)\right] = m\sigma \sum_{k=1}^{\infty} \left(1 - e^{-\frac{\lambda q_k}{m} t}\right) \\
&= m\sigma \sum_{k=1}^{\infty} \int_k^{k+1} \left(1 - e^{-\frac{\lambda q_k}{m} t} du\right) \geq m\sigma \sum_{k=1}^{\infty} \int_k^{k+1} \left(1 - e^{-\frac{\lambda q_u}{m} t}\right) du \\
&= m\sigma \int_0^{\infty} \left(1 - e^{-\frac{\lambda}{m} \frac{c}{u^\alpha} t}\right) du = \left(\frac{\lambda c}{m} t\right)^{1/\alpha} \frac{m\sigma}{\alpha} \int_0^{\frac{\lambda c}{m} t} v^{-1/\alpha+1} (1 - e^{-v}) dv \\
&\sim \Gamma\left(1 - \frac{1}{\alpha}\right) m\sigma \left(\frac{\lambda c t}{m}\right)^{1/\alpha} \quad t \to +\infty.
\end{aligned} \tag{6}
$$

Similarly one can derive the following upper bound:

$$
\begin{aligned}
\mathbb{E}[S(0,t)] &= m\sigma \left(1 - e^{-\frac{\lambda}{m} ct}\right) + m\sigma \sum_{k=1}^{\infty} \int_k^{k+1} \left(1 - e^{-\frac{\lambda q_k}{m} t} du\right) \\
&\leq m\sigma \left(1 - e^{-\frac{\lambda}{m} ct}\right) + m\sigma \sum_{k=1}^{\infty} \int_k^{k+1} \left(1 - e^{-\frac{\lambda q_u}{m} t}\right) du \\
&= m\sigma \left(1 - e^{-\frac{\lambda}{m} ct}\right) + m\sigma \int_0^{\infty} \left(1 - e^{-\frac{\lambda}{m} \frac{c}{u^\alpha} t}\right) du \\
&= m\sigma \left(1 - e^{-\frac{\lambda}{m} ct}\right) + \left(\frac{\lambda c}{m} t\right)^{1/\alpha} \frac{m\sigma}{\alpha} \int_0^{\frac{\lambda c}{m} t} v^{-1/\alpha+1} (1 - e^{-v}) dv \\
&\sim m\sigma + \Gamma\left(1 - \frac{1}{\alpha}\right) m\sigma \left(\frac{\lambda c t}{m}\right)^{1/\alpha} \quad t \to +\infty.
\end{aligned} \tag{7}
$$

As a consequence, if we consider $\lim_{t\to\infty} \frac{\mathbb{E}[S(0,t)]^\alpha}{t}$, both upper and lower bounds coincide with

$$
1/g \equiv \lim_{t\to\infty} \frac{\mathbb{E}[S(0,t)]^\alpha}{t} = \lambda c \sigma^\alpha m^{\alpha-1} \Gamma\left(1 - \frac{1}{\alpha}\right)^\alpha .
$$

□

Further, recall that

$$
S^{(i(c_k))}(0,t) = \sum_{k'=1}^{K} \left(\sum_{c'_k=1}^{m} \sum_{j(c'_k)=1}^{N_{ch}} B^{j(c'_k)}(0,t) \right)
$$

Let also denote by $\{t_n^{(i_k)}\}_{n>0}$ the time sequence of requests for the given chunk i (of content c_k in class k) and $\{t_n^{(i_k)}\}_{n>0}$ the inter-arrival time sequence, where $\tau_n^{(i_k)} = t_{n+1}^{(i_k)} - t_n^{(i_k)}$. Note that we omitted the indication of the content, c_k to simplify the notation. The following lemma characterizes the miss process for a given chunk i (of content c_k) in class k and states the independence of $S^{(i(c_k))}$ over time intervals among two misses of the given chunk.

Lemma 5.3.

$$\mathbb{P}[\textit{a miss of chunk } i \textit{ of content } c_k \textit{ in class } k \textit{ occurs at times} \quad t_1^{(i_k)}, t_2^{(i_k)}, ..., t_n^{(i_k)}] = \mathbb{P}[S^{(i_k)}(0, \tau^{(i_k)}) > x]^n \tag{8}$$

where $\tau^{(i_k)}$ is exponentially distributed with parameter λ_k/m (the inter-arrival among two requests for the same chunk within class k).

Proof. Due to the property of LRU replacement policy, chunk i_k is moved to the front of the list at each $t_n^{(i_k)}$. A miss event is defined by $\{S^{(i_k)}(t_{n-1}^{(i_k)}, t_n^{(i_k)}) \geq x\}$, i.e. the sum of different chunks requested after the previous miss event for chunk i is larger than the cache size, x. Therefore

$$\begin{aligned}
&\mathbb{P}[\text{a miss of chunk } i \text{ of content } c_k \text{ in class } k \text{ occurs at times} \quad t_1^{(i_k)}, t_2^{(i_k)}, ..., t_n^{(i_k)}] = \\
&\mathbb{P}[S^{(i_k)}(t_1^{(i_k)}, t_2^{(i_k)}) \geq x, S^{(i_k)}(t_2^{(i_k)}, t_3^{(i_k)}) \geq x, ..., S^{(i_k)}(t_{n-1}^{(i_k)}, t_n^{(i_k)}) \geq x]
\end{aligned}$$

The independence among $S^{(i_k)}(t_l^{(i_k)}, t_{l+1}^{(i_k)})$ and $S^{(i_k)}(t_{l+j}^{(i_k)}, t_{l+j+1}^{(i_k)})$, $j \geq 1$ is a consequence of the memory-less Poisson property at content level and of the geometric content size distribution in terms of requested chunks for a given content request. By recalling its definition (4), we have

$$\begin{aligned}
\mathbb{P}[S^{(i(c_k))}(t_1^{(i_k)}, t_2^{(i_k)}) > x] &= \sum_{v_1+v_2+...v_K=1}^{M} \sum_{k'=1}^{K} \mathbb{P}[B^{j_{k'}}(0,t) = 1]^{v_{k'}} \mathbb{P}[vN_{ch} \geq x] \\
&= \sum_{v_1+v_2+...+v_K=1}^{M} \sum_{k'=1}^{K} \left(1 - e^{-\frac{\lambda}{m} q_{k'}(t_2^{(i_k)} - t_1^{(i_k)})}\right)^{v_{k'}} \mathbb{P}[vN_{ch} \geq x] \\
&= \sum_{v_1+v_2+...+v_K=1}^{M} \sum_{k'=1}^{K} \left(1 - e^{-\frac{\lambda}{m} qk'(\tau_1^{(i_k)})}\right)^{v_{k'}} \mathbb{P}[N_{ch} \geq x/v],
\end{aligned}$$

where we sum over the probability to have at least one chunk for v_1 contents in class $k = 1$, v_2 contents in class $k = 2$ and so on, so that $v_1 + v_2 + ... + v_K = v$, times the conditioned probability to have more than x different chunks when having v different content requests. The memory-less property of the Poisson process at content level and of the geometric distribution at chunk level allow to conclude that $\mathbb{P}[S^{(i(c_k))}(t_1^{(i_k)}, t_2^{(i_k)}) > x]$ only depends on the length of the interval $\tau_1^{(i_k)}$, which is equivalent to say that in steady state,

$$\mathbb{P}[\text{a miss of chunk } i \text{ of content } c_k \text{ in class } k \text{ occurs at times} \quad t_1^{(i_k)}, t_2^{(i_k)}, ..., t_n^{(i_k)}] = \mathbb{P}[S^{(i_k)}(0, \tau^{(i_k)}) > x]^n$$

□

The next proposition characterizes the stationary distribution of $S^{i(c_k)}$ among two miss events for chunk $i(c_k)$, $k = 1, ..., K$.

Proposition 5.4. *If $q_k = \frac{c}{k^\alpha}$, $\alpha > 1$, $c > 0$, $i = \lfloor \delta x \rfloor$, $t_n^{(i)}$ is the n-th arrival time of a given chunk i belonging to class k, then*

$$\lim_{x\to\infty} \mathbb{P}[S^{(i)}(t_n^{(i)}, t_{n+1}^{(i)}) \geq x] \sim \mathbb{P}[\tau^{(k)} \geq g x^\alpha] = e^{-\lambda q_k g x^\alpha} = e^{-\frac{m}{k^\alpha \Gamma(1-1/\alpha)^\alpha}\left(\frac{x}{\sigma m}\right)^\alpha}, \quad x \to \infty \tag{9}$$

with $1/g = \lambda c \sigma^\alpha m^{\alpha-1} \Gamma\left(1 - \frac{1}{\alpha}\right)^\alpha$, and $\tau^{(k)}$ denoting the inter-arrival among two subsequent requests of chunks of class k.

Proof. We first evaluate the upper bound.

$$\begin{aligned}
\mathbb{P}[S^{(i)}(t_n^{(i)}, t_{n+1}^{(i)}) \geq x] &= \mathbb{P}[S^{(i)}(0, \tau^{(k)}) \geq x] \\
&= \mathbb{P}[S^{(i)}(0, \tau^{(k)}) \geq x, \tau^{(k)} \geq g x^\alpha] + \mathbb{P}[S^{(i)}(0, \tau^{(k)}) \geq x, \tau^{(k)} < g x^\alpha] \\
&\leq \mathbb{P}[S^{(i)}(0, \tau^{(k)}) \geq x, \tau^{(k)} \geq g x^\alpha] + \mathbb{P}[S^{(i)}(0, g x^\alpha) > x] \\
&\leq \mathbb{P}[\tau^{(k)} \geq g x^\alpha] + \mathbb{P}[S(0, g x^\alpha) > x] \sim \mathbb{P}[\tau^{(k)} \geq g x^\alpha], \quad x \to \infty
\end{aligned} \tag{10}$$

where we have used monotonicity of $S(0,t)$ in t as well as $\mathbb{P}[S(0, g x^\alpha) > x] \to 0, x \to \infty$.
The lower bound can also be obtained

$$\begin{aligned}
\mathbb{P}[S^{(i)}(0, \tau^{(k)}) \geq x] \geq \mathbb{P}[S^{(i)}(0, \tau^{(k)}) \geq x, \tau^{(k)} > g x^\alpha] &\geq \mathbb{P}[S(0, g x^\alpha) > x, \tau^{(k)} > g x^\alpha] \\
&= \mathbb{P}[S(0, g x^\alpha) \geq x] \mathbb{P}[\tau^{(k)} > g x^\alpha] \sim \mathbb{P}[\tau^{(k)} \geq g x^\alpha], \quad x \to \infty
\end{aligned} \tag{11}$$

Where we have used the law of large numbers for $S(0, g x^\alpha)/x \to \mathbb{E}[S(0, g x^\alpha)]/x = 1$ in probability. □

The main statement of Prop.5.1 follows from the next theorem which is based on previous results and allows to conclude on the miss rate at the cache from the superposition of the miss sequences of each chunk.

Theorem 5.5. *$\forall k = 1, ..., K$ let $i(1_k), i(2_k), ..., i(m_k)$ denote the i^{th} chunk of content $1, 2, ..., m$ in class k. Given that all chunks have the same size, each one can be expressed as a function of cache size x as $i(1_k) = \lfloor \delta x \rfloor$. Then, if we denote by $\mathcal{M}(t_l^{i(c_k)})$ the l^{th} the event of a miss at time $t_l^{i(c_k)}$ for chunk i of content c in class k we have that*

$$\begin{aligned}
&\mathbb{P}[\mathcal{M}(t_1^{1(1_1)}), \mathcal{M}(t_2^{1(1_1)}), ..., \mathcal{M}(t_{k_1}^{1(1_1)}), \mathcal{M}(t_1^{2(1_1)}), \mathcal{M}(t_2^{2(1_1)}), ..., \mathcal{M}(t_{k_1}^{2(1_1)}), ..., \\
&\mathcal{M}(t_1^{1(2_1)}), \mathcal{M}(t_2^{1(2_1)}), ..., \mathcal{M}(t_{k_1}^{1(2_1)}), ..., \mathcal{M}(t_1^{1(m_1)}), \mathcal{M}(t_2^{1(m_1)}), ..., \mathcal{M}(t_{k_1}^{1(m_1)}), ..., \\
&\mathcal{M}(t_1^{1(m_2)}), \mathcal{M}(t_2^{1(m_2)}), ..., \mathcal{M}(t_{k_2}^{1(m_2)}), ..., \mathcal{M}(t_1^{1(m_K)}), \mathcal{M}(t_2^{1(m_K)}), ..., \mathcal{M}(t_{k_K}^{1(m_K)})] \\
&= e^{-\lambda c g \frac{k_1}{\delta^\alpha}} e^{-\lambda c g \frac{k_2}{\delta^\alpha}} \cdots e^{-\lambda c g \frac{k_K}{\delta^\alpha}},
\end{aligned} \tag{12}$$

that is the probability to have misses for the chunk i of the content c in class k at $(t_1^{i(c_k)}, t_2^{i(c_k)}, ..., t_k^{i(c_k)})$ is equal to the product of the probabilities of having that miss sequence for one chunk of each content in each class.

Proof. The proof is divided into steps:

1. we show that $\forall k = 1, ..., K$, $\forall c_k$ (i.e. content in class k),

$$\mathbb{P}[\mathcal{M}(t_1^{i(c_k)}), \mathcal{M}(t_2^{i(c_k)}), ..., \mathcal{M}(t_l^{i(c_k)}), \mathcal{M}(t_1^{(i+1)(c_k)}), \mathcal{M}(t_2^{(i+1)(c_k)}), ..., \mathcal{M}(t_l^{(i+1)(c_k)})]$$
$$\mathbb{P}[\mathcal{M}(t_1^{i(c_k)}), \mathcal{M}(t_2^{i(c_k)}), ..., \mathcal{M}(t_l^{i(c_k)})], \qquad \forall l \geq 1;$$

2. we apply Prop.5.4 to explicitly compute the miss probabilities.

1. Prop.5.4 allows to state that $\mathbb{P}[\mathcal{M}(t_l^{i(c_k)})] = \mathbb{P}[S^{i(c_k)}(t_{l-1}^{i(c_k)}, t_l^{i(c_k)}) \geq x] = \mathbb{P}[\tau^{i(c_k)} > gx^\alpha]$, $\forall l \geq 1$. Now, if the interval between two consecutive misses for chunk i is $\tau^{i(c_k)}$, the interval between two consecutive misses for the following chunk, $i+1$ of that specific content can not be shorter than $\tau^{i(c_k)}$. In fact, either the chunk $i+1$ is requested after chunk i both one $VRTT_k$ after $t_{l-1}^{i(c_k)}$ and one $VRTT_k$ after $t_l^{i(c_k)}$ or in one of the two cases the content request stops at chunk i. Given that $\tau^{(i+1)(c_k)} \geq \tau^{i(c_k)} > gx^\alpha$, the miss event for chunk i implies a miss event for chunk $i+1$ (when requested) with probability equal to one, since $\tau^{(i+1)(c_k)} \geq \tau^{i(c_k)}$, which concludes the first step of the proof.
2. The second step follows by Prop.5.4, when applied on the first chunk of each content of each popularity. The independence between misses of different content requests as $x \to \infty$ can be proved by using lower and upper bounds as in the proof of Prop.5.4 and it happens to be a consequence of the memoryless Poisson property of content-level requests. □

As a consequence of the previous theorem, the miss sequences for contents in popularity classes of the order of x are asymptotically mutually independent. Thus the miss process at content level, which constitutes the input process for caches at second hop (level), has an intensity

$$\mu_k = \lambda_k \exp\left\{ -\frac{\lambda}{m} q_k g x^\alpha \right\}, \qquad \forall k = 1, ..., K. \tag{13}$$

Observation 5.6. *The output process of the first node can be well approximated by a renewal process. Let us see why. Consider the sequence of chunks inter-arrivals of a given class k. After a chunk miss a request is sent upstream and is followed by a chunk miss of the same class with probability p_k or by a hit with probability $1-p_k$. The next miss arrives after a geometric number h of hits with probability $(1-p_k)^h p_k$. The next miss can be approximated by a geometric sum of exponential independent inter-arrivals exponentially distributed of parameter λ_k with a support translated by gx^α which might be seen as a critical value that indicates that two consecutive misses cannot take place before gx^α.*

□

5.1 Content requests aggregation

In chunk-based content delivery systems a chunk request is issued and forwarded through the network until the given chunk is found. A fundamental feature to avoid chunk requests flooding is interest aggregation in CCN, which is a mechanism taking trace of pending chunk requests at each node and preventing the

dispatch of new requests for the same chunk. This mechanism has to be accounted when studying the interaction between transport and caching in such chunk-based content delivery systems as it impacts both throughput and network caches behavior.
Let us denote by Δ the time window where requests for the same chunk are aggregated and hence all requests after the first one are not propagated.

Observation 5.7. *In steady state, the effective timescale aggregation for chunk requests of content of class k is $\Delta_k(i) = \min(\Delta, RVRTT_k(i))$, where $RVRTT_k(i)$ represents the residual virtual round trip time of class k at hop i, that is the time that elapses on average until the reception at node i of the chunk of data requested. Clearly, $VRTT_k = RVRTT_k(1)$.*

Indeed, one can reasonably assume Δ is taken larger than the virtual round trip time in order to effectively aggregate requests, but in practice the request aggregation is done until the on-the-fly request is satisfied and the data chunk is received. From that time on, the chunk is stored in cache, and whether removed by the replacement policy a new chunk request has to be forwarded.
The request aggregation or filtering clearly impacts the miss rate of a given cache. In fact, when a chunk of class k request arrives at one cache, it can generate a *hit* if the chunk is found in cache, otherwise it generates a *miss*. In the latter case, the request is filtered only if a previous request for the same chunk have been emitted and the chunk has not yet been received (e.g. the time elapsed by the previous chunk request emitted is smaller than Δ_k).
Let us now compute the filtering probability and thus the filtered miss rate at the lowest level of caches in Fig.3(b).

Proposition 5.8. *Given the content request process defined in sec.4.2, the filtering probability associated to class k at the first hop is,*

$$p_{filt,k}(1) = \frac{1-b_k}{1-(1-1/\sigma)b_k} \quad k = 1, ..., K \tag{14}$$

with $b_k = e^{-\Delta_k \lambda_k}$, and $\Delta_k = \min(\Delta, VRTT_k)$.

Proof. First, observe that if the l-th chunk of a given content is filtered, then chunk $l+1$ is filtered with $1 \leq l \leq \sigma - 1$. This means that the l-th chunk is filtered when the following event is verified: $F_l = \bigcup_{m=1}^{l-1} E_m$, $E_m = \{\text{chunk } m \text{ is filtered and chunk } m-1 \text{ is not filtered}\}$.
$\mathbb{P}\{E_m\} = p(1-p)^{m-1}$ where p is the probability that the first chunk is filtered, easily computed by observing that content request inter-arrivals $\sim \text{Exp}(\lambda_k)$, hence $p = 1 - e^{-\lambda_k \Delta_k}$ is the probability that the inter-arrival among two request of the first chunk of the same content is smaller than Δ_k with $\Delta_k = \min(\Delta, VRTT_k)$. Therefore $F_l = \sum_{m=1}^{n} p(1-p)^{m-1} = 1-(1-p)^n$, being n the number of chunks. Defining $b_k = 1-p = e^{-\lambda_k \Delta_k}$ and averaging over the number of chunks belonging to a content $N_c h$, which is geometrically distributed with mean σ, we obtain

$$p_{filt,k}(1) = \mathbb{E}[1-b_k^{N_{ch}}] = 1 - \mathbb{E}[b_k^{N_{ch}}] = 1 - \frac{b_k/\sigma}{1-(1-1/\sigma)b_k} = \frac{1-b_k}{1-(1-1/\sigma)b_k}$$

Just using the fact that the p.g.f of the geometric distribution $\mathbb{E}[z^{N_{ch}}] = \frac{z/\sigma}{1-(1-1/\sigma)z}$. □

6 Network of caches

In this section we apply the results in Sec.4 to the study of the topologies in Fig.3 and derive analytical expressions for the miss probabilities/rates at every node (level) (Sec.6.1). Furthermore we obtain for the throughput (Sec.6.3) in steady state with and without request aggregation. As in previous sections, we consider the case $M_k = m = M/K, \forall k = 1, ..., K$, and $q_k = c/k^\alpha$.

We introduce here the following notation. $\mu(i)$ indicates the miss rate at a node i while $\lambda(i)$ indicates the request rate at node i. This means that for a line $\lambda(i+1) = \mu(i)$ and for a binary tree $\lambda(i+1) = 2\mu(i)$. We set $\mu(0) = \lambda(1) = \lambda$.

6.1 Miss rate characterization

In order to derive the stationary miss probabilities at hop i, with $i > 1$ for the topology under study, we need the following auxiliary result which extends Lemma 5.2 to hops after the first.

Lemma 6.1. *Given a MMRP content request process with rate $\lambda(i)$ and popularity distribution $q_i(k) = \prod_{j=1}^{i-1} p_k(j) q_k / \sum_{l=1}^{K} \prod_{j=1}^{l-1} p_l(j) q_l, \quad k = 1, ..., K$, (where $q_k = c/k^\alpha$, $\alpha > 1$) as input at the content store at i^{th} (level) hop and defined $S_i(0,t)$ the number of different chunks requested in the open interval $(0,t]$ at the i^{th} node, as $K \to \infty$, $t \to \infty$, it holds*

$$1/g(i) \equiv \lim_{t\to\infty} \frac{\mathbb{E}[S_i(0,t)]^\alpha}{t} = \frac{\lambda(i)}{\mu(i-1)} \lambda c \sigma^\alpha m^{\alpha-1} \Gamma\left(1 - \frac{1}{\alpha}\right)^\alpha = \frac{\lambda(i)}{\mu(i-1)} 1/g. \tag{15}$$

Proof. Let us consider $i = 2$ and proceed as for the proof of Lemma 5.2 by constructing a lower and an upper bound for $\mathbb{E}[S_2(0,t)]$.
The input rate at the second content store, here denoted as $\lambda(2)$, is a function of the miss rate of the first cache and of the topology under study.

Under the assumption of a MMRP miss process with intensity equal to $\mu(1) = \lambda \sum_j p_j(1) q_j$, the miss rate for popularity class k is $\mu_k(1) = \mu \frac{p_k(1) q_k}{\sum_j p_j(1) q_j}$. The popularity distribution of content requests in the miss process results, therefore, to be modified since probability to have a request for a content of class k becomes: $q_2(k) \equiv \frac{\mu_k(1)}{\mu(1)} = \frac{p_k(1) q_k}{\sum_j p_j(1) q_j}, \forall k = 1, ..., N$. Thus, the input content request rate at the second hop is equal to $\lambda(2) = \mu(2) = \lambda \sum_j p_j(1) q_j$ for the topology in Fig.3(a) and to $\lambda(2) = 2\mu(1) = 2\lambda \sum_j p_j(1) q_j$ for the topology in Fig.3(b). For both topologies, one can write

$$\mathbb{E}[S_2(0,t)] = \sum_{k=1}^{\infty} \sum_{c_k=1}^{m} \mathbb{E}\left[\sum_{j(c_k)=1}^{N_{ch}} B^{j(c_k)}(0,t)\right] = \sum_{k=1}^{\infty} m\sigma \mathbb{E}[B_k^{(j)}(0,t)] = m\sigma \sum_{k=1}^{\infty} \frac{\lambda(2) q_2(k)}{\lambda q_k} \left(1 - e^{-\frac{\lambda q_k}{m} t}\right) \tag{16}$$

Now, if $\lambda(2) = \mu(1) = \lambda \sum_j p_j(1) q_j$,

$$\mathbb{E}[S_2(0,t)] = m\sigma \sum_{k=1}^{\infty} p_k(1) \left(1 - e^{-\frac{\lambda q_k}{m} t}\right) = m\sigma \sum_{k=1}^{\infty} e^{-\frac{\lambda q_k}{m} g x^\alpha} \left(1 - e^{-\frac{\lambda q_k}{m} t}\right)$$
$$\geq \left(\frac{\lambda c}{m} t\right)^{1/\alpha} \frac{m\sigma}{\alpha} \int_0^{\frac{\lambda c}{m} t} v^{-1/\alpha+1} (1 - e^{-v}) e^{-v \frac{g x^\alpha}{t}} dv \sim \Gamma\left(1 - \frac{1}{\alpha}\right) \left(\frac{\lambda c t}{m}\right)^{1/\alpha} m\sigma, \;\; t \to +\infty \tag{17}$$

Using the dominated convergence theorem, one can easily obtain the upper bound as for the proof in Lemma 5.2

$$\sup \mathbb{E}[S_2(0,t)] \sim m\sigma + \Gamma\left(1 - \frac{1}{\alpha}\right) m\sigma \left(\frac{\lambda c t}{m}\right)^{1/\alpha} \tag{18}$$

Similarly when $\lambda(2) = 2\mu(1) = 2\lambda \sum_j p_j(1) q_j$ (tree topology), one gets

$$\sup \mathbb{E}[S_2(0,t)] \sim m\sigma + \Gamma\left(1 - \frac{1}{\alpha}\right) m\sigma \left(\frac{2\lambda c t}{m}\right)^{1/\alpha}$$

Therefore, regardless of the topology, it results

$$1/g(2) \equiv \lim_{t\to\infty} \frac{\mathbb{E}[S_2(0,t)]^\alpha}{t} = \frac{\lambda(2)}{\mu(1)} \lambda c \sigma^\alpha m^{\alpha-1} \Gamma\left(1 - \frac{1}{\alpha}\right)^\alpha = \frac{\lambda(2)}{\mu(1)} 1/g \tag{19}$$

By iteration, it suffices to consider the input rate at node $i > 2$, $\lambda(i)$ and the modified popularity profile at node i, $q_i(k) = \prod_{j=1}^{i-1} p_k(j) q_k / \sum_{l=1}^{K} \prod_{j=1}^{l-1} p_l(j) q_l, \quad k = 1, ..., K$, to derive the general statement. □

Let us now state the main result on the miss probabilities at hop $i > 1$ characterization for topology (a) in Fig.3 in absence of aggregation.

Proposition 6.2. *Given a cascade of N caches as in Fig.3(a) and a MMRP content request process with rate $\lambda(i)$ and popularity distribution*

$$q_k(i) = \frac{\prod_{j=1}^{i-1} p_k(j) q_k}{\sum_{l=1}^{K} \prod_{j=1}^{l-1} p_l(j) q_l}, \quad k = 1, ..., K$$

(where $q_k = c/k^\alpha$, $\alpha > 1$) as input at the content store at i^{th} (level) hop, then $\forall\, 1 < i \leq N$ it holds

$$\log p_k(i) = \log p_k(1) \prod_{l=1}^{i-1} p_k(l) \tag{20}$$

Proof. Given a class k of content popularity, thanks to Prop.5.1

$$p_k(1) = e^{-\frac{\lambda}{m} q_k g x^\alpha}, \quad k = 1, ..., N$$

Under the MMRP assumption on the miss process and hence on the input process of the second (level) cache, one can compute the miss probability for class k at the second (level) by accounting for the modified rate and modified popularity in the miss rate. It results:

$$p_k(2) = e^{-\frac{\lambda(2)}{m} q_k g(2) x^\alpha \frac{p_k(1)}{\sum_{j=1}^K q_j p_j(1)}} = e^{-\frac{\lambda}{m} \sum_j p_j(1) q_j q_k g(2) x^\alpha \frac{p_k(1)}{\sum_{j=1}^K q_j p_j(1)}} = p_k(1)^{\frac{g(2)}{g} p_k(1)} = p_k(1)^{p_k(1)}. \tag{21}$$

where we used $g(i) = g$, $i \geq 1$, as it is the case for the considered topology (Fig.3(a)). By iteration, one gets the expression of the miss probability at node i, $i > 1$, which depends on previous nodes' miss probabilities in the following manner:

$$p_k(i) = p_k(1)^{\frac{g(i)}{g(1)} \prod_{l=1}^{i-1} p_k(l)} = p_k(1)^{\prod_{l=1}^{i-1} p_k(l)} \tag{22}$$

or alternatively, $\log p_k(i) = \log p_k(1) \left(\prod_{l=1}^{i-1} p_k(l) \right)$. □

It is worth to observe how $p_k(i)$, for a general popularity class k at the i^{th} node depends on $p_k(1)$. From Th.6.2 it results that $p_k(i)$ can be expressed as a function of $p_k(1)$ as it follows:

$$p_k(i) = p_k(1)^{\sum_{j=1}^{i-1} p_k(1)^{*j}} \tag{23}$$

where p^{*j} denotes the operation p^p iterated j times. Similarly, one can study the *binary tree* topology in fig.3(b). where the only difference with respect to the case of a cascade is the input rate at upper level caches, $\lambda(i) = 2\mu(i)$.

Corollary 6.3. *Given a binary tree with $2^N - 1$ caches (that is with N levels) as in Fig.3(b) and a MMRP content request process with rate $\lambda(i)$ and popularity distribution,*

$$q_i(k) = \frac{\prod_{j=1}^{i-1} p_k(j) q_k}{\sum_{l=1}^K \prod_{j=1}^{l-1} p_k(j) q_l}, \quad k = 1, ..., K$$

(where $q_k = c/k^\alpha$, $\alpha > 1$) as input at the content store at i^{th} (level) hop, then $\forall\ 1 < i \leq N$ it holds $\log p_k(i) = \prod_{l=1}^{i-1} p_k(l) \log p_k(1)$.

Proof. Let us consider firs the case $i = 2$. Note that $g(i)/g = \frac{\mu(i)}{\lambda(i)} = 1/2$ in the binary tree topology (Fig.3(b)). So we have

$$p_k(2) = e^{-\frac{\lambda(2)}{m} q_k g(2) x^\alpha \frac{p_k(1)}{\sum_{j=1}^K q_j p_j(1)}} = e^{-\frac{2\lambda}{m} \sum_j p_j(1) q_j q_k g(2) x^\alpha \frac{p_k(1)}{\sum_{j=1}^K q_j p_j(1)}} = p_k(1)^{2\frac{g(2)}{g} p_k(1)} = p_k(1)^{p_k(1)}. \tag{24}$$

The iteration procedure for any $i > 1$, follows the same calculations made for the linear topology. □

Let us now move to the case with request aggregation.

6.2 Miss rate characterization with request aggregation

For the topology in Fig. 3(a) the request aggregation takes place at first cache only. In fact, as observed in 5.1 the effective timescale of aggregation for requests of class k at node i in the linear topology is $\Delta_k(i) = \min(\Delta, \text{RVRTT}_k(i))$, where $\text{RVRTT}_k(i)$ is defined in (29). This implies that once requests are aggregated at the first cache, in a topology with no exogenous request arrival after the first hop, the request process is not filtered anymore. In the rest of the paper for the ease of notation we will omit the i in $\Delta_k(i)$.

Proposition 6.4. *Given a cascade of N caches as in Fig.3(a), a MMRP content request arrival process as described in Sec.6.1 and a timescale aggregation for content request Δ, then it holds*

$$p_k^f(i) = p_k^f(1)^{\prod_{l=1}^{i-1} p_k(l)} = p_k(i)^{1-p_{filt,k}(1)}. \tag{25}$$

Proof. Given the aggregation at the first hop only, we have $p_k^f(1) = p_k(1)$ since the filtering acts on the miss rate, whereas the miss rate filtered $\mu_k^f(1) = \mu_k(1)(1 - p_{filt,k}(1))$. The miss probability at the second node results to be modified according the new popularity defined in $\mu_k^f(1)$. As in (24), one can derive

$$\begin{aligned} p_k^f(2) &= e^{-\frac{\mu^f}{m} \sigma_k q_k g(2) x^\alpha \frac{p_k(1)(1-p_{filt,k}(1))}{\sum_{j=1}^K q_j p_j(1)(1-p_{filt,j}(1))}} = e^{-\frac{\lambda}{m} \sigma_k q_k g(2) x^\alpha p_k(1)} \\ &= p_k(1)^{\frac{g(2)}{g} p_k(1)(1-p_{filt,k}(1))} = p_k(2)^{(1-p_{filt,k}(1))}. \end{aligned} \tag{26}$$

By iterating eq.(26) one obtains the statement (25). □

For the topology in Fig.3(b) the request aggregation can take place at several hops depending on traffic and cache parameters.

Proposition 6.5. *Given a binary tree with $2^N - 1$ caches (that is with N levels) as in Fig.3(b), a MMRP content requests arrival process as described in Sec.6.1 and a timescale aggregation for content request Δ, then it holds*

$$p_k^f(i) = p_k^f(1)^{\prod_{l=1}^{i-1} p_k^f(l)(1-p_{filt,k}(l))}, \tag{27}$$

where $\forall i > 1$

$$p_{filt,k}(i) = 1 - \frac{b_k(i)(1 - b_k(i)^{\sigma_k})}{(1 - b_k(i))\sigma_k}, \quad k = 1, ..., N,$$

$$b_k(1) = e^{-\Delta_k \lambda_k / m}, \quad b_k(i) = e^{-\Delta_k 2\mu_k^f(i-1)/m}, \quad i > 1,$$

$$\mu_k^f(1) = \lambda_k p_k(1)(1 - p_{filt,k}(1)),$$
$$\mu_k^f(i) = 2\mu_k^f(i-1)p_k^f(i)(1 - p_{filt,k}(i)), \quad i > 1. \tag{28}$$

Proof. The proof is a simple extension of Th.6.4 proof. □

6.3 Throughput characterization

Recalling the definition of the VRTT (eq.(1)) where R_i denotes the value of twice the one-way delay among node 1 and node i (similarly among level 1 and level i nodes in the binary tree topology), one has

$$\text{VRTT}_k = \sum_{i=1}^{N} R_i(1 - p_k(i)) \prod_{j=1}^{i-1} p_k(j)$$

The residual round trip time at node $i > 1$ represents the virtual round trip that one would have if node (level) i would be the first node (level), hence if follows from (1),

$$\text{RVRTT}_k(i) = \sum_{j=i}^{N} (R_j - R_{i-1})(1 - p_k(j)) \prod_{l=i}^{j-1} p_k(l). \tag{29}$$

To define VRTT_k^f, $\text{RVRTT}_k^f(i)$ in the case with request aggregation (filtering), it suffices to replace the miss probabilities in absence of filtering with those defined in Theorems 6.4/6.5. From the VRTT formula above, one can directly derive the formula of the stationary average throughput for contents of class k, X_k, in absence of congestion,

$$X_k = \frac{W}{\text{VRTT}_k} \tag{30}$$

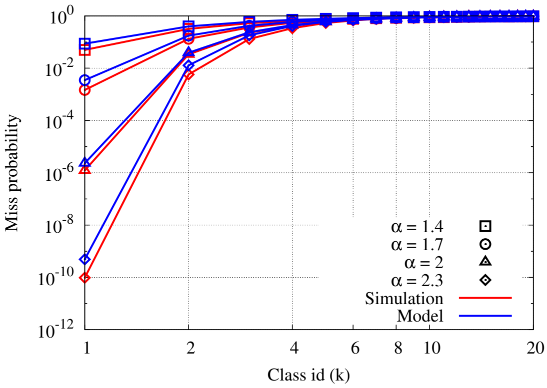
where W is the chunk transmission window, i.e. W chunk requests are issued in parallel. One can easily infer then the average content delivery time as a function of the content size and of the average throughput. Indeed, for a content of class k, this results to be $T = (\sigma/W)/X$. Eq.(30) is a powerful tool for cache sizing when some throughput or content delivery time guarantees have to be respected. In sec.8 we give hints on cache size / link bandwidth dimensioning, once explained the underlying performance/resources trade-off.

7 Numerical results

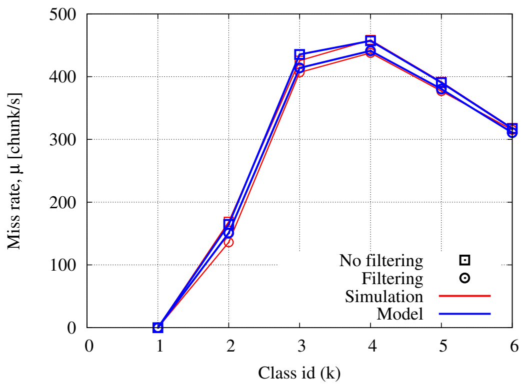
In this section we present simulative results to corroborate the above theoretical results first in the case of a single content store (sec.7.1), then in the case of a network of content stores as in CCN (sec.7.2).

7.1 Single cache

This section gathers numerical results obtained by means of chunk-level simulations that corroborate the analytical formulae derived in Sec.5. To this purpose we developed an ad-hoc C++ event-driven simulator, implementing data caching and forwarding as well as the receiver driven transport protocol. For the present paper we assume a simple transport protocol using a fixed window size for chunk requests as in Parc's CCN. Nodes' forwarding tables are computed according to the publish & subscribe routing protocol implemented by Carzaniga et al. in content-based networking simulator [22,23]. We consider a population of $M = 20000$



(a)



(b)

Figure 4: Single-cache scenario. Miss probability as a function of class popularity (a); miss rate with and without request filtering for $\alpha = 2$ (b).

content items, organized in $K = 400$ classes of decreasing popularity, each one with $m = 50$ items. Content popularity is Zipf distributed, i.e. content items in class $k = 1, ..., K$ are requested with probability $q_k = c/k^\alpha$, $c > 0$, with $\alpha \in (1, 2.5)$. Clearly, a given content in class k is requested with probability q_k/m. We suppose content items are split in chunks of $10kB$ each, and their size is geometrically distributed with average 690 chunks (6.9MB). These are typical values for a UGC applications (see [24]). Users generate content requests according to a Poisson process of intensity $\lambda = 40$ content/sec, and the chunk transmission window size is $W = 1$. We suppose a cache of size $x = 200000$ chunks (2GBs) which implements the LRU replacement policy. The simulated time is almost 7 hours.

In fig. 4(a) we show the miss probability as a function of the popularity distribution for different values of the Zipf parameter α in absence of request aggregation. Fig. 4(b) reports the miss rate for $\alpha = 2$ with and without request filtering. Results are reported for the most popular classes for model and simulations, so confirming model accuracy in predicting miss probability/rate (eq.2). The major discrepancy between model and simulations can be observed when the miss probability is very small, that is on the most popular classes for very skewed popularity. In such cases, cache misses are very rare events which are difficult to observe over a limited simulation time.

We notice that the miss probability is affected by content popularity for the 8 most popular classes; classes $k > 8$ count few requests for the considered α values and therefore result in miss probabilities close to 1. As expected, for these most popular classes the miss probability increases as α decreases; a smaller α parameter leads to a flatter popularity, which means an increasing number of different chunks flowing trough the cache and eventually an higher miss probability. The miss rate decreases when requests are aggregated, with a reduction of about 5% for classes 2 to 4.

In Fig. 5 we illustrate the impact of the cache size on the miss probability for $\alpha = 2$. Model accuracy can be appreciated for all considered size values. As expected, miss probability decreases as cache size increases. In fact, for a given content size, the miss probability depends onthe ratio cache size over content size.

7.2 Network of caches

Let us now analyze the network case by means of chunk-level simulations in order to assess model accuracy. Consider the topology reported in Fig.3(b), a $N = 3$ levels *binary tree*, where data is stored at the root of the tree while leaf nodes are the entry points of user content requests. All links have the same capacity of $10Gbps$ and the same round trip delay equal to $2ms$. Every node is equipped with a cache of size $x = 200000$ chunks (2GB) implementing LRU replacement policy. This topology is meant to represent a typical aggregation network collecting requests coming from different DSLAMs. The root of the tree serves as gateway for content retrieved from the rest of the Internet.

The aggregate content request process at every leaf node is Poisson distributed with intensity $\lambda = 40$ content/sec. Unless otherwise specified, we set the chunk transmission window size $W = 1$ and the simulated time to 7 hours. Content population characteristics are the same as in the single cache scenario (Sec. 7.1) with Zipf parameter $\alpha = 2$, and chunk size is $10kB$. Fig.6 compares the miss probabilities at different nodes (from the 1^{st} to the 3^{rd} level) without request aggregation. The comparison outlines the

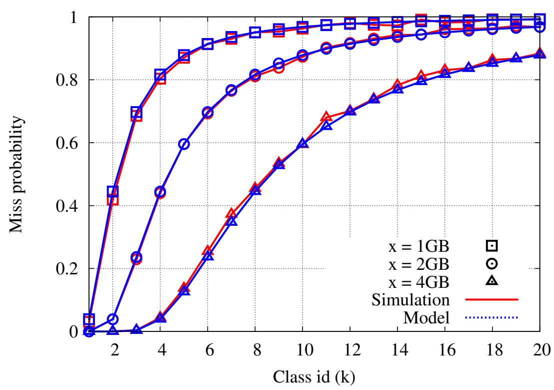


Figure 5: Single-cache scenario. Miss rate for different cache sizes.

good match between model predictions Eq.(2) at level $i = 1$, Eq.(20)at level $i > 1$ and simulations. From Fig.6 it can be also observed how content popularity changes along the path. Requests for content items of the most popular class are almost completely served by caches of first nodes. As a consequence, the miss probability for class $k = 1$ is nearly 1 at upper levels. Content items of class 2 are mainly cached at first and second level, whereas less popular classes, represented in the queue of the curves, are very rarely cached as hardly requested.

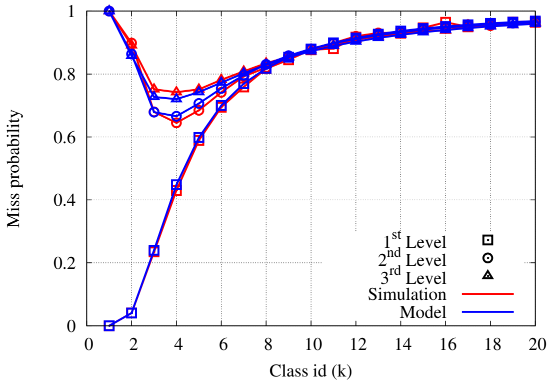
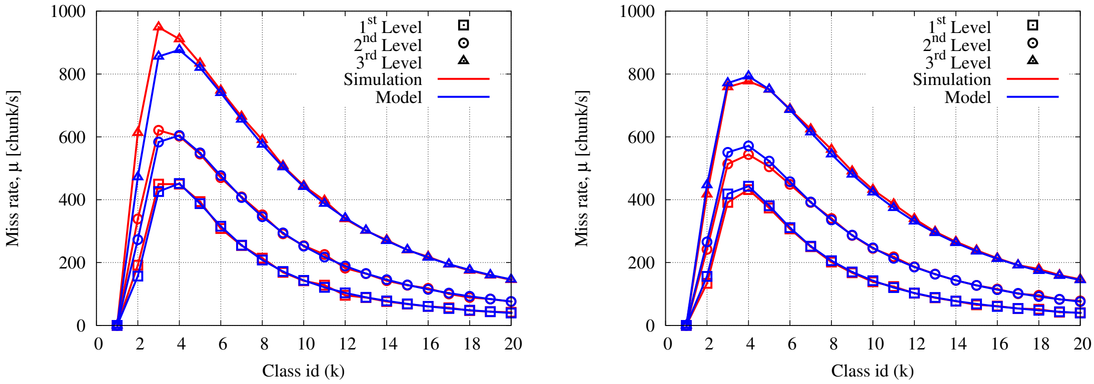


Figure 6: Miss probability at different levels for topology Fig.3-b, with no filtering.



(a) (b)

Figure 7: Miss rate with filtering disabled (a) and enabled (b), for the binary tree topology in Fig.3(b).

Fig. 7 reports the miss rate as a function of content popularity with and without request aggregation. Besides the good match between analytical and simulation values, it is important to remark that the miss rate reduction, already observed in Sec. 7.1 for the single cache scenario, is more significant at higher levels of the network, with a maximum decrease of about 20% at level $i = 3$.

In Fig.8 we show the virtual round trip time, VRTT_k, as a function of content popularity with and without request aggregation. The virtual round trip time measures the average distance between the user and the content as a function of the miss probabilities along the path. Therefore, it represents a suitable metric to evaluate data transfer performance, as it quantifies average chunk delivery time.

As previously noticed, chunks of the most popular content items are cached at first level nodes, so that $\text{VRTT}_1 \approx$ 2ms, whereas the rarest items are not cached within the network, with a consequent round trip time of about 8ms (4 hops). The figure also highlights that the content aggregation has no or little impact on the stationary VRTT, even if it helps in strongly reducing the chunk request traffic as also showed in Fig. 7.

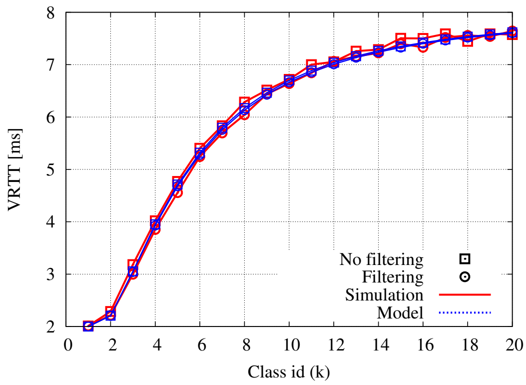


Figure 8: VRTT experienced by end-users in topology Fig.3(b).

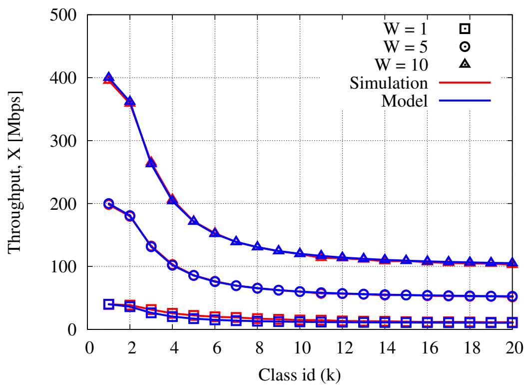
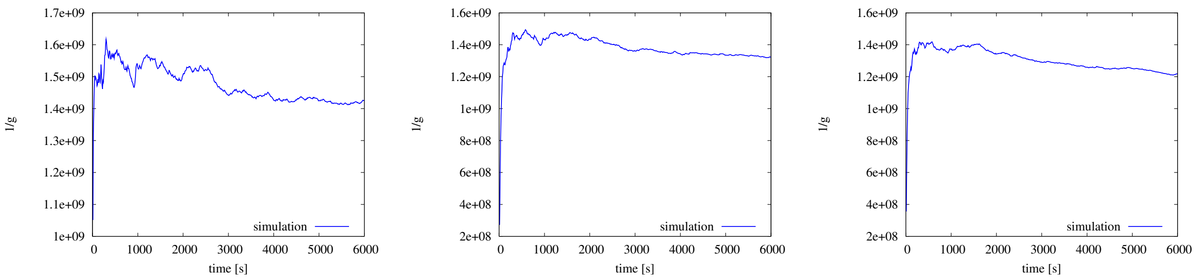


Figure 9: User's throughput expressed in chunk/s in topology Fig.3(b) varying the window size of the transport protocol.

The mean stationary throughput is reported in Fig.9 as a function of content popularity for several values of the chunk transmission window size W, in a scenario where requests are not aggregated. As expected from Eq. (30), the throughput decreases as k increases, because of the VRTT increase. Similarly, with larger windows sizes, higher throughput can be achieved as multiple chunks can be retrieved in parallel. The throughput gain due to parallel downloading is more important for most popular classes due to a smaller VRTT.

7.3 Convergence time

In this section we consider the convergence rate to the steady state of the simulations runs On of the most interesting metric to consider is $S(0,t)^{\alpha}/t$ that indicates the number of different chunks per seconds requested in the open interval $(0,t)$. In Fig.10 we report $S(0,t)^{\alpha}/t$ as a function of time during a simulation run. This metric is reported for each level of the binary tree.



(a) (b) (c)

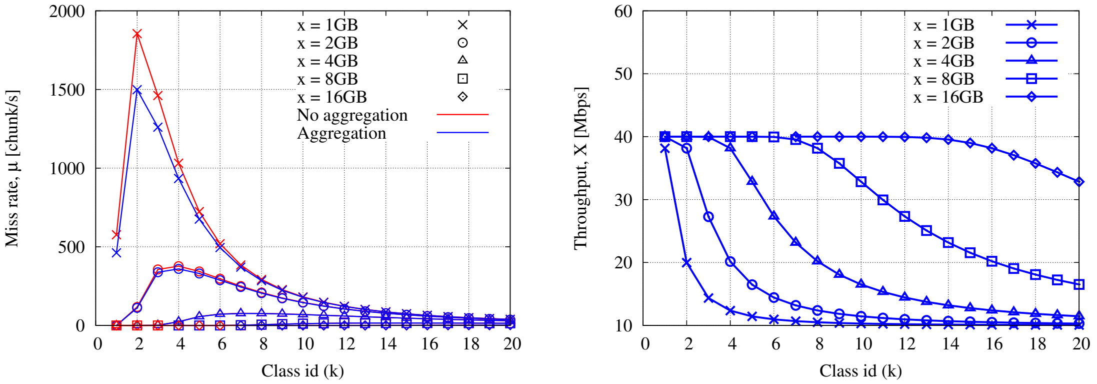
Figure 10: Binary tree scenario. Convergence evolution of $\mathbb{E}[S_1(0,t)]^{\alpha}/t$ at the first level nodes (a); $\mathbb{E}[S_2(0,t)]^{\alpha}/t$ at the second level does (b); $\mathbb{E}[S_3(0,t)]^{\alpha}/t$ at the root node (c).

From the graphs we notice that $\mathbb{E}[S_1(0,t)]^{\alpha}/t$, reported in Fig.10, has the same behavior for all three tree levels: it has a fast increase in the first few seconds and then attains a stationary regime that lasts about 2000 seconds: the most popular contents are requested during this phase with little contributions from non popular classes. At a certain point more or less the same contents are periodically requested with little refresh because the number of different contents is finite in a simulation run. Such limitation can be observed in a second regime that $S(0,t)^{\alpha}/t$ enters when $S(0,t)$ does not increase anymore as new contents are hardly requested. During the decreasing regime there are two factor to observe : from one side popular contents have all been requested while non popular contents are requested with low probability. $S(0,t)^{\alpha}/t$ has an overall decreasing trend because $S(0,t)$ has no contribution from popular content as they all have been requested at the end of the stationary phase; so $S(0,t)^{\alpha}/t$ decreases. However much less popular content may be requested and this is detectable when $S(0,t)^{\alpha}/t$ has a relatively small increase.

8 Dimensioning and Performance Trade-offs

Bandwidth and storage capacity are the most critical resources in a content-centric architecture, and represent the network cost for an operator to deploy this kind of distribution infrastructures. On the other hand, it is of significant interest to establish a direct link between resources and performance. The model can be used as a tool to *quantify network costs* for a provider, to guarantee certain performance to users or, inversely, to *predict performance* users perceive for a given amount of resource devoted to a specific service.

As an example, consider the binary tree topology in Fig.3(b), representative of a typical aggregation



(a) (b)

Figure 11: Total miss rate (a) and mean stationary throughput (b) as a function of the cache size in the binary tree scenario.

network, where the root of the tree can serve as data termination point for content retrieved from outside the network. Suppose all caches have the same storage capacity and consider users of a UGC application (e.g. YouTube, Daily Motion), where content popularity is Zipf distributed with $\alpha = 2.3$ and the average content size is 6.9MB (See [24]). In Fig.11 we report the traffic at the gateway (a) (total miss rate flowing upstream in the network) and the users' throughput (b) as a function of content popularity for cache size ranging from 1GB to 16GB. Fig.11(a) shows the *throughput-storage capacity* trade-off: given a target throughput, the minimum storage capacity needed to provide minimum performance can be evaluated or inversely, given a the storage size, the maximum expected miss rate or minimum expected throughput. In Fig.11(b) we quantify, through the model, the trade-off *performance-network cost*: we accurately predict how much users' throughput can be increased with additional network resources. Moreover, it is possible to deduce some system properties which can serve as design guidelines: e.g. (i) request aggregation has a significant impact on the bandwidth required at the gateway only for $1GB$ caches; (ii) the maximum throughput is limited to $40Mbps$ even for very large caches as $W = 1$.

9 Conclusions

Content-centric network proposals bring fresh thinking into Internet architecture design by re-centering communication principles around current usage, mostly content dissemination and retrieval. However, a comprehensive study of content-centric networks still lacks and key features of such systems are poorly understood.

Unlike traditional end-to-end transport, in CCN data transfer is realized through a receiver-centric paradigm, whose performance is closely tied to network cache dynamics. Indeed, storage capabilities are embedded at network level and every node acts as a cache for flowing data.

In this paper we developed an analytical model for the performance evaluation of content transfer in CCN that allows an explicit characterization of steady state dynamics. A closed form expression for the stationary throughput, and hence for the content delivery time, is provided, showing the dependence from key system parameters such as content popularity, content size and cache size.

Differently from previous work, our model allows to capture chunk-level dynamics and thus to account for the correlation in the content request process either temporal, due to the receiver-driven transport protocol, and spatial, as an effect of content request aggregation.

Thanks to these features, the analytical framework proposed in this paper constitutes an essential building block for the design of a receiver-driven transport control protocol aimed at avoiding network congestion while realizing specific fairness criteria. Moreover, the model is suitable to account for specific forwarding techniques employing multiple-paths to route content requests. Other possible future research directions are the study of system time evolution, useful for analyzing flash crowd scenarios, and of different cache replacement policies.

10 Acknowledgement

This work has been partially funded by the French national research agency (ANR), CONNECT project, under grant number ANR-10-VERS-001, and by the European FP7 IP project SAIL under grant number 257448. We thank Bruno Kauffmann for his useful comments.

References

[1] "The global environment for network innovations," (GENI) http://www.geni.net.

[2] "European future internet assembly," (FIA) http://www.future-internet.eu.

[3] V. Jacobson, D. Smetters, J. Thornton, M. Plass, N. Briggs, and R. Braynard, "Networking named content," in *Proc. of CoNEXT '09*.

[4] M. Ain and al., "Architecture definition, component descriptions, and requirements," in *PSIRP project deliverable D2.3*, 2009.

[5] T. Koponen, M. Chawla, B. Chun, A. Ermolinskiy, K. Kim, S. Shenker, and I. Stoica, "A data-oriented (and beyond) network architecture," in *Proc. of ACM SIGCOMM*, 2007.

[6] K. Park and V. S. Pai, "Scale and performance in the coblitz large-file distribution service," in *Proc. of NSDI'06*.

[7] Z. Al-Qudah, S. Lee, M. Rabinovich, O. Spatscheck, and J. Van der Merwe, “Anycast-aware transport for content delivery networks,” in *in Proc. of WWW ’09*.

[8] S. Arianfar, P. Nikander, and J. Ott, “On content-centric router design and implications,” in *Proc. of ACM ReArch*, 2010.

[9] U. Lee, I. Rimac, and V. Hilt, “Greening the internet with content-centric networking,” in *Proc. of ACM E-Energy*, 2010.

[10] G. Carofiglio, V. Gehlen, and D. Perino, “Experimental evaluation of storage management in conten-centric networking,” in *Proc. of IEEE ICC*, 2011.

[11] W. King, “Analysis of paging algorithms,” in *Proc. of IFIP*, 1971.

[12] P. Flajolet, D. Gardy, and L. Thimonier, “Birthday paradox, coupon collectors, caching algorithms and self-organizing search,” *Discrete Appl. Math.*, vol. 39, no. 3, pp. 207–229, 1992.

[13] D. Starobinski and D. Tse, “Probabilistic methods for web caching,” *Perform. Eval.*, vol. 46, no. 2-3, pp. 125–137, 2001.

[14] E. Coffman and P. Jelenković, “Performance of the move-to-front algorithm with markov-modulated request sequences,” *Oper. Res. Lett.*, pp. 109–118, 1999.

[15] P. R. Jelenković, “Asymptotic approximation of the move-to-front search cost distribution and least-recently-used caching fault probabilities,” *The Annals of Applied Probability*, vol. 9, no. 2, pp. 430–464, 1999.

[16] P. R. Jelenković and X. Kang, “Characterizing the miss sequence of the lru cache,” *ACM SIGMETRICS*, vol. 36, no. 2, pp. 119–121, 2008.

[17] 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, 2002.

[18] A. Dan and D. Towsley, “An approximate analysis of the lru and fifo buffer replacement schemes,” *ACM SIGMETRICS*, 1990.

[19] E. J. Rosensweig, J. Kurose, and D. Towsley, “Approximate models for general cache networks,” in *Proc. of IEEE INFOCOM ’10*.

[20] T. E. Stern and A. I. Elwalid, “Analysis of separable markov-modulated rate models for information-handling systems,” *Advances in Applied Probability*, vol. 23, no. 1, pp. 105–139, 1991.

[21] E. Chlebus and J. Brazier, “Nonstationary poisson modeling of web browsing session arrivals,” *Information Processing Letters*, vol. 102, no. 5, pp. 187 – 190, 2007.

[22] A. Carzaniga, M. Rutherford, and A. Wolf, “A routing scheme for content-based networking,” in *Prof. of IEEE INFOCOM 2004*.

[23] A. Carzaniga, “Content-based networking,” 2003, http://www.inf.usi.ch/carzaniga/cbn/.

[24] 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 *Proc. ACM SIGCOMM IMC*, 2007.