Modeling Data Transfer in Content-Centric Networking

Giovanna Carofiglio*, Massimo Gallo†, Luca Muscariello† and Diego Perino*.
* Bell Labs, Alcatel-Lucent, France, † Orange Labs, France Telecom, France
first.last@alcatel-lucent.com, first.last@orange-ftgroup.com

***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 ori ented 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 content/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.**

I. 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 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 projects have been funded in the last years focusing on the definition of novel architectures for the future Internet (e.g. US NSF GENI or EU FIA). In this research arena, *content-centric* proposals, as Parc's CCN [1], PSIRP (now in PURSUIT [1]), or DONA [2], 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[2] uniquely identified and independently 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 system performance.

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 requests, needs to be precisely modeled in order to define the modified request process feeding upstream caches.

Transport performance is affected by caching dynamics and a unified modeling framework is necessary to evaluate data transfer 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 [3] or Anycast CDN [4], 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 a 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. II provides system description. Sec.III is devoted to related work, while Sec. IV introduces notation and main modeling

[1] www.fp7-pursuit.eu

[2] Chunks are intended to be packet-size entities in content centric networks.

assumptions. In Sec. V we report the main analytical results about single cache dynamics, while in Sec. VI we extend the model to the network case and provide explicit formulae for stationary throughput. Sec. VII presents an example of model application while Sec. VIII concludes the paper.

II. System description

In this paper we primarily focus on the content-centric networking (CCN) proposal by Parc [1], 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 as *virtual round trip time* (VRTT) in analogy with connection-based transport protocols like TCP.

III. Related Work

Previous work on content-centric networks has mainly focused on global architecture design [1], [2] while less effort has been devoted to analyze caching and transport mechanisms in such architectures. More recently, Somaya *et al.* [5] analyze the feasibility of caching in routers at line-speed, while Lee *et al.* [6] 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 [7]. However, none of the aforementioned work 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 model 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 the tail of the search cost distribution for the Move-To-Front (MTF) searching algorithm [8]. In [9] 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 content items. Alternative combinatorial approaches are developed in [10]-[11]. In [12], authors give an asymptotic characterization, for a large number of content items, 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 [13] 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. 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.

IV. Model Description

A. Assumptions and notation

Using the notation described in table I, in our setting, we make the following assumptions: (i) 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 $q_k,\ k \geq 1$. The notion of popularity class allows to group all equivalently popular content items accounted for by the popularity histograms inferred by network measurements. In the rest of the paper, we assume a Zipf popularity distribution, hence $q_k = c/k^\alpha,\ k \geq 1$ with parameter $\alpha > 1$. (ii) Content items are segmented into chunks and have different sizes: σ denotes the average content size in terms of number of chunks (chunks are fix sized). (iii) Each node in the network has a cache (also referred to as content store in Parc's CCN) of size x chunks. (iv) We focus on different topologies (Fig.1), meant to represent (segments of) the aggregation network which gathers a large number of users' requests over time. (v) 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.

$$\mathrm{VRTT}_k = \sum_{i=1}^{N} R_i(1-p_k(i)) \prod_{j=1}^{i-1} p_k(j),\ \ k = 1, \dots, K \quad (1)$$

$VRTT_k$ 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. Similarly, we define the residual virtual round trip time at node (level) i as

$$\mathrm{RVRTT}_k(i) = \sum_{j=i}^{N} (R_j - R_{i-1})(1-p_k(j)) \prod_{l=i}^{j-1} p_k(l). \quad (2)$$

$\mathrm{RVRTT}_k(i)$ at node $i > 1$ represents the virtual round trip that one would have if node (level) i would be the first

TABLE I

N	Number of network nodes
K	Number of different classes
M	Number of different content items ($m = M/K$)
x	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
$q_k, q_k(i)$	Popularity distribution for class k at level 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	Imposed and effective filtering time window for class k
X_k	Chunk delivery rate or throughput of class k

node (level). Clearly, $\text{VRTT}_k = \text{RVRTT}_k(1)$. More details are provided in Sec.VI.

B. Content request process

The content request process is structured in two levels, *content* and *chunk*, which are important to characterize. In this section we build a fluid model of such two-level request process capturing the first-order system dynamics in steady state.

The request arrival process is modeled through a Markov Modulated Rate Process (MMRP) [14]: requests for content items in class k are generated according to a Poisson process of intensity $\lambda_k = \lambda q_k$, and the content to be requested is uniformly chosen among the m different content items in class k. 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. Content size (in number of chunks), N_{ch}, is assumed to be geometrically distributed with mean σ, i.e. $\mathbb{P}(N_{ch} = s) = \frac{1}{\sigma}\left(1 - \frac{1}{\sigma}\right)^{s-1}, \quad s \geq 1$. However, chunk-level simulations outlined that average dynamics do not vary significantly in presence of heavy-tailed content size distribution with average σ chunks. The inter-arrival between the requests of two subsequent chunks of the same content is supposed to be deterministic[3] and equal to the average Virtual Round Trip Time of class k, VRTT_k. The superposition of different content requests defines the MMRP process, whose underlying Markov chain is represented in [15]. 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 [16]. *At chunk level* we suppose to look at system dynamics in steady state where the cache dynamics and hence the chunk level rate X_k, $k = 1, ..., K$, have converged to a stationary value.

[3] cfr.[15] for further details about this assumption.

The large number of content requests served by the aggregation network justifies the fluid assumption behind the MMRP [14]. It is worth to remark that we do not assume any other temporal correlation in the input process, but we model the temporal correlation induced by content request aggregation (filtering). Such a 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.V-A.

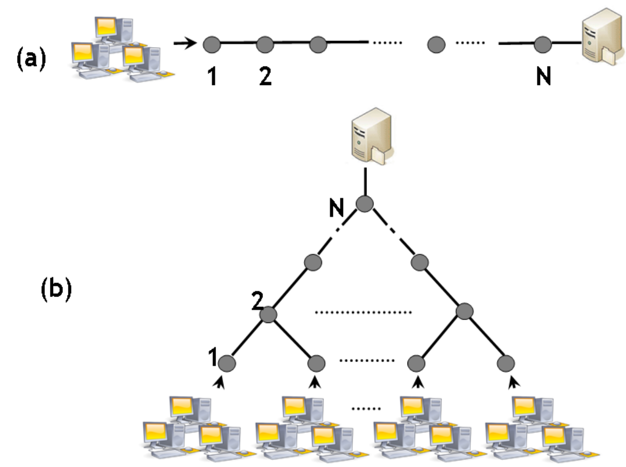

Fig. 1. Network topologies: cascade (a), binary tree (b).

V. 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 [13], authors give an analytical characterization of the miss probability and hence of the miss rate for the case of a Poisson content request process when the cardinality of the set of different content items M tends to infinity. In this section we extend their result to the case of:

(i) MMRP request process (content/chunk levels) with content request rate $\lambda_k = \lambda q_k$, where $q_k = c/k^\alpha$.

(ii) K popularity classes with the same number of content items $m = M/K$ in each one.

(iii) Non-unitary content size (σ).

(iv) Pending request aggregation (filtering) over a time interval Δ (sec.V-A).

Proposition 5.1: Given a MMRP request arrival process as described in Sec.IV-B 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

$$p_k \equiv p_k(1) \sim e^{-\frac{\lambda}{m} q_k g x^\alpha} \tag{3}$$

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

Due to lack of space we provide here the intuition behind the proof that the reader can find in [15]. A request for a given

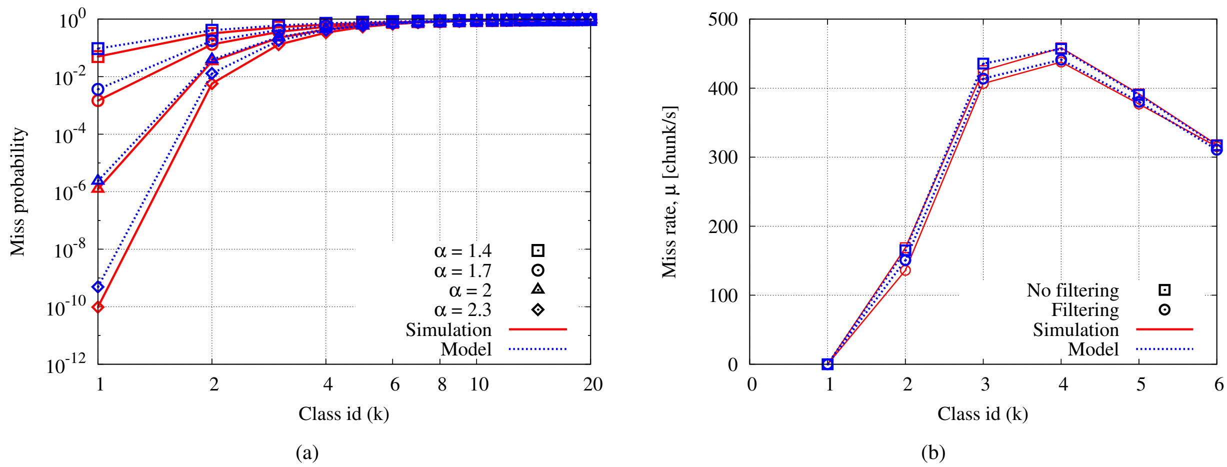

Fig. 2. Single-cache scenario. Miss probability as a function of class popularity (a); miss rate with and without request filtering for $\alpha = 2$ (b).

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 Poisson and the geometric distribution memory-less property, for a given chunk in class k the number of different chunk requested in the open interval (τ_{n-1}, τ_n) is independent from the number of different chunk requested in (τ_n, τ_{n+1}), where $\{\tau_1, \tau_2, ...\}$ is the sequence of points at which a miss for that specific chunk occurs. The miss sequence of a specific chunk of a content in class k determines the miss sequence for all chunks of that content. Finally, the global miss process is given by the superposition of the miss sequences of different content items in all popularity classes.

A. Content request aggregation

A fundamental feature of CCN to avoid request flooding is the aggregation, which is a mechanism taking trace of pending chunk requests at each node and preventing the dispatch of new requests for the same data. This mechanism should be considered when studying the interaction between transport and caching in content oriented 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 5.2: In steady state, the effective aggregation timescale for chunk requests of content of class k is $\Delta_k(i) = \min(\Delta, \text{RVRTT}_k(i))$, where $\text{RVRTT}_k(i)$ is defined in Eq.(2).

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 pending request is satisfied and the data is received. From that time on, the chunk is stored in cache, and a new chunk request has to be forwarded only if data has been removed by the replacement policy.

The request aggregation clearly impacts the miss rate of a given cache. In fact, when a chunk request of class k 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 has been emitted and the chunk has not been received yet (i.e. 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.1(b).

Proposition 5.3: Given the request process defined in sec.IV-B, the filtering probability associated to class k at the first hop is,

$$p_{filt,k}(1) = \frac{1 - b_k}{1 - (1 - 1/\sigma)b_k} \quad k = 1, ..., K \tag{4}$$

with $b_k = e^{-\Delta_k \lambda_k}$, and $\Delta_k = \min(\Delta, \text{VRTT}_k)$.
The proof is reported in [15].

B. Numerical results

This section gathers a set of numerical results obtained by means of chunk-level simulations that corroborate the analytical formulae derived in Sec.V.

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. 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 [17][4].

We consider a population of $M = 20000$ 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).

[4]Content-Based Networking www.inf.usi.ch/carzaniga/cbn

Users generate content requests according to a Poisson process of intensity $\lambda = 40$ content/sec, and the interest transmission window size is $W = 1$. We suppose a cache of size $x = 200000$ chunks (2GBs) which implements the LRU replacement policy. Notice that caches are assumed to be initially empty, while statistics are collected in steady state only.

In fig. 2(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. 2(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.3). 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

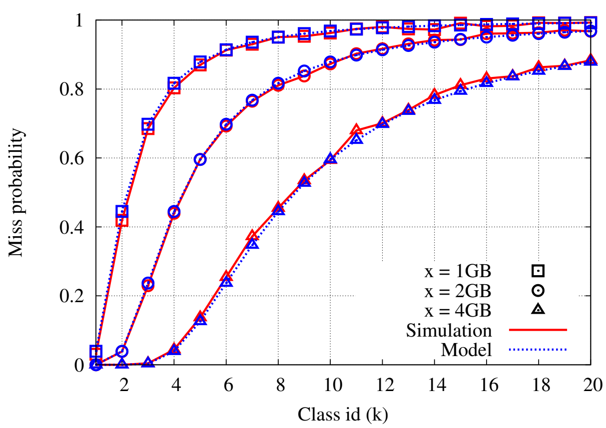

Fig. 3. Single cache scenario. Miss probability for different cache sizes.

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. 3 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 on the ratio cache size over content size.

VI. Network of caches

In this section we consider the topologies reported Fig.1, and we extend the results presented in Sec.IV to derive analytical expressions for the miss probabilities/rates at every node (level) of these networks (Sec.VI-A). Furthermore we derive a closed-form expression of the stationary throughput (Sec.VI-C) with and without request aggregation.

A. Miss rate characterization

In order to derive the stationary miss probabilities $p_k(i)$ at hop $i > 1$, we need the following auxiliary result.

Lemma 6.1: Given a MMRP request arrival process with rate $\lambda(i)$ and popularity distribution $q_k(i) = \prod_{j=1}^{i-1} p_k(j) q_k / \sum_{l=1}^{K} \prod_{j=1}^{l-1} p_l(j) q_l$, $k = 1, ..., K$, (where $q_k(1) \equiv q_k = c/k^\alpha$, $\alpha > 1$) as input at the cache 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 $K \to \infty$, $t \to \infty$. Be $\mu(i)$ the miss rate at node i ($\mu(i) = \lambda(i) \sum_k p_k(i) q_k(i)$), then it holds

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

i.e. $g(i)/g \equiv \mu(i-1)/\lambda(i)$, being $g(1) \equiv g$. The proof is reported in [15]. Let us now state the main result on the miss probabilities at hop $i > 1$ for topology (a) in Fig.1 in absence of aggregation.

Proposition 6.2: Given a cascade of N caches as in Fig.1(a) and a MMRP request arrival process as described in Sec.IV-B, then $\forall\ 1 < i \le N$ it holds

$$\log p_k(i) = \frac{g(i)}{g(1)} \prod_{l=1}^{i-1} p_k(l) \log p_k(1) = \prod_{l=1}^{i-1} p_k(l) \log p_k(1) \tag{5}$$

Similarly, one can study the *binary tree* topology in Fig.1(b), that has no difference with the line topology in the homogeneous case expect that $\lambda(i) = 2\mu(i-1)$.

Corollary 6.3: Given a homogeneous binary tree with $2^N - 1$ caches (that is with N levels) as in Fig.1(b) and a MMRP requests arrival process as described in Sec.IV-B, then $\forall\ 1 < i \le N$ it holds

$$\log p_k(i) = \frac{\lambda(i) g(i)}{\mu(i-1) g(1)} \prod_{l=1}^{i-1} p_k(l) \log p_k(1) = \prod_{l=1}^{i-1} p_k(l) \log p_k(1)$$

B. Miss rate characterization with request aggregation

For the topology in Fig. 1(a) the request aggregation takes place at first cache only. In fact, as observed in V-A the effective timescale of aggregation for requests of class k at node i in the linear topology is $\Delta_k(i) = \min(\Delta, \text{RVRTT}_k(i))$. Recall that $\text{RVRTT}_k(i)$ is defined as

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

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.1(a), a MMRP request arrival process as described in Sec.IV-B and an aggregation timescale Δ, then it holds

$$p_k^f(i) = p_k^f(1)^{\prod_{l=1}^{i-1} p_k(l)} = p_k(i)^{1-p_{filt,k}(1)}. \qquad (6)$$

The proof is provided in [15]. For the topology in Fig.1(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.1(b), a MMRP request arrival process as described in Sec.IV-B and an aggregation timescale for content request Δ, then it holds $p_k^f(i) = p_k^f(1)^{\prod_{l=1}^{i-1} p_k^f(l)(1-p_{filt,k}(l))}$, $i > 1$,

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

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

$$\mu_k^f(i) = \begin{cases} \lambda_k p_k(1)(1 - p_{filt,k}(1)) & \text{if } i = 1 \\ 2\mu_k^f(i-1)p_k^f(i)(1 - p_{filt,k}(i)) & \text{if } i > 1 \end{cases}$$

The proof is a simple extension of Prop.6.4 (see [15]).

C. Throughput characterization

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

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 Propp. 6.4/6.5. From the VRTT formula above, one can directly derive the formula of the average stationary throughput for content items of class k, X_k, in absence of congestion,

$$X = \frac{W}{\text{VRTT}_k} \qquad (7)$$

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.(7) is a powerful tool for cache sizing when some throughput or content delivery time guarantees have to be respected. In sec.VII we give hints on cache size/link bandwidth dimensioning, once explained the underlying performance/resources trade-off.

D. Numerical results

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.1(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 $W = 1$, we assume caches are initially empty, while statistics are collected in steady state only.

Content population characteristics are the same as in the single cache scenario (Sec. V-B) with Zipf parameter $\alpha = 2$, and chunk size is $10kB$. Fig.4 compares the miss probabilities at different nodes (from the 1^{st} to the 3^{rd} level) without request aggregation. The comparison outlines the good match between model predictions Eq.(3) at level $i = 1$, Eq.(5) at level $i > 1$ and simulations. From Fig.4 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.

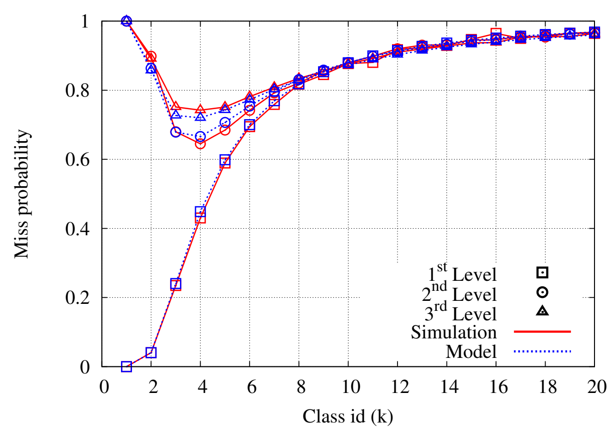

Fig. 4. Miss probability at different levels for topology Fig.1(b), with no filtering.

Fig. 5 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. V-B 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.6 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 $\text{VRTT}_1 \approx 2\text{ms}$,

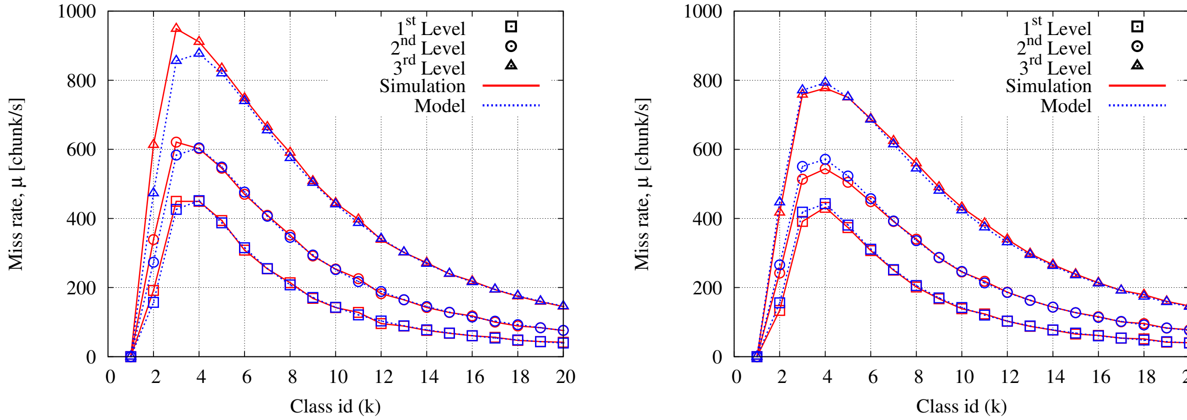

Fig. 5. Miss rate with filtering disabled (on the left) and enabled (on the right), for the binary tree topology in Fig.1(b).

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. 5. The mean stationary throughput is reported in

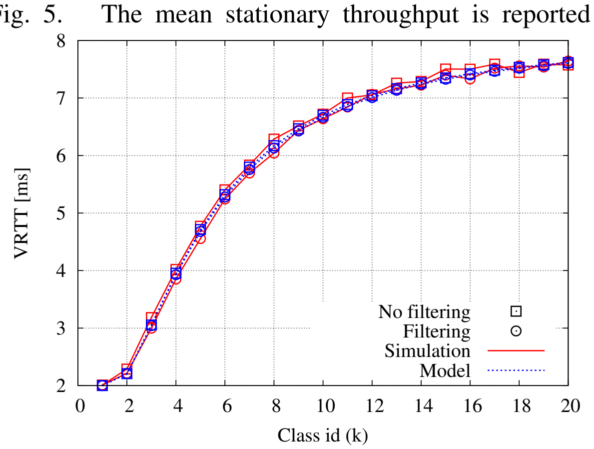

Fig. 6. VRTT experienced by end-users in topology Fig.1(b).

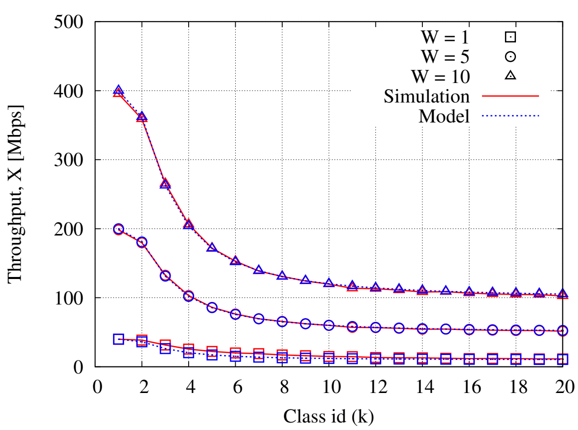

Fig. 7. User's throughput expressed in Mbps in topology Fig.1(b) varying the window size of the transport protocol.

Fig.7 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. (7), 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.

VII. 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.1(b), representative of a typical aggregation network, where the root of the tree can serve as data termination point for content retrieved from the rest of the Internet. Suppose all caches have the same storage capacity and consider users of a web media application, where content popularity is Zipf distributed with $\alpha = 1.6$ and the average content size is 14MB (See [18]). In Fig.8 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.8(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.8(b) we quantify, through the model, the *user performance-network cost* trade-off : 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

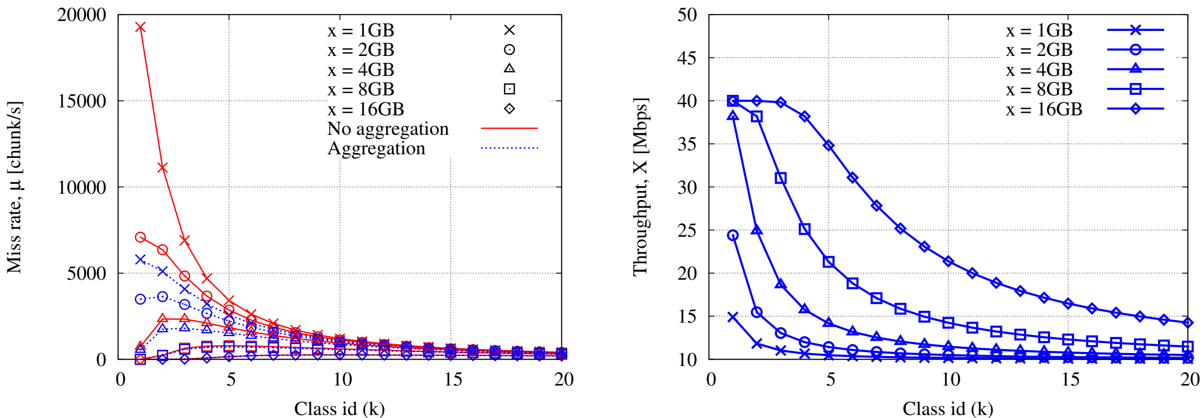

Fig. 8. Total miss rate (on the left) and mean stationary throughput (on the right) as a function of the cache size in the binary tree scenario.

impact on the bandwidth required at the gateway only for small caches (1 and 2GB); (ii) the maximum throughput is limited to $40Mbps$ even for very large caches as $W = 1$.

VIII. 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 control transport 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.

Acknowledgements

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] V. Jacobson, D. Smetters, J. Thornton, M. Plass, N. Briggs, and R. Braynard, "Networking named content," in *Proc. of ACM CoNEXT '09*.
[2] 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.
[3] K. Park and V. S. Pai, "Scale and performance in the coblitz large-file distribution service," in *Proc. of Usenix NSDI'06*.
[4] Z. Al-Qudah, S. Lee, M. Rabinovich, O. Spatscheck, and J. Van der Merwe, "Anycast-aware transport for content delivery networks," in *Proc. of WWW '09*.
[5] S. Arianfar, P. Nikander, and J. Ott, "On content-centric router design and implications," in *Proc. of ACM ReArch*, 2010.
[6] U. Lee, I. Rimac, and V. Hilt, "Greening the internet with content-centric networking," in *Proc. of ACM E-Energy*, 2010.
[7] G. Carofiglio, V. Gehlen, and D. Perino, "Experimental evaluation of storage management in conten-centric networking," in *Proc. of IEEE ICC*, 2011.
[8] W. King, "Analysis of paging algorithms," in *Proc. of IFIP*, 1971.
[9] 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.
[10] D. Starobinski and D. Tse, "Probabilistic methods for web caching," *Perform. Eval.*, vol. 46, no. 2-3, pp. 125–137, 2001.
[11] E. Coffman and P. Jelenković, "Performance of the move-to-front algorithm with markov-modulated request sequences," *Oper. Res. Lett.*, pp. 109–118, 1999.
[12] 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.
[13] P. R. Jelenković and X. Kang, "Characterizing the miss sequence of the lru cache," in *in Proc. of ACM SIGMETRICS, MAMA Workshop*, 2008.
[14] 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.
[15] G. Carofiglio, M. Gallo, L. Muscariello, and D. Perino, "Modeling data transfer in conten-centric networking (extended version)," in *Research report, available at http://perso.rd.francetelecom.fr/muscariello*, 2011.
[16] E. Chlebus and J. Brazier, "Nonstationary poisson modeling of web browsing session arrivals," *Information Processing Letters*, vol. 102, no. 5, pp. 187 – 190, 2007.
[17] A. Carzaniga, M. Rutherford, and A. Wolf, "A routing scheme for content-based networking," in *Proc. of INFOCOM 2004*.
[18] L. Cherkasova and M. Gupta, "Characterizing locality, evolution, and life span of accesses in enterprise media server workloads," in *Proc. of ACM NOSSDAV*, 2002.