Bandwidth and Storage Sharing Performance in Information Centric Networking

Giovanna Carofiglio
Bell Labs, Alcatel-Lucent
giovanna.carofiglio@alcatel-lucent.com

Massimo Gallo
Orange Labs, France Telecom
massimo.gallo@orange-ftgroup.com

Luca Muscariello
Orange Labs, France Telecom
luca.muscariello@orange-ftgroup.com

ABSTRACT

Internet usage has dramatically evolved towards content dissemination and retrieval, whilst the underlying infrastructure remains tied up to hosts interconnection. Information centric networking (ICN) proposals have recently emerged to rethink Internet foundations and design a natively content-centric network environment.

Important features of such networks are the availability of built-in network storage and of receiver-driven chunk-level transport, whose interaction significantly impacts overall system and user performance. In the paper, we provide an analytical characterization of statistical bandwidth and storage sharing, under fairly general assumption on total demand, topology, content popularity and limited network resources. A closed-form expression for average content delivery time is derived and its accuracy confirmed by event-driven simulations. Finally, we present some applications of our model, leveraging on explicit formulae for the optimal dimensioning and localization of storage resources.

Categories and Subject Descriptors

C.2.1 [**Computer-Communication Networks**]: Network Architecture and Design—*Packet-switching networks*

General Terms

Performance

Keywords

Information-centric networking

1. INTRODUCTION

In recent years we assisted to a shift in the communication paradigm used by the most popular Internet services from client-server towards publication, dissemination and retrieval of information.

With the spread of information-centric services, the need for a content-aware infrastructure has been addressed to a certain extent through application-layer solutions like CDNs, P2P overlays and HTTP proxies, deployed on top of the current infrastructure. If the evolution of the Internet architecture has been a fruitful sequence of incremental extensions and enhancements, there is today a number of recognized functional limitations in terms of performance, cost and complexity, originating in the superposition of solutions inconsistent with the original host-centric design. Pioneering ideas have been proposed, as Content-Centric Networking [9] (CCN, now considered in the NDN [1] project), DONA [10], PSIRP [2], SAIL [3], and significant effort devoted by Future Internet programs, as GENI and FIND (in the USA) and FIA (in Europe), all steering towards a network of information where content becomes first class network citizens. As a shared principle of ICN proposals, content (the information object) is uniquely identified, addressed and retrieved by its name independently from its location, and storage capabilities are distributed across the network.

To meet the challenges of ICN proposals, transport protocols need to natively support content dissemination and retrieval based on names instead of end-point addresses, and to exploit content-awareness for efficient usage of bandwidth and storage resources. A preliminary understanding of transport and caching issues in ICN appears to be necessary to quantify potential benefits and to guide optimized protocol design.

In this paper, we focus on the performance evaluation of ICN networks and develop an analytical model of bandwidth and storage sharing under limited resources. Each user is supposed to implement a receiver-driven flow control protocol yielding fair and efficient bandwidth utilization along the path to the content, while content storage is managed by least-recently used (LRU) per-chunk replacement policy. Under these assumptions, a closed-form characterization of the average content delivery time is provided, which captures the tradeoff between user performance and limited network resources. An interesting application of our model is the optimal dimensioning and localization of storage resources in a given bandwidth capacity setting.

The remainder of the paper is organized as follows. In Sec.2 we introduce ICN related work and summarize system description. Sec.3 describes the analytical model and provides a closed-form expression for the average stationary data delivery time, whose accuracy is confirmed by chunk-

---

[1] Named Data Networking www.named-data.net
[2] FP7 PubSub Internet Routing Paradigm www.psirp.org
[3] Network of Information www.sail-project.eu

level simulations in Sec.5. Some applications of the model are presented in Sec.6, before concluding the paper in Sec.7.

2. BACKGROUND

There have been several proposals of ICN architectures ranging from evolutionary to clean slate designs, all aimed at defining the building blocks of an information-centric network. The shift towards name-based networking is initially advocated in TRIAD [4], then reaffirmed in DONA [10], where authors introduce a new naming scheme and a route-by-name protocol built upon flat, self-certifying labels on top of the IP layer. Carzaniga et al. [7] define an alternative name-based forwarding and routing protocol solution, where names follow a general structure based on attributes/values couples.

CCN [9],[13] and more recently the NDN project, enlarge the scope of these works by adding two important features for a purely content-based network, say systematic in-network storage capabilities and receiver-driven chunk-based transport. The naming is hierarchical and modeled over the standard form of a URI (Universal Resource Identifier). Content items are split into self-identified chunks sent upon end-user request (interest) once the content is published into a network repository. Every node receiving an interest from an input interface, verifies if the given chunk is present in its local cache, otherwise it forwards it to the interface(s) indicated by the Forwarding Information Base (FIB). Ongoing requests are tracked by a Pending Interests Table (PIT) in order to send back data through interests' reverse path.

Similar principles inspire the clean-slate design of PSIRP (now in PURSUIT [5]) based on Publish-Subscribe routing made at rendez-vous resolution points and realized through source-routing. The NetInf design in [1] is more focused on the session layer in ICN: object search and name-resolution while the transport layer is currently studied in the SAIL project. Recent works, evaluating aspects of ICN architectures, explore content router issues [3], secure naming [14],[8] as well as congestion control [2]. Finally, an experimental evaluation of chunk-level caching in CCN is presented in [6]. Despite the differences between ICN proposals, a number of invariants can be recognized that we consider as the starting point for our study:

- named objects segmented in uniquely identified chunks,
- receiver-driven chunk-based transport protocol,
- routers with in-network per-chunk storage capabilities
- name-based routing and forwarding primitives.

More precisely, in this paper we assume a receiver-driven transport mechanism as that defined in the CCN architecture proposed in [9], which requires the expression of an interest per chunk and utilizes a pipelining of interests determined by a constant window W. We de not make any assumption on naming and routing, but rather focus on transport and caching interaction. To this purpose, we suppose each router to be equipped with a finite size LRU cache in order to store incoming data chunks to serve interests for stored data.
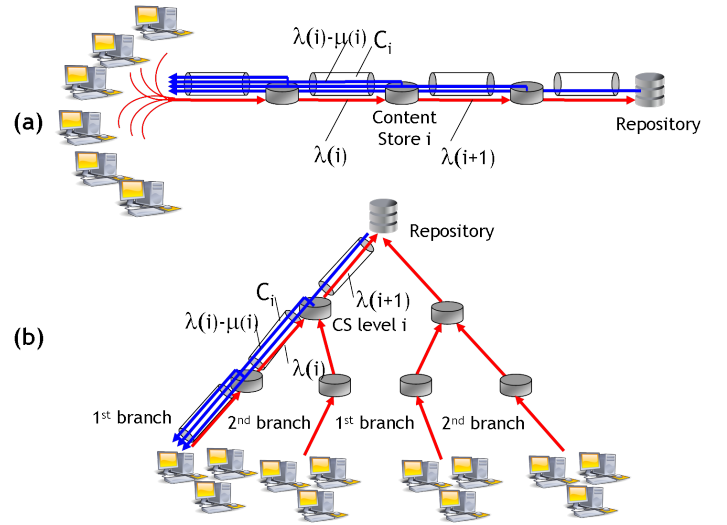

Figure 1: Linear (a) and binary tree (b) topologies with bandwidth-limited down-link.

3. ICN NETWORK MODEL

In this section, we present an analytical model of a network of caches as in fig.1 working under the above described ICN paradigm. Notation is summarized in tab.1.

Sec.3.1 introduces the content request process and provides an explicit characterization of the hit/miss probabilities at the caches along the path, under the assumption of no upstream bandwidth limitations. Instead, the downstream path to the user is characterized by finite link capacity, C_i for down-link i, and downstream bandwidth allocation is modeled by fluid max-min fair-share as described in Sec.3.2.

3.1 Storage sharing model

The main assumptions are the following:

- M different content items are equally partitioned in K classes of popularity, i.e. content items of class k are requested with probability $\{q_k\}_{k=1,...,K}$.
  We assume a Zipf popularity distribution, $q_k = \frac{c}{k^\alpha}$, $c = (\sum_{k=1}^{K} 1/k^\alpha)^{-1}$, with $\alpha > 1$.

- Content items are segmented into chunks (of size P) and the content size distribution is geometric with average σ chunks.

- N network nodes (or levels for tree topologies). Node i, $i = 1, ..., N$ is equipped with a cache of size x_i chunks. The repository is modeled as a cache of infinite size, as it stores all content items.

- The request arrival process is modeled through a Markov Modulated Rate Process (MMRP) [5] where 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 in the popularity class ($\lambda \equiv \lambda(1)$).
  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 (e.g. $W = 1$).

Under these assumptions, in [5, 4], we derived the miss probabilities in steady state for the topologies in fig.1 without considering any uplink bandwidth limitation. We report here the main results that we generalize to the case of possibly different cache sizes, Propp.3.1-3.2.

PROPOSITION 3.1. *Under previous assumptions, the stationary miss probability for chunks of class k, $p_k(1)$, is given by*

$$p_k(1) \sim e^{-\frac{\lambda}{m} q_k g x_1^{\alpha}} \qquad (1)$$

for large x_1, where $1/g = \lambda c \sigma^{\alpha} m^{\alpha-1} \Gamma \left(1 - \frac{1}{\alpha}\right)^{\alpha}$, and $q_k = c/k^{\alpha}$.

As shown in [5, 4], the miss process at node 1, and more generally $i \geq 1$, can be well approximated with a MMRP of intensity $\mu(i) = \sum_{k=1}^{K} p_k(i)\lambda_k(i)$. For the linear topology in fig.1(a) the miss rate at node i also coincides with the input rate at node $i + 1$, $i \geq 1$, i.e. $\mu(i) = \sum_{k=1}^{K} p_k(i)\lambda_k(i) \equiv \lambda(i + 1)$, whilst for the symmetric binary tree topology in fig.1(b), $\lambda(i + 1) = 2\mu(i)$, for $i \geq 1$.

The content popularity distribution also changes at hops $i > 1$ as a result of fact that only missing chunk requests are forwarded upstream, then for $k = 1, ..., K$, and using $q_k \equiv q_k(1)$,

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

For the linear topology in fig.1(a), the stationary miss probability values at hops $i > 1$ are consequently derived.

PROPOSITION 3.2. *Given a cascade of N nodes as in fig.1(a), a MMRP content request process with intensity λ and a popularity distribution q_k, $k = 1, ..., K$, $\forall 1 < i \leq N$ it holds*

$$\log p_k(i) = \prod_{l=1}^{i-1} \left(\frac{x_{l+1}}{x_l}\right)^{\alpha} p_k(l) \log p_k(1) \qquad (2)$$

PROOF. The proof is a straightforward extension of the result we proved for $x_i = x$, $\forall i \geq 1$, in Proposition 6.2 in [5, 4]. $\square$

COROLLARY 3.3. *Given a binary tree with $2^N - 1$ nodes (that is with N levels) as in fig.1(b) , a MMRP content request process with intensity λ and a popularity distribution q_k, $k = 1, ..., K$ then $\forall 1 < i \leq N$ it holds*

$$\log p_k(i) = \prod_{l=1}^{i-1} \left(\frac{x_{l+1}}{x_l}\right)^{\alpha} p_k(l) \log p_k(1). \qquad (3)$$

Notice that we have reported the results in the case without request aggregation in PITs to simplify the notation in the rest of the paper. Next results are still valid in presence of request aggregation by simply replacing the miss probabilities p_k with those computed in [5, 4] under request aggregation, p_k^f.

3.2 Bandwidth sharing model

In ICN, a receiver can download data from multiple nodes and share bandwidth with other concurrent transfers. User-perceived performance critically depends on the way bandwidth is shared between parallel downloads, A generally accepted fairness objective is max-min, which aims at rate

Table 1: Notation.

N	# of network nodes (or levels for tree topologies)
M	# of different content items (m per class)
σ	Average content size [chunks]
P	Packet size [bits]
$q(i)$	Content popularity distribution at node i
x_i	Cache size at node i [chunks]
$\lambda(i)$	Content request rate at node i
$\mu(i)$	Content miss rate at node i
$p_k(i)$	Miss probability for class k at node i
C_i	Capacity of link i [bps]
$R(i)$	Round trip delay between user and node i [s]
$\gamma(i)$	Max-min fair share on route i [bps]
T_k	Content delivery time of class k

equalization and can be realized either trough flow control at the receiver or by fair-queuing scheduling in network links (cfr.[11]).

In the previous section, we have computed the probability to find a given chunk in the cache of a given node i through the miss probabilities $p_k(i)$, $i = 1, ..., N$. Data chunks are sent back to the end-user following the reverse path of chunk requests. This implies that a request satisfied by node i triggers a corresponding data transmission to the users over the route that we denote with i, from node i to the end-user (blue arrows in fig.1). We make an abuse of notation by tagging every route with the same index i used for nodes, by meaning that route i is the set of links $l \ni i$ from the user to the i-th node along the path that goes up to the repository. Conversely $i \ni l$ indicates all routes traversing through link l.

Every route i is characterized by a number of parallel transfers, $\mathcal{N}_i$, sharing the same route and therefore the limited bandwidth.

$\mathcal{N}_i$ is a random process that varies according to the demand λ and the service rate each transfer in progress gets. $\mathcal{N}_i$ is a birth and death Markov process, with birth rate $\mu(i - 1) - \mu(i)$, with $\mu(0) \equiv \lambda$, and death rate $n_i \gamma(i)/(\sigma P)$ when n_i parallel transfers are in progress. Recall that $\mu(i - 1) - \mu(i)$ denotes the rate of the Poisson process of content requests satisfied by node i, under the assumption of a MMRP miss process of intensity $\mu(i)$ at node i. The death rate, $n_i \gamma(i)/(\sigma P)$, is determined by the max-min fair shared bandwidth $\gamma(i)$ in bps associated to each of the n_i content transfers in parallel over route i. Indeed, in this paper we assume that bandwidth is fairly shared among each transfer in the max-min sense. This implies that, under the capacity constraints

$$\sum_{i \ni l} n_i \gamma(i) \leq C_l, \qquad \text{for all links } l, \qquad (4)$$

there exists at least one bottleneck link $l \ni i$ for every route i serving such transfers at the maximum rate, i.e. $\exists l \ni i$ such that

$$\sum_{i \ni l} n_i \gamma(i) = C_l \qquad \text{and} \qquad \gamma(i) = \bigvee_{i' \ni l} \gamma(i') \qquad (5)$$

where $\bigvee$ denotes the maximum operator. Such conditions uniquely define the max-min share $\gamma(i)$ related to route i.

4. AVERAGE CONTENT DELIVERY TIME

Let us now apply bandwidth and storage models to determine average content delivery time. As illustrated in fig.1,

down-link $i > 1$ is shared among all routes j, with $j > i$. At chunk level, the data rate is given by $\sigma \sum_{j=i}^{N}(\mu(j-1)-\mu(j))$, which accounts for all data chunks flowing from upper caches (up to the repository) down to the receiver.

Let us denote with ρ_i the traffic load at link i,

$$\rho_i \equiv \frac{\sigma P}{C_i} \sum_{j=i}^{N}(\mu(j-1)-\mu(j)), \qquad (6)$$

Clearly, ρ_i must satisfy the stability condition $\rho_i < 1 \ \forall i = 1,\ldots,N$ to keep $\mathcal{N}_j$, $\forall j \ni i$ finite. With no loss of generality, we assume that all the significant source of delay is generated at the bottleneck link encountered along the path and consider any constant propagation delay negligible either in the upstream and downstream. Therefore, the average delay between user and node i, $R(i)$,

$$R(i) = PD_{up} + PD_{down} + P/\gamma(i) \approx P/\gamma(i), \qquad (7)$$

where we defined it as the sum of a constant up-link, PD_{up}, and down-link propagation delay, PD_{down}, plus the transmission delay of data chunks along route i.

PROPOSITION 4.1. *If $\rho_i < 1$, $\forall i \geq 1$, the average delivery time for content items in popularity class k, $\forall k = 1,...,K$ is*

$$\mathbb{E}[T_k] = \sum_{i=1}^{N} \bigvee_{j \ni i} \frac{\sigma P}{C_j(1-\rho_j)}(1-p_k(i)) \prod_{j=1}^{i-1} p_k(j), \qquad (8)$$

PROOF. In [5, 4], we derive the average content delivery time under no bandwidth limitation as $\mathbb{E}[T_k] = \sigma\mathrm{VRTT}_k/W$ where VRTT is the virtual round trip time defined as

$$\mathrm{VRTT}_k = \sum_{i=1}^{N} R(i)(1-p_k(i)) \prod_{j=1}^{i-1} p_k(j). \qquad (9)$$

In such case, the choice of a window of interests $W > 1$ makes a fundamental difference, as the system is not limited in bandwidth and $E[T_k]$ reduced of a factor W. In the case of a bandwidth-limited downstream path, the max-min fair share is determined by the number of transfers in progress (which does not vary with W) and by the link capacities so that eqq.(4-5) are satisfied. Under these conditions, allowing for a higher W does not enhance content delivery performance as the receiver is already exploiting all the available bandwidth. As a consequence, $\mathbb{E}[T_k] = \sigma\mathrm{VRTT}_k$ as if $W = 1$ (in the fluid approximation), with $R(i) \approx P/\gamma(i)$ as above explained. $\gamma(i)$ can be easily computed using the processor sharing model for parallel downloads assuming equal share of available bandwidth (see [11]). The average per data chunk delay is then given by $P/\gamma(i)$ with

$$\gamma(i) = \bigwedge_{j \ni i} C_j(1-\rho_j) \qquad (10)$$

where $\bigwedge$ denotes the minimum operator. Such allocation verifies conditions (4),(5). The statement follows by invoking Eqs.(7) and (10) in Eq.(9). □

5. SIMULATION RESULTS

To support analytical findings and to assess model accuracy, we implemented name-based forwarding and routing upon the simulator released by Carzaniga [6] et al. and performed event-driven simulations. Additional modules for the

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

transport layer have been developed realizing receiver-based transport, in-networks storage as well as output link queuing. Naming is adapted to CCN. More in detail, the receiver-driven transport protocol makes use of a simple fixed-size sliding window to send sequential requests. We also implemented forwarding queues with Deficit Round Robin scheduler [12] to impose fairness among parallel transfers at every link. Let us first consider the linear topology in fig.1(a), where content requests are forwarded through two nodes before reaching the content repository.

By varying link capacities, four configurations can be distinguished, depending on the bottleneck link which determines max-min fair shares for routes $1,2,3$:

(i) $\quad C_1(1-\rho_1) < C_2(1-\rho_2)$, $C_1(1-\rho_1) < C_3(1-\rho_3)$. In this case, eq. (8) reduces to

$$\mathbb{E}[T_k] = \frac{\sigma P}{C_1(1-\rho_1)},$$

imposing the same delivery time for different popularity classes.

(ii) $\quad C_2(1-\rho_2) < C_1(1-\rho_1)$, $C_2(1-\rho_2) < C_3(1-\rho_3)$. In this case,

$$\mathbb{E}[T_k] = \frac{\sigma P}{C_1(1-\rho_1)}(1-p_k(1)) + \frac{\sigma P}{C_2(1-\rho_2)}p_k(1).$$

(iii) $\quad C_3(1-\rho_3) < C_2(1-\rho_2) < C_1(1-\rho_1)$. Here,

$$\mathbb{E}[T_k] = \frac{\sigma P}{C_1(1-\rho_1)}(1-p_k(1)) + \frac{\sigma P}{C_2(1-\rho_2)}p_k(1)(1-p_k(2))$$
$$+ \frac{\sigma P}{C_3(1-\rho_3)}p_k(1)p_k(2)$$

(iv) $C_3(1-\rho_3) < C_1(1-\rho_1) < C_2(1-\rho_2)$. Here,

$$\mathbb{E}[T_k] = \frac{\sigma P}{C_1(1-\rho_1)}(1-p_k(1)p_k(2)) + \frac{\sigma P}{C_3(1-\rho_3)}p_k(1)p_k(2).$$

Fig. 2 reports analytical and simulated average content delivery time as a function of the popularity distribution and in a subset of configurations. We consider M = 20000 files, equally partitioned into K = 2000 classes, of average size σ = 100 chunks, of size P = 10kBytes. Content popularity distribution is assumed to be Zipf(α) with α = 2. Network links have a propagation delay equal to 10 μs, while nodes are equipped with an LRU cache of x = 5000 chunks (90% of the most popular contents in the catalog). Finally, content request rate is λ = 1 content/s.

Case (i) is illustrated in fig. 2 where we set (C_1, C_2, C_3) = $(10, 20, 30)$Mbps. As expected, the content delivery time is the same for all popularity classes, being link 1 the bottleneck for all routes, and a good match between model and simulation results can be observed.

Fig. 2 shows $\mathbb{E}[T]$ in configuration (ii) with (C_1, C_2, C_3) = $(30, 10, 20)$ Mbps, and (iii) with (C_1, C_2, C_3) = $(30, 20, 10)$ Mbps. The case (iv) is analyzed in Sec.6. In configurations (ii)-(iii) we remark a significant reduction of the average content delivery time w.r.t. to case (i), which benefits in a larger proportion to more popular classes, whose requests are more likely to be satisfied by the first cache.

Let us now analyze the binary tree topology described in fig. 1(b), in a symmetric and asymmetric setting depending on link capacities across the network and on content request rates. In the symmetric case, we set λ = 1 content/s, (C_1, C_2, C_3) = $(30, 20, 10)$ Mbps respectively for the
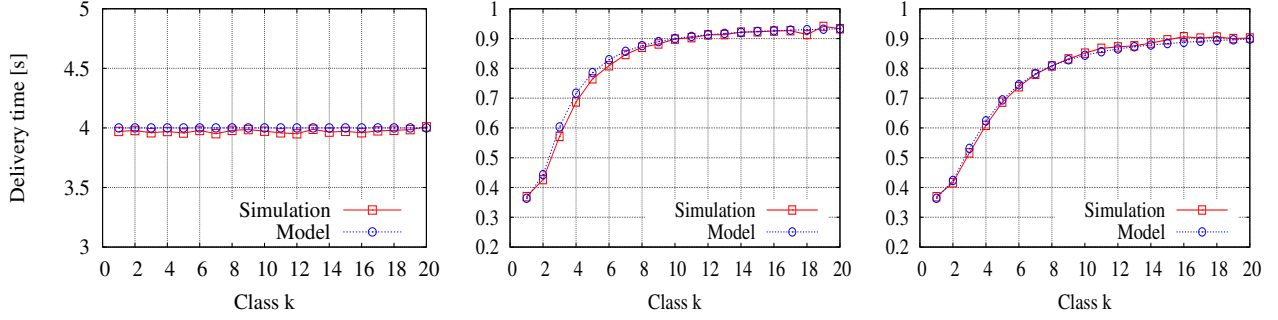
Figure 2: Average content delivery time as a function of the popularity class k for the linear topology in fig.1(a) in cases (i)-(ii)-(iii) from left to right.

first, second and third level links, while $x_1 = x_2 = 5000$ chunks. In the asymmetric case, instead, we differentiate between first and second branch of the tree as denoted in fig. 1(b). For the first branch, $\lambda_1 = 1$ content/s, $(C_{11}, C_{21}) = (30, 20)$ Mbps, whereas for the second one, $\lambda_2 = 2$ content/s, $(C_{12}, C_{22}) = (30, 30)$ Mbps (the second index refers to the branch number). The third level links, shared by the first and second branch have capacities $C_3 = 20$ Mbps. Moreover, cache sizes are: $(x_{11}, x_{12}, x_2) = (5000, 1000, 1000)$ chunks. With such parameters, first and second branch are in configurations (iii) and (iv), respectively, being both characterized by a bottleneck in the same shared link 3. Compared to the symmetric tree, the second branch is under-provisioned in terms of storage capacity. This results in higher traffic load upstream and consequent larger delivery time for the second branch. The first branch performance is, however, almost unchanged. If we compare the two tree setups in fig.3 (top and bottom plots) we see that performance for user's belonging to the first branch are insensitive to upstream overload due to misconfiguration of the storage in the second branch. We keep the evaluation of larger networks for future work due to lack of space.

6. APPLICATIONS

In this section, we focus on possible applications of our analysis to optimal bandwidth/storage dimensioning. We assume known average demand λ, topology and routing. The objective is twofold: first, to determine the minimum amount of network resources (links bandwidth, nodes storage) needed for guaranteeing some average performance metrics. As illustrated in previous sections, in fact, in-path storage has the effect of distributing traffic load across links along the path. The larger the storage space in a node the less the traffic load upstream and viceversa.

The second objective is to decide where and in which proportion the storage space at different network levels is more effective and enhances user's performance. Indeed, a typical problem faced by network operators is to decide how to distribute storage capabilities across the network under a fixed total amount of memory.

Let us consider the tree aggregation network depicted in fig.1(b). End-users are usually characterized by a small access bandwidth ($C_1 < C_2, C_3$, like for ADSL links) and by low local storage capacity (in home gateways and DSLAMs). In fact, storage space available at end-users is not shared
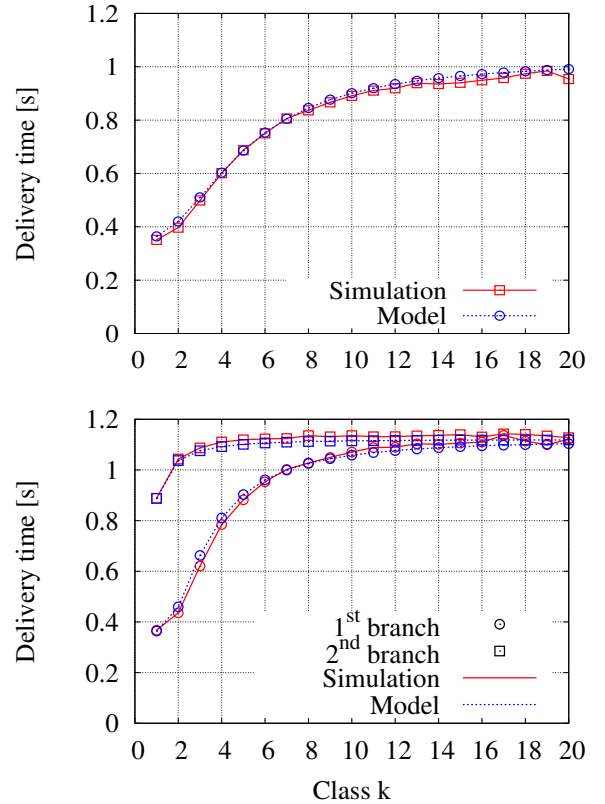

Figure 3: Average content delivery time as a function of the class k, for the binary tree topology in fig.1(b) under symmetric (top) and asymmetric (bottom) configurations.

among different clients and therefore it appears less attractive to network operators than the deployment of a larger amount of storage space in the back-haul where it can be exploited by a large number of end-users. Indeed, significant storage sharing takes place at node 2 aggregating requests forwarded upstream by different DSLAMs. Dimensioning of x_2 is, therefore, of crucial importance to "de-congestion" upper links and provide at the same time content delay guarantees. In terms of link capacity, link 3, relying content

requests to content repository, usually has a larger capacity $C_3 > C_2 > C_1$, but it also serves all requests coming from the two children nodes (say DSLAMs) on the left at the bottom of fig.1(b), so that $C_3(1-\rho_3) < C_1(1-\rho_1) < C_2(1-\rho_2)$. Given such configuration, a reasonable objective in terms of storage dimensioning in the back-haul is to provide the optimal x_2^* that minimizes $\mathbb{E}[T]$. By recalling and specifying eq.(8) in this scenario, we have

$$\mathbb{E}[T] = \sum_{k=1}^{K} q_k \mathbb{E}[T_k] = \frac{\sigma P(1 - \phi(x_2/x_1))}{C_1 - \lambda} + \frac{\sigma P \phi(x_2/x_1)}{C_3 - 2\lambda \phi(x_2/x_1)} \tag{11}$$

with $\phi(x_2/x_1) = \sum_{k=1}^{K} q_k p_k(1) p_k(2)$, where we used eq.(6) to define the traffic loads ρ_i, $i = 1, 2, 3$. Given the latter condition and the miss probabilities definition $p_k(2) = p_k(1)^{(\frac{x_2}{x_1})^{\alpha} p_k(1)}$, $\forall k = 1, ..., K$, it can be shown that $\mathbb{E}[T]$ has a minimum in

$$x_2^* = \beta x_1 \tag{12}$$

$$\beta = \phi^{-1}\left(\frac{C_3 + \sqrt{(C_1 - \lambda)C_3}}{2\lambda}\right)$$

by solving $\frac{\partial \mathbb{E}[T](x_2)}{\partial x_2} = 0$ in x_2 and by verifying usual conditions, i.e. that $\frac{\partial^2 \mathbb{E}[T](x_2)}{\partial x_2^2} \geq 0$ in x_2^*. In the same example, under the additional constraint of a total storage amount X,

$$x_1 + x_2 = X, \tag{13}$$

our model predicts the optimal partitioning among x_1 and x_2 that minimizes content delivery time, as expressed in (11). Indeed, by solving the linear system given by eqq. (12, 13), we obtain $x_1^* = X\beta/(1 + \beta)$ and $x_2^* = X/(1 + \beta)$.

7. CONCLUSIONS

The ICN paradigm radically changes transport networks: we move from the original TCP/IP model of a flow of bytes from a source to a destination to receiver-driven retrieval of named data pieces triggered by end-user requests. Ephemeral data carried by agnostic IP packets are replaced with self-identified and self-certified data chunks stored by forwarding network nodes and accessible by users' requests. Distributed in-network caching is an attractive property of ICN architectures as it avoids inefficient retransmissions of the same data from the source multiple times and naturally favors multipoint to multi-point communication. Independently of the specific ICN architecture considered, the definition and evaluation of transport and caching protocols is of the utmost importance to prove the viability of the entire concept.

In this paper, we provide an analytical framework for the evaluation of average content delivery performance under statistical bandwidth and storage sharing, which explicitly captures the underlying bandwidth/storage tradeoff. We further give hints on how to use our model as network planning tool in an aggregation network. As future work we intend to extend the model to a general class of topologies with a number of repositories distributed across the network.

Our analysis, though not tailored to a specific transport protocol, lies on the assumption of an optimal receiver-driven flow control yielding full and fair resource utilization. In the context of ICN proposals, preliminary results have appeared in [2]. However, a comprehensive ICN transport protocol design still lacks and we believe such research issue needs further investigation.

To this purpose, our model constitutes a reference for the evaluation of the newly designed flow control protocol in terms of efficiency and fairness. We leave for future work the analysis and planning of more complex network topologies, in comparison with real measured content requests.

Acknowledgments

8. REFERENCES

[1] B. Ahlgren and al. Design considerations for a network of information. In *Proc. of ACM CoNEXT'08*.

[2] S. Arianfar, P. Nikander, L. Eggert, and J. Ott. Contug: A receiver-driven transport protocol for content centric networks. In *Proc. of IEEE ICNP'10. Poster session*.

[3] S. Arianfar, P. Nikander, and J. Ott. On content-centric router design and implications. In *Proc. of ACM ReArch'10*.

[4] G. Carofiglio, M. Gallo, L. Muscariello, and D. Perino. Modeling data transfer in content centric networking. In *Proc. of 23rd International Teletraffic Congress 2011 (ITC23)*.

[5] G. Carofiglio, M. Gallo, L. Muscariello, and D. Perino. Modeling data transfer in content centric networking (extended version). In *Technical Report http://perso.rd.francetelecom.fr/muscariello/report-itc-transport.pdf*.

[6] G. Carofiglio, V. Gehlen, and D. Perino. Experimental evaluation of storage management in content-centric networking. In *Proc. of IEEE ICC*, 2011.

[7] A. Carzaniga, M. Rutherford, and A. Wolf. A routing scheme for content-based networking. In *Proc. of IEEE INFOCOM 2004*.

[8] C. Dannewitz, J. Golic, B. Ohlman, and B. Ahlgren. Secure naming for a network of information. In *IEEE Global Internet 2010*.

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

[10] 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.

[11] L. Massoulié and J. Roberts. Bandwidth sharing: objectives and algorithms. *IEEE/ACM Transactions on Networking*, 2002.

[12] M. Shreedhar and G. Varghese. Efficient fair queuing using deficit round-robin. *IEEE/ACM Transactions on Networking*, 1996.

[13] D. Smetters and V. Jacobson. Securing network content. In *PARC Technical Report*, 2009.

[14] W. Wong and P. Nikander. Secure naming in information-centric networks. In *Proc. of ACM ReArch'10*.