Enabling opportunistic search and placement in cache networks

Guilherme Domingues [a,b], Edmundo de Souza e Silva [a], Rosa M.M. Leão [a,*],
Daniel S. Menasché [a], Don Towsley [c]

[a] *Federal University of Rio de Janeiro, Brazil*
[b] *State University of Rio de Janeiro, Brazil*
[c] *University of Massachusetts at Amherst, USA*

ABSTRACT

Content distribution networks have been extremely successful in today's Internet. Despite their success, there are still a number of scalability and performance challenges that motivate clean slate solutions for content dissemination, such as content centric networking. In this paper, we address two of the fundamental problems faced by any content dissemination system: content search and content placement. We consider a multi-tiered, multi-domain hierarchical system wherein random walks are used to cope with the tradeoff between exploitation of known paths towards custodians versus opportunistic exploration of replicas in a given neighborhood. TTL-like mechanisms, referred to as reinforced counters, are used for content placement. We propose an analytical model to study the interplay between search and placement. The model yields closed form expressions for metrics of interest such as the average delay experienced by users and the load placed on custodians. Then, leveraging the model solution we pose a joint placement-search optimization problem. We show that previously proposed strategies for optimal placement, such as the square-root allocation, follow as special cases of ours, and that a bang-bang search policy is optimal if content allocation is given.

© 2017 Elsevier B.V. All rights reserved.

1. Introduction

Content distribution is in the vogue. Nowadays, virtually everybody can create, distribute and download content through the Internet. It is estimated that video distribution will alone account for up to 80% of global traffic by 2017 [1]. Despite the success of the current Internet infrastructure to support user demand, scalability challenges motivate clean slate approaches for content dissemination, such as information centric networking.

In information centric networks (ICNs), the focus is on content, rather than on hosts [2,3]. Each content has an identification and is associated to at least one custodian. Once a request for a content is generated it flows towards a custodian through routers equipped with caches, referred to as *cache-routers*. (As such, ICNs are also referred to as cache networks. In this paper, we use both terms interchangeably.) A request that finds the content stored in a cache-router does not have to access the custodian. This alleviates the load at the custodians, reduces the delay to retrieve the content and the overall traffic in the network. To achieve performance gains with respect to existing architectures, in information

centric networks cache-routers must efficiently and distributedly determine how to route content requests and where to place contents.

ICN architectures, such as NDN [2], are promising solutions for the future Internet. Still, it is unclear the scope at which the proposed solutions are feasible [4]. Incrementally deployable solutions are likely to prevail [5], and identifying the simplest foundational attributes of ICN architectures is essential while envisioning their Internet scale deployment.

The efficient management of distributed storage resources in the network coupled with the routing of requests for information retrieval are of fundamental importance [6,7]. However, the interplay between search and placement is still not well understood, and there is a need to study search and placement problems under a holistic perspective. In fact, an adequate framework within which to assess the overall performance gains that ICNs can provide is still missing [6].

In this paper, we propose and study a simple cache-network architecture comprising of a logical hierarchy of cache-routers divided into tiers, where each tier is subdivided into one or more logical domains (Fig. 1). In-between domains, requests are routed from users towards custodians which are assumed to be placed at the top of the hierarchy.

* Corresponding author.
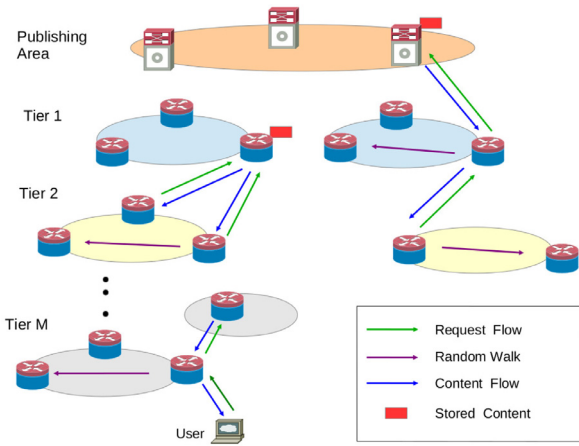 *E-mail address:* rosammleao@gmail.com (R.M.M. Leão).

Fig. 1. System diagram.

To route content requests from users to custodians, a random lookup search takes place in the vicinity of the logically connected cache-routers (horizontal arrows in Fig. 1). Cache-routers within a domain are assumed to form a logical *clique*. As such, a request that does not find the searched content in a cache-router is forwarded to one of the remaining cache-routers in the same domain. The goal is to opportunistically explore the presence of content replicas in a given domain. If a copy is found in the domain within a reasonable time interval, the content is served. Otherwise, requests are routed from users towards custodians (vertical arrows in Fig. 1). Custodians as well as the name resolution system (NRS) are supplied by third parties at the publishing area, and we focus our attention on the infrastructure from users to publishing areas.

By using random walks to opportunistically explore the presence of content replicas closer to users, we avoid content routing tables and tackle the scalability challenge posed in [8]. An alternative would be to adopt scoped-flooding [9]. However, scoped-flooding is more complex than random walks and requires some level of synchronization between caches. In addition, random walks have been show to scale well in terms of overhead [10].

To efficiently and distributedly place content in the cache network, we consider a flexible content placement mechanism inspired by TTL caches. At each cache, a counter is associated to each content stored there, which we refer to as *reinforced counter* (RC). Whenever the RC surpasses a given threshold, the corresponding content is stored. The RC is decremented at a given established rate, until reaching zero, when the content is evicted.

Focusing on two of the simplest possible mechanisms for search and placement, namely random walks and TTL-like caches, our benefits are twofold. From a practitioners point of view, the proposed architecture is potentially deployable at the Internet scale [4]. From the performance evaluation perspective, our architecture is amenable to analytical treatment. Our quantitative analysis provides closed-form expressions for different metrics of interest, such as the average delay experienced by users.

Given such an architecture, we pose the following questions,

1. How long should the random-walk based search last at each domain so as to optimize the performance metrics of interest?
2. How should the reinforced counters be tuned so as to trade-off content retrieval delay with server load at the custodian?
3. What parameters have the greatest impact on the performance metrics of the proposed cache network architecture?

To answer these questions, we introduce an analytical model that yields: a) the expected delay to find a content (average search time) and; b) the rate at which requests have to be satisfied by custodians. While the expected delay is directly related to users quality of experience, the rate of accesses towards the custodian is associated with publishing costs. The model yields simple closed-form expressions for the metrics of interest.

Using the model, we study different tradeoffs involved in the setting of the parameter values. In particular, we study the tradeoff between the time spent in opportunistic exploration around the vicinity of the user in order to find content and the custodian load.

In summary, our key contributions are the following:

Cache-network architecture: we propose a simple multi-tiered cache network architecture based on random walks and TTL-like caches. Simplicity eases deployment and allows for analytical treatment, while capturing essential features of other cache-network architectures such as the tension between opportunistic exploration of replicas closer to users and exploitation of known paths towards custodians.

Analytical model: we introduce a simple analytical model of the proposed cache network architecture that can be helpful in the performance evaluation of ICNs. In particular, we consider the interplay between content placement and search. Using the model we show that we can achieve performance gains using a simple search strategy (random walks) and a logical hierarchical storage organization. Although our analysis is focused on the proposed architecture, we believe that the insights obtained are more broadly applicable to other architectures as well, such as scoped-flooding [9].

Parameter tuning: we formulate an optimization problem that leverages the closed-form expressions obtained with the proposed model to determine optimal search and placement parameters under storage constraints. We show that previously proposed strategies for optimal placement, such as the square-root allocation, follow as special cases of our solution, and that a bang-bang search policy is optimal if content allocation is given.

Performance studies: we investigate how different parameters impact system performance under different assumptions regarding the relative rate at which requests are issued and content is replaced in the cache-routers.

The remainder of this paper is organized as follows. After introducing background in Section 2, we describe the system studied in this paper in Section 3. An analytic model of this system is presented in Section 4. The joint placement and search optimization problem is posed and analyzed in Section 5 and numerical evaluations are presented in Section 6. Further discussions are presented in Sections 7 and 8 concludes.

2. Background and related work

In this section we introduce the background used in this paper. We start by describing previous works on the modeling and analysis of cache networks. Then, in Section 2.2 we present previously proposed ICN architectures and in Section 2.3 we indicate some of the challenges they pose.

2.1. Modeling and analysis of cache networks

The literature on the modeling and analysis of cache networks, accounting for TTL caches [11,12], random replacement [13], complexity aspects [14] and optimal caching [15] is rapidly growing. There has been also recent efforts on determining unified frameworks to study and compare different replacement policies [16]. To the best of our knowledge, none of these previous works considered the joint problem of optimally determining search and placement strategies for TTL-like cache networks.