Content Placement in Heterogeneous End-to-End Virtual Networks

Kostas Katsalis,
Univ. of Thessaly, Greece.
kkatsalis@uth.gr

Vasilis Sourlas
CERTH-ITI, Greece.
vsourlas@uth.gr

Thanasis Papaioannou
CERTH-ITI, Greece.
thanasis.papaioannou@iti.gr

Thanasis Korakis
Univ. of Thessaly, Greece.
korakis@uth.gr

Leandros Tassiulas
Yale, USA.
leandros.tassiulas@yale.edu

ABSTRACT

One of the distinctive features of the new virtualized ecosystem, is the multi-stakeholder participation in the way cloud services are designed, deployed and offered. In this work, we address the emerging content replication problem a Content Delivery Network (CDN) provider needs to consider, when deploying its network over multi-domain, heterogeneous environments, where virtual network operators utilize both the CDN services and the virtualized infrastructures. In our model, the benefit that the CDN provider enjoys may be different per network operator for the same request, while our model takes into account the replication cost to every domain, as well as the user mobility, besides physical storage limitations. Since the optimal placement of the objects at the caches of the various domains resembles the multiple knapsack problem, which is NP-complete, we provide two approximate solutions to the emerging content placement problem. We evaluate the proposed policies through extensive simulations and we compare them against a myopic method, where a domain is unaware of the caching strategy of the other domains that is connected to.

Keywords

Wireless Network Virtualization, Content Delivery Networks, Content Placement, Heterogeneous Networks.

1. INTRODUCTION

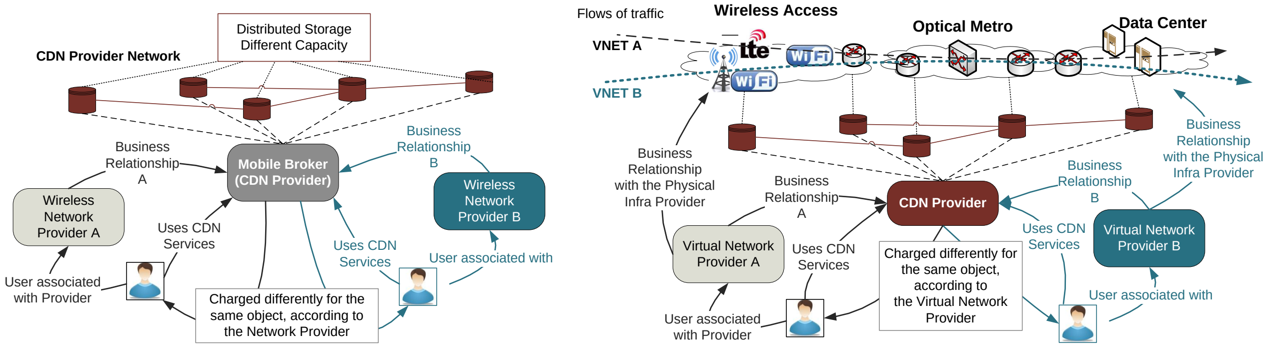
Cloud computing technologies offer the necessary infrastructures to rapidly deploy large distributed systems and applications and together with the recent advances in the way wireless access technologies evolve to provide ubiquitous wireless access, a technological breakthrough is on the go. The potential for collaboration between cloud/virtualization technologies and ubiquitous wireless access networks is enormous and this “coupling” constitutes a true paradigm shift over which services can be build. The services we focus in this paper are CDN services provided over multiple provider networks, virtual or not, where their access networks are deployed in wireless heterogeneous networks (HetNets).

In particular, we study replication/caching strategies for efficient content placement, where the objective is to maximize the *net benefit* of the CDN provider (i.e., the total user utility minus the total cost) under the following assumptions. Every user/subscriber is associated with a network provider and the CDN's content is distributed among all the subscribers. Also, we assume that there exists a different retrieval cost per object depending on the network provider each object is requested from. This is a highly reasonable assumption, since under multiple CDN schemes different business relationships exists between the CDN provider, the network providers and the network operators. Furthermore, in the wireless domain we assume a wireless HetNet [1], where a mobile subscriber can potentially change access network, e.g., for offloading purposes or for any other optimization criterion.

We also describe two schemes that motivate our study. The first is a classic one, where a mobile broker is acting as the CDN provider and distributes content among subscribers associated with Mobile Operators and Mobile Virtual Operators (MVNOs), whereas the second scheme is about a modern virtualized environment, where different business relationships exists between the physical providers, the virtual network operators (VNOs) and the actual service providers. For instance, in the latter one a discount may be applied for CDN services to virtual network operators that sign for high bandwidth (and thus costly) SLAs with the physical network providers.

Our contributions are the following. Firstly, we develop a mathematical framework for efficient content placement in multi-domain environments. The model takes into account the probability distribution of users belonging to one access network or another, while the network providers (operators) that the users are associated with, have different business relationships with the physical providers and the CDN providers; these relationships affect the cost of content retrieval. The proposed model also considers the content placement cost in every domain, besides physical storage size limitations. Then, a greedy centralized approach and a distributed content placement/replacement scheme (low overhead and easily implementable) are proposed and evaluated through extensive simulations. Note that besides virtual

SAC'15 April 13-17, 2015, Salamanca, Spain.
Copyright 2015 ACM 978-1-4503-3196-8/15/04$15.00
http://dx.doi.org/10.1145/2695664.2695673


Distributed Storage
Different Capacity
CDN Provider Network
Business Relationship A
Mobile Broker (CDN Provider)
Business Relationship B
Wireless Network Provider A
Wireless Network Provider B
Uses CDN Services
Uses CDN Services
User associated with
User associated with Provider
Charged differently for the same object, according to the Network Provider
Flows of traffic
Wireless Access
Optical Metro
Data Center
VNET A
VNET B
Business Relationship with the Physical Infra Provider
Business Relationship A
CDN Provider
Business Relationship B
Business Relationship with the Physical Infra Provider
Virtual Network Provider A
Virtual Network Provider B
Uses CDN Services
Uses CDN Services
Charged differently for the same object, according to the Virtual Network Provider
User associated with Provider
User associated with

(a) Mobile Broker as CDN provider.

(b) CDN provider and Virtual Networks.

Figure 1: Network Architecture

end-to-end networking, the concept of cloud CDN providers [2] has also emerged. In this work, we focus on a single CDN provider (virtual or not) and the content placement problem (over physical or virtual infrastructures), since the CDN content placement and provisioning is important to be understood and optimally controlled. Investigation of scenarios with multiple CDN providers are left for future work.

The rest of the paper is organized as follows. In Section 2, we motivate our work, and survey related work. In Section 3, we formulate the problem of content placement in multi-domain environments, whereas in Section 4 we present the content placement policies. In Section 5 we evaluate through simulations the proposed policies, while we conclude the paper and give pointers for future work in Section 6.

2. MOTIVATION AND RELATED WORK

Under the concept of wireless HetNets, macro base stations, pico base stations alongside with femtocells and WiFi APs, aim to deliver the future wireless access infrastructure [1]. We are motivated by the HetNet concept, where multiple wireless access technologies (i.e. WiFi) are used to offload traffic from LTE networks. Despite the fact that work already exists in literature on offloading decision making and policies for inter-network hand-offs [1][3], very few results exist on the way services are build over the converged infrastructure. In this work, we focus on such a problem, namely the problem of cache management and content placement that a CDN provider (or a virtual CDN provider) addresses, when CDN services are provided to multiple network operators (virtualized or not). In the following, we discuss two application scenarios under which the modeling assumption of different cost per provider is essential. This assumption is already present in practice, where CDN service providers address the problem of content placement under different business relationships with Mobile Network Providers and will further emerge as we enter the cloud computing/SDN era.

A Mobile Broker as CDN provider: The network architecture of a Mobile Broker is shown in Fig. 1(a). We refer as Mobile Broker, a mobile, social and application-based service provider, that enables operators to offer content that owns, as well as to distribute content (video, music, etc.) to their subscribers through its own CDN network. Every mobile user is a subscriber of the Broker and is also associated with a Mobile Network Provider (physical or virtual). Depending on the business relationship between the Provider and the Broker, the subscriber may enjoy different charges from the Broker (and the Broker different benefits) for the same content. This model is highly realistic and is the one that is also found in practice (e.g., in [4]).

A CDN Provider and Virtual end-to-end Networks: The recent advances in network virtualization and SDN technologies are paving the way to true end-to-end virtualization. The technology to build end-to-end virtual networks that exploit wireless, optical and data center infrastructures based on the SDN paradigm, although not standardized, is currently available (i.e. [5] and [6]). The network architecture we consider is depicted in Fig. 1(b). A number of access networks are connected through multiple optical networks up to the data center(s) and virtual end to end networks (VNets) operate over converged virtualized infrastructures [5]. Any user is logically associated to a VNet (owned by a Virtual Network Operator), but physically served by a number of access domains.

A single CDN provider owns content that users from all the VNets can access, while the CDN provider can establish different business relationships with a) every physical storage provider regarding the placement cost of content and b) the VNet operators. The problem under consideration is to weigh the trade-off between speed of content access and increased user QoE, that we translate in higher revenue for the CDN provider, with the cost of content placement in every physical domain in a way that will lead in profit maximization for the CDN provider. In the rest of the paper we focus on the analysis of the VNets/CDN scenario, since it is more generic and of high importance to the building of end-to-end virtual networks.

3. SYSTEM MODEL AND PROBLEM STATEMENT

Let $\mathcal{K} = \{1, 2, \cdots, K\}$ denote the set of all the available domains (e.g., optical A, optical B, WiFi, etc) and $\mathcal{L} = \{1, 2, \cdots, L\}$ denote the set of all the access domains (e.g., LTE A, LTE B, WiMAX, WiFi A, etc), where $\mathcal{L} \subseteq \mathcal{K}$. Also let $\mathcal{V} = \{1, 2, \cdots, V\}$ denote the set of all end-to-end virtual networks. We assume that every mobile user is associated with a single VNet. We also assume that a single CDN provider offers content services with $\mathcal{M} = \{1, 2, \cdots, M\}$ objects. In our model, all the objects are accessible by all users belonging in all VNets; however, the utility for an object i enjoyed by a CDN user belonging in VNet j is different across VNets. This is reasonable in our end-to-end virtualized model, since the physical providers and their business relationship with the CDN provider may be also affected by

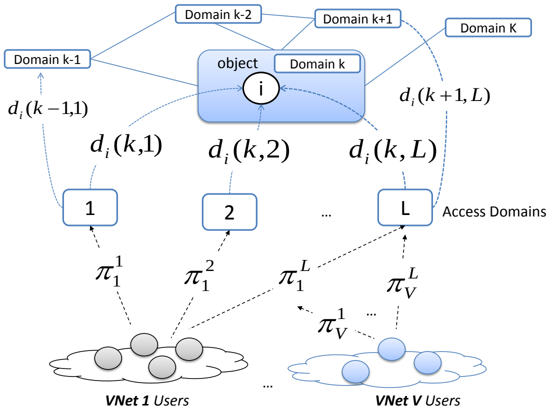


Figure 2: System Model

the business relationship between them and the VNet operator. We use $u_{i,j}$ to denote the willingness to pay (in cost units) of a user belonging in VNet j for accessing object i.

In the access domain, a lot of research has been done regarding the optimal access network selection [3] and rate distribution in HetNets [1], while various mobility models exhibit the characteristics of temporal dependency, spatial dependency and geographic constraint mobility [7]. Nevertheless, in this work in order to handle user mobility and also be aligned with the HetNet concept, we are only interested in the steady state probability of a user using a number of access networks. We adopt a simple Markov model for any user, where states represent the access networks that a user/subscriber is physically served from. These steady state probabilities (of using one physical access network or another) are then used, to "split" the user's total request traffic to the various access networks it enables.

Let $r_{i,j}$ denote the request rate for object i from all the users associated with VNet j. If we let π_j^l denote the steady state probability of any user that belongs to VNet j to be served by access network l, then the request rate distribution is equal to $r_{i,j}^l = \pi_j^l \cdot r_{i,j}$, $\forall l \in \mathcal{L}$. Also, we define $d_i(k,l)$ as the distance between the closest domain $k \in K$ where object i is placed and the access domain $l \in \mathcal{L}$ where the request originates.

We also use the following notation: c_i^k is the cost of placing object i in domain k (c_i^k also includes the transfer cost that the CDN provider pays to the physical network provider(s) to transfer object i in domain k), p_i^k is the probability of finding object i in domain k (and is our control variable as we show later), s_i is the size of object i and S^k is the storage capacity of domain $k \in \mathcal{K}$ in bits accordingly. Fig. 2 depicts the system model under consideration and Table 1 provides a notation summary.

3.1 Problem Statement

We define the utility $U_j(l)$ enjoyed by the CDN provider when the objects are accessed by users of VNet j residing at the $l \in \mathcal{L}$ access network as

$$U_j(l) = \sum_{i=1}^{|\mathcal{M}|} u_{i,j} \cdot r_{i,j}^l \cdot (1 - f(D_i(l)) + \Delta_i^*) \tag{1}$$

where $D_i(l)$ is the minimum distance between the domain k where object i is stored, and the requester when the requester is served by access network l. We define this distance as

$$D_i(l) = \min_{k:p_i^k=1} d_i(k,l) \tag{2}$$

Table 1: Notation Summary

Not.	Description
$\mathcal{K}$	set of domains (optical, WiFi etc)
$\mathcal{L}$	set of access domains (e.g LTE,WiFi), $\mathcal{L} \subseteq \mathcal{K}$
$\mathcal{V}$	set of Virtual Network Operators (VNO)
$\mathcal{M}$	set of objects
$u_{i,j}$	willingness to pay of a user in VNet j for object i
$r_{i,j}$	request rate for object i by users associated to VNO j
π_j^l	steady state prob. of a user in VNet j to be served by access network l
$d_i(k,l)$	distance between domain k where object i is placed and access domain l
c_i^k	cost of placing object i in domain k
p_i^k	probability of finding object i in domain k
s_i	size of object i
S^k	storage size of domain k
$U_j(l)$	Utility $U_j(l)$ the CDN provider enjoys from VNet j when the objects are accessed by access network l

where $d_i(k,l)$ is the distance expressed by hop counts.

In Eq.(1) function $f : R_+ \rightarrow [0,1]$ is used to normalize the distance values in a range between $[0,1]$. The intuition behind Eq.(1) and Eq.(2) is that whenever a user retrieves content from an access domain and the objects retrieved are also stored in this domain then $f(D_i(l)) = 0$. In this case, the maximum net benefit (i.e., utility minus cost) occurs for the CDN (content is retrieved as fast as possible and thus user willingness to pay is maximized); thus, CDN provider can make higher profit and increase demand for CDN services. In the case where the requested content can be found only in the data center, then $f(D_i(l)) = 1$. In this case, we use Δ_i^* to describe that a minimum gain (satisfaction) is achieved even when the object is found in the data center.

The probability p_i^k of finding object i in domain k is defined as

$$p_i^k = \begin{cases} 1, & \text{if } k = 1 \text{ (the datacenter)}, \\ \in \{0,1\}, & \text{otherwise.} \end{cases} \tag{3}$$

With the above definition we assume that domain $k = 1$ is the data center and that all objects are stored in the data center, while is up to the used cache management algorithm to decide in which other domain(s) to replicate the content (object i in this case). Note that $p_i^k = 0$ means that the object i is not available in domain k, while $p_i^k = 1$ means that the object i is available in domain k. These values are used to define the binary matrix $P = [p_i^k]$ of size $K \times M$, that is the control variable to the following optimization problem

$$\underset{P}{\text{maximize}} \quad \sum_{l=1}^{|\mathcal{L}|} \sum_{j=1}^{|\mathcal{V}|} U_j(l) - \sum_{i=1}^{|\mathcal{M}|} \sum_{k=1}^{|\mathcal{K}|} p_i^k \cdot c_i^k \tag{4}$$

$$\text{subject to} \quad \sum_{i=1}^{|\mathcal{M}|} p_i^k \cdot s_i \leq S^k, \forall k \in \mathcal{K} \tag{4a}$$

$$\sum_{k=2}^{|\mathcal{K}|} p_i^k \leq |\mathcal{K}| - 1, \forall i \in \mathcal{M} \tag{4b}$$

where c_i^k is the cost of placing object i in domain k. Note that we have also included the transfer cost in c_i^k. The physical interpretation of this inclusion is that in order to cache an object closer to the requester and increase its QoE and the CDN's profit, a higher transfer cost based on the business relationships of the CDN provider with the network provider or the ISP would be observed. This inclusion makes the model even more generic, since complex relationships can also be defined. The first restriction Eq.(4a) is set in order to meet the capacity constraints in every domain, whereas the second restriction Eq.(4b) means that any object i can be stored in up to $|\mathcal{K}| - 1$ domains, besides the data center where it is already stored. Eq.(4) provides the *maximum net benefit* of the CDN provider.

4. CONTENT PLACEMENT POLICIES

Effective implementation of the vision for inter-domain cloud-based CDNs requires the formulation of a robust methodology regarding the coordination between multiple providers (e.g. access, optical, ISPs) and the integration of multiple cloud and virtualization technologies. Inter-domain CDNs rely on the interaction between ISPs and CDNs and cache management schemes over this converged environment are presented in [8] and [9], while replica-selection decisions coordination is presented in [10] in a distributed environment, focusing in cloud services. Also, today caches use dedicated hardware on a per-CDN provider and per-operator basis [11]. In the new virtualized environment, by applying the NFV paradigm in order to utilize and deploy virtualized caches, the underlying hardware resources could be consolidated and shared among multiple CDN providers, improving resources usage [12].

An autonomic cache management framework for future Internet was presented in [13], while a work on cache management over converged end-to-end virtual networks was presented in [14]. In our modeling, in contrast to [13]-[14] we consider different cost per end-to-end VNet and in addition, we consider user mobility, multiple access domains, while we take into account the placement cost, besides physical storage limitations. Placement algorithms that use workload information, such as distance from the storage points and request rates, to make the placement decision are investigated in [15].

Even at steady state (e.g., static object access rates, fixed geographical distribution of users, fixed storage costs, etc.) optimal placement of the objects at the caches of the various domains resembles the multiple knapsack problem, which is NP-complete. Hence, we provide two approximate solutions to the problem of object placement.

Centralized Approach - Greedy

In [16], a greedy approximation algorithm has been proposed based on a cost metric related to the distance of the object storage points to the end-users and the object request rates. According to [16], the greedy algorithm has a median performance of $(1.1 - 1.5) \cdot OPT$ and a worst case of $4 \cdot OPT$.

We adapt the greedy approximation algorithm of [16] to our problem as follows. The greedy algorithm works in rounds/iterations. At each round, for each object at the data center, the net benefit gain of its replica placement at every feasible domain is calculated, given that the rest of the objects are already cached at each domain. The object whose replication at a domain gives the highest net benefit gain is selected to be replicated at that domain. The process is repeated for all objects until all the available storage capacity of every possible domain is full. This algorithm requires $\min\{M, \sum_{k=2}^{K} S^k/\bar{s}\}$ iterations, where $\bar{s}$ is the mean object size.

The greedy approximation algorithm is a centralized and static approach that should be re-executed for new objects, for objects whose request rates are significantly modified or whenever other significant problem parameters are modified (e.g. cache storage/placement costs).

Decentralized Approach - Holistic

In this approach, we assume a cache manager at each domain (other than the data center) that acts as autonomous agent and takes decisions, so as to host the objects that maximize the overall net benefit. Periodically, the cache manager of a domain may decide to fetch new objects from the data center or remove cached objects according to the following process:

1. Calculate the net benefit decrease arising from the removal of an object stored at the domain. Sort objects in ascending order of the net benefit decrease in list D.
2. Calculate the net benefit increase by caching a new object at the domain. Sort objects in descending order of the net benefit increase in list I.
3. Select object o^* with size s_{o*} at the top of list I, whose insertion results to the maximum net benefit increase.
4. Starting from the beginning of list D, select d objects, so that their total size is greater or equal than the size of object o^*, i.e., $\sum_{i=1}^{d} s_i \geq s_{o*}$.
5. If the total net benefit decrease arising from the removal of d objects is lower than the net benefit increase by the insertion of the new object o^*, then replace the d objects with object o^*.
6. If there is some extra storage space left at the domain by the removal of the d objects, i.e., $\sum_{i=1}^{d} s_i - s_{o*} = e^* > 0$, then go through the list I and fetch every new object that fits and remove its size from the extra space e^*, until no new object fits any more or the extra space is exhausted.
7. Repeat the above steps until no further object replacements can be made among domains and increase the overall net benefit.

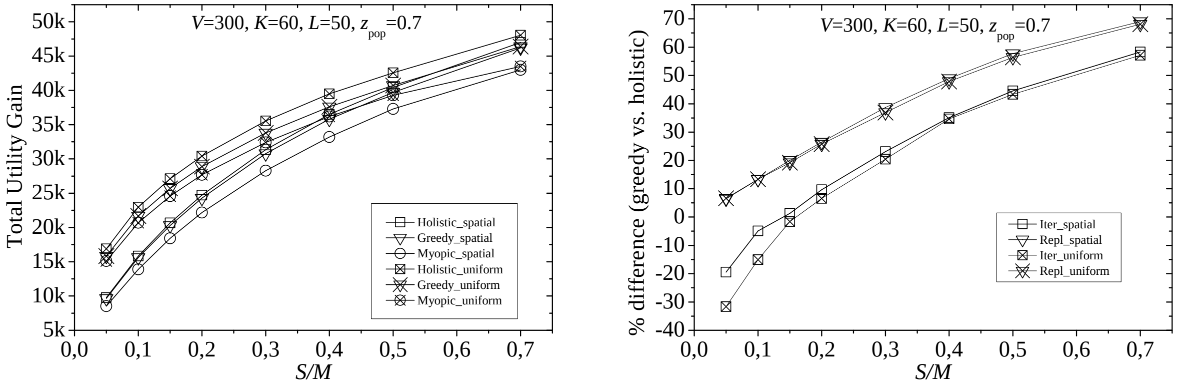
Only one domain (manager at each domain) is allowed to perform object replacements at each iteration by means of a distributed consensus algorithm, i.e., Paxos [17], until a steady state is reached where no further object replacements occur.

5. PERFORMANCE EVALUATION

In this section, we evaluate through simulations the performance of the two cache management algorithms (*Greedy* and *Holistic*) and we compare them against a *Myopic* algorithm. In the *Myopic* algorithm each domain caches objects with the objective to maximize the overall net benefit based on the observed request pattern, without having any knowledge of the caching decisions of the other domains, i.e. an intermediate domain does not know the caching strategy of the access domains that are connected to it.

For the performance evaluation we assume that all domains have the same caching capacity $S^k = S$, $\forall k \in \mathcal{K}$. We also consider the scenario of $|\mathcal{M}| = 10^6$ different unit sized objects, where the request rate for each object from each VNet $j \in \mathcal{V}$ is determined by its popularity. Here we approximate the popularity of the objects by a Zipf law of exponent z_{pop} (file popularity in the Internet follows Zipf distribution [18]-[19]). The request rate of each object at each VNet varies from $0 - 20$ reqs/sec according to its popularity and ranking.

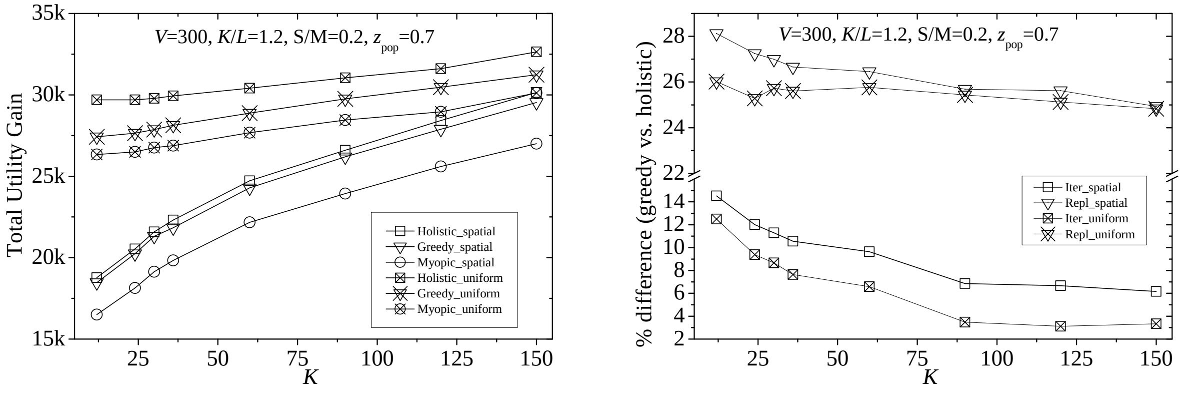
For comparison reasons we depict the performance of the proposed algorithms both with the spatial locality workload



(a) Total utility gain

(b) Iterations and object exchanges relative gain

Figure 3: The performance of the cache management algorithms vs. the fraction (S/M) of the objects that can be stored in each one of the domains.



(a) Total utility gain

(b) Iterations and object exchanges relative gain

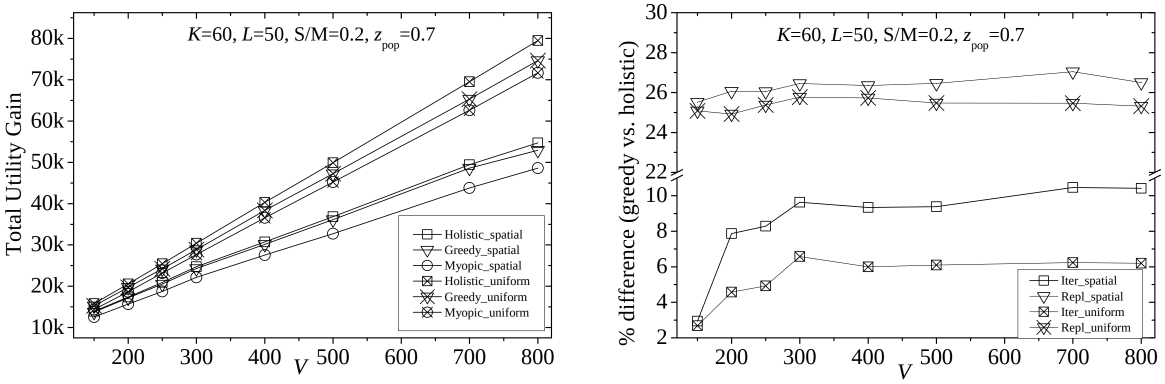
Figure 4: The performance of the cache management algorithms vs. the number of domains $\mathcal{K}$.

(noted as *spatial* in the figures), as well as when the VNets follow the same popularity distribution and the same object ranking (noted as *uniform* in the figures). The reason is that the popularity of each object may differ between different virtual networks, a phenomenon that is referred to as locality of interest (*spatial locality* in [20]). In our experiments, the workload is tuned from a localized subscription model, where at each virtual network (VNet) the popularity of the requests follow the same Zipf law distribution of exponent z_{pop}. Nevertheless, the ranking of the objects within this distribution is different among the virtual networks. This means that an object $i \in \mathcal{M}$ that is the most popular object in some VNet might not be the most popular in a different VNet, where another object, may be the most popular. Thus in our evaluation model all objects follow the same z_{pop} popularity distribution in the various VNets, but with different ranking.

In our system model we use a generic mathematical formulation where the request distribution for every VNet depends on the steady state probability of users/subscribers using a specific access domain. Due to page size limitations, we examine the performance of the proposed schemes, in the case where a user is served by a single access domain (so depending on the VNet j, $\pi_j^l = 1$ for a single l and zero elsewhere). More specifically, in order to assign the VNets to the access domains we assume a square area of 300 distance units, where the access domains are uniformly deployed. Each access domain $l \in \mathcal{L}$ covers a circle area of 50 distance units. This means that every VNet that is deployed within this area can be assigned to access domain l. Each VNet j can communicate with a subset of access domains (at least one), and randomly chooses one of them to be assigned. We also assume that each access domain $l \in \mathcal{L}$ can use a random number of other domains to connect with the data center. Finally, we assume that the utility $u_{i,j}$ that a user of VNet j enjoys for retrieving object i from domain k, as well as the the cost c_i^k of caching object i in domain k varies from $0-10$ cost units.

For each proposed algorithm the following performance metrics are used to describe the algorithm's performance: 1) the *Total Utility Gain* at the stationary point and 2) the *percentage difference* of the algorithms regarding
a) the number of iterations $= \frac{(\text{Iter-grd} - \text{Iter-hol})}{\text{Iter-grd}}$
b) the number of replacements $= \frac{(\text{Repl-grd} - \text{Repl-hol})}{\text{Repl-grd}}$.

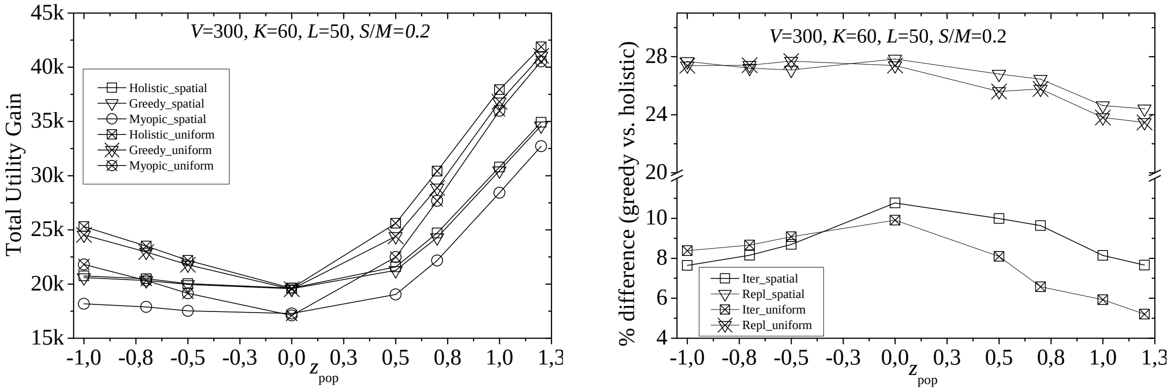
The number of iterations is indicative of the difference of the algorithms regarding their running time. Regarding the holistic algorithm, the number of object replacements is the number of object fetches (from the data center) that have to be performed once the algorithm has converged, i.e. how many objects have to be replaced in the caches compared to the initial cache assignment, whereas the greedy and the myopic algorithms always start from an empty cache and



(a) Total utility gain

(b) Iterations and object exchanges relative gain

Figure 5: The performance of the cache management algorithms vs. the number of virtual networks $\mathcal{V}$.



(a) Total utility gain

(b) Iterations and object exchanges relative gain

Figure 6: The performance of the cache management algorithms vs. the popularity exponent z_{pop}.

have to fetch every object from the data center. Since the holistic algorithm assumes an initial cache assignment in the caches of the domains, each point of the following figures is the mean value out of 100 executions starting from different initial cache assignments.

Impact of domain's caching capacity: In Fig. 3 we depict the impact of the cache capacity, expressed as the fraction of the objects that can be stored at each domain $k \in \mathcal{K}$, on the performance of the examined algorithms. Regarding the Total Utility Gain we observe that the the holistic algorithm performs $\approx 2\%$ better than the greedy one when the workload with the spatial locality is used, and $\approx 5\%$ better when uniform popularity is assumed. The holistic algorithm performs also better than the myopic algorithm regarding the utility gain $8\% - 15\%$. From Fig. 3 we also observe a linear increase regarding their difference in the number of iterations and object fetches. As we relax the storage capacity constraint and allow more objects to fit in the cache of each domain, the holistic algorithm only needs to make small adjustments in the caches to maximize the utility gain, whereas the greedy algorithm every time starts with an empty cache and its complexity is strongly coupled with the size of the caches. In more details, in the holistic algorithm the number of object replacements/fetches per domain decreases almost linearly as the capacity of the caches increase, since the availability of more cache slots enables more objects to be stored and hence less replacements are required to reach the selected assignment. Only for very small sizes of the caching capacity of each domain the greedy algorithm requires less iterations than the holistic, which implies that terminates faster, but at any case the holistic requires always less object fetches (less traffic in the system).

Impact of the number of domains: In Fig. 4 we depict the impact of the number of the domains $\mathcal{K}$, on the performance of the examined algorithms. We assume that the access domains $\mathcal{L}$ are equal to $|\mathcal{L}| = |\mathcal{K}|/1.2$. Regarding the utility gain we observe an almost linear increase as we increase the number of domains, since more domains means that each VNet has more choices of access domains to be assigned to, which further implies that the total load could be distributed more balanced between the access domains. Also, more domains means more storage capacity between the VNets and the data center. As in the previous figure the holistic algorithm requires $\approx 25\%$ less object fetches than the greedy algorithm and $3\% - 15\%$ less iterations regardless the used workload setup (spatial or uniform).

Impact of the number of VNets: In Fig. 5 we depict the impact of the number of VNets V in the performance of the examined algorithms. We notice that the total utility gain metric increases linearly with the number of VNets. This means that the algorithms are not affected by the number of the VNets in the system and they manage to accommodate the extra load within the domains, without having to request more objects from the data center. In more details, the holistic algorithm performs $1.5\% - 5\%$ better than the greedy algorithm depending on the used workload setup, and requires almost 10% less iterations and $\approx 28\%$ less object fetches from the data center. This further implies that even if the usage of the holistic algorithm only provides marginal differences in the utility gain compared to the greedy algorithm, its execution converges faster and produces less overhead cost/traffic (object fetches).

Impact of the Zipf's exponents value: In Fig. 6 we investigate the performance of the algorithms as the popularity (exponent z_{pop}) of the request pattern at each VNet change. As with the previous figures we observe that the newly proposed holistic algorithm performs slightly better than the greedy, but requires almost 10% less iterations and approximately 25% less object fetches at the stationary point.

6. CONCLUSIONS AND FUTURE WORK

In this work we addressed the emerging content replication problem a CDN provider needs to consider, when virtual network operators utilize both the CDN services and end-to-end virtualized infrastructures. We presented a model where the utility (i.e., value) enjoyed by the users of the CDN provider may be different per virtual network operator for the same requests, while our model takes into account the data placement cost to every domain, the user mobility and user's ability to use multiple access networks, besides physical storage limitations. The heuristic policies proposed scale well for various system parameters. More complex business relationships between the various players (physical infrastructure providers, CDN providers and virtual network operators) in more complex network architectures with multiple CDN providers will be investigated in the future. Moreover, content migration cost considerations and thorough investigation of the utility function definition to reflect real market conditions will also be part of our future research, as well as cases of content delivery failure that can be caused by the convergence of multiple domains.

Acknowledgements

This work has been supported by the EU Project n. 318514 “Convergence of wireless optical network and IT resources in support of cloud services” (CONTENT).

7. REFERENCES

[1] S. Singh, H.S. Dhillon, and J.G. Andrews. Offloading in heterogeneous networks: Modeling, analysis, and design insights. *Wireless Communications, IEEE Transactions on*, 12(5):2484–2497, 2013.

[2] Onapp cdn. `http://onapp.com/`.

[3] Lusheng Wang and G.-S.G.S. Kuo. Mathematical modeling for network selection in heterogeneous wireless networks; a tutorial. *Communications Surveys Tutorials, IEEE*, 15(1):271–292, 2013.

[4] Internetq. `http://www.internetq.com/`.

[5] K. Katsalis et al. Content project: Considerations towards a cloud-based internetworking paradigm. In *Future Networks and Services (SDN4FNS), 2013 IEEE SDN for*, pages 1–7, 2013.

[6] Ankur Singla and Bruno Rijsman. Contrail architecture. `http://www.juniper.net/us/en/local/pdf/whitepapers/2000535-en.pdf`, 2013.

[7] Fan Bai and Ahmed Helmy. A survey of mobility models. *Wireless Adhoc Networks. University of Southern California, USA*, 206, 2004.

[8] Kideok Cho, Hakyung Jung, Munyoung Lee, Diko Ko, T.T. Kwon, and Yanghee Choi. How can an isp merge with a cdn? *Communications Magazine, IEEE*, 49(10):156–162, 2011.

[9] Wenjie Jiang, Rui Zhang-Shen, Jennifer Rexford, and Mung Chiang. Cooperative content distribution and traffic engineering in an isp network. *SIGMETRICS Perform. Eval. Rev.*, 37(1):239–250, 2009.

[10] Patrick Wendell, Joe Wenjie Jiang, Michael J. Freedman, and Jennifer Rexford. Donar: decentralized server selection for cloud services. *SIGCOMM Comput. Commun. Rev.*, 41(4), 2010.

[11] Akamai cdn. `http://www.akamai.com`.

[12] M. Chiosi et al. Network functions virtualisation. an introduction, benefits, enablers, challenges & call for action. *SDN and OpenFlow World Congress, Darmstadt-Germany*, 2012.

[13] V. Sourlas, L. Gkatzikis, P. Flegkas, and L. Tassiulas. Distributed cache management in information-centric networks. *Network and Service Management, IEEE Transactions on*, 10(3):286–299, 2013.

[14] K. Katsalis, V. Sourlas, T. Korakis, and L. Tassiulas. Cloud-based content replication framework over multi-domain environments. In *International Conference on Communications (ICC),IEEE*, 2014.

[15] Lili Qiu, V.N. Padmanabhan, and G.M. Voelker. On the placement of web server replicas. In *INFOCOM 2001. Twentieth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, volume 3, pages 1587–1596, 2001.

[16] Jussi Kangasharju, James Roberts, and Keith W. Ross. Object replication strategies in content distribution networks. *Comput. Commun.*, 25(4):376–383, 2002.

[17] L. Lamport. The part-time parliament. *ACM Transactions on Computer Systems*, 16, 1998.

[18] L. Breslau, Pei Cao, Li Fan, G. Phillips, and S. Shenker. Web caching and zipf-like distributions: evidence and implications. In *INFOCOM '99. Eighteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, volume 1, pages 126–134, 1999.

[19] Lada A. Adamic and Bernardo A. Huberman. Zipf's Law and the Internet. *Glottometrics*, 3:143–150, 2002.

[20] K.V. Katsaros, G. Xylomenos, and G.C. Polyzos. Globetraff: A traffic workload generator for the performance evaluation of future internet architectures. In *New Technologies, Mobility and Security (NTMS), 2012 5th International Conference on*, pages 1–5, 2012.