Proactive Multi-tenant Cache Management for Virtualized ISP Networks

Maxim Claeys*, Daphne Tuncer†, Jeroen Famaey*, Marinos Charalambides†,
Steven Latré‡, George Pavlou† and Filip De Turck*
*Department of Information Technology, Ghent University - iMinds, Belgium
Email: maxim.claeys@intec.ugent.be
†Department of Electronic & Electrical Engineering, University College London, United Kingdom
‡Department of Mathematics and Computer Science, University of Antwerp - iMinds, Belgium

***Abstract*—The content delivery market has mainly been dominated by large Content Delivery Networks (CDNs) such as Akamai and Limelight. However, CDN traffic exerts a lot of pressure on Internet Service Provider (ISP) networks. Recently, ISPs have begun deploying so-called Telco CDNs, which have many advantages, such as reduced ISP network bandwidth utilization and improved Quality of Service (QoS) by bringing content closer to the end-user. Virtualization of storage and networking resources can enable the ISP to simultaneously lease its Telco CDN infrastructure to multiple third parties, opening up new business models and revenue streams. In this paper, we propose a proactive cache management system for ISP-operated multi-tenant Telco CDNs. The associated algorithm optimizes content placement and server selection across tenants and users, based on predicted content popularity and the geographical distribution of requests. Based on a Video-on-Demand (VoD) request trace of a leading European telecom operator, the presented algorithm is shown to reduce bandwidth usage by 17% compared to the traditional Least Recently Used (LRU) caching strategy, both inside the network and on the ingress links, while at the same time offering enhanced load balancing capabilities. Increasing the prediction accuracy is shown to have the potential to further improve bandwidth efficiency by up to 79%.**

I. Introduction

The rise of Internet-based over the top multimedia services has put immense strain on the resources of Internet Service Provider (ISP) networks. This has resulted in increasing operating costs but also in decreasing revenues of traditional traffic forwarding services. As a consequence, ISPs have started exploring alternative business models and service offerings. This has lead to the deployment of Telco Content Delivery Networks (CDNs), which allow content to be cached deep inside the ISP network. For the operators, Telco CDNs reduce bandwidth demand on their backbone infrastructure and open up new business models. For the end-users, Quality of Service (QoS) can be significantly improved, as content is stored nearby and the ISP has full control over the network infrastructure.

Currently, the only way for traditional CDNs, such as Akamai or Limelight, to bring their content to the edge of the network and reduce delivery times, is to physically place one of their servers inside the ISP network or connect it to a nearby Internet exchange point, through manually-negotiated contractual agreements. However, the advent of cloud computing and software-defined networking (SDN) technologies enable ISPs to virtualize their networks [1] and by extension, their Telco CDN infrastructures. This opens up new business models and allows ISPs to dynamically offer virtual storage and content delivery services at the edge of the network, redeeming traditional CDN and content providers from installing additional hardware.

Previous work in the literature has proposed ISP-operated content delivery services [2], [3] and has also investigated various content management strategies based on the deployment of distributed storage within an ISP network, e.g. [4]. Furthermore, the approach proposed by co-authors of this paper in [5] involves operating a limited capacity CDN service within ISP networks. Lightweight content placement strategies were used to show that the proposed in-network caching functionality can enable ISPs to have better control over the utilization of their network resources. These solutions, however, assume that the Telco CDN infrastructure is only used by a single entity, which poses significant limitations in a realistic scenario. To address this important shortcoming, we propose a management framework for an ISP-operated content delivery infrastructure that supports multiple tenants simultaneously leasing part of the available storage resources. The tenants specify the amount of resources they want to lease, while the management framework optimizes bandwidth utilization by intelligently placing the content of each tenant based on popularity and the geographical distribution of requests.

The main contributions of this paper are as follows. We formally model the multi-tenant content placement and server selection problems by means of an Integer Linear Program (ILP) formulation. The developed algorithm based on this model, determines where to store which content item of each tenant (content placement) and from which location to satisfy each request (server selection), with the objective of minimizing bandwidth consumption in the network while maximizing the cache hit ratio. To compute a new configuration, the model requires predicted values concerning content popularity and the geographical distribution of requests, for which we employ a prediction strategy. An extensive performance evaluation of the proposed algorithm is presented, which is compared to that of the traditional Least Recently Used (LRU) caching strategy. The analysis of the influence of the practical prediction strategy on the performance is also considered. In addition, given that the proposed solution actively pushes content onto specific servers, the trade-off between optimality and content migration overhead is investigated. To the best of the authors' knowledge, this paper is the first to quantitatively evaluate the effects of

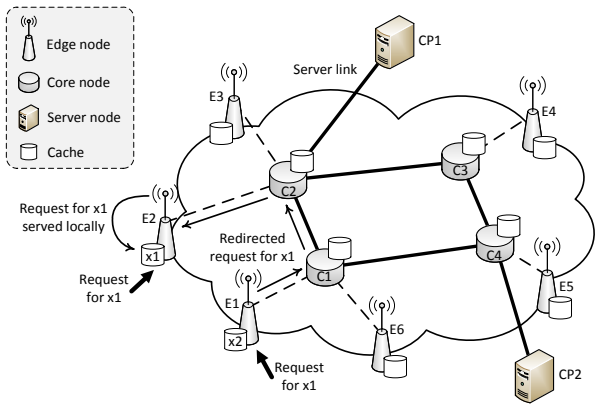


Fig. 1. Overview of the proposed ISP-based multi-tenant caching infrastructure.

migration overhead on the network resources.

In previous work [6], an algorithm was presented to determine the theoretical optimum for the content placement and server selection policy at every moment in time. This was used to determine the maximum theoretical performance gain achievable by a proactive approach. However, this theoretical optimum is associated with a prohibitive migration overhead, which makes the application of such an approach infeasible in practice. The algorithm proposed here overcomes these limitations. It reduces the migration overhead by executing the reconfiguration less frequently (at fixed time intervals, e.g. every 12 hours) and employs predictions to estimate content popularity and the geographical distribution of requests for the next provisioning period.

The remainder of this paper is structured as follows. Section II describes the scenario under study and introduces the proposed management architecture. Subsequently, the problem is formally modelled as an ILP in Section III and an algorithm based on this model is proposed. Next, details on the use case, considered in this paper, are presented in Section IV. Section V evaluates the performance of the proposed algorithm and compares it to the traditional LRU caching strategy. Section VI highlights related work in the area of content placement in distributed storage infrastructures. Finally, the main conclusions are presented in Section VII.

II. Scenario Description

We consider a scenario where a large-scale ISP operates a limited capacity CDN service by deploying caching points within its network, as depicted in Fig. 1. The set of network nodes consists of edge nodes, which represent access networks connecting multiple users in the same region, and core nodes, which interconnect the access networks. Each network node is associated with caching capabilities, which enable a set of content items to be stored locally. The local caches can be external storage modules attached to routers or, with the advent of flash drive technology, integrated within routers.

All content requests are received at the edge nodes. If a requested content item is available in the local cache of the corresponding edge node, it is served locally. Otherwise, the request is redirected to one of the caches in the network where the requested item is stored. In case a copy of the item is not available in the ISP network, the request is served from outside of the network by a content provider (CP). As depicted in Fig. 1, the request for content $x1$ received at edge node $E2$ is served locally, whereas the request for content $x1$ received at edge node $E1$ is redirected to and served by node $E2$. In line with previous related research by Applegate et al. [7], we assume shortest path routing is used, which has been shown to be more realistic than arbitrary routing [8].

In our scenario, the ISP leases the caching space in its network to multiple CPs. Each CP specifies the amount of caching capacity it wishes to lease for storing part of its content catalog. The optimal placement of content, in terms of network resource utilization, depends on the geographical distribution of requests, the content popularity and the network topology. Based on the availability of such information to the ISP, we propose a novel proactive multi-tenant cache management approach where the ISP controls the partitioning of available storage space between multiple CPs. More specifically, the proposed solution focuses on the following management decisions: (i) where to allocate the leased capacity, (ii) where to store which content (content placement) and, (iii) from where to serve user requests (server selection), with the objective of minimizing ISP network resource utilization, while simultaneously maximizing the total cache hit ratio (i.e., reducing the number of requests that have to be served from outside the network). New caching configurations are periodically computed by a central manager, based on predicted values of content popularity and geographical distribution of requests for the next provisioning period.

III. ILP Formulation

To model the considered problem as an ILP, the network is represented by a directed graph $G = (V, E)$ with V and E representing the set of nodes and links, respectively. The set of nodes contains both the nodes V_{ISP}, belonging to the ISP, and an external server node S, logically representing the Internet, containing all contents of all providers. V_{ISP} can further be divided in a set of core nodes V_C and edge nodes V_E. The links E can be divided into a set of links, E_S, connected to the external server (i.e. the ingress links), and, ISP-managed links, E_{ISP}, connecting core and edge nodes. For each node $n \in V$, we define a caching capacity $c_n \in \mathbb{N}^+$ and a set of incoming and outgoing links, denoted by $I_n \in E$ and $O_n \in E$, respectively. For every link $e \in E$, the available bandwidth capacity is denoted as $b_e \in \mathbb{N}^+$. The routing strategy, applied in the network is represented by a forwarding path $R_{n,n'} \subset E$, for every source-destination pair $(n, n') \in V \times V$. The forwarding path can be divided into a set of server links $R^S_{n,n'} \subset E_S$, containing the links in the forwarding path connected to the external server node S, and a set $R^{ISP}_{n,n'} \subset E_{ISP}$ containing the other links in the forwarding path, inside the ISP network.

A set of content providers P lease caching space from the ISP. For each content provider $p \in P$, the leased amount of caching space and the set of offered content items are denoted by $d_p \in \mathbb{N}^+$ and O^p, respectively. $O = \bigcup_{p \in P} O^p$ represents the entire set of offered content. Every content item $o \in O$ has an associated size $s_o \in \mathbb{N}^+$ and bitrate $B_o \in \mathbb{N}^+$.

The objective of the proposed approach is to periodically compute a new caching configuration based on the estimation

of content popularity and geographical distribution of requests for the next provisioning interval. As such, a prediction of the request pattern for the considered time interval is required by the algorithm at each reconfiguration step to determine a new content placement and server selection strategy. We note T the finite set of n timepoints ($|T| = n$) at which a request is predicted to start or to finish. For every timepoint $t \in T$, $r_{o,d,t} \in \mathbb{N}$ denotes the active number of requests for content $o \in O$, originating from edge node $d \in V_E$. $V_t^o \subset V_E$ represents the set of edge nodes requesting content $o \in O$ at timepoint $t \in T$. The *validity period* of timepoint $t_i \in T$ is defined as $(t_{i+1} - t_i)$ and is denoted as δ_{t_i}. The validity period of the last timepoint $t_n \in T$ is defined to be 1 ($\delta_{t_n} = 1$).

A solution to the content placement problem can be translated into binary decision variables $x_{n,o} \in \{0, 1\}$ defining if an ISP node $n \in V_{ISP}$ is used to store content $o \in O$. In addition, auxiliary decision variables $z_{n,o,d} \in \{0, 1\}$ are introduced to represent the server selection strategy. These define if a node $n \in V_{ISP}$ is used to store content $o \in O$ to be delivered to edge node $d \in V_E$. In order to determine a valid solution, the auxiliary variables satisfy the constraints $z_{n,o,d} \leq x_{n,o}$ and $x_{n,o} \leq \sum_{d \in V_E} z_{n,o,d}$ (content o is stored at node n if and only if at least one edge node d requests o from n).

Different optimization criteria, such as cache hit ratio maximization or delivery delay minimization, have been considered in the literature [7], [9], [10]. In this paper, we focus on reducing the ISP resource usage. As such, we define the optimal solution to the problem as the one minimizing the bandwidth usage inside the ISP network. This can be represented by the minimization of the objective function defined in (1). A weighting factor $\alpha \in [0; 1]$ is used to define the importance of ingress link usage. Higher values of α will result in minimizing the usage of ingress link bandwidth, leading to a higher cache hit ratio.

$$\sum_{\substack{t \in T, n \in V, \\ o \in O, d \in V_t^o}} \left(\sum_{e \in R_{n,d}^S} \alpha + \sum_{e \in R_{n,d}^{ISP}} (1 - \alpha) \right) \times \delta_t \times r_{o,d,t} \times B_o \times z_{n,o,d} \tag{1}$$

Multiple constraints are considered to define the set of valid solutions to the considered optimization problem. A valid solution is so that the caching space reserved for each content provider $p \in P$ is at most equal to the leased capacity, while satisfying the storage capacity limitations. These constraints are modelled in (2) and (3), respectively.

$$\forall p \in P : \quad \sum_{n \in V_{ISP}} \sum_{o \in O^p} s_o \times x_{n,o} \leq d_p \tag{2}$$

$$\forall n \in V_{ISP} : \quad \sum_{o \in O} s_o \times x_{n,o} \leq c_n \tag{3}$$

Furthermore, the bandwidth limitations should be enforced at any point in time of the considered period. To simplify the formulation of this constraint, an additional notation U_e is introduced for every link $e \in E$ to represent the set of source-destination pairs $(s, d) \in V \times V$ which are routed over link e. The bandwidth limitation constraint is then defined as shown in (4).

$$\forall t \in T, \forall e \in E : \sum_{(s,d) \in U_e} \sum_{o \in O} r_{o,d,t} \times B_o \times z_{s,o,d} \leq b_e \tag{4}$$

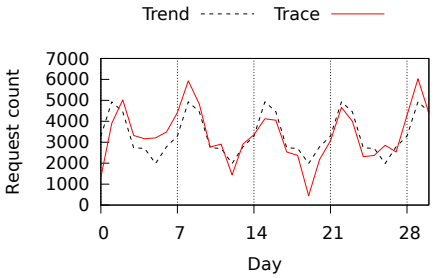


Fig. 2. Number of requests per day in the considered VoD trace.

Finally, constraint (5) ensures that every request is served from exactly one location.

$$\forall o \in O, \forall d \in V_E : \sum_{n \in V} z_{n,o,d} = 1 \tag{5}$$

Solving the ILP using *IBM ILOG CPLEX Optimization Studio 12.4*, results in a storage profile, represented by the values of $x_{n,o}$, and a server selection strategy, represented by the values of $z_{n,o,d}$, which minimizes the objective function in (1), while satisfying constraints (2) – (5). Every request from edge node $d \in V_E$ for content $o \in O$ is served from node $n \in V$ where $z_{n,o,d} = 1$, using the shortest path $R_{n,d}$.

IV. Use case

We evaluate the proposed approach based on a Video-on-Demand (VoD) use case, for which we used a request trace of the VoD service of a leading European telecom operator. We first discuss the characteristics of this trace and then describe the request prediction method we use.

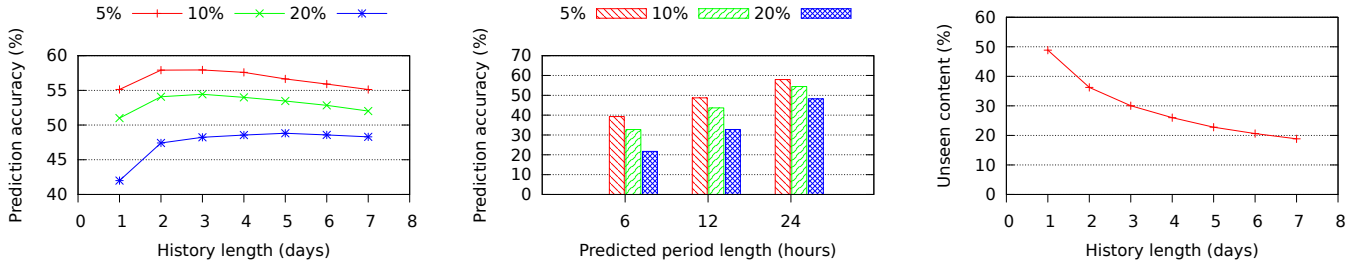
A. VoD trace characteristics

The trace was collected over a period of 31 days between Saturday February 6, 2010 and Sunday March 7, 2010. During the considered period, 104,217 requests for 5644 unique movies were monitored, sent by 8825 unique users, originating from 12 cities. All movies are considered to have an equal length of 90 minutes and a bitrate of 1Mbit/s ($b_o = 1 \; \forall o \in O$). Each movie thus has a size of 5.4Gbit ($s_o = 5400 \; \forall o \in O$) and is requested by the user in segments of 1 second each.

Fig. 2 depicts the evolution of the number of requests per day over the considered time period. As can be observed, this exhibits a weekly pattern. The five peaks in Fig. 2 correspond to the five weekends, with increased activity on Friday, Saturday and Sunday. In addition to the weekly pattern, a diurnal trend also exists, which is not visible in Fig. 2 due to the per-day data aggregation used. For Wednesdays and Sundays, the activity peak is between 4:30pm and 6:30pm. For the other days of the week, the largest number of requests is reported between 8pm and 10pm.

B. Request prediction

As discussed in Section III, the ILP requires the request rates $r_{o,d,t}$ for a content $o \in O$, originating from edge node $d \in V_E$ at a future time point $t \in T$ as input variables. In this paper, we use a simple request prediction method,



(a) Average prediction accuracy as a function of the history length.

(b) Average prediction accuracy with a history length of 3 days.

(c) Average percentage of unseen content as a function of the history length.

Fig. 3. Analysis of the prediction accuracy for the VoD trace.

which is based on the observed characteristics of the VoD trace (cf. Section IV-A), to estimate the future request pattern. While more sophisticated solutions can be used [7], popularity prediction is a complex issue and a more elaborate mechanism is therefore outside the scope of this paper. The influence of the prediction accuracy on optimality is discussed in Section V-C.

The request pattern contains various types of information: (i) the request intensity over time, i.e. the total number of requests in the network for every point in time, (ii) the geographical distribution of this request intensity and (iii) the content popularity, i.e. the distribution of requests amongst all contents. As explained in Section IV-A, weekly patterns can be identified for the request intensity in the considered VoD use case. In order to predict the request pattern for a specific time period, the request intensity and its geographical distribution over the same period of the previous week are used. However, given the highly dynamic nature of video popularity, such an approach cannot be used to predict content popularity. Although some request intensity trends can be identified for each day of the week, it is more likely that the change in popularity of a given content item will be more significant over a week than between two consecutive days. As such, we use the content popularity of the x previous days to predict the popularity of specific content items.

To estimate the prediction accuracy, we defined an accuracy metric. When the total amount of leased capacity is equal to the combined size of y videos, the videos that are more likely to be cached for a specific day are the y most popular videos of that day. We define the accuracy of a prediction for a specific day, based on the x previous days, to be the percentage of the y most popular content items of the previous x days that are still in the top y most popular content items during the considered time period. For example, when we consider a history of the 2 previous days and a total amount of leased caching capacity of 100 videos, a prediction accuracy of 50% means that 50 of the 100 most popular videos of the 2 previous days are in the top 100 most popular videos of the considered time period. Fig. 3a shows the average prediction accuracy for the VoD trace in terms of the history length for the scenario where the content providers lease a caching capacity equal to 5%, 10% or 20% of their total content catalogue. Fig. 3a shows that on average, the highest prediction accuracy can be achieved when using 3 days of history. Furthermore, increasing leased capacity leads to lower accuracy, which indicates that the more requested content items are easier to predict than less popular ones.

Fig. 3b shows the prediction accuracy, using a history length of 3 days, in terms of the length of the considered period. It can be observed that the prediction accuracy decreases when shorter time periods are considered: the content popularity is easier to predict for an entire day than for a short time period. This is even more distinct when the leased capacity increases. Finally, Fig. 3c shows the limitation of pure history-based prediction. In the VoD use-case, the number of daily new content items is significant. On average, 30% of all content requested on a given day has not been requested in the previous 3 days. Requests for these content items are therefore impossible to predict based on history only. Techniques that predict the popularity of new content, based on additional information, have been proposed in the literature [7], but are out of the scope of this research.

V. Evaluation Results

In this section, we thoroughly evaluate the proposed ILP approach. First, the evaluation setup is described in Section V-A. Next, a parameter analysis is performed in Section V-B to find the preferred parameter configuration for the ISP. The performance of the proposed proactive approach using this preferred configuration is compared to a reactive approach in Section V-C. Finally, the influence of the number of tenants is discussed in Section V-D.

A. Evaluation setup

We evaluate the performance of the proposed ILP-based algorithm using the GÉANT topology[1], which consists of 23 nodes. The employed VoD request trace contains 12 cities that we map onto 12 edge nodes ($V_E = \{E1, .., E12\}$). One node is assigned as server node S, while the 10 remaining nodes are modelled as core nodes ($V_C = \{C1, .., C10\}$). The 10 most connected nodes were selected as core nodes. The resulting topology is shown in Fig. 4. The links interconnecting core nodes and the links connected to the server have a bandwidth capacity of 1Gbit/s. All other links have a capacity of 500Mbit/s. Since fixed shortest path routing is used, the exact bandwidth capacities are of minor importance, as long as they suffice to serve the requests. The server node S hosts all the available content of all CPs. In each experiment, the storage capacity of each core node was set high enough to be able to accommodate the leased capacity of all tenants, i.e. for every core node $n \in V_C$, the caching capacity was

[1]GÉANT Project – http://www.geant.net

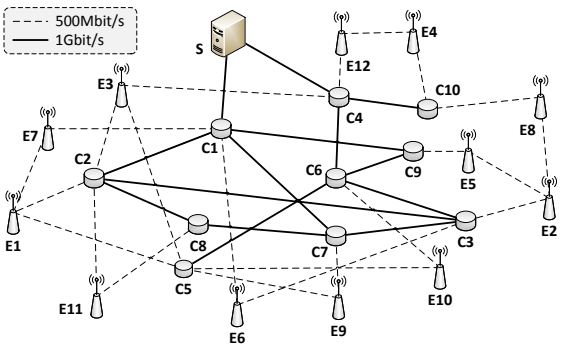


Fig. 4. Evaluated scenario based on the GÉANT topology

defined as $c_n = \sum_{p \in P} d_p$. Preliminary simulations have been performed using other capacities for the core nodes, but no significant influence could be observed. The storage capacity of the edge nodes and the amount of leased storage space is varied throughout the experiments. Unless otherwise stated, the number of tenants is equal to two.

The performance of the proposed proactive approach is compared to the one of a reactive approach following the commonly used LRU caching strategy in the VoD use case described in Section IV. For the reactive approach, the leased capacity of every content provider is uniformly split over the 22 ISP managed nodes. All requests are sent to the origin server, applying reactive caching for future requests. Each content item is considered to consist of fixed duration segments (i.e., 1 second each), which is often the case in modern streaming technologies (e.g., Apple HLS, MPEG DASH). As a consequence, the reactive approach can result in a configuration where only parts of a movie are available at a given node. In contrast, the proactive approach either places an entire movie at a specific node, or does not store it there at all. Shortest path routing, based on hop count, is used in both approaches.

We performed experiments for all of the 31 days in the VoD trace, while only evaluating the last 24 days (from February 13, 2010 to March 7, 2010). The first 7 days of the trace were used for obtaining the request prediction for the ILP approach, but also for *warming up* the LRU caches.

B. Parameter analysis

Using the proactive content placement approach, described in Section III, the ISP can decide on multiple design parameters to optimize the performance of the cache management. The server link weight α in objective function (1) can influence the placement decisions, based on the cost of using the ingress link, while the frequency of reconfiguration defines the trade-off between the optimality of the decisions and the management overhead. In addition, given that a high cost can be associated with the provisioning of caching capacity at the network edge, it is important to evaluate the influence of the edge node capacity on the performance of the system. To determine the impact of each of these decisions, we analyzed a wide range of possible parameter configurations, as presented in Table I. The results of this analysis are subsequently used to define the preferred configuration for the ISP.

TABLE I. EVALUATED PARAMETER CONFIGURATIONS

Parameter	Values
α	0.50, 0.75, 0.90
Reconfiguration frequency	6h, 12h, 24h
Edge node capacity*	1%, 5%, 10%, 20%, 50%

*Relative to the core node capacity.

To assess the influence of the request prediction strategy, we evaluated the ILP approach, using both the request prediction algorithm described in Section IV-B, as well as the (theoretical) perfect prediction where the actual trace data is used. For completeness, the comparison between these approaches is always incorporated in the presented graphs. A performance comparison between the reactive LRU approach and the ILP approach using both prediction strategies is presented in Section V-C. The performance is evaluated in terms of (i) the peak bandwidth usage of the links, i.e. the maximum bandwidth usage on every link during the evaluated period, (ii) the average bandwidth usage inside the entire ISP network, (iii) the average bandwidth usage on the ingress links, (iv) the bandwidth required to migrate the content at each reconfiguration period and (v) the time required by the ILP-based algorithm to calculate a solution. In order to analyze the influence of a specific parameter, average performance measures have been calculated over all configurations that have the same value for that parameter (e.g., all configurations that include $\alpha = 0.5$).

1) Server link weight α: The server link weight α in objective function (1) defines the trade-off between minimizing the bandwidth usage inside the ISP network and minimizing the ingress link usage. A value of $\alpha = 0.50$ indicates that the same weight is applied to both objectives. As the value of α increases, more weight is applied by the system to the minimization of the ingress link usage, which results in the maximization of the cache hit ratio. This can be observed in Fig. 5, which shows the influence of the server link weight α on the bandwidth usage of the ingress links and of the links inside the ISP network. Lower values of α lead to the placement of more popular content items at multiple locations (i.e. higher replication degree), while less popular items have to be fetched from the origin server. This results in shorter delivery paths for most requests and thus lower bandwidth usage inside the ISP network, at the expense of ingress link usage given that a smaller number of distinct contents can be cached. In contrast, with higher values of α, the minimization of the ingress link usage is given more weight. This forces the system to place a larger number of contents inside the ISP network, but with a lower replication degree. While this leads to higher bandwidth usage inside the ISP network (the content is further away from the users), it reduces the ingress link usage. Similar observations can be made for the peak bandwidth usage inside the ISP network and on the ingress links. Furthermore, the results show that, for a value $\alpha = 0.90$, the average overhead incurred by the migration of the content at each reconfiguration interval is 4.75% higher compared to the case with $\alpha = 0.50$ (graph omitted due to space limitations). More specifically, in this case, the algorithm decides to cache a larger number of unique contents inside the ISP network, which have to be migrated from the origin server. Finally, it was observed that the value of α does not significantly affect the time required by the

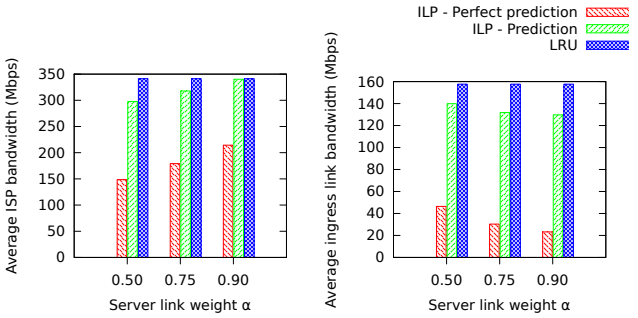


Fig. 5. Influence of the server link weight α on the average bandwidth usage

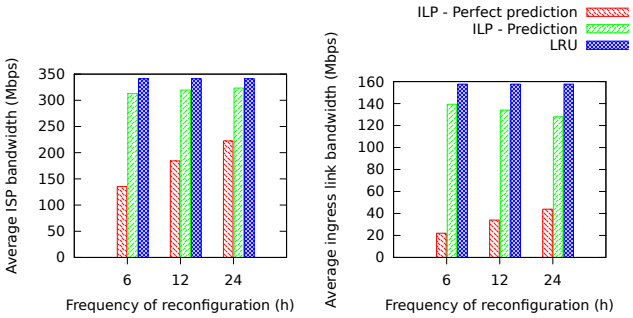


Fig. 6. Influence of the reconfiguration frequency on the average bandwidth usage.

algorithm to compute a feasible solution of the ILP model.

Considering this trade-off, a value of $\alpha = 0.50$ is preferred. The average total bandwidth usage (i.e., the sum of the bandwidth usage inside the ISP network and on the ingress links) is 6.90% lower compared to $\alpha = 0.90$, while having 4.53% less migration overhead.

2) Frequency of reconfiguration: The frequency at which the proactive algorithm computes new content placement defines the trade-off between optimality and overhead. While more frequent reconfigurations allow the system to be more reactive with respect to changes in the request pattern, this comes at the cost of more frequent content migrations. Fig. 6 shows the influence of the frequency of the reconfiguration on the average bandwidth usage inside the ISP network and the ingress links. As can be observed in the figure, less frequent reconfigurations lead to a performance degradation in the case of the perfect request prediction. This, however, is not the case when using our request prediction strategy. Although the average bandwidth usage inside the ISP network slightly increases when the frequency of reconfiguration decreases, the increase is limited to 3.44% when the reconfiguration period goes from 6h to 24h. In contrast, the average bandwidth usage on the ingress links decreases by 8.25% when the reconfiguration period increases from 6h to 24h. This counterintuitive result can be explained by the characteristics of the VoD trace and our prediction strategy, discussed in Section IV. As shown in Fig. 3b, content popularity is easier to predict for longer time periods. In the case of very frequent reconfigurations, the low accuracy of the prediction means that a significant number of items has to be fetched from the origin server, which results in high ingress link bandwidth usage. Similar observations were made for the peak link usage inside the ISP network and on the ingress links (graphs omitted due to space limitations).

Given that the number of items and requests considered in the ILP model directly depends on the frequency of the reconfiguration, this can influence the time needed to solve it. The values of the average and maximum solving time obtained for the different reconfiguration frequencies are reported in Table II. Although the time required to find a solution increases when the reconfiguration frequency decreases, the complexity can be considered acceptable with respect to the length of the provisioning period. For instance, in the case of a reconfiguration period of 24h, the ILP has to be solved once every 24h only. In terms of the number of contents that cross a single link during content migration on a daily basis, Fig. 7 shows that the overhead is 67.33% lower when using a reconfiguration

TABLE II. ILP SOLVING DURATION IN TERMS OF THE RECONFIGURATION PERIOD.

Reconfiguration period	Average solving duration	Maximum solving duration
6h	4min 34s	21min 51s
12h	10min 17s	34min 35s
24h	23min 51s	51min 55s

period of 24h compared to the one of 6h. Based on the above findings, a reconfiguration every 24h is preferred.

3) Edge node capacity: The deployment of caching capacity at the edge nodes allows the content to be placed close to the end users, resulting in shorter delivery paths and lower bandwidth usage. However, to be able to provide caching capacity at the edge of the network, storage nodes need to be deployed at distributed street cabinets. This significantly increases deployment and maintenance costs compared to providing storage in centralized data centers in the core of the ISP network. As shown in Fig. 8, low edge node capacities result to increasing bandwidth usage inside the ISP network. This is caused by longer delivery paths since contents are cached further away from the end users. However, since content is more likely to be cached in the network core, the replication degree is likely to be lower. Therefore, more distinct contents can be cached inside the network, which reduces the average ingress link usage. Similar observations were made for the peak bandwidth usage (graphs omitted due to space limitations). Lower edge node capacities lead to slightly higher peak bandwidth usages inside the ISP network (1.65% higher with an edge node capacity of 1% compared to an edge node capacity of 50%), while the peak bandwidth usage of

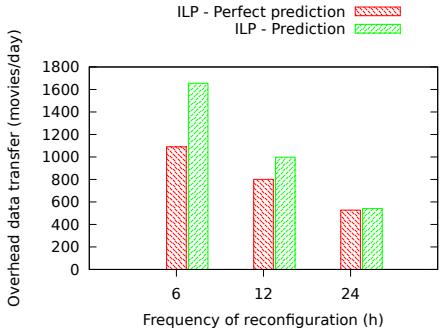


Fig. 7. Influence of the reconfiguration frequency on the content migration overhead.

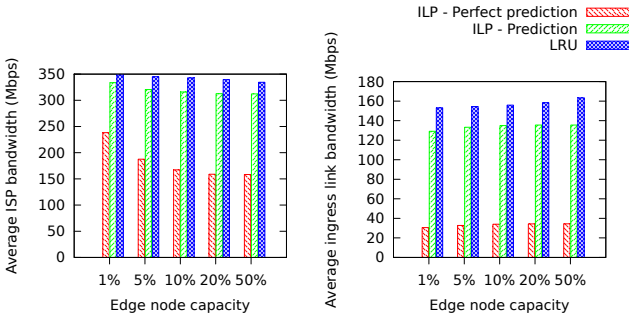


Fig. 8. Influence of the edge node capacity on the average bandwidth usage.

TABLE III. PERFORMANCE RESULTS FOR MULTIPLE AMOUNTS OF LEASED CAPACITY USING THE SELECTED PARAMETER CONFIGURATION.

	5%‡		10%‡		20%‡	
Criteria	**LRU**	**ILP***	**LRU**	**ILP***	**LRU**	**ILP***
Avg. ISP link usage°	360	317 (236)	344	292 (174)	323	268 (98)
Avg. ingress link usage°	175	147 (95)	157	136 (63)	135	127 (28)
Migration overhead ISP†	n/a	334 (333)	n/a	669 (677)	n/a	1333 (1370)
Migration overhead server†	n/a	113 (178)	n/a	221 (364)	n/a	420 (739)
Max. peak link usage°	832	664 (431)	710	583 (293)	572	548 (188)
Avg. peak link usage°	197	51 (38)	180	47 (29)	163	44 (19)

*The values between brackets are for the scenario with perfect prediction.
°Bandwidth in Mbps.
‡Relative to the total catalog size.
†Upper limit of migration overhead bandwidth in Mbps, needed to migrate all content in 60 minutes when all migrations would need to cross the same link.

the ingress link is reduced (5.17% lower with an edge node capacity of 1% compared to an edge node capacity of 50%). In addition, given that most of the items tend to be cached in the core nodes, the paths used for migrating the content tend to be shorter. As a result, slightly lower migration overhead can be observed in the case of low edge node capacities. It can also be observed that there is no significant performance difference between the different configurations when the capacity of the edge nodes is more than 10% of the core node capacity. Finally, the choice of the capacity of the edge nodes does not significantly influence the time required to solve the ILP (graphs omitted due to space limitations). Based on these findings, an edge node capacity equal to 10% of the core node capacity is preferred. Further increasing the edge node capacity comes at a high provisioning cost without further performance gain.

C. Performance comparison

Based on the above parameter analysis, the preferred configuration for the ISP has the following parameter values: a server link weight α of 0.5, a reconfiguration period of 24h and an edge node capacity equal to 10% of the core node capacity. The leased capacity is decided by the content providers and is expressed relative to their total catalog size. Table III shows a performance comparison, based on multiple criteria, between the reactive LRU approach and the proposed ILP approach using the preferred configuration. For the proactive ILP approach, results are shown both using a perfect prediction and the prediction strategy discussed in Section IV-B.

The results show that with the proposed ILP approach, the average total bandwidth usage inside the entire ISP network can be reduced by 12%-17%, depending on the amount of leased caching capacity. However, with perfect prediction, these reductions could be further increased by 34% up to 70%, based on the amount of leased caching capacity. The bandwidth usage on the ingress links can, on average, be reduced by 6% to 17%, depending on the amount of leased capacity. With a perfect request prediction, the performance increase could amount to 46% up to 79%.

We also investigated the impact of the overhead in terms of resource utilization of the content migration that needs to be performed at each reconfiguration interval. This is computed based on the available bandwidth required to perform all content migrations within 60 minutes. For different migration periods, the overhead bandwidth scales linearly (e.g., the overhead bandwidth doubles when the migration has to be done in 30 minutes instead of 60 minutes). However, given that some assumptions are made while calculating this overhead, the presented values represent a broad and pessimistic upper bound. For example, the calculations assume that every content migration traverses a single link in the network and that all migrations are performed in parallel. In practice, content migration can be optimized by serialization. As an example, we consider the topology in Fig. 4. If a new content item has to be migrated from the server node S to the core nodes $C7$ and $C3$, the overhead estimation assumes that the content is fetched twice from the origin server in parallel. In practice, however, it may be more efficient to first migrate the content from S to $C7$ and then duplicate the content from $C7$ to $C3$. As such, the bandwidth overhead will be significantly lower than the presented upper bound.

Given that the number of content items to be cached increases when more capacity is leased, the migration overhead scales with the leased capacity. As observed in Table III, the migration overhead inside the ISP network is similar when using a practical or perfect request prediction. However, the migration overhead on the ingress link is significantly higher when a perfect prediction can be made. Given that the predicted content popularity is based on a history of three days, the daily predicted number of new content items (i.e. which have to be migrated from the server node at each reconfiguration interval) is significantly lower than the actual number of new contents. Given that the most popular content items are more likely to remain the same over time, compared to the less popular ones, this difference further increases when a larger amount of caching capacity is leased.

The performance is also evaluated in terms of the peak link usage. The maximum peak link usage is the peak usage of the most loaded link in the network, while the average peak link usage is the peak usage averaged over all links in the network. Given that the content items that are not stored inside the ISP network have to be fetched from the origin server for every edge node, and, as such, traffic has to be routed over the ingress links, the latter will always be the most heavily loaded links in the network. Therefore, the maximum peak link usage is measured on the ingress links. As shown in Table III, the maximum peak link usage can be reduced by 4% to 20%, compared to a reactive LRU approach, depending on the amount of leased capacity. In the case of the perfect request prediction, the maximum peak usage is decreased by 48%-67%. In addition, the performance that can be achieved in terms of average peak link usage can be improved by

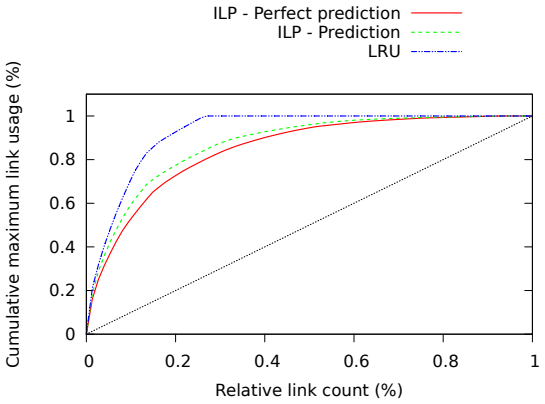


Fig. 9. Peak link load division for the reactive LRU approach and the proposed proactive ILP approach, using the selected parameter configuration and a leased capacity of 5%.

around 73%. In the case of the perfect request prediction, the performance gain is between 81% and 88%.

Finally, the results of the evaluations showed that using the proactive ILP approach can lead to more balanced link load distribution inside the ISP network. This is shown in Fig. 9, which depicts the relative cumulative maximum link usage as a function of the relative number of links, ordered by decreasing link load, for a leased capacity of 5%. The dotted diagonal line represents a uniform load distribution, corresponding to a situation where each link is equally loaded. The horizontal line $y = 1$ in the figure represents an extreme scenario where only one link is used. The graph shows that with the LRU approach only 27% of the links are used, while 95% of the links are used with the proactive ILP approach. To measure how the network load is balanced, we define the *unfairness degree* as the sum of the deviations from the cumulative maximum link usage to the uniform cumulative link usage. Graphically, this corresponds to the area between the cumulative maximum link usage curve and the dotted diagonal. The *unfairness degree* is scaled to be 0% when the load is perfectly balanced (all links are equally loaded) and to be 100% when only one link is loaded. Using the proactive ILP approach, the degree of unfairness can be reduced from 86% to 75%, compared to the reactive LRU approach. With a perfect request prediction, the degree of unfairness could be further decreased to 70%.

D. Influence of the number of tenants

To show the general applicability of the proposed approach, evaluations have been performed for multiple tenants leasing storage capacity from the ISP. For each of these evaluations, the VoD catalog of 5644 movies is uniformly split amongst the tenants. As the content characteristics are unchanged, performance deviations are only due to the multi-tenancy. Table IV shows the performance for 1 up to 5 tenants, using the preferred parameter configuration and a leased capacity of 5% of the total catalog size for every tenant. Due to space limitations, the results using perfect request prediction are omitted. The results show that the performance of the proposed ILP approach is not influenced by the number of tenants.

TABLE IV. PERFORMANCE RESULTS FOR MULTIPLE NUMBERS OF TENANTS, USING THE SELECTED PARAMETER CONFIGURATION AND A LEASED CAPACITY OF 5%.

	1		2		3		4		5	
Criteria	**LRU**	**ILP**	**LRU**	**ILP**	**LRU**	**ILP**	**LRU**	**ILP**	**LRU**	**ILP**
Avg. ISP link usage°	360	317	360	317	360	317	361	318	361	319
Avg. ingress link usage°	175	146	175	147	175	147	176	147	175	147
Migration overhead ISP†	n/a	334	n/a	334	n/a	333	n/a	333	n/a	332
Migration overhead server†	n/a	112	n/a	113	n/a	113	n/a	115	n/a	114
Max. peak link usage°	832	655	832	664	830	661	835	659	838	659
Avg. peak link usage°	196	51	197	50	197	51	197	51	198	51

°Bandwidth in Mbps.
†Upper limit of migration overhead bandwidth in Mbps, needed to migrate all content in 60 minutes when all migrations would need to cross the same link.

VI. RELATED WORK

The problem of how to allocate capacity resources to different nodes was considered by Laoutaris et al. in [11] and [12]. The authors focused on the design of algorithms for the joint optimization of capacity allocation and object placement decisions under known topological and user demand information. The objective is to determine the placement of objects selected from a set of available content items and also the proportion of the total storage capacity to be allocated at each potential caching location so that the network cost is minimized.

While the solutions proposed in [11] and [12] take into account a global capacity constraint on the total storage space available in the network, the constraint is not applied on a per-node basis. This may not be realistic in practice given that a node may not have enough capacity to accommodate all content items. In contrast, we formulate the problem by considering the capacity constraint at the node level. In addition, we extend the optimization problem by also considering from where to serve user requests. It should also be noted that Laoutaris et al. focused on hierarchical caching infrastructures, whereas this restriction does not apply in our work.

The issue of partitioning the storage space available at each caching location has been considered from a different perspective in [7] and [10]. In these works, the authors propose to pre-partition the local storage capacity according to fixed ratios in order to implement different caching strategies and investigate the effects of such a scheme on network performance. In [7], Applegate et al. present an approach to solve the combined problem of content placement and assignment of requests to caching locations for a large-scale VoD system. The problem is formulated as a mixed integer program which takes into account storage capacity and link bandwidth constraints, as well as content popularity. The authors discuss some simple strategies to estimate the popularity and determine the frequency of updates. Optimal solution structures for the combined problem have also been proposed in [13] and [14]. In contrast to our approach, however, these consider a single provider scenario only, which can be considered as a subset of the problem we investigate.

Various content placement approaches have been proposed in the context of a single provider scenario, e.g. in [4], [9], [15], [16]. These focused on intelligent techniques to replicate content across different network locations in order to better utilize network resources. A distributed content placement strategy is proposed in [9] in the context of distributed repli-

cation groups. Sourlas et al. present in [4] an autonomic cache management framework for information-centric networks. Specific placement approaches have also been considered for hierarchical network infrastructures, especially in the context of IPTV [15], [16]. Chun et al. analyzed the content placement problem and the effect of coordination among caches using a game theoretic model [17]. However, they assume caches have unlimited capacity. In parallel to the placement problem, previous research efforts (e.g. [18], [19]) have also focused on the server selection issue and have proposed new mechanisms to manage the redirection of user requests.

Given that current content delivery services can adversely affect the utilization of ISP networks, some research efforts have been investigating new models and frameworks to support the interaction between ISPs and CDNs. These range from ISP-centric caching approaches (e.g. [2], [3]), which exclude CDNs from the delivery chain, to collaborative solutions (e.g. [19], [20]), which define new models of cooperation between ISPs and CDNs in order to improve content delivery performance.

More recently, the approach developed in [5] by co-authors of this paper has proposed the operation of a limited capacity CDN service within ISP networks by deploying caches at the network edges. Such a service can allow ISPs to implement their own cache management strategy. This paper extends the framework to a multi-tenant scenario.

VII. Conclusions

In this paper, we presented a proactive cache management approach for ISP networks in a scenario where multiple content providers lease caching capacity. Based on predictions of the content popularity and the geographical distribution of requests, this approach employs an ILP-based optimization algorithm to perform content placement and server selection. The approach has been thoroughly evaluated in a VoD use-case. Evaluations have shown that this reconfiguration should ideally be performed every 24 hours, due to the trade-off between solution optimality and migration overhead. Furthermore, we have shown that using the proposed ILP-based algorithm, the average bandwidth usage can be reduced by up to 17%, both inside the ISP network and on the ingress links, compared to a reactive LRU approach. Moreover, a more balanced load distribution is achieved, reducing the average peak link usage inside the ISP network by 73%, while reducing the maximum peak link usage by 20%. However, the performance of the proposed approach strongly depends on the accuracy of the prediction of future request patterns. Results have shown that under the assumption of perfect prediction, average bandwidth usage within the ISP network can be reduced by up to 70% compared to LRU. Future work consists of focusing on more advanced prediction methods and evaluate their obtained performance.

Acknowledgment

M. Claeys is funded by a grant of the Agency for Innovation by Science and Technology in Flanders (IWT). This work was partly funded by FLAMINGO, a Network of Excellence project (318488) supported by the European Commission under its Seventh Framework Programme.

References

[1] S. Davy, J. Famaey, J. Serrat, J. L. Gorricho, A. Miron, M. Dramitinos, P. M. Neves, S. Latré, and E. Goshen, "Challenges to support edge-as-a-service," *IEEE Communications Magazine*, vol. 52, no. 1, pp. 132–139, 2014.

[2] N. Kamiyama, T. Mori, R. Kawahara, S. Harada, and H. Hasegawa, "ISP-operated CDN," in *Proc. of the IEEE INFOCOM Workshops*, 2009, pp. 49–54.

[3] K. Cho, H. Jung, M. Lee, D. Ko, T. Kwon, and Y. Choi, "How can an ISP merge with a CDN?" *IEEE Commun. Mag.*, vol. 49, no. 10, pp. 156–162, oct. 2011.

[4] V. Sourlas, P. Flegkas, L. Gkatzikis, and L. Tassiulas, "Autonomic cache management in information-centric networks," in *Proc. IEEE Network Operations and Management Symposium (NOMS'12)*, apr. 2012, pp. 121–129.

[5] D. Tuncer, M. Charalambides, R. Landa, and G. Pavlou, "More control over network resources: An ISP caching perspective," in *Network and Service Management (CNSM), 2013 9th International Conference on*, Oct 2013, pp. 26–33.

[6] M. Claeys, D. Tuncer, J. Famaey, M. Charalambides, S. Latré, F. De Turck, and G. Pavlou, "Towards multi-tenant cache management for ISP networks," in *European Conference on Networks and Communications (EUCNC)*, 2014.

[7] D. Applegate, A. Archer, V. Gopalakrishnan, S. Lee, and K. K. Ramakrishnan, "Optimal content placement for a large-scale vod system," in *Proc. Co-NEXT '10*, Philadelphia, Pennsylvania, 2010, pp. 1–12.

[8] N. Garg and J. Könemann, "Faster and simpler algorithms for multicommodity flow and other fractional packing problems," *SIAM Journal on Computing*, vol. 37, no. 2, pp. 630–652, 2007.

[9] N. Laoutaris, O. Telelis, V. Zissimopoulos, and I. Stavrakakis, "Distributed selfish replication," *IEEE Trans. Parallel Distrib. Syst.*, vol. 17, no. 12, pp. 1401–1413, dec. 2006.

[10] A. Sharma, A. Venkataramani, and R. K. Sitaraman, "Distributing Content Simplifies ISP Traffic Engineering," in *Proceedings of the ACM SIGMETRICS/International Conference on Measurement and Modeling of Computer Systems*, ser. SIGMETRICS '13, 2013, pp. 229–242.

[11] N. Laoutaris, V. Zissimopoulos, and I. Stavrakakis, "Joint object placement and node dimensioning for internet content distribution," *Inf. Process. Lett.*, vol. 89, no. 6, pp. 273–279, Mar. 2004.

[12] ——, "On the optimization of storage capacity allocation for content distribution," *Comput. Netw.*, vol. 47, no. 3, pp. 409–428, Feb. 2005.

[13] I. Baev, R. Rajaraman, and C. Swamy, "Approximation Algorithms for Data Placement Problems," *SIAM J. Comput.*, vol. 38, no. 4, pp. 1411–1429, aug. 2008.

[14] T. Bektaş, J.-F. Cordeau, E. Erkut, and G. Laporte, "Exact algorithms for the joint object placement and request routing problem in content distribution networks," *Comput. Oper. Res.*, vol. 35, no. 12, pp. 3860–3884, dec. 2008.

[15] S. Borst, V. Gupta, and A. Walid, "Distributed caching algorithms for content distribution networks," in *Proc. IEEE INFOCOM'10*, mar. 2010, pp. 1–9.

[16] J. Dai, Z. Hu, B. Li, J. Liu, and B. Li, "Collaborative hierarchical caching with dynamic request routing for massive content distribution," in *Proc. IEEE INFOCOM'12*, mar. 2012, pp. 2444–2452.

[17] B. Chun, K. Chaudhuri, H. Wee, M. Barreno, C. H. Papadimitriou, and J. Kubiatowicz, "Selfish caching in distributed systems: a game-theoretic analysis," in *Proceedings of the 23rd annual ACM Symposium on Principles of Distributed Computing (PODC'04)*, 2004, pp. 21–30.

[18] V. Valancius, B. Ravi, N. Feamster, and A. C. Snoeren, "Quantifying the Benefits of Joint Content and Network Routing," in *Proceedings of the ACM SIGMETRICS/International Conference on Measurement and Modeling of Computer Systems*, ser. SIGMETRICS '13, 2013, pp. 243–254.

[19] B. Frank, I. Poese, G. Smaragdakis, S. Uhlig, and A. Feldmann, "Content-aware traffic engineering," *SIGMETRICS Perform. Eval. Rev.*, vol. 40, no. 1, pp. 413–414, jun. 2012.

[20] W. Jiang, R. Zhang-Shen, J. Rexford, and M. Chiang, "Cooperative content distribution and traffic engineering in an ISP network," in *Proc. SIGMETRICS '09*, Seattle, WA, USA, 2009, pp. 239–250.