On Fair Network Cache Allocation to Content Providers

Sahar Hoteit, Mahmoud El Chamie, Damien Saucez, Stefano Secci

On Fair Network Cache Allocation to Content Providers

Sahar Hoteit[a], Mahmoud El Chamie[b], Damien Saucez[c], Stefano Secci[d]

[a]*LSS, Centrale-Supelec, Gif sur Yvette, France, email: sahar.hoteit@lss.supelec.fr*
[b] *Department of Aerospace Engineering and Engineering Mechanics, University of Texas at Austin, TX 78712, USA, email: melchami@utexas.edu*
[c]*Inria Sophia Antipolis, 2004 Route des Lucioles, B.P. 93, 06902 Sophia Antipolis Cedex, France, email: damien.saucez@inria.fr*
[d]*Sorbonne Universités, UPMC Univ Paris 06, UMR 7606, LIP6, F-75005, Paris, France, email: stefano.secci@upmc.fr*

Abstract

In-network caching is an important solution for content offloading from content service providers. However despite a rather high maturation in the definition of caching techniques, minor attention has been given to the strategic interaction among the multiple content providers. Situations involving multiple Content Providers (CPs) and one Internet Service Provider (ISP) having to give them access to its caches are prone to high cache contention, in particular at the appealing topology cross-points. In this paper, we propose a resource allocation and pricing framework to support the network cache provider in the cache allocation to multiple CPs, for situations where CPs have heterogeneous sets of files and untruthful demands need to be avoided. As cache imputations to CPs need to be fair and robust against overclaiming, we evaluate common proportional and max-min fairness (PF, MMF) allocation rules, as well as two coalitional game rules, the Nucleolus and the Shapley value. We find that the naive least-recently-used-based cache allocation approach provides proportional fairness. Moreover, the game-theoretic rules outperform in terms of content access latency the naive cache allocation approach as well as PF and MMF approaches, while sitting in between PF and MMF in terms of fairness. Furthermore, we show that our pricing scheme encourages the CPs to declare their truthful demands by maximizing their utilities for real declarations.

Keywords: In-network Caching, Information Centric Networking, Mechanism Design, Game Theory, Cache Allocation.

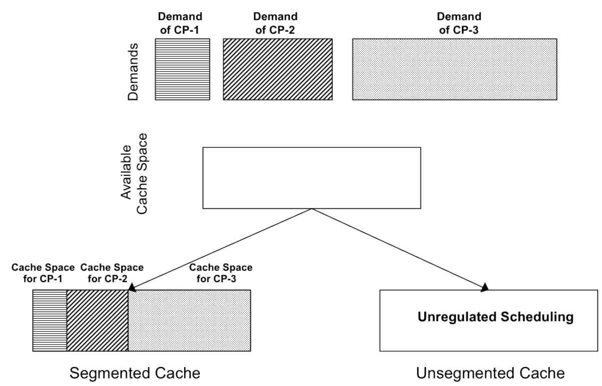


Figure 1: Representation of segmented and unsegmented caches with many content providers (CPs).

1. Introduction

With the advent of broadband and social networks, the Internet became a worldwide content delivery platform ([1, 2]), with high bandwidth and low latency requirements. To meet the always increasing demand, contents are pushed as close as possible to their consumers and *Content Providers* (CP) install dedicated storage servers directly in the core of Internet Service Provider (ISP) networks [3]. However, the TCP/IP protocol suite uses a conversational mode of communication between hosts that can be considered not appropriate for content delivery [2]. Therefore, a complex machinery is developed (around the Domain Name System, DNS, protocol and the HyperText Transfer Protocol, HTTP) to compensate the limitations of the TCP/IP protocol suite. Conscious of the mismatch between the network usage and its conception, the research community recently proposed the concept of in-network caching (e.g., Information Centric Networking (ICN) [2, 4]). For instance, in ICN, content objects can be accessed and delivered natively by the network according to their name rather than relying on IP addresses [2]. Hence, this technology removes the concept of location or topology from communication primitives and uses the notion of contents and their name instead. These contents can therefore be found potentially anywhere in the network, moved or replicated at different locations [5, 4, 6].

ISP networks then become native distributed storage systems, i.e., network cache providers that can directly sell caching capabilities to content providers instead of hosting their servers. However, it is most probable that the storage demand exceeds the total ISP storage offer, at least for the content caching locations the closest to the users. So far, the contention is solved by considering each storage as one autonomic and self managed cache (e.g., using a LRU, least-recently-used, mechanism), as depicted in the rightmost part of Figure 1. With this approach CPs are unable to provision their own infrastructure accurately as they cannot predict what contents will be cached by the ISP as it depends on the workload of the other CPs using the ISP infrastructure.

In this paper, we propose to address this contention situation by segmenting the storage on a per-content provider basis, as depicted in the leftmost part of Figure 1. Each content provider receives a portion of the storage space depending on its storage demand. For this, based on application of results in economics and game theory to the target problem, we propose a 2-step *mechanism design* ([7, 8]) that computes a fair and rational sharing of resources between CPs. The first step relies on a content cache allocation algorithm where, as a function of content cache demands coming from CPs, the network cache provider decides the *imputation* of cache spaces to CPs. The second step uses a predefined *payment rule* by auctions to decide the selling price of the storage unit in the network; its purpose is to prevent content providers from lying about their true demands.

The paper is organized as follows. Section 2 presents an overview of related works. In Section 3, we analytically introduce the context of our work: Section 3.2 presents the resource allocation problem by modeling it as a cooperative game, and Section 3.4 develops our pricing scheme based on mechanism design theory. Section 4 presents the implementation of our proposed pricing scheme for the different cache imputations. Section 5 compares the proposed cache allocation rules with other schemes. Finally, Section 6 concludes the paper.

2. Background

Several researches have recently proposed various cache allocation solutions. Rossi and Rossini compare the in-network caching performance in homogeneous (i.e., where the routers have the same overall cache size) and heterogeneous cache deployments (i.e., where the routers have not the same cache size) [9]. In the latter case, they propose to allocate cache capacity proportionally to the router centrality metric measured according to different criteria: degree, stress, betweenness, closeness, graph and eccentricity centrality. Authors of both [9] and [10] show that allocating cache capacity across the network in a heterogeneous manner slightly improves network performance compared to the homogeneous manner; however, the benefits of heterogeneous deployments become apparent with larger networks (e.g., more than 100 nodes). Moreover, Wang et al. study the influence of content popularity distribution on network performance showing that (i) for uniformly distributed content demands (e.g., catch-up TV), pushing caches into the core yield better performance while (ii) highly skewed popularity request patterns (e.g., YouTube, mobile VoD system or Vimeo) are better served by edge caching [10]. This latter point is confirmed by Fayazbakhsh et al. [11].

Recently, there has been significant interest in applying game theory to the analysis of communication networks, with the aim to identify rational strategic solutions for multiple decision-maker situations. Indeed, as opposed to mono-decision maker problems, game-theoretic approaches adopt a multi-agent perspective to account for different objective functions and counter objections to rationally non justified solutions [12]. Thus far, many papers from the literature have tackled game-theoretic approaches for cache allocation using non-cooperative game theory. These papers consider servers or routers or networks as selfish entities seeking to maximize their own profit at the expense of globally optimum behavior. For example Pacifici and Dan study a non-cooperative game to characterize the problem of replication of contents by a set of selfish routers aiming to minimize their own costs [13]. In the same context, Chun et al. characterize the caching problem among selfish servers using a non-cooperative game [14]. For each content in the network, selfish servers have two possible actions: either caching the content if all its replicas are located too far away or not caching it if one of its replicas is located at a nearby node. As in [13], they show the existence of pure strategy Nash equilibrium of the caching game.

Motivated by the intuition that forms of collaboration between different network cache providers could yield an enhancement in network performance, in [15] the authors propose a game whereby the routers behave as rational selfish agents that seek to minimize their aggregate content access cost. Going beyond routers, [16] describes how content providers could shape their content access prices and discounts to favor the emergence of cache space distribution overlays across independent networks, toward the formation of incentive-prone overlay equilibria.

Under a similar rationale, yet a broader context, in this paper we investigate how the network cache provider provider, modeling CPs as players of a game, can design a cache allocation framework so that cache imputations to CPs are strategically fair and robust against cache space over-claiming, while outperforming legacy approaches in terms of content access latency. Up to our knowledge, there are no other works precisely addressing this problem, despite the above-cited works do share similar concerns in cache allocation and component sharing. As detailed in the following, we propose various cache allocation rules, including coalitional game theory rules for *bankruptcy* situations [17] to solve the atomic cache contention problem, motivated by the fact that a similar algorithmic approach has shown high performances in strategic shared spectrum allocation problems [18].

3. Cache Allocation Framework and Rules

In the context of a network cache provider, the cache capacity is used to host content files in order to enhance users' quality of experience by decreasing content access latency. Assuming contents are owned by external CPs, the network cache provider would need to offer a neutral interface to access its caches, guaranteeing a fair allocation of caches with respect to cache space demands, which are in turn a function of content popularity. In this section, we formulate the problem, and then we detail the cache allocation algorithm and the corresponding pricing framework.

3.1. Problem Formulation

Let us assume that there are n *Content Providers* (CPs), and each CP owns a given number of files. With the possibility to cache some files in the network between them and the users (by renting cache space from the network cache provider), the CPs can reduce their CAPEX by reducing the load on their servers and enhance their users' quality of experience by decreasing content access latency. Depending on how much cache space each provider is willing to pay for, the demand for a cache space by each content provider may not cover all its catalog size. Let d_i be the cache space demand of the i^{th} CP, indicated in the following as CP i. Moreover, different files can have different popularity. In the following $\vec{d}$ denotes the vector of all demands.

We denote by E the global cache space of the network cache provider. We target the expected situation for an economically viable cache deployment in which the network cache provider receives more demands than what it can satisfy, i.e., $\sum_{i=1}^{n} d_i \geq E$. If this was not the case, i.e., if the total demand is less than the available space, then the network cache provider would be able to allocate for every CP the exact space demanded. Contention would likely still manifest for at least those few best nodes that are at the most attractive cross-points of users' demands (as far as these few best nodes would not be able alone to satisfy the whole demand).

In this context, there is a competition in accessing the network caches. Even if unlikely, the risk from a network cache provider perspective is that CPs partially ally between each other, forming sub-coalitions when designing their respective demands. To be robust to such behavior and avoid the formation of oligopolies, the network cache provider shall take into account the possible sub-coalitions in the allocation of cache sizes to CPs, designing an appropriate pricing framework. More precisely, the network cache provider (e.g., ICN provider) has to:

1. decide on the *allocation rule*, i.e., how to assign cache space to each CP based on CPs individual content cache size demands.
2. decide on the *payment rule*, i.e., how to fix prices for the allocated space given by step 1.

To emphasize the need of these two separate provisioning rules, let us explain the rationale with the following three interaction cases (unrealistic, naive, and wise cases). First, let us consider the (unrealistic) case where the network cache provider announces that the space is given for free for the highest demand: every CP would then have an incentive to announce a very high demand, lying on the value of their real needs, to get free space. Suppose now another (more realistic, but naive) case with an announced fixed price per unit of cache: also in this case, because the space is limited, each CP has an incentive to announce a higher *untruthful* demand so that it can get more space. In order to avoid these situations, the network cache provider should (wisely) design both steps in advance to make sure that the outcome of the overall scheme is a desired one. For this purpose, we propose to adopt *mechanism design* theory concepts [7]. In particular, we refer to approaches for single-dimensional environments to make sure that the allocation scheme provides strong performance guarantees (as explained hereafter, performance guarantees are based on fairness criteria), and at the same time it provides strong incentives for the CPs to be truthful in communicating their real demand.

The allocation and payment rules are interrelated in general. However, the mechanism design theory successfully deals with the two steps in a consecutive manner. First we suppose that the CPs are communicating their truthful demand. Based on these demands, we design a cache allocation scheme giving each CP its share of the limited resource E. Then, we design a payment rule for the CPs such that the dominant strategy for the CPs is to send their real demand (i.e., with no incentives to lie about it). Under this approach, the network cache provider can shape a strategic allocation making its provisioning architecture rationally acceptable and attractive for additional CP customers.

3.2. Cache Allocation to Content Providers

An *allocation rule* is a function f having as an input the demands of the CPs (the demand vector $\vec{d} \in \mathbb{R}_+^n$) and the total available cache space $E \in \mathbb{R}_+$, and giving as output an *imputation vector* $\vec{x} \in \mathbb{R}_+^n$ containing the cache space portion to allocate to each CP (i.e., the values in $\vec{x}$ ranges between 0 and E such that $\sum_{i=1}^{n} x_i = E$), i.e., $f : (\vec{d}, E) \rightarrow \vec{x}$.

Let $\vec{d}_{-i}$ be the vector of demands of all the CPs other than i. With a little abuse of notation, let us indicate the imputation for CP-i. as $x_i = f_i(d_i, \vec{d}_{-i}, E)$. For convenience, we also define $\bar{x}_i = x_i/E$ as the normalized imputation, i.e., the proportion of E allocated to CP-i. Let us give the following definition.

Definition 1. *(Monotone Allocation Rule) An allocation rule is monotone if for each* $(\vec{d}, E)$ *and for each CP-i the following statement holds:*

$$\text{If } d_i' > d_i, \text{ then } f_i(d_i', \vec{d}_{-i}, E) \geq f_i(d_i, \vec{d}_{-i}, E), \tag{1}$$

In other words, fixing all the other CPs demands $\vec{d}_{-i}$, if the demand of CP i increases from d_i to d_i', then the imputation x_i should not decrease. Monotonicity plays an important role in designing the payment rule (we get back to this issue in Section 3.4).

The allocation of resources to those claiming higher demands than what is available is referred to in the literature as a *bankruptcy* problem (the term derives from the evident connection with the problem of bankruptcy where a person or other entity cannot repay the debts claimed by creditors). For this reason, in the following we sometimes refer to the CPs as *claimants*, or the total available cache space to partition as the *estate*.

There are different possible approaches from the literature that can be used as allocation rules for a bankruptcy situation. We present thereafter the most common.

3.2.1. Allocation by Proportional Fairness (PF)

Proportional fairness distributes the resources proportionally to the demands subject to total space constraint [19], i.e.,

$$\frac{f_i(\vec{d}, E)}{d_i} = \frac{f_j(\vec{d}, E)}{d_j} \quad \text{for any pair of CPs } (i, j).$$

It is straightforward to note that PF is monotone.

3.2.2. Allocation by Max-Min Fairness (MMF)

MMF maximizes the profit of the lowest claimant, then it maximizes the second lowest demand in the game, and so on [20]. Formally, if we order the CPs according to their increasing demand, i.e., $d_1 \leq d_2 \leq \cdots \leq d_n$, then MMF allocates the available space E as follows:

$$f_i(\vec{d}, E) = \min\left(d_i, \frac{E - \sum_{j=1}^{i-1} f_j(\vec{d}, E)}{n - i + 1}\right) \text{ for } i = 1, \ldots, n.$$

Intuitively, MMF gives the lowest claimant (assuming $\min_i d_i \leq \frac{E}{n}$) its total demand and evenly distributes unused resources to the other users. It is also straightforward to note that MMF is monotone.

Both MMF and PF allow computing fair imputations without considering the possibility that CPs could ally when formulating their demands. Alternatively, game theoretic allocation rules can be attractive toward the computation of a strategically fair imputation. Before presenting some game-theoretic allocation rules, let us formally define the bankruptcy game for our settings where the CPs are the players.

Definition 2 (Bankruptcy Game [17])**.** *A bankruptcy game, denoted by $G(\mathcal{N}, v)$, is a cooperative game where $\mathcal{N}$ represents the set of claimants of the bankruptcy situation (i.e., the CPs with $|\mathcal{N}| = n$) and v is the characteristic function of the game given in Eq.* (2) *that associates to each coalition S its worth defined as the part of the estate (i.e., the global cache space), not claimed by its complement.*

$$v(S) = \max(0, E - \sum_{i \in \mathcal{N} \setminus S} d_i)\,, \forall S \subseteq \mathcal{N} \setminus \{\emptyset\} \tag{2}$$

where $E \geq 0$ is the estate that has to be divided among the members of $\mathcal{N}$, S is a coalition of players, and $\sum_{i=1}^{n} d_i \geq E$.

After defining the characteristic function of each possible coalition in the game by Eq. (2), then $f(\vec{d}, E)$ gives the imputation using well known fairness concepts in cooperative games. Imputations for cooperative games are essentially qualified with respect to the satisfaction of individual and coalitional rationality constraints, desirable properties and existence conditions. Among them, the Shapley value and the Nucleolus are attractive allocation rules that give a unique imputation while satisfying desirable properties.

3.2.3. Allocation by Shapley Value

The Shapley Value [21] is the center of gravity of the *core*[1] of a bankruptcy game. It is defined as:

$$f_i(\vec{d}, E) = \sum_{S \subset \mathcal{N} \setminus \{i\}} \frac{|S|!(|\mathcal{N}| - |S| - 1)!}{|\mathcal{N}|!}[v(S \cup \{i\}) - v(S)] \tag{3}$$

In other terms, the Shapley value is computed by averaging the marginal contributions of each player in the game in each strategic situation (i.e., players' permutation). The Shapley value has been already proposed for a variety of situations in networking, such as inter-domain routing [22] and network security [23], because it shows desirable properties in terms of correct modeling of null player situations, symmetry, individual fairness, and additivity.

Moreover, the Shapley value allocation rule for bankruptcy games is monotone because Eq. (3) can be rewritten using equation (2) as follows:

$$f_i(b_i, \vec{b}_{-i}, E) = \sum_{S \subset \mathcal{N} \setminus \{i\}} \alpha_S \phi_S(b_i), \tag{4}$$

where $\alpha_S = \frac{|S|!(|\mathcal{N}|-|S|-1)!}{|\mathcal{N}|!}$ and:

$$\phi_S(b_i) = \begin{cases} b_i & \text{if} \quad b_i \leq \max(0, E - \sum_{j \in \mathcal{N} \setminus \{S,i\}} b_j) \\ \max(0, E - \sum_{j \in \mathcal{N} \setminus \{S,i\}} b_j) & \text{otherwise} \end{cases} \tag{5}$$

so by fixing $\vec{b}_{-i}$, the function $\phi_S(b_i)$ is a non-decreasing function in b_i for any set S. Thus the Shapley value allocation is monotone.

[1]The core of a game contains the imputations satisfying coalitional rationality and efficiency constraints, such that no player or coalition gains by seceding from the grand coalition, i.e., the core is a stable set. The core in general might not exist, but for bankruptcy games it does (i.e., it is not empty).

3.2.4. Allocation by Nucleolus

The Nucleolus [24] is the unique consistent solution in bankruptcy games that minimizes the worst inequity. The Nucleolus lies in the core and it is computed by minimizing the largest excess of different coalitions of the game. The excess is expressed as:

$$e\left(f(\vec{d}, E), S\right) = v(S) - \sum_{j \in S} x_j \ \ , \forall S \subseteq \mathcal{N} \tag{6}$$

This excess measures the amount by which the coalition S falls short of its potential $v(S)$ in the imputation $\vec{x}$.

To give the formal definition of the Nucleolus for bankruptcy games, denote $O(\vec{y}) = (e(\vec{y}, S_1), e(\vec{y}, S_2), \ldots, e(\vec{y}, S_{2^n}))$, where $e(\vec{y}, S_k) \geq e(\vec{y}, S_{k+1}), k = 1, \ldots, 2^n - 1$. Among all the imputations $\vec{y}$ satisfying: $\sum_{i=1}^{n} y_i = v(\mathcal{N}) = E$, the Nucleolus gives the *unique* imputation $\vec{x}$ such that $O(\vec{x}) <_L O(\vec{y}) \quad \forall \vec{y}$, where $<_L$ is the lexicographic order.[2] In other terms, the Nucleolus is the solution that improves the situation of the player in the worst case; so it introduces a degree of fairness in its imputation. It is monotone thanks to the intrinsic consideration of individual rationality constraints. The Nucleolus is used for instance in strategic transmission computation ([25, 26, 18]) because it satisfies desirable properties, e.g., it improves the situation of the player that is worst off while being consistent (i.e., no player or group of players can gain more by unilaterally deviating from an imputation).

3.3. Cache allocation algorithm

The total cache space in the network is formed from the collection of the router caches. These caches are distributed in heterogeneous locations in the network. For example, it might be more convenient for CPs to be allocated a cache space closer to the end users (thus their contents are closer to clients reducing content access latency). Therefore, it is important that the network cache provider distributes a homogeneous cache space to CPs (every unit of cache space should have the same value from the content providers perspective). In this respect, the cache provider should *cluster* routers that have similar properties for CPs. According to [9] and [10], three commonly

[2] We say that a vector $\vec{u}$ is *lexicographically* larger than $\vec{v}$ (denoted by $\vec{v} <_L \vec{u}$) if there exists k such that $u_i = v_i$ for all $i \in \{1, 2, \ldots, k-1\}$ and $u_k > v_k$.

accepted criteria for grouping the routers are: the proximity to the user-network edge, the router degree, and the router centrality (betweenness). More precisely, the *contention* metrics that we investigate are defined as follows:

- *Router Proximity to network edge (RP)*: the number of hops separating a router from network edge.
- *Router Degree (RD)*: the number of links incident to a router.
- *Router Betweenness (RB)*: the number of times a node is along the shortest path between two other nodes.

Following the ranking of routers according to the contention metric, we propose the following allocation algorithm that can be performed upon significant changes of content providers' demands.

Algorithm 1 Cache Allocation Algorithm

1: Form clusters of routers by grouping together those having the same contention metric, and order these clusters from the highest importance (in terms of total cache space in each cluster of routers) to the lowest one;
2: Take the cluster with the highest importance and apply the allocation rule to routers of the cluster;
3: Decrease the demand of each CP by the amount allocated in the cluster;
4: Take the next cluster and apply the allocation rule;
5: Stop when all clusters are treated or there is no remaining demand.

For the game-theoretic allocation rules, this corresponds in iterating a game $G(\mathcal{N}, v)$ differing in that, at each iteration:

- $\mathcal{N}$ includes all the CPs, but with different demands d_i.
- The available cache size (E), varies as a function of the cluster size and the capacities of routers in the cluster. For instance, if the cache capacity of each router is given by C_r, the corresponding estate is given by: $E = \sum_{r \in \mathcal{R}_c} C_r$ where $\mathcal{R}_c$ is the set of routers in the cluster c.

It is worth noting that since the routers within the same cluster have the same contention metric, the allocated cache space to each CP in a cluster can be evenly allocated from any cache among the routers in that cluster.

Remark. Algorithmic game theory adds one more requirement to the design of the system: the complexity of obtaining the allocation should be computationally efficient. As a matter of fact, the computation of the Shapley value is generally done using (3); however, in games with a large number of players the computational complexity of the Shapley value grows significantly. In our instance this does not cause a real problem because the number of CPs asking for the resource in a network is typically low (less than 10) and the complexity of the allocation scheme is a function of the number of CPs (and not a function of the potentially huge number of content files). □

For computing the Shapley value in reasonable time, several analytical techniques have been proposed such as multi-linear extensions [12], and sampling methods for simple games [27], among others. The process for computing the Nucleolus is however more complex than for the Shapley value. It is described as follows. First, we start by finding the imputations that distribute the worth of the grand coalition in such a way that the maximum excess (dissatisfaction) is minimized. In the event where this minimization has a unique solution, this solution is the nucleolus.[3] Otherwise, we search for the imputations which minimize the second largest excess. The procedure is repeated for all subsequent excesses, until finding a unique solution which would be the nucleolus. These sequential minimizations are solved using linear programming techniques [28].

3.4. Pricing Framework

As already argued, a robust pricing framework needs to be designed by the network cache provider to ensure true demands are formulated by CPs. Actually, the same unit of cache space may have different values for the different CPs, those with higher traffic (i.e., higher demand) are willing to pay more for a cache space unit to accommodate the high traffic volume. Taking into account this design goal, in our model we consider that the value of a unit of cache space for a given CP is a given function of its clients' traffic.

Along with the fairness of the allocation scheme, the payment rule should be designed to give strong guarantees that the CPs are truthful in communicating their real demand. Under this perspective, it becomes natural to think of the demands as bids (as in auctions), and the cache partitioning as the allocation outcome from an auction. The demand vector is given by $\vec{d}$

[3] For the class of Bankruptcy games, the nucleolus always exists.

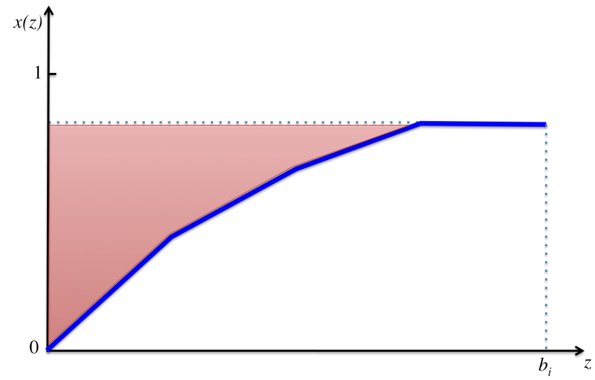


Figure 2: The solid curve (in blue) is the piecewise-linear allocation function $\bar{x}_i$ given by $x(z)$ for CP i when varying its demand from 0 to b_i (z axis). The area above the curve (in red) is the payment rule (price to pay by the CP).

where d_i is the (true) demand by CP-i (also considered as the private value of i). The bid vector is given by $\vec{b}$ where b_i is the value communicated by CP-i to the network cache provider (could be equal to d_i if i declares the truth).

The truthful communication of demands should be a dominant strategy. This is known as the *dominant-strategy incentive-compatible* (DSIC) property [8, p. 415]. The normalized allocation $\bar{x}_i$ is the proportion of the full available cache space allocated to content provider i (i.e., $\bar{x}_i$ ranges in the interval $[0, 1]$). The payment rule is given by $\vec{p}$, where p_i is the price of the allocation paid by CP-i. The utility of a content provider is given by:

$$U_i = V_i(d_i, f_i(b_i, \vec{b}_{-i}, E)) - p_i(b_i, \vec{b}_{-i}, E) \tag{7}$$

where $V_i(d_i, f_i(b_i, \vec{b}_{-i}, E)) = d_i\bar{x}_i$ is the value of the allocated space from the CP-i perspective,[4] and $p_i(b_i, \vec{b}_{-i}, E)$ is the price paid. Every CP has the incentives to maximize its utility.

Definition 3 (DSIC)**.** *The tuple* $(\vec{x}, \vec{p})$ *is DSIC if: 1) each truth-telling CP*

[4]Since $V_i = d_i\bar{x}_i$, the value of a unit storage is considered proportional to the demand. In case of a generic function $g_i(.)$, d_i would be replaced by $g_i(d_i)$ and what follows hold with minor changes.

is guaranteed a non-negative utility and 2) each CP has as dominant strategy the communication of its truthful demand, i.e., for all CPs, and for any b_i*,*

$$V_i(d_i, f_i(d_i, \vec{b}_{-i}, E)) - p_i(d_i, \vec{b}_{-i}, E) \geq V_i(d_i, f_i(b_i, \vec{b}_{-i}, E)) - p_i(b_i, \vec{b}_{-i}, E)$$

Then, the tuple $(\vec{x}, \vec{p})$ is DSIC if when $b_i = d_i$, this strategy maximizes the utility of CP-i no matter what the other CPs do. Being that the utility $U_i = d_i\bar{x}_i - p_i$, for example with the pricing rule $p_i = b_i\bar{x}_i$, no one has an incentive to communicate the true demand. Because with that pricing rule, the utility would be $U_i = 0$ for truth-tellers while it can be increased if everyone declared a slightly lower demand. This would lead to a situation where everyone declares a lower demand than their real one. On the other hand, for a fixed price per storage space (i.e., $p_i = \alpha\bar{x}_i$ for a given $\alpha \in \mathbb{R}_+$) every CP having $d_i > \alpha$ has an incentive to increase its communicated demand (b_i) to receive more space increasing its utility. We thus have to determine what pricing rule ensures that the CPs have no incentives to lie (given the Shapley and the Nucleolus-based allocation rules). It turns out that by Myerson's Lemma [29] from mechanism design theory we can design the prices to meet our objective:

Theorem 1 (Myerson's Lemma [29])**.** *If* $\vec{x}$ *is monotone, then there is a unique payment rule* $\vec{p}$ *such that the mechanism* $(\vec{x}, \vec{p})$ *is DSIC.*

The monotonicity is given by Definition 1, and the four presented allocation rules are monotone as already discussed. The price is given by Myerson's Lemma [29] as follows:

$$p_i(b_i, \vec{b}_{-i}, E) = b_i \frac{f_i(b_i, \vec{b}_{-i}, E)}{E} - \frac{1}{E}\int_0^{b_i} f_i(z, \vec{b}_{-i}, E)\mathrm{d}z \tag{8}$$

The allocation as function of the demand looks as in Fig. 2. The price can be interpreted as an area above the curve (as given by Fig. 2). Notice that by considering this pricing rule, each content provider maximizes its utility U_i by communicating its true demand no matter what others do, i.e., U_i is maximized when $b_i = d_i$ for every $\vec{b}_{-i}$.

Remark. For the Shapley value allocation, the allocation is piece-wise linear as function of b_i and we can identify precisely the points where the curve in Fig. 2 changes its slope, and thus closed-form pricing equations can be derived for Shapley value. Closed-form pricing equations can also be derived for PF and MMF allocations. The transition points of the curve in the case of Nucleolus allocation cannot be found in closed-form, and thus we refer to numerical methods as we demonstrate in the next section. □

As a result, the network cache provider can declare a pricing accordingly to (8) to all the CPs, so that none of the CPs has an incentive to declare a different demand than their real one, and based on these (truthful declarations) the allocation using the proposed cache allocation algorithm is carried out. It is important to note that this pricing framework does not necessarily maximize the profit for the network cache provider, but it is the *unique* pricing rule [29] that provides strong incentives for truthful declaration of demands by the CPs. Any other pricing rule can cause the CPs to communicate false demands to maximize their utilities.

Remark. In reality, many companies use business models that do not necessarily maximize their profits. For example, eBay online auctions (using proxy bidding feature) resembles a theoretical second-price sealed-bid auction closely. Its purpose is not to maximize the profit of the company but to have participants bid their real values of the items [30]. Another example is Google sponsored search auction that identifies which advertisers' links are shown and in what order after every search query to Google engine. Also in this model, Google uses "generalized second price"auction format whose primary objective is not to maximize the profit, but for bidders to give their real value for the position of their link (this model gave over 98% of Google total revenue in 2005 [31]). □

4. Pricing Implementation

The pricing rule given in the paper is of the form

$$p_i(b_i, \vec{b}_{-i}, E) = b_i \frac{f_i(b_i, \vec{b}_{-i}, E)}{E} - \frac{1}{E} \int_0^{b_i} f_i(z, \vec{b}_{-i}, E) \mathrm{d}z \tag{9}$$

Thus we need to calculate $f_i(z) = f_i(z, \vec{b}_{-i}, E)$ as an intermediate step in calculation of the price. Closed-form equations can be found for PF, MMF, and Shapley value. For the case of the Nucleolus, f_i can be calculated only for a given z, so numerical methods are needed to give an approximation of the price.

4.1. Proportional Fairness

The allocation for proportional fairness is given by

$$\frac{f_i}{b_i} = \frac{f_j}{b_j} \text{ for all CPs } i, j,$$

then since $\sum_i f_i = E$ we can write

$$f_i(z) = \frac{Ez}{z + \sum_{j \neq i} b_j}. \tag{10}$$

Then

$$\int_0^{b_i} f_i(z) = Eb_i - E(\sum_{j \neq i} b_j) \log\left(\frac{\sum_j b_j}{\sum_{j \neq i} b_j}\right),$$

and the resulting price to pay by CP-i knowing that the bids are $b_1, \ldots, b_n$ is

$$P_i^{(prop)} = \frac{b_i^2}{\sum_j b_j} - b_i + (\sum_{j \neq i} b_j) \log\left(\frac{\sum_j b_j}{\sum_{j \neq i} b_j}\right). \tag{11}$$

An interesting observation about this pricing rule is that it is independent of E. Which means that as long as the ratio between the demands is the same, even if E increased the allocation changes, but the price remains the same.

4.2. Max-Min Fairness

Assuming that the bids are placed in increasing order $b_1 \leq b_2 \leq \cdots \leq b_n$, then the allocation of max-min is given by

$$f_i(z) = \min\left(z, \frac{E - \sum_{j=1}^{i-1} f_j(\vec{d}, E)}{n - i + 1}\right) \text{ for } i = 1, \ldots, n.$$

The equation shows that as we increase z from 0 to b_i we have

$$f_i(z) = \begin{cases} z & \text{if } z \leq C_i \\ C_i & \text{if } z > C_i, \end{cases} \tag{12}$$

where C_i is the critical point when the curve becomes constant to be determined. To find C_i we can calculate f_i for any sufficiently large z. This sufficient large number can be chosen to be E because $\frac{E-\sum_{j=1}^{i-1} f_j(\vec{d},E)}{n-i+1} \leq E$ for any vector $\mathbf{b}$. Then

$$C_i = f_i(E).$$

We can now calculate the following integral:

$$\int_0^{b_i} f_i(z) = \begin{cases} \frac{b_i^2}{2} & \text{if } b_i \leq C_i \\ \frac{C_i^2}{2} + (b_i - C_i)C_i & \text{if } b_i > C_i, \end{cases}$$

and the corresponding price for CP-i is

$$P_i^{(maxmin)} = b_i \frac{\min(b_i, C_i)}{E} - \frac{(\min(b_i, C_i))^2}{2E}.$$

4.3. Shapley value fairness

The allocation for Shapley value is given by equations (4) and (5). In order to determine the price; equation (4) can be reformulated using Eq. (5) as follows:

$$f_i(z, \vec{b}_{-i}, E) = g(\vec{b}_{-i}) + \left(\sum_{T \in \mathcal{T}} \alpha_T \right) z \quad (13)$$

where $g(\vec{b}_{-i})$ is a scalar function independent of z, and $\mathcal{T}$ is a relevant set of the sets $T \subseteq \mathcal{N} \backslash \{i\}$. Equation (13) demonstrates that the curve of Fig. 2 is piece-wise linear for the Shapley value allocation.

For every content provider i and for any set $S \in \mathcal{N} \backslash i$, we can define a function $q_i(S) = \max(0, E - \sum_{j \in \mathcal{N} \backslash \{S,i\}} b_j)$. Since the domain of definition of $q_i(.)$ has finite elements, then we can define a vector $\Phi \in \mathbb{R}^{2^{n-1}}$ to be the image of the function (i.e., for any $S \in \mathcal{N} \backslash i$, there exists an index m such that $\Phi_m = q_i(S)$). Then each element of this vector corresponds to a set of CPs without CP-i. Define Θ a vector that has elements $\alpha_S = \frac{|S|!(|\mathcal{N}|-|S|-1)!}{|\mathcal{N}|!}$ where S is the corresponding set index.

Let us define $\hat{\Phi}$ the vector having the elements of Φ sorted in increasing order. And define $\hat{\Theta}$ to be the vector having the elements for the corresponding α_S (note that $\hat{\Theta}$ is not necessarily in increasing order).

The allocation function $f_i(z)$ is piece-wise linear defined on the interval $[0, b_i]$ having $f_i(0) = 0$ and slopes given as follows:

$$\frac{\partial f_i(z)}{\partial z} = \begin{cases} \sum_{j=1}^{2^{n-1}} \hat{\Theta}_j & \text{for } 0 < z < \hat{\Phi}_1 \\ \sum_{j=k+1; k=1,\ldots,2^{n-1}-1}^{2^{n-1}} \hat{\Theta}_j & \text{for } \hat{\Phi}_k < z < \hat{\Phi}_{k+1} \\ 0 & \text{for } \hat{\Phi}_{2^{n-1}} < z < b_i. \end{cases} \quad (14)$$

Note that the points that the curve changes its slope are the points $z \in [0, b_i]$ such that $z = \hat{\Phi}_k$. As we know that the function satisfies $f_i(0) = 0$, then we can use (14) in a recursive way for the exact calculation of the integral $\int_0^{b_i} f_i(z)$ and the corresponding price.

4.4. Nucleolus fairness

In case of nucleolus, the curve $f_i(z)$ is also piece-wise linear with z. But the critical points for which the slope can change cannot be given in closed form solution. The integral can then be numerically approximated. Since we know that the slope cannot change more than 2^{n-1} times, we can divide the interval $[0, b_i]$ into $2^{n-1} + 1$ equal intervals where the length of an interval is given by

$$\Delta = \frac{b_i}{2^{n-1} + 1}.$$

Then the integral can be discretized and approximated as follows:

$$\int_0^{b_i} f_i(z) \approx \sum_{k=0}^{2^{n-1}} \left(f_i(k\Delta)\Delta + \frac{\Delta}{2} \left[f_i((k+1)\Delta) - f_i(k\Delta) \right] \right)$$

and the resulting price follows directly from (9).

5. Performance Evaluation

We consider a network composed of 25 routers of same caching capacity C (i.e., homogeneous cache size). We consider two networks with a tree (where there is only one path from an end-user to a CP) and a partial mesh (where there can be multiple paths from an end-user to a CP), having both an edge-to-CP shortest path length up to 6 hops. To have comparable results that are independent of the CPs' locations and their connections to the network, we use symmetric topologies. This is especially important as the results obtained through assymetric topologies highly depend on the way each CP is connected to the network. For this aim, in the simulations, the tree topology consists of connecting the CPs to the root router of the tree while connecting the end-users to its leaves. Besides, in the partial mesh, the CPs are all connected to one router in the network, while the end users are connected randomly to some of the other router nodes of the network.

We include 5 CPs, denoted CP-i for $i = 1, \ldots, 5$, connected all to the same router and each supplying different contents (i.e., files). We assume that each content j has a uniform size (1 MB for example) and a popularity $P_j \in [0; 1]$ reflecting the request frequency made by end-users for the content (i.e., the number of times end users issue 'interest' messages to retrieve the content) [2]. The sum of all file popularities in the network is equal to 1, i.e., $\sum_j P_j = 1$.

In the simulations, we model a practical network scenario of ICN with high heterogeneity in content popularity. The popularity of contents in the network are determined using the Zipf's law [32] that quantifies the frequencies of occurrence of the contents in the network (we set the Zipf's law exponent to 1). Each CP runs the LRU cache replacement policy that we model using the Che approximation [33].

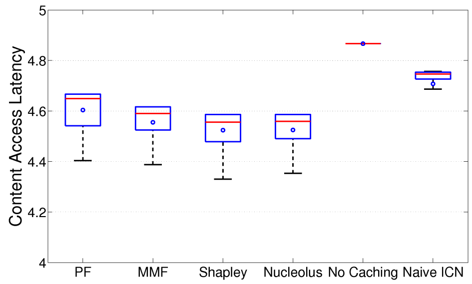
Contents are always delivered via a shortest path. We recall we assume that each content is offered by only one CP. To take into account network cases with a heterogeneous set of demands, we suppose that, among the five CPs, the CP-1 has the lowest demand d_1, and that CP-2, CP-3, CP-4, and CP-5 have, respectively, three, five, seven and nine times the demand of CP-1. The overall demand of CP-1 is set to 80 files (i.e., 80 MB).[5] The *Contention Level* in the network is then computed as: $C_L = 1 - (25C/\sum_{i=1}^{5} d_i)$.

We do compare the results under different allocation rules also for the case of a network without in-network caching, i.e., in which end user requests need to go all the way up from the edge to the CP containing the needed file at the network provider CP edge. Moreover, for the in-network caching cases, we include a naive cache allocation approach in which there is no router clustering and there is no CP-specific cache allocation [4]; instead, contents are delivered following the shortest path and cached on-the-fly by the LRU caches collocated on the traversed routers. As a reminder, we evaluate the four allocation schemes listed in Section 3.2: PF, MMF, Shapley Value, and Nucleolus. The following evaluation focuses on a performance analysis based on content access latency reduction, on fairness analysis and on the benefits of declaring truthful demands.

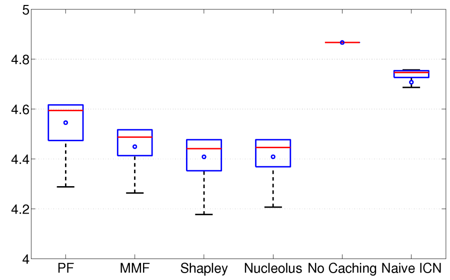
5.1. Content Access Latency

We evaluate the performance of different approaches with respect to the most important user's quality of experience metric i.e., the content access

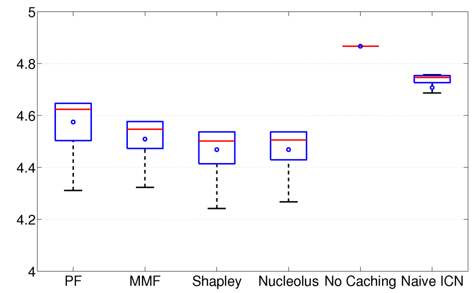
[5] We note that the demand of the CPs can be, in general, a generic function of the number of files, traffic volume and contents priorities.



(a) Router Proximity to network edge (RP)



(b) Router Degree (RD)



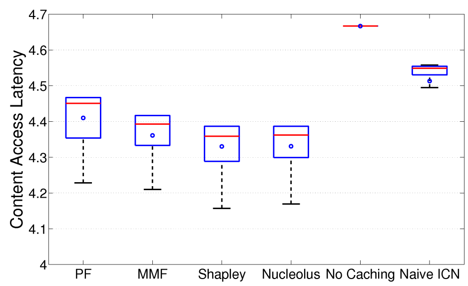
(c) Router Betweenness (RB)

Figure 3: Content access latency distributions for a tree topology and with different router clustering metrics.

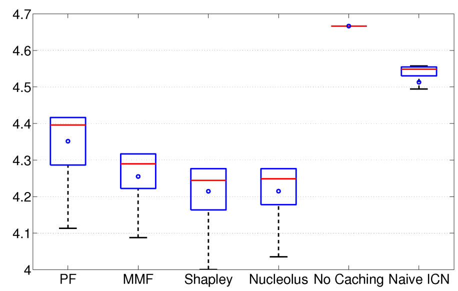
latency. We compute the average content access latency (expressed in number of hops) as a function of the edge-to-content path, and the average hit ratio on each router along the path as given by the Che approximation [33]. To model the case of high cache contention situation, we set C_L to 80% (i.e., the total cache space is equal to only 20% of the total CPs demands).

Fig. 3 and Fig. 4 show the boxplot statistics (max, min, quartiles, median as a red line, average as a star) of the content access latency for the network contents using the above mentioned metrics for the tree and partial mesh topology, respectively. We can notice that:

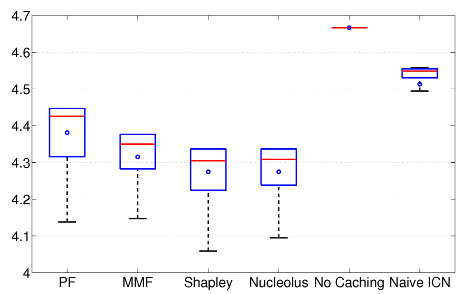
- Comparing in-network caching approaches to the one without caching, the former outperforms the latter one for all the cases; e.g., for the partial mesh topology and using the RD metric, the median content access latency decreases, from the approach without caching, by 9% with the game-theoretic approaches, 8% for the MMF, 4% with PF,



(a) Router Proximity to network edge (RP)



(b) Router Degree (RD)



(c) Router Betweenness (RB)

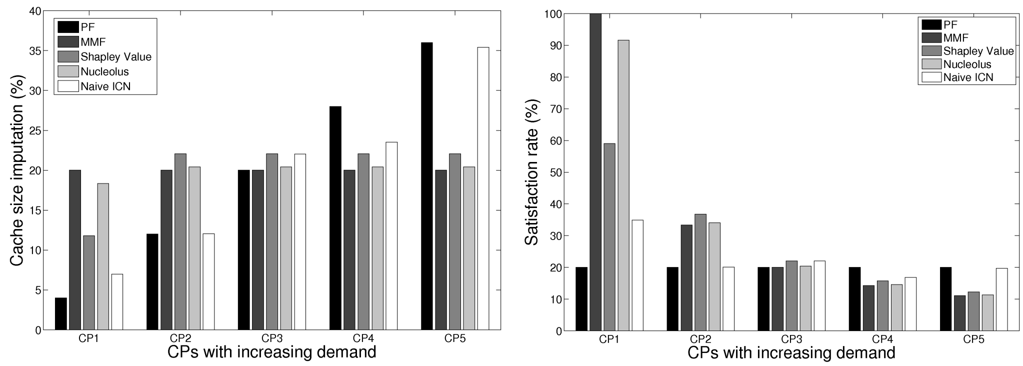
Figure 4: Content access latency distributions for a partial mesh topology and with different router clustering metrics.

and 2.5% for the naive ICN approach.

- Comparing the naive ICN approach to the router aggregation case with the four allocation rules, the content access latency decreases with the latter one for all the cases (e.g., for the partial mesh topology and RD metric, the median access latency decreases from basic ICN by 3% with PF, 5.6% with MMF and 6.5% for game-theoretic approaches).

- The game-theoretic approaches, Nucleolus and Shapley value, give very close performances for the different cases. They outperform the PF and MMF approaches for all the cases; e.g., for the tree topology and using the RB metric, the median content access latency is lower by 2.5% with respect to PF, and by 1.6% with respect to MMF.

- The partial mesh topology outperforms the tree one, likely because it allows multiple paths between network routers differently than the tree



(a) Cache size distribution

(b) Satisfaction rates

Figure 5: Cache size distribution and satisfaction rates, as a function of the CP demand, for a partial mesh topology using the RD metric.

topology with a single path from each router to the root.

- The RD router clustering metric outperforms the other metrics for all the in-network caching cases; e.g., in the mesh topology, the content access latency for the Nucleolus decreases from the RP by 3% and 1.25% to the RD and RB metrics, respectively. This somehow confirms previous findings of [9] where RD was shown to be superior to all other metrics. As a new insight, the gain of RD with respect to RB is less important than with respect to RP.

All in all, these highlights show that game-theoretic approaches increase content access performance. It is also worth mentioning that even if naive LRU driven in-network caching permits to reduce latency, it does not accomplish as much one could expect, mostly because of the potentially high replication of contents in the network [34].

5.2. Fairness of Cache Imputations

In order to further investigate on the cache allocation results, Figure 5 shows the imputation distribution (i.e., the ratio of the cache each CP obtains as a function of the total available cache) as well as the satisfaction rate (i.e., the ratio of the cache each CP obtains as a function of its demand), for the different allocation cases (PF, MMF, Nucleolus, Shapley value, and naive ICN). The partial mesh topology with the RD metric case is considered (similar results are obtained for the tree topologies). We can observe that the Nucleolus and Shapley value give the lowest claimant (i.e., CP-1) an imputation in-between those obtained by PF and MMF: CP-1 gets by Nucleolus and Shapley value 18% and 11% respectively of the total estate, while PF and MMF give respectively 5% and 20% of the total estate (20% corresponds actually to the totality of its demand, indeed the satisfaction rate of CP-1 is 100% with MMF). The same behavior can be seen also for the highest claimant (CP-5) whose imputation by Nucleolus and Shapley value is in-between those of MMF and PF. This indicates that game-theoretic approaches do not favor low demands as MMF does, or high demands as PF does, but instead distribute the estate in a way that discourages too greedy demands at the benefit of lower demands.

It is also worth noting that the naive approach with ICN is closer to the PF approach than the others. Intuitively, this can be explained by the fact that as the claim increases, the probability of finding claimant's files in the network likely proportionally increases.

Furthermore, in order to qualify the fairness of the solutions, we evaluate them with respect to two notable fairness indexes: the Jain's fairness index (J_I) [35] that rates the fairness of a set of values and defined as:

$$J_I = \left(\sum_{i=1}^{n} (x_i/d_i) \right)^2 / \left(n \sum_{i=1}^{n} (x_i/d_i)^2 \right) \tag{15}$$

which in fact has been conceived to be better the closer the solution is to the PF, and the Atkinson's index (A_I) [36] which is one of the commonly used measure of inequality, computed as follows:

$$A_I = 1 - \frac{n}{\sum\limits_{i=1}^{n} x_i} \left(\frac{1}{n} \sum_{i=1}^{n} x_i^{(1-\epsilon)} \right)^{1/(1-\epsilon)} \tag{16}$$

which conversely has been conceived to be better the closer the solution is to an even division ($A_I = 0$ means perfect equality while $A_I = 1$ expresses maximal inequality). ϵ is chosen in practice between 0.5 and 1.5 (we set a value of 1.5 in our case).

Figure 6 shows the fairness index results, as a function of the contention level C_L. We can state that:

- Fairness indexes confirm the close behavior between naive ICN and PF. Both appear as independent of the contention level - PF gives the best

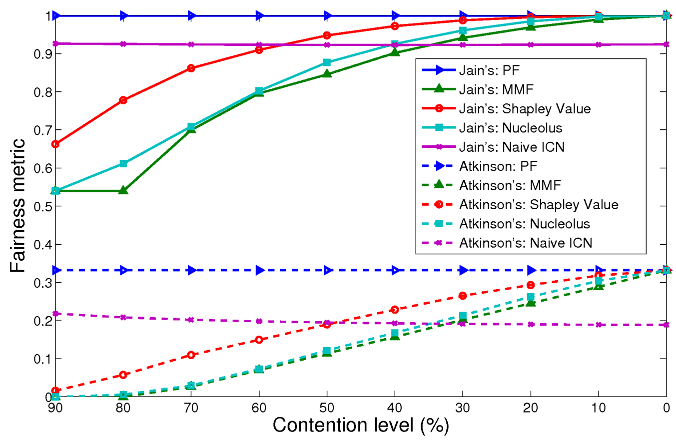


Figure 6: Fairness indexes as a function of the contention level (the lower the contention level, the higher the available cache size with respect to demands), for different allocation rules.

for the Jain's index and the worst for the Atkinson's one, and naive ICN gives better Atkinson's index values than PF.

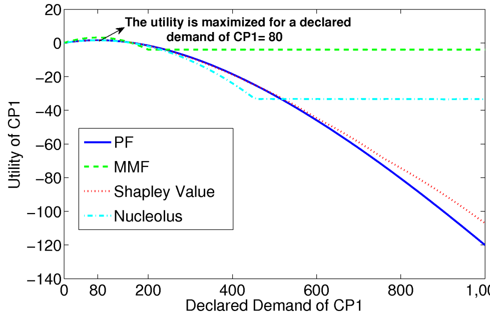
- Comparing the Nucleolus and the Shapley value for both metrics, the latter is strictly the closer one to the PF, while the former is closer to MMF. The gap between them, PF and MMF strictly decreases as the contention level decreases.

Overall, depending on the desired fairness behavior, PF or MMF, the network provider can refer to the Shapley as the one closer to PF, and the Nucleolus closer to MMF, being reassured about the fact that they bring a gain in terms of content access latency. Simply using the naive ICN approach would be a good approximation of the PF rule, with however a lower content access performance.

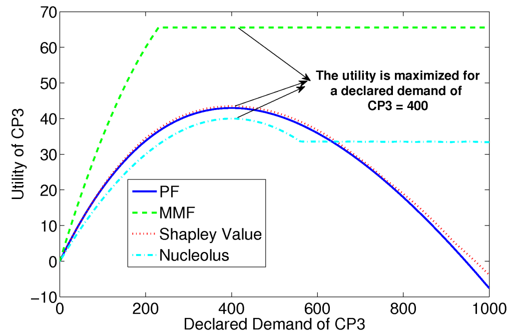
5.3. Utility Maximization by Truthful Declaration

The pricing criteria given in (8) is based on mechanism design theory. Its objective is to prevent the content providers to lie about their real demand value. In this subsection we study the utility of the content providers as function of their declaration. We consider the same simulation scenario where five CPs whose demands are given as follows:

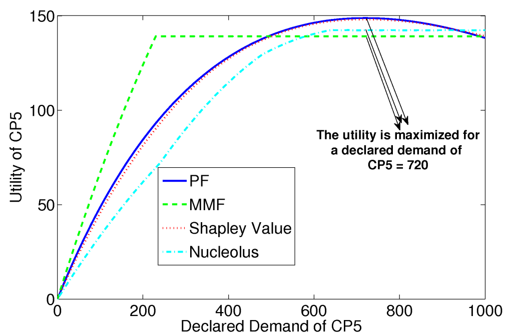
$$\mathbf{d} = 80 \times [1, 3, 5, 7, 9]^T,$$



(a) Utility of content provider CP1



(b) Utility of content provider CP3



(c) Utility of content provider CP5

Figure 7: The utility of different content providers as a function of their declared demands. The total available space is $E = 1000$

where d_i is the real demand of CP i. The price to pay for the allocated cache to a CP depends not only on the allocated space, but also on the claimed demand. Given the allocation and the price, the utility of the content provider is the difference between the value the CP evaluates the allocated space and the price the CP has to pay for the ISP (given by (7)). Figure 7 shows that, given that the pricing equation is known to all content providers, if any of the content providers declares a demand that is different from its real one ($b_i \neq d_i$), its utility does not increase. In other terms, the utilities of different content providers are maximized by announcing their real demands(e.g., the utility of CP1 is maximized when it declares a truthful demand that is equal to 80). This shows that the proposed pricing rule gives an equilibrium where the CPs have no incentives to deviate from declaring their truthful demands as they will not gain in terms of utility. That encourages all content providers to declare their real demands (Theorem 1). The figure also reveals some robustness properties of these equilibrium points. They show that the Shapley and PF provide more robust equilibrium than MMF and Nucleolus because shifting slightly away from the equilibrium point (by declaring slightly different demand than the truthful one) causes the utilities of the Shapley and PF to *strictly* decrease which is not always the case for MMF (see CP-3 and CP-5 utilities) and Nucleolus (see CP-5 utility).

5.4. ISP Profit

We further investigate the pricing rule for the different allocation schemes. The price is not designed to maximize the ISP profit, but rather to drive the CP to be truthful. However, different allocation schemes can give different profit. Figure 8 shows the total profit of the ISP as function of his total caching space (estate). In particular we identify some interesting points from the figure:

- The profit due to proportional fairness allocation does not change with increasing the estate, this is because equation (11) is independent of E. This shows that PF gives a "monopoly" pricing when the available cache space (the estate) is small because the ISP pricing in this case depends only on the CP demands with no considerations to the available caching space. Therefore, applying such a pricing rule in a multi-supplier market can lead to clients shifting to another ISP.

- MMF gives the lowest profit for the ISP. This is consistent with the interpretation that MMF favors, in its allocation, the low demand CPs that have less purchasing power with respect to CPs with high demand.

- The Shapley allocation provides a better profit than the Nucleolus and MMF for small estates. The profit is monotonically increasing with the estate size, however the slope of the profit is higher for low estate sizes (≤ 700) then it starts to decrease with high estate sizes.

- The profit due to Nucleolus provides an interesting behavior. It shows that the ISP profit increases with the estate until a point where it reaches a maximum, then it decreases again. From the ISP perspective, this counter-intuitive result shows that adding more cache space in the network can lead to lower profit. This can be interpreted by the fact that the pricing of Nucleolus balances between the fairness and the

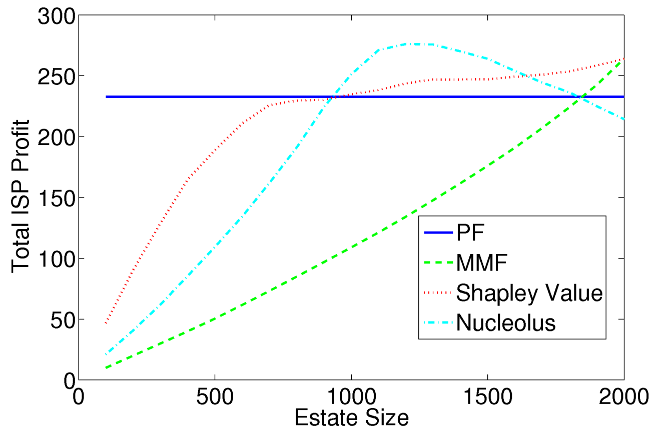


Figure 8: The total profit for the ISP for different allocation approaches.

contention level of CPs, so when the available cache size is high, the prices decrease to achieve fairness. It can also provide the ISP with an important information about how to dimension his network given the demands to maximize his profit. According to the figure, in our network scenario, the ISP should place around a total of 1200 MB available cache memory to CPs to maximize his profit.

6. Conclusion

Novel technologies are difficult to adopt as it has to be proven that they are incentive compatible for all the involved stakeholders. In this paper, we address a multi-stakeholder situation (i.e., involving more than one provider) that appears as a win-win setting toward ICN deployment, i.e., the case of an Internet Network Service Provider deploying ICN for external content providers, offering a neutral interface and pricing to multiple content providers. The network cache provider hence allocates to external content providers spaces in its ICN router caches for content delivery.

In this context, we argue that the proper way the network cache provider shall design the cache allocation framework and model the behavior of external content providers is game theory, so as to qualify and counter-balance their natural tendency to form oligopolies and to ally to have a stronger position in getting the available caching resources. We investigate the application of well-known concepts from cooperative game-theory showing desirable properties, the Nucleolus and the Shapley value, as well as other principles commonly adopted in networking research, the proportional fairness (PF) and the max-min fairness (MMF). We propose a cache allocation algorithm, applied in the context of ICN, that can be performed upon significant changes of content providers' demands. This algorithm is able to incorporate these different allocation rules applying them to clusters of routers ordered with respect to centrality metrics suggested in the literature. Moreover, we propose a pricing framework that, taking advantages of the monotonicity of the presented cache allocation rules, correctly nullifies the threat of malicious behaviors in formulating content caching demands.

Results from simulations show that the game-theoretic approaches offer a (not so straightforward) sensible access latency gain with respect to both PF and MMF, and the naive ICN approach (without cache allocations and using least-recently-used cache management) to content providers. Among the Nucleolus and the Shapley value approaches, the former could be considered more interesting given that it maximizes the ISP profit for a well dimensioned caching space in the network. In terms of fairness, the Nucleolus and the Shapley values sit in-between PF and MMF allocation rules, balancing their well-known weaknesses and strengths, so that the Shapley value is close to PF and the Nucleolus very close to MMF. It is also valuable to report that the naive ICN approach permits to approximate PF without having to compute cache imputation (at the expense, however, of worse content access performance). Moreover, we show that declaring truthful demands yields better CPs' utilities for the different cache imputations where the Shapley and PF are more robust than Nucleolus and MMF in terms of utility maximization for truthful declaration.

As further work, we are planning to generalize the results to settings where the content providers have overlapping contents and the contents from the content providers are dynamic. The positive performance of the game-theoretic approaches, which balance the strengths and weakness of both PF and MMF in terms of fairness, opens the way to revisiting former applications of PF and MMF to other networking situations (scheduling, load-balancing, resource reservation), in which behind the network decision rational and independent agents can be identified.

Acknowledgment

This work was partially supported by the EU FP7 IRSES MobileCloud Project (Grant No. 612212).

7. References

[1] C. Labovitz, S. Iekel-Johnson, D. McPherson, J. Oberheide, and F. Jahanian. Internet inter-domain traffic. *SIGCOMM Comput. Commun. Rev.*, 40(4):75–86, August 2010.

[2] G. Xylomenos, C. N. Ververidis, V. A. Siris, N. Fotiou, C. Tsilopoulos, X. Vasilakos, Konstantinos V. Katsaros, and G. C. Polyzos. A Survey of Information-Centric Networking Research. *IEEE Communications Surveys and Tutorials*, 16(99):1–26, 2013.

[3] E. Nygren, R. K. Sitaraman, and J. Sun. The akamai network: A platform for high-performance internet applications. *SIGOPS Oper. Syst. Rev.*, 44(3):2–19, August 2010.

[4] V. Jacobson, D. K. Smetters, J. D. Thornton, M. F. Plass, N. H. Briggs, and R. L. Braynard. Networking named content. In *Proc. of the 5th International Conference on Emerging Networking Experiments and Technologies*, CoNEXT '09, New York, NY, USA, 2009. ACM.

[5] A. Detti, M. Pomposini, N. Blefari-Melazzi, S. Salsano, and A Bragagnini. Offloading cellular networks with information-centric networking: The case of video streaming. In *World of Wireless, Mobile and Multimedia Networks (WoWMoM), 2012 IEEE International Symposium on a*, June 2012.

[6] Guoqiang Zhang, Yang Li, and Tao Lin. Caching in information centric networking: A survey. *Computer Networks*, 57(16):3128 – 3141, 2013.

[7] J. Bulow and J. Roberts. The Simple Economics of Optimal Auctions. *Journal of Political Economy*, 97(5):1060–90, January 1989.

[8] N. Nisan, T. Roughgarden, E. Tardos, and V. V. Vazirani. *Algorithmic Game Theory.* Cambridge University Press, New York, NY, USA, 2007.

[9] D. Rossi and G. Rossini. On sizing ccn content stores by exploiting topological information. In *Computer Communications Workshops (INFOCOM WKSHPS), 2012 IEEE Conference on*, March 2012.

[10] Y. Wang, Z. Li, G Tyson, S. Uhlig, and G. Xie. Allocation for Content-Centric Networking. *IEEE Intl. Conference on Network Protocols*, 2013.

[11] S. K. Fayazbakhsh, Y. Lin, A. Tootoonchian, A. Ghodsi, T. Koponen, B. Maggs, K.C. Ng, V. Sekar, and S. Shenker. Less pain, most of the gain: Incrementally deployable icn. In *Proc. of the ACM SIGCOMM 2013 Conference on SIGCOMM*, SIGCOMM '13, New York, NY, USA, 2013. ACM.

[12] G. Owen. *Game Theory*. Academic, New York, 1982.

[13] V. Pacifici and G. Dan. Selfish content replication on graphs. In *Proc. of Teletraffic Congress (ITC), 2011 23rd International*, Sept 2011.

[14] B. G. Chun, K. Chaudhuri, H. Wee, M. Barreno, C. H. Papadimitriou, and J. Kubiatowicz. Selfish caching in distributed systems: A game-theoretic analysis. In *Proc. of the Twenty-third Annual ACM Symposium on Principles of Distributed Computing*, PODC '04, New York, NY, USA, 2004.

[15] E. Jaho, M. Karaliopoulos, and I Stavrakakis. Social similarity favors cooperation: The distributed content replication case. *Parallel and Distributed Systems, IEEE Transactions on*, 24(3):601–613, March 2013.

[16] D. Saucez, S. Secci, and C. Barakat. On the Incentives and Incremental Deployments of ICN technologies for OTT Services. *IEEE Network Magazine*, 28(3):20–25, May 2014.

[17] R. J. Aumann and M. Maschler. Game theoretic analysis of a bankruptcy problem from the Talmud. *Journal of Economic Theory*, 36(2):195–213, August 1985.

[18] S. Hoteit, S. Secci, R. Langar, and G. Pujolle. A nucleolus-based approach for resource allocation in ofdma wireless mesh networks. *Mobile Computing, IEEE Transactions on*, 12(11):2145–2154, Nov 2013.

[19] H. J. Kushner and P. A. Whiting. Convergence of proportional-fair sharing algorithms under general conditions. *Wireless Communications, IEEE Transactions on*, 3(4):1250–1259, July 2004.

[20] S. Keshav. *An Engineering Approach to Computer Networking: ATM Networks, the Internet, and the Telephone Network*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 1997.

[21] L. S. Shapley. A value for n-person games. In *The Shapley value*, pages 31–40. Cambridge University Press, 1988. Cambridge Books Online.

[22] M. Mycek, S. Secci, M. Pioro, J. Rougier, A Tomaszewski, and A Pattavina. Cooperative multi-provider routing optimization and income distribution. In *Design of Reliable Communication Networks, 2009. 7th Int. Workshop on*, Oct 2009.

[23] T. Alpcan and T. Başar. A game theoretic approach to decision and analysis in network intrusion detection. In *42nd IEEE Conference on Decision and Control CDC 2003*, pages 2595–2600, Dec 2003.

[24] D. Schmeidler. *The Nucleolus of a Characteristic Function Game*. Defense Technical Information Center, 1966.

[25] Zhu H. and H.V. Poor. Coalition games with cooperative transmission: a cure for the curse of boundary nodes in selfish packet-forwarding wireless networks. *Communications, IEEE Transactions on*, 57(1):203–213, January 2009.

[26] W. Saad, H. Zhu, M. Debbah, A Hjorungnes, and T. Basar. Coalitional game theory for communication networks. *Signal Processing Magazine, IEEE*, 26(5):77–97, September 2009.

[27] J. Castro, D. Gomez, and J. Tejada. Polynomial calculation of the shapley value based on sampling. *Computers & Operations Research*, 36(5), 2009.

[28] J. Derks and J. Kuipers. *Implementing the Simplex Method for Computing the Prenucleolus of Transferable Utility Games*. Reports in operations research and systems theory. Maastricht University, 1996.

[29] R. B. Myerson. Optimal auction design. *Mathematics of Operations Research*, 6(1):58–73, 1981.

[30] RodneyJ. Garratt, Mark Walker, and John Wooders. Behavior in second-price auctions by highly experienced ebay buyers and sellers. *Experimental Economics*, 15(1):44–57, 2012.

[31] Benjamin Edelman, Michael Ostrovsky, and Michael Schwarz. Internet Advertising and the Generalized Second-Price Auction: Selling Billions of Dollars Worth of Keywords. *American Economic Review*, 97(1):242–259, March 2007.

[32] G.K. Zipf. *Human behavior and the principle of least effort: an introduction to human ecology*. Addison-Wesley Press, 1949.

[33] H. Che, Y. Tung, and Z. Wang. Hierarchical web caching systems: Modeling, design and experimental results. *IEEE J.Sel. A. Commun.*, 20(7):1305–1314, September 2006.

[34] W. K. Chai, D. He, I. Psaras, and G. Pavlou. Cache "less for more" in information-centric networks. In *Proc. of the 11th International IFIP TC 6 Conference on Networking - Volume Part I*, IFIP'12, Berlin, Heidelberg, 2012. Springer-Verlag.

[35] R. Jain, D.M. Chiu, and W.R. Hawe. *A Quantitative Measure of Fairness and Discrimination for Resource Allocation in Shared Computer System*. Eastern Research Laboratory, Digital Equipment Corporation, 1984.

[36] A. B. Atkinson. On the measurement of inequality. *Journal of Economic Theory*, 2(3):244–263, September 1970.