Efficient Joint Bandwidth and Cache Leasing in Information Centric Networks

Michele Mangili[*‡], Fabio Martignon[*§], Stefano Paris[†] and Antonio Capone[‡]

[*] LRI, Université Paris-Sud
michele.mangili@lri.fr
fabio.martignon@lri.fr
[§] IUF, Institut Universitaire de France

[†] Université Paris-Descartes
Lab. d'Informatique Paris Descartes
stefano.paris@parisdescartes.fr

[‡] Politecnico di Milano
DEIB, Dip. Elettronica,
Informazione e Bioingegneria
capone@elet.polimi.it

***Abstract*—Information Centric Networking (ICN) is a novel paradigm that aims at improving the performance of today's Internet by supporting universal caching and multicast content delivery features on every network device.**

In this paper we propose a strategy to stimulate third parties to jointly lease the unused bandwidth and storage available on wireless access points in an ICN. We formulate this problem as a combinatorial reverse auction run by a content provider willing to increase the number of users reached by his service. We first show that the optimal allocation algorithm is NP-hard, we then provide greedy heuristics that guarantee the *individual rationality* and *truthfulness* properties and we compare their performance numerically. Finally, we evaluate the benefits of our proposed mechanisms in terms of the computational time necessary to execute the allocation algorithms, as well as the cost savings for the content provider obtained by offloading the backhaul connections using the distributed caches to directly serve users' requests.

The novelty of our approach is to simultaneously take into account in the mechanism design the bandwidth as well as the storage available at the access network, and to study their mutual interaction.

I. Introduction

The way customers use Internet nowadays has radically changed with respect to the original design goals that had driven the development of the TCP/IP protocol stack [1]. As a matter of fact, although the Internet was initially conceived as a means to establish a point-to-point communication between a client machine and a remote server, it is currently used mostly as a content distribution infrastructure [2].

In order to better handle this usage shift, new Information Centric Networking (ICN) design proposals for the Future Internet are recently gaining momentum. These designs have specifically been formulated to create new protocols centered around the concept of *what* the user is willing to retrieve, rather than *where* the data can be found. In fact, their distinctive characteristic is that they replace at the packet level the source and destination addresses with content names. Routing and forwarding are thus performed according to the name of the data that has to be delivered [3].

In this paper we analyze a relevant scenario where the content provider has the simultaneous objectives of 1) extending his mobile customer base by offering the users an ubiquitous access to the provided content and 2) performing server offloading by exploiting the built-in caching features of ICNs.

In our vision, the user buys a digital content, and the content provider offers the connectivity to retrieve the corresponding data. As an example, we believe that this *"wireless shopping"* business model can be effectively applied to online content stores such as e-book libraries, music and video streaming services, online magazines and newspapers as well as application stores. Some steps towards this direction have been done by Amazon, which is providing to Kindle users the "Free 3G" mobile broadband connection service to wirelessly browse the store, purchase and download the content.

Since the content provider should not bear the costs of the realization of a new access network, we propose the creation of a marketplace where third party access point owners will offer their unexploited bandwidth and storage resources as they will receive economic incentives for their cooperation. However, misbehaving access point owners may jeopardize the efficiency of the allocation mechanism by choosing strategically to declare false valuations for the offered resources. In order to solve this issue, *auction theory* provides insights for the design of tamper-proof mechanisms characterized by the fact that the dominant strategy of every bidder is to declare the real valuation for the provided resources.

The contribution of this paper is threefold:

1) We design an optimal mechanism that can be used to motivate access point owners to jointly lease their unused bandwidth capacities and cache storage, in exchange for economic incentives. The mechanism is a *reverse auction* that guarantees the *individual rationality* and *truthfulness* properties as it forces the owners to declare their real valuations for the provided resources.
2) We show that the proposed optimal algorithm is NP-hard. In order to cope with the computational complexity, we then provide three variants of a greedy algorithm that can be used to obtain a sub-optimal solution in polynomial time.
3) We provide performance comparisons of the proposed strategies. We show that all the variants of the greedy algorithm outperform the optimal one in terms of execution time; however, this comes at an increased economical cost that the provider needs to face, due to the sub-optimal property of the discovered solution.

This paper is structured as follows: in Sec. II we discuss related work. Sec. III describes the network architecture and motivates our proposal. Sec. IV formulates the optimal combinatorial reverse auction as an optimization model, while

Sec. V proposes three greedy algorithms to solve the allocation problem in polynomial time. Numerical results are analyzed in Sec. VI. Finally, concluding remarks are discussed in Sec. VII.

II. Related Work

In this section, we briefly survey the literature on Information Centric Networks (ICNs). Sec. II-A reviews the ICN design principles, while Sec. II-B presents auction mechanisms that can be extended to ICNs in order to stimulate resource leasing.

A. Information Centric Network Solutions

Information Centric Networks have been designed with the ambitious goal to provide a new networking paradigm that could better accommodate new performance, security, mobility and scalability requirements imposed on the Internet by its users. Several architectures for ICNs have been proposed in the literature and in ongoing research projects: DONA [4], CONET [5], PURSUIT [6], and NDN/CCN [7].

For the sake of simplicity, we specifically design our proposal for NDN/CCN (Named Data Networking/Content-Centric Networking), since a widely used and open implementation of this model is available[1]. Additionally, to the best of our knowledge, it is the proposal that has received most of the attention from the scientific community. However, we would like to point out the fact that our model can also be adapted to other ICN designs in a straightforward way, since the only requirement that needs to be enforced is the possibility to implement content caching on an access point.

Addressing the content rather than the location has the main advantage of improving the performance by making universal in-network caching as well as multicast delivery easily implementable in the network architecture [8]. For this reason, it becomes vital to understand the effect of in-network caching in order to study the behavior of an ICN. In particular, in [9], Fricker et al. propose a model to compute the hit rate for a single and two-layer cache hierarchy, given a Zipf content popularity distribution. While, Psaras et al. [10] studied caching performance for a tree-like CCN topology, by means of representing the state of every router using Markov chains.

B. Auction Theory in Communication Networks

Auction theory has been used to design efficient allocation mechanisms in several network contexts and it is extremely appealing to model the problem of spectrum leasing to secondary users tackled in cognitive radio networks [11], [12]. Besides these classical scenarios, auctions are becoming an even more interesting tool to model bandwidth allocation.

Dai et al. in [13] presented a collaborative caching auction system for wireless video streaming based on the Vickrey-Clarke-Groves (VCG) mechanism. Their proposal fosters the cooperation between cache servers by ensuring that every bidder declares his real private valuation for the auctioned resource, thus ensuring the *truthfulness* property.

Unlike the surveyed literature, our approach not only takes into account the available bandwidth but it also considers the

[1]https://www.ccnx.org

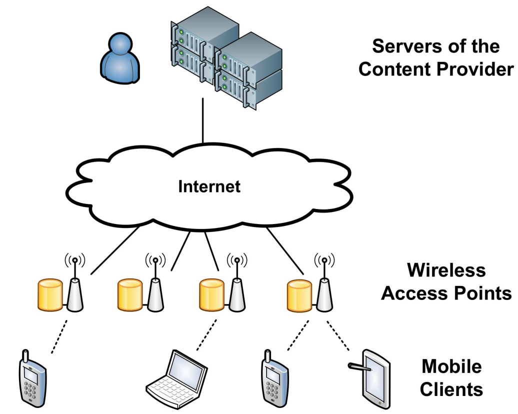


Figure 1. *Network architecture*. A single content provider (*CP*) is considered in our design. Access points (*APs*) are equipped with caching storage. Mobile clients (*MCs*) connect to a subset of the available *APs*. The demand of a *MC* will be satisfied either by cached content, or by retrieving the data directly from the servers of the *CP* using the available backhaul Internet connection. Our mechanism derives the *MCs* to *APs* allocation that minimizes the total cost.

storage space offered by every access point owner. Given the fact that we study the performance of this mechanism for a CCN network, the availability of caches not only reduces the backhaul capacity used at every access point, but it is also beneficial to the content provider himself, since it offloads its servers.

III. System Model

In this section we illustrate the principles, definitions and assumptions characterizing the communication network of our scenario. Sec. III-A will describe the network architecture and discuss the benefits provided by our allocation algorithm to the content provider, while Sec. III-B will clarify the structure of the economic incentives to the access point owners and explain the properties enforced by the auction mechanism.

A. Advantages for the Content Provider

A graphical representation of the system model we consider in this paper is shown in Fig. 1. A single content provider (CP) wants to lease a set of wireless access points (APs) in order to increase his customer base by providing an ubiquitous content delivery service to mobile clients (MCs).

The core business of the provider is to distribute contents that will be accessed by the customers under payment of a fee. The hardware infrastructure owned by the CP does not comprise the radio access network, since it is beyond the scope of the service it provides; however, we assume that the CP owns a set of content distribution servers reachable through any Internet connection.

Raising the number of customers reached by the service jointly increases the *remuneration* of the CP as well as the *operational costs* (OPEX) due to the increased load of the computational infrastructure. An interesting feature offered by ICN that can mitigate the effects of this trade-off is *universal caching*: each AP is equipped with a variable quantity of storage that is used to memorize content previously forwarded to the destinations. For this reason, we propose that the operator leases not only the radio access and backhaul connection capacities, but also portions of the caches available at the access points. By exploiting the caching feature of ICN we make the content move across the network towards the locations where most of the users are requesting it.

B. Economic Incentives to the Access Point Owners

We assume that the APs are owned by third party users. They can participate to the bandwidth allocation by submitting to the CP the bid $[b_j, s_j]$ where b_j is the price at which the j-th AP agrees to jointly share the unexploited bakhaul and wireless bandwidth as well as a quantity s_j of cache. The main difference with respect to classical optimization techniques is that in our case the real valuation v_j of each owner is kept hidden; thus, in the most general case, $v_j \neq b_j$.

Let $p_j \in \mathbb{R}^+$ be the price paid by the CP to lease the j-th AP. The utility function of the AP owner is such that:

$$u_j = \begin{cases} p_j - v_j & \text{if AP } j \text{ is selected} \\ 0 & \text{otherwise.} \end{cases} \quad (1)$$

We say that *individual rationality* holds if the utility of each player is always non-negative: $\forall j \in \mathcal{A}, u_j \geq 0$. When the access point is not selected, the owner's utility is null since he doesn't incur additional costs given by incoming traffic and at the same time he doesn't receive any economic incentive. Meanwhile, a rational player will choose to play only if $p_j \geq v_j$ and thus $u_j \geq 0$.

When the CP has collected all the sealed bids of the APs, it will in turn select a set of access points that should be rewarded, with the joint aim to satisfy the traffic demands d_i of the mobile clients and minimizing the sum of incentives paid to the AP owners and the operational cost (OPEX) of the computational infrastructure. Thus, we would like to design a two step mechanism that 1) chooses which wireless access points should be selected among those that participated to the allocation, and 2) computes the rewards paid to the corresponding winners, in such a way that every user is forced to declare a price equal to the true valuation of his offer.

We make the assumption that each access point is directly connected to a backhaul Internet connection. For the time being, we assume that this is a symmetric channel whose capacity is known to the operator. We also assume that it is very unlikely that the AP owner declares a false quantity of available cache storage since this misbehavior can be easily detected and punished by the content provider. Notwithstanding, the mechanism needs to force the AP owners to declare their real valuations, since they may be tempted to lie in order to increase their utilities by behaving strategically.

IV. Optimal Allocation and Payment Scheme

This section illustrates the optimal mobile clients allocation and access point payment algorithm. We will derive the optimal allocation rule in terms of: 1) the prices paid by the content provider (CP) to remunerate the selected access point owners and 2) the operational cost (OPEX) faced by the CP due to the load on his computational infrastructure. The notation used in this paper is summarized in Table I.

Table I. Summary of the notation used in this paper.

Parameters of the ILP model	
$\mathcal{M}$	Set of Mobile Clients
$\mathcal{A}$	Set of Access Points
b_j	Bid of AP j
s_j	Storage space for caching offered by AP j
h_j	Average hit-rate for AP j
R_j	Backhaul bandwidth available at AP j
L_j	Maximum coverage radius of AP j
$l_{i,j}$	Distance between MC i and AP j
$r_{i,j}$	Maximum Wi-Fi rate of the MC i when it is connected to AP j
d_i	Bandwidth demand of MC i
C	Cache miss cost per unit of bandwidth, paid by the CP

Variables of the ILP Model	
$x_{i,j}$	Binary variable that indicates whether MC i is assigned to AP j
y_j	Binary variable that indicates whether AP j is selected or not

Parameters of the Auction	
v_j	Private valuation of AP j
p_j	Actual price paid by the CP to the j-th AP owner
u_j	Utility function of the j-th AP owner

We denote with $\mathcal{M}$ the set of mobile clients (MCs), while $\mathcal{A}$ is the set of access points (APs). Each MC $i \in \mathcal{M}$ generates a traffic demand d_i that might be satisfied by at most one AP $j \in \mathcal{A}$.

Due to locality constraints, the APs have a limited maximum coverage radius, that we denote with L_j for AP j. Let $l_{i,j}$ be the distance between MC i and AP j, and let $r_{i,j}$ be the maximum rate of MC i when connected to AP j. If MC i is beyond the range of AP j, the corresponding rate will be null: $r_{i,j} = 0, \forall\, i \in \mathcal{M},\ j \in \mathcal{A} \mid l_{i,j} > L_j$.

In order to participate to the auction, AP owners are required to declare s_j, the amount of storage they will offer for caching purposes. This quantity will be used by the content provider (CP) in order to compute the average cache hit rate for AP j, denoted by h_j. If $h_j = 0$ this means that AP j does not cache any object.

Cache misses increase the load on the computational infrastructure of the CP. We denote with C the cost per unit of bandwidth due to cache misses. Lastly, let R_j be the bandwidth of the backhaul connection available at AP j.

The binary variable $x_{i,j} \in \{0, 1\}$ is used to represent the assignment between MC i and AP j, and is such that:

$$x_{i,j} = \begin{cases} 1, & \text{if MC } i \text{ is assigned to AP } j \\ 0, & \text{otherwise.} \end{cases} \quad (2)$$

Furthermore, we denote with the binary variable y_j the set of APs selected by the auction:

$$y_j = \begin{cases} 1, & \text{if AP } j \text{ was selected by the auction} \\ 0, & \text{otherwise.} \end{cases} \quad (3)$$

Given the above definitions and assumptions, the mobile clients allocation problem (MCAP) can be formulated as follows:

$$\min \sum_{\forall j \in \mathcal{A}} y_j b_j + \sum_{\substack{\forall i \in \mathcal{M} \\ \forall j \in \mathcal{A}}} x_{i,j} d_i (1 - h_j) C \tag{4}$$

s.t.

$$\sum_{\forall j \in A} x_{i,j} = 1 \qquad \forall i \in \mathcal{M} \tag{5}$$

$$\sum_{\forall i \in \mathcal{M}} \frac{d_i x_{i,j}}{r_{i,j}} \leq 1 \qquad \forall j \in A \tag{6}$$

$$\sum_{\forall i \in \mathcal{M}} d_i x_{i,j} (1 - h_j) \leq R_j \qquad \forall j \in A \tag{7}$$

$$x_{i,j} \leq y_j \qquad \forall i \in \mathcal{M}, j \in \mathcal{A} \tag{8}$$

$$x_{i,j} \in \{0, 1\} \qquad \forall i \in \mathcal{M}, j \in \mathcal{A} \tag{9}$$

$$y_j \in \{0, 1\} \qquad \forall j \in A \tag{10}$$

The objective function (4) minimizes the total cost of the content provider, which is given by:

1) the incentives paid to remunerate the selected access points:
$$\sum_{\forall j \in \mathcal{A}} y_j b_j; \tag{11}$$
2) the infrastructure costs due to cache misses:
$$\sum_{\substack{\forall i \in \mathcal{M} \\ \forall j \in \mathcal{A}}} x_{i,j} d_i (1 - h_j) C. \tag{12}$$

Eq. (5) expresses the set of *full coverage* constraints which force every mobile client to be assigned to exactly one access point.

Constraints (6) and (7) limit the number of MCs assigned to each AP, given that the radio access network and the backhaul Internet connection have a bounded capacity. While constraints (6) impose that the total client demand served by an access point doesn't exceed the capacity of the radio access network, constraints (7) consider the fact that the backhaul Internet connection serves only the aggregate demand that generates a cache miss.

Constraints (8) make sure that only the APs that are serving at least one MC are selected by the mechanism. Finally, the set of constraints (9) and (10) express the integrality conditions on the decision variables.

After having defined the ILP model used to solve MCAP, we now illustrate the algorithm that forces the AP owners to bid their real valuations. *Algorithm 1* formalizes the steps that the auctioneer should follow in order to determine the APs that should be selected as well as their payments for the provided resources. It returns the list of selected APs y_j, the assignment matrix of MCs to APs x_{ij} and the corresponding payments p_j, where as usual $i \in \mathcal{M}, j \in \mathcal{A}$.

The algorithm proceeds in three steps. In step 1, the ILP model is solved and the minimum cost allocation $(y_j, x_{i,j})$ is computed. In step 2 the solution of the ILP model without the j-th AP is determined, in other terms, the algorithm solves again the ILP model (Eq. (4)-(10)) with the additional constraint $y_j = 0$. Finally, step 3 computes the optimal price for the j-th AP according to the Vickrey-Clarke-Groves mechanism with Clarke pivot rule,

Algorithm 1: Optimal and Truthful Reverse Auction

Input : $\mathcal{M}, \mathcal{A}, b_j, s_j, d_i, r_{ij}, h_j, R_j$
Output: y_j, p_j, x_{ij}
1 $(y_j, x_{ij}) \Leftarrow$ Solve the ILP model ;
foreach $j \in \mathcal{A} : y_j = 1$ **do**
2 $(y_j^{'}, x_{ij}^{'}) \Leftarrow$ Solve the ILP model without AP j ;
3 $p_j \Leftarrow$ $\sum_{\substack{n \in \mathcal{A} \\ n \neq j}} \left(y_n^{'} b_n^{'} - y_n b_n \right) + \sum_{\substack{i \in \mathcal{M} \\ n \in \mathcal{A}}} (x_{in}^{'} - x_{in}) d_i (1 - h_n) C$;
end

which ensures *individual rationality* and *truthfulness* [14]. The optimal price represents the "*opportunity cost*" that the presence of the j-th AP causes to the other bidders.

It can be shown that Alg. 1 is NP-hard, in fact, it is straightforward to derive a polynomial time procedure that can reduce the Multidimensional Knapsack Problem to the ILP model (4)-(10), as their structure is similar. Therefore, since solving the MKP is at least as difficult as solving the optimal allocation problem, and MKP is known to be NP-hard [15], Alg. 1 is NP-hard.

V. Greedy Algorithms

The computational complexity of Alg. 1 is such that, as the number of MCs m and APs a increases, the completion time of the optimal algorithm raises exponentially. In order to find an allocation strategy that can effectively scale up to large network instances, this section provides a computationally efficient, polynomial-time algorithm with three alternative strategies, that still guarantees *truthfulness* and *individual rationality*. We begin by describing the *greedy MC* alternative, while we will talk about the other greedy strategies at the end of the section.

Let $k_j = \frac{b_j}{|\mathcal{M}_j|}$ be the ratio between the bid of the j-th AP (b_j) and the number of MCs that it can potentially serve $|\mathcal{M}_j|$. The quantity k_j can be interpreted as the price per mobile client that should be paid to the j-th AP if it served all the MCs in its range. Let c be the index of the *critical access point*, which is defined as the first AP that has not been selected by the mechanism.

In the same way as in the optimal auction, the greedy algorithm (*Algorithm 2*) is characterized by the following two phases: 1) the *allocation* phase, which selects the APs characterized by the lowest k_j until all the demands of MCs are satisfied and 2) the *payment* phase, which determines the price paid to the selected APs as a function of the critical access point c.

Algorithm 2: Greedy Cache and Bandwidth Auction

Input : $\mathcal{M}, \mathcal{A}, b_j, s_j, d_i, r_{ij}, h_j, R_j$
Output: y_j, p_j, x_{ij}
1 $(y_j, x_{ij}, c) \Leftarrow$ Greedy_Allocation($\mathcal{M}, \mathcal{A}, b_j, s_j, d_i, r_{ij}, h_j, R_j$);
foreach $j \in \mathcal{A} : y_j = 1$ **do**
2 $p_j \Leftarrow \frac{b_c}{|\mathcal{M}_c|} |\mathcal{M}_j|$;
end

Algorithm 3: Greedy Allocation (Step 1 of Alg. 2)

Input : $\mathcal{M}, \mathcal{A}, b_j, s_j, d_i, r_{ij}, h_j, R_j$
Output: y_j, x_{ij}, c
1 $L \Leftarrow Sort\left(j \in \mathcal{A}, \frac{b_j}{|\mathcal{M}_j|}, \text{“non-decreasing”}\right)$;
$U \Leftarrow \mathcal{M}$;
while $U \neq \emptyset$ **do**
  $j \Leftarrow Next(L)$; $y_j \Leftarrow 1$;
2   **while** *Is_Feasible_Solution*$(\mathcal{M}, \mathcal{A}, b_j, s_j, d_i, r_{ij}, h_j, R_j)$ **do**
    $\mathcal{V}_j \Leftarrow \text{Sort}\left(i \in \mathcal{M}_j, \frac{d_i}{r_{ij}}, \text{“non-decreasing”}\right)$;
    $i = \text{Next}(\mathcal{V}_j)$;
    if $y_i = 0$ **then**
      $x_{ij} \Leftarrow 1$; $U = U \setminus \{i\}$;
    end
  end
end
3 $c \Leftarrow Next(L)$;

The greedy allocation of MCs to APs, as well as the identification of the critical access point c, are performed by *Algorithm 3*. The allocation procedure starts by sorting the APs in non-decreasing order according to the corresponding k_j values (Step 1). This is done with the aim of selecting the *“most promising”* APs in terms of the declared price per number of reachable mobile clients. We then iteratively allocate to the APs the largest number of unassigned mobile clients by choosing the MCs which have lower capacity demand $\frac{d_i}{r_{ij}}$, while still preserving the feasibility of the solution (Step 2). Alg. 3 completes the execution by assigning all the mobile clients to a subset of the available APs, and by determining the critical access point c (Step 3), where the critical access point is the first AP that hasn't been selected by the algorithm (the first looser).

Finally, *Algorithm 4* checks if the solution is feasible, by determining whether the constraints of the ILP model hold. In detail, Step 1 checks if the MC is assigned to at most one AP as imposed by constraints (5), Step 2 checks whether the available Wi-Fi bandwidth is not saturated as in constraints (6), and lastly Step 3 checks if the backhaul connection can accommodate the allocated traffic as in constraints (7).

Alg. 2 implements a *truthful* auction since 1) the allocation phase respects the monotonicity property as the APs are sorted in non-increasing order of their bids per number of covered mobile customers, and 2) there exists a critical value which determines whether the AP has been selected or not. As demonstrated in [14] the previous two conditions ensure the truthfulness of the greedy algorithm.

By jointly substituting Step 2 of Alg. 2 and Step 1 of Alg. 3,

Algorithm 4: Is Feasible Solution (Step 2 of Alg. 3)

Input : $\mathcal{M}, \mathcal{A}, b_j, s_j, d_i, r_{ij}, h_j, R_j$
Output: t
1 $c_1 \Leftarrow \sum_{\forall j \in A} x_{i,j} \ \leq \ 1$;
2 $c_2 \Leftarrow \sum_{\forall i \in \mathcal{M}} \frac{d_i x_{i,j}}{r_{i,j}} \ \leq \ 1$;
3 $c_3 \Leftarrow \sum_{\forall i \in \mathcal{M}} d_i x_{i,j}(1 - h_j) \ \leq \ R_j$;
$t \Leftarrow c_1 \wedge c_2 \wedge c_3$;

Table II. SUMMARY OF THE 3 PROPOSED VARIANTS FOR THE GREEDY ALGORITHM

Greedy Variant	Step 2, Alg. 2	Step 1, Alg. 3
MC (G.m.)	$p_j \Leftarrow \frac{b_c}{|\mathcal{M}_c|}|\mathcal{M}_j|$	$L \Leftarrow \text{Sort}\left(j \in \mathcal{A}, \frac{b_j}{|\mathcal{M}_j|}, \text{“non-decr”}\right)$
Cache (G.c.)	$p_j \Leftarrow \frac{b_c}{h_c} h_j$	$L \Leftarrow \text{Sort}\left(j \in \mathcal{A}, \frac{b_j}{h_j}, \text{“non-decr”}\right)$
Backhaul (G.b.)	$p_j \Leftarrow \frac{b_c}{R_c} R_j$	$L \Leftarrow \text{Sort}\left(j \in \mathcal{A}, \frac{b_j}{R_j}, \text{“non-decr”}\right)$

we can change the optimization strategy of the greedy algorithm; in particular, we propose the following three variants:

1) *Greedy MC* (G.m.);
2) *Greedy cache* (G.c.);
3) *Greedy backhaul* (G.b.).

As discussed previously, the *greedy MC* (G.m.) alternative gives higher priority to the APs that can potentially serve more MCs, due to their space location. On the other hand, the *greedy cache* (G.c.) selects the APs that are leasing larger caching storage; lastly, the *greedy backhaul* (G.b.) chooses the APs with a faster backhaul Internet connection. Table II summarizes these three variants, which will be compared numerically in Sec. VI. Finally, we observe that all the variants we propose do not modify the monotonicity property of the allocation phase; therefore, it is straightforward to show that *truthfulness* and *individual rationality* still hold.

VI. NUMERICAL RESULTS

In this section, we analyze and discuss the numerical results obtained solving our proposed models and heuristics in realistic, large-size network scenarios. More specifically, we first illustrate (Sec. VI-A) the experimental methodology used to evaluate the proposed algorithms, and then (Sec. VI-B) we discuss the obtained results.

A. Methodology

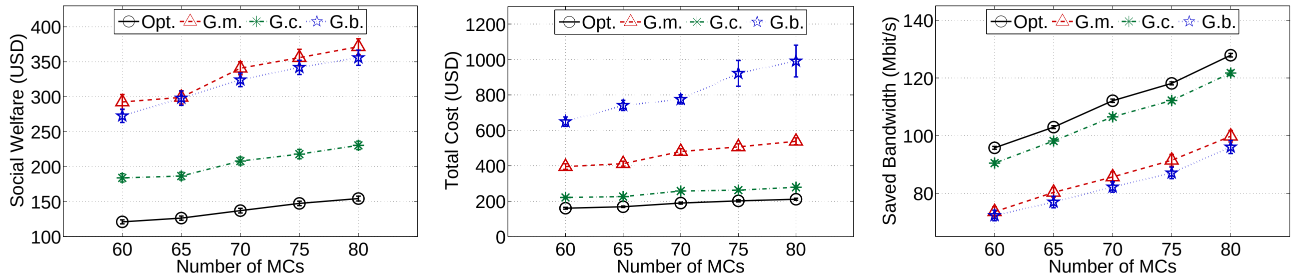
Simulations were conducted by distributing 50 APs uniformly in a 300×300 m^2 grid, while the locations of MCs were selected as a bi-variate Gaussian distribution centered on uniformly chosen APs. The bids of the APs are selected uniformly in the $[7, 15]$ USD range, whereas the demands of every MC were generated in the range $[0.5, 3]$ Mbit/s. We performed 50 runs for every scenario, computing the narrow 95% confidence intervals reported in each figure.

At each AP, we chose the backhaul Internet bandwidth randomly in the set $\{1; 6; 8; 20; 100\}$ Mbit/s, while the size of cache storage was uniformly selected in the range $[10, 100]$ GB. We set the average object size to 11 Kbytes and the Zipf popularity exponent to 0.8, according to the standard web-content scenario investigated in [9]. The hit rate is then computed using the single-layer cache model proposed by [9] for a LFU cache.

We define the *normalized cache size*, N, as the ratio between the average cache size and the size of all the objects:

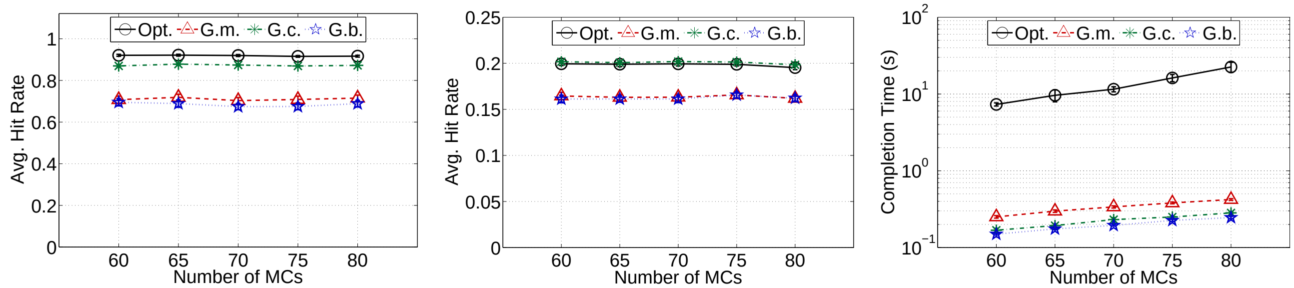
$$N = \frac{\text{Avg. cache size}}{\text{Objects cardinality} \cdot \text{Average object size}}. \qquad (13)$$

Since we assume that the object cardinality can either be 10^7 or 10^9, the normalized cache sizes are thus equal to 0.524 and 0.005 respectively.



(a) Social Welfare (b) Total Cost (c) Bandwidth Saved for Caching

Figure 2. *Social Welfare*, *Total Cost* and *Bandwidth Saved for Caching* as a function of the number of mobile clients. The number of access points is set to 50 and the objects cardinality is fixed and equal to 10^7, which corresponds to a Normalized Cache Size N of 0.524; this means that, on average, caches can store 52% of the total number of objects.



(a) Avg. Hit Rate; 10^7 objects; $N = 0.524$ (b) Avg. Hit Rate; 10^9 objects; $N = 0.005$ (c) Completion Time

Figure 3. *Average Hit Rate* and *Completion Time*, as a function of the number of mobile clients. The number of access points is set to 50, the objects cardinality is equal to 10^7 for Figures 3a and 3c while it is equal to 10^9 for Fig. 3b.

Regarding the three variants of the algorithms presented in Table II, we denote the greedy *MC*, *cache* and *backhaul* with G.m., G.c. and G.b., respectively. We analyze and compare the performance of the different algorithms according to the following metrics:

- *Social welfare* (SW): which is the objective function of the ILP model (4), and has a monetary currency as unit of measurement;

$$SW = \sum_{\forall j \in \mathcal{A}} y_j b_j + \sum_{\substack{\forall i \in \mathcal{M} \\ \forall j \in \mathcal{A}}} x_{i,j} d_i (1 - h_j) C$$

- *Total cost* (TC): which has the same expression as the SW apart from the fact that instead of considering the *bids*, it takes into account the *actual price* paid to the AP owners;

$$TC = \sum_{\forall j \in \mathcal{A}} y_j p_j + \sum_{\substack{\forall i \in \mathcal{M} \\ \forall j \in \mathcal{A}}} x_{i,j} d_i (1 - h_j) C$$

- *Bandwidth saved for caching* (SB): which is the total amount of traffic directly served by caches, measured in Mbit/s;

$$SB = \sum_{\substack{\forall i \in \mathcal{M} \\ \forall j \in \mathcal{A}}} x_{i,j} d_i h_j$$

- *Average hit rate* ($\bar{h}$): defined as the ratio between the content served by the caches and the sum of all the clients' demands;

$$\bar{h} = \frac{SB}{\sum_{\substack{\forall i \in \mathcal{M} \\ \forall j \in \mathcal{A}}} x_{i,j} d_i}$$

- *Completion time* (t): the time (expressed in seconds) spent by the algorithm to compute the allocation and determine the incentives paid to the AP owners.

B. Performance Analysis

Unless otherwise specified, the figures illustrated in the following are related to a network scenario with 10^7 different objects; hence, the normalized cache size is $N = 0.524$. A similar trend has been observed for 10^9 objects ($N = 0.005$), but it is not shown due to space constraints.

First of all, the *social welfare* (SW) as a function of the number of MCs is shown in Fig. 2a. As expected, such metric has an increasing trend, since the higher the number of MCs, the higher the costs that the content operator has to face. We also note that the optimal allocation performs better than all the greedy strategies. Quantitatively, the SW generated by the *greedy cache* is on average 50% higher than the optimal solution, while the other two variants of the greedy algorithm are comparable to each other, and generate more expensive allocations (the worst of which being 150% costlier than the optimal).

Nevertheless, the gap between the greedy and the optimal algorithm is further reduced for the *total cost* metric (TC). Fig. 2b shows the *total cost* paid by the content provider, by jointly considering the actual price to remunerate the selected APs as well as the cache miss costs. As illustrated in the figure, the *backhaul* variant of the algorithm is not only less efficient with respect to the other alternatives, but it also produces results with higher variance. Still, the *greedy cache* algorithm proves to generate interesting solutions, being only 34% higher than the optimal. In addition to that, we observed even a smaller gap between the two solutions when the cardinality of objects was 10^9, being the greedy cache algorithm only 22% costlier than the optimal.

The robustness of the *greedy cache* strategy is further confirmed by the *bandwidth saved for caching* (SB) metric, shown in Fig. 2c. Since the model assumes that the characteristics of the popularity distribution of the requests do not change as the number of MCs increases, the overall SB increases linearly. Interestingly, Fig. 2c further shows that if the main aim of the content provider is to offload the servers, the *greedy cache* variant well approaches the optimal solution. In particular, the *greedy cache* algorithm saves on average only 5% less bandwidth than the optimum, while the same metric raises to 25.5% for the worst case represented by the greedy backhaul strategy.

Fig. 3a and Fig. 3b show the *average hit rate* ($\bar{h}$), respectively with 10^7 and 10^9 objects. First of all, we note that while in Fig. 3a the average hit rate for the optimal algorithm is above 80%, in Fig. 3b it decreases below 25%. Nevertheless we underline that achieving a hit rate that approaches 25%, while having cache sizes that can only store 0.5% of the available objects, is remarkable. This behavior can be explained due to two reasons: 1) the power-law characteristic of the Zipf distribution models the fact that, in the Internet, few objects are extremely popular, while many objects are rarely accessed and 2) our mechanism selects the APs that are most valuable in terms of the available cache. Apparently, Fig. 3b shows an unexpected result: the *greedy cache* seems to behave better than the optimal algorithm. This result, however, is not surprising since, as a matter of fact, the optimal algorithm minimizes the sum of the prices paid to AP owners as well as the cache miss costs, while Fig. 3a and Fig. 3b portrait only the cache miss costs, which is only a *part* of the total cost.

Finally, as shown in Fig. 3c, the greedy algorithm beats the optimal allocation procedure by one order of magnitude in terms of the *completion time* (t). As expected, the greedy MC is slower than the other choices, since it needs to additionally compute the set of MCs that each AP has to serve.

VII. CONCLUSION

This paper proposed a novel mechanism for Information Centric Networks to stimulate wireless access point owners to jointly lease their unused bandwidth and storage space to a content provider.

We provided an algorithm to determine the optimal allocation of mobile clients to access points that ensures the *individual rationality* as well as the *truthfulness* properties by forcing the AP owners to bid the real valuation for the offered resources. We showed that the optimal allocation problem is NP-hard, and provided three efficient alternatives of a greedy algorithm that computes a sub-optimal solution of the problem in polynomial time, while still guaranteeing the *individual rationality* as well as the *truthfulness* properties.

Finally, numerical results demonstrated that the performance of the greedy cache algorithm well approaches the results obtained by the optimal solution. In particular, while the greedy cache algorithm raises the total cost for the content provider of 34%, it is only 5% worse than the optimal solution with respect to the saved bandwidth, while being one order of magnitude faster than the optimal algorithm.

In future works, we would like to further address the routing problem in ICNs: the presence of the distributed cache as well as the stateful information available at every router poses additional questions on how these characteristics can affect the performance of this novel paradigm.

ACKNOWLEDGMENT

This work was partially supported by Italian MIUR and French ANR in the framework of the PRIN Gatecom and ANR Green-Dyspan projects.

REFERENCES

[1] M. Gritter and D. R. Cheriton, "An Architecture for Content Routing Support in the Internet," *Proceedings of the 3rd conference on USENIX Symposium on Internet Technologies and Systems - Volume 3*, 2001.

[2] J. Pan, S. Paul, and R. Jain, "A Survey of the Research on Future Internet Architectures," *Communications Magazine, IEEE*, 2011.

[3] V. Jacobson, D. K. Smetters, J. D. Thornton, M. Plass, N. Briggs, and R. Braynard, "Networking named content," *Commun. ACM*, vol. 55, no. 1, pp. 117–124, 2012.

[4] T. Koponen, M. Chawla, B.-G. Chun, A. Ermolinskiy, K. H. Kim, S. Shenker, and I. Stoica, "A Data-Oriented (and Beyond) Network Architecture," *SIGCOMM Comput. Commun. Rev.*, 2007.

[5] A. Detti, N. Blefari Melazzi, S. Salsano, and M. Pomposini, "CONET: a content centric inter-networking architecture," *Proceedings of the ACM SIGCOMM workshop on Information-centric networking*, pp. 50–55, 2011.

[6] "FP7 PURSUIT Project," http://www.fp7-pursuit.eu, Last accessed: Mar. 2013.

[7] V. Jacobson, D. K. Smetters, N. H. Briggs, M. F. Plass, P. Stewart, J. D. Thornton, and R. L. Braynard, "VoCCN: voice-over content-centric networks," *Workshop on Re-architecting the internet*, pp. 1–6, 2009.

[8] D. Rossi and G. Rossini, "On sizing CCN content stores by exploiting topological information," *IEEE INFOCOM, NOMEN Workshop*, pp. 280 – 285, 2012.

[9] C. Fricker, P. Robert, J. Roberts, and N. Sbihi, "Impact of traffic mix on caching performance in a content-centric network," *IEEE INFOCOM, NOMEN Workshop*, pp. 310 –315, 2012.

[10] I. Psaras, R. G. Clegg, R. Landa, W. K. Chai, and G. Pavlou, "Modelling and evaluation of CCN-caching trees," *Proceedings of the 10th international IFIP TC 6 conference on Networking - Volume Part I*, pp. 78–91, 2011.

[11] X. Zhou and H. Zheng, "TRUST: A General Framework for Truthful Double Spectrum Auctions," *INFOCOM 2009, IEEE*, pp. 999 –1007, 2009.

[12] G. S. Kasbekar and S. Sarkar, "Spectrum Auction Framework for Access Allocation in Cognitive Radio Networks," *IEEE/ACM Trans. Netw.*, vol. 18, no. 6, pp. 1841–1854, 2010.

[13] J. Dai, F. Liu, B. Li, B. Li, and J. Liu, "Collaborative Caching in Wireless Video Streaming Through Resource Auctions," *Selected Areas in Communications, IEEE Journal on*, pp. 458 –466, 2012.

[14] N. Nisan, T. Roughgarden, E. Tardos, and V. V. Vazirani, *Algorithmic Game Theory*. Cambridge University Press, 2007.

[15] J. Puchinger, G. R. Raidl, and U. Pferschy, "The Multidimensional Knapsack Problem: Structure and Algorithms," *INFORMS J. on Computing*, vol. 22, 2010.