

Content-peering Dynamics of Autonomous Caches in a Content-centric Network

Valentino Pacifici and György Dán
ACCESS Linnaeus Center, School of Electrical Engineering
KTH, Royal Institute of Technology, Stockholm, Sweden
E-mail: {pacifici,gyuri}@kth.se

Abstract—A future content-centric Internet would likely consist of autonomous systems (ASes) just like today’s Internet. It would thus be a network of interacting cache networks, each of them optimized for local performance. To understand the influence of interactions between autonomous cache networks, in this paper we consider ASes that maintain peering agreements with each other for mutual benefit, and engage in content-level peering to leverage each others’ cache contents. We propose a model of the interaction and the coordination between the caches managed by peering ASes. We address whether stable and efficient content-level peering can be implemented without explicit coordination between the neighboring ASes or alternatively, whether the interaction needs to rely on explicit announcements of content reachability in order for the system to be stable. We show that content-level peering leads to stable cache configurations, both with and without coordination. If the ASes do coordinate, then coordination that avoids simultaneous updates by peering ISPs provides faster and more cost efficient convergence to a stable configuration. Furthermore, if the content popularity estimates are inaccurate, content-level peering is likely to lead to cost efficient cache allocations. We validate our analytical results using simulations on the measured peering topology of more than 600 ASes.

I. INTRODUCTION

Recent proposals to re-design the Internet with the aim of facilitating content delivery share the common characteristic that caches are an integral part of the protocol stack [1], [2], [3]. In these content-centric networks users generate interest messages for content, which are forwarded until the content is found in a cache or the interest message reaches one of the content’s custodians. The resulting network is often modeled as a network of interacting caches. Several recent works aimed at optimizing the performance of a cache network through dimensioning cache sizes as a function of their location in the cache network [4], by routing interest messages to efficiently find contents [5] and by tuning the cache eviction policies used by the individual caches [6].

Similar to the structure of today’s Internet, a future content-centric network is likely to be a network of autonomous systems (AS). ASes are typically profit seeking entities and use an interior gateway protocol (IGP) for optimizing their internal routes. Nevertheless, they maintain client-provider and peering business relations with adjacent ASes [7], and they coordinate with each other using the Border Gateway Protocol (BGP), which allows them to exchange reachability information with their neighbors. The effect of BGP coordination on the stability

and performance of global IP routing has been extensively investigated, e.g., the negative impact of damping route flaps [8], [9], the number of updates needed for BGP convergence [10], and general conditions for cycle-free IP routes [11].

ASes are likely to play a similar role in a future content-centric Internet as they do today, and thus, instead of a single cache network dimensioned and managed for optimal global performance, the content-centric Internet will be a network of cache networks, each of them optimized for local performance. To make such a network of cache networks efficient, we need to understand the potential consequences of interaction between the individual cache networks in terms of stability and convergence of the cache contents, and the potential impact of coordination between the networks of caches.

In this work we consider a network of ASes that maintain peering agreements with each other for mutual benefit. The traffic exchanged between the peering AS is not charged, unlike the traffic that each AS exchanges with its transit provider. The ASes maintain their own cache networks, and they engage in *content-level peering* in order to leverage each others’ cache contents, which in principle should enable them to decrease their transit traffic costs. The interaction between the caches could, however, lead to unforeseen instability and oscillations, as in the case of BGP. Thus, a fundamental question that one needs to answer is whether stable and efficient content-level peering can be implemented without explicit coordination between the neighboring cache networks. Alternatively, does the interaction need to rely on explicit announcements of content reachability, resembling the BGP announcements in today’s Internet, and if so, is the system going to be stable.

In this paper we address these questions by proposing a model of the interaction and the coordination between the caches managed by peering ASes. We show that, with or without coordination, content-peering leads to stable cache configurations. Furthermore, we investigate how the convergence speed and the cost efficiency of the reached cache configuration are affected by coordination. Finally, we give insight into the structure of the most likely cache allocations in the case of inaccurate estimation of the arrival rate of user requests. We illustrate the analytical results using simulations on the measured peering topology of more than 600 ASes.

The rest of the paper is organized as follows. In Section II we describe the system model. In Section III we consider caching

under perfect information, and in Section IV we consider the case of imperfect information. In Section V we present numerical results, and in Section VI we review related work. Section VII concludes the paper.

II. SYSTEM MODEL

We consider a set N of autonomous ISPs. Each ISP $i \in N$ is connected via peering links to some ISPs $j \in N$. We model the peering links among ISPs by an undirected graph $\mathcal{G} = (N, E)$, called the *peering graph*. We call $\mathcal{N}(i)$ the set of neighbors of ISP $i \in N$ in the peering graph, i.e. $\mathcal{N}(i) = \{j | (i, j) \in E\}$. Apart from the peering links, every ISP can have one or more transit links.

A. Content Items and Caches

We denote the set of content items by \mathcal{O} . We follow common practice and consider that every item $o \in \mathcal{O}$ has unit size [12], [13], which is a reasonable simplification if content is divisible into unit-sized chunks. Each item $o \in \mathcal{O}$ is permanently stored at one or more *content custodians* in the network. We denote by \mathcal{H}_i the set of items kept by the custodians within ISP i . Since the custodians are autonomous entities, ISP i cannot influence the set \mathcal{H}_i . Similar to other modeling works, we adopt the Independent Reference Model (IRM) [14], [12], [13] for the arrival process of interest messages for the items in \mathcal{O} generated by the local users of the ISPs. Under the IRM, the probability that the next interest message at ISP i is for item o is independent of earlier events. An alternative definition of the IRM is that the inter-arrival time of interest messages for item o at ISP i follows an exponential distribution with distribution function $F_i^o(x) = 1 - e^{-w_i^o x}$, where $w_i^o \in \mathbb{R}_+$ is the average arrival intensity of interest messages for item o at ISP i .

Each ISP $i \in N$ maintains a network of content caches within its network, and jointly engineers the eviction policies of the caches, the routing of interest messages and the routing of contents via the caches to optimize performance. The set of items cached by ISP i is described by the set $\mathcal{C}_i \in \mathcal{C}_i = \{\mathcal{C} \subset \mathcal{O} : |\mathcal{C}| = K_i\}$, where $K_i \in \mathbb{N}_+$ is the maximum number of items that ISP i can cache. A *summary cache* in each ISP keeps track of the configuration of the local caches and of the content stored in local custodians, it thus embodies the information about what content is available within ISP i . We call $\mathcal{L}_i = \mathcal{C}_i \cup \mathcal{H}_i$ the set of items available within ISP i .

We denote by $\alpha_i > 0$ the unit cost of retrieving an item from a local cache. We consider that retrieving an item from a peering ISP is not more costly than retrieving it locally. The assumption of equal local and peering cost is justified by the fact that in general, once a peering link has been established, there is no additional cost for traffic. The traffic on the transit link is charged by volume with unit cost γ_i , and we make the reasonable assumption that $\gamma_i > \alpha_i$.

B. Content-peering

Upon receiving an interest message for an item, ISP i consults its summary cache to see if the item is available locally. If it is, ISP i retrieves the item from a local cache. Otherwise, before

ISP i would forward the interest message to its transit provider, it can leverage its neighbors' caches according to one of two scenarios.

a) *Uncoordinated Content-peering*: Without coordination, if ISP i finds that an item o is not available locally, it forwards the interest message to all of its neighbors $j \in \mathcal{N}(i)$. If a neighbor has the item in cache, it returns the item to ISP i . If none of the neighbors has the item, ISP i forwards the interest message to its transit provider.

b) *Coordinated Content-peering*: In the case of coordinated peering ISPs *synchronously exchange information* about the contents of their summary caches *periodically*, at the end of every time slot. If, upon an interest message for item o , ISP i finds that item o is not available locally, it consults its most recent copy of the summary caches of its peering ISPs $\mathcal{N}(i)$. In case a peering ISP $j \in \mathcal{N}(i)$ is caching the item, ISP i forwards the request to ISP j and fetches the content. If not, the interest message is sent to a transit ISP through a transit link.

Using the above notation, and denoting by \mathcal{C}_{-i} the set of the cache configurations of every ISP other than ISP i , we can express the cost of ISP i to obtain item $o \in \mathcal{O}$ as

$$C_i^o(\mathcal{C}_i, \mathcal{C}_{-i}) = w_i^o \begin{cases} \alpha_i & \text{if } o \in \mathcal{L}_i \cup \mathcal{R}_i \\ \gamma_i & \text{otherwise,} \end{cases} \quad (1)$$

where $\mathcal{R}_i = \bigcup_{j \in \mathcal{N}(i)} \mathcal{L}_j$ is the set of items ISP i can obtain from its peering ISPs. The total cost can then be expressed as

$$C_i(\mathcal{C}_i, \mathcal{C}_{-i}) = \alpha_i \sum_{\mathcal{L}_i \cup \mathcal{R}_i} w_i^o + \gamma_i \sum_{\mathcal{O} \setminus \{\mathcal{L}_i \cup \mathcal{R}_i\}} w_i^o, \quad (2)$$

which is a function of the cache contents of the peering ISPs $\mathcal{N}(i)$.

C. Caching Policies and Cost Minimization

A content item o that is not available either locally or from a peering ISP is obtained through a transit link, and is a candidate for caching in ISP i . The cache eviction policy of ISP i determines if item o should be cached, and if so, which item $p \in \mathcal{C}_i$ should be evicted to minimize the expected future cost. There is a plethora of cache eviction policies for this purpose, such as Least recently used (LRU), Least frequently used (LFU), LRFU (we refer to [15] for a survey of some recent algorithms). We model the eviction decision as a comparison of the estimate \bar{w}_i^o of the arrival intensity w_i^o for the item o to be cached and that for the items p in the cache, \bar{w}_i^p .

Perfect information: Under perfect information $\bar{w}_i^o = w_i^o$, and only the items with highest costs $C_i^o(\mathcal{C}_i, \mathcal{C}_{-i})$ are cached.

Imperfect information: Under imperfect information \bar{w}_i^o is a random variable with mean w_i^o , and we assume that the probability of misestimation decreases exponentially with the difference in arrival intensities, that is, for $w_i^o > w_i^p$ we have

$$P(\bar{w}_i^o < \bar{w}_i^p) \propto e e^{-\frac{1}{\beta}(w_i^o - w_i^p)}. \quad (3)$$

This assumption is reasonable for both the LRU and the LFU cache eviction policies. Under LRU the cache miss rate was shown to be an exponentially decreasing function of the item popularity [13]. Under a perfect LFU policy, if we denote the

interval over which the request frequencies are calculated by τ , then \bar{w}_i^p follows a Poisson distribution with parameter $w_i^p \tau$. The difference $k = \bar{w}_i^o \tau - \bar{w}_i^p \tau$ of two estimates thus follows the Skellam distribution [16] with density function

$$f(k, w_i^o \tau, w_i^p \tau) = e^{-\tau(w_i^o + w_i^p)} \left(\frac{w_i^o}{w_i^p} \right)^{k/2} I_{|k|}(2\tau \sqrt{w_i^o w_i^p}),$$

where $I_{|k|}(\cdot)$ is the modified Bessel function of the first kind. The probability of misestimation is $\sum_{k=-\infty}^{-1} f(k, w_i^o \tau, w_i^p \tau)$, which decreases exponentially in $w_i^o - w_i^p$ for $\tau > 0$.

III. CONTENT-PEERING UNDER PERFECT INFORMATION

We start the analysis by considering the case of perfect information, that is, when the cache eviction policies are not prone to misestimation, and we first consider the case of coordinated peering.

The key question we ask is whether the profit-maximizing behavior of the individual ISPs would allow the emergence of an equilibrium allocation of items. If an equilibrium cannot be reached then content-peering could potentially lead to increased costs for the peering ISPs, as shown by the following simple example in which as a consequence of coordination every ISP evicts and fetches the same items repeatedly over transit connections, thereby increasing their traffic costs compared to no content-peering.

Example 1. Consider two ISPs and $\mathcal{O} = \{1, 2\}$. Let $K_1 = K_2 = 1$. Without content peering both ISPs cache their most popular item and forward interest messages to their transit provider for the least popular item. Their cost is thus $C_i = \alpha_i w_i^{h_i} + \gamma_i w_i^{l_i}$, where $w_i^{h_i} > w_i^{l_i}$. With content peering, if the initial allocation strategies are $\mathcal{C}_1 = \mathcal{C}_2 = \{1\}$, then the cache contents of the ISPs will evolve indefinitely as $(\{1\}, \{1\}) \rightarrow (\{2\}, \{2\}) \rightarrow (\{1\}, \{1\})$, etc. The average cost for the ISPs is thus $C'_i = \alpha_i \left(\frac{w_i^{h_i} + w_i^{l_i}}{2} \right) + \gamma_i \left(\frac{w_i^{h_i} + w_i^{l_i}}{2} \right) > C_i$.

This simple example illustrates that content peering could potentially lead to undesired oscillations of the cache contents of the ISPs, with the consequence of increased traffic costs. Ideally, for a stationary arrival of interest messages the cache contents should stabilize in an equilibrium state that satisfies the ISPs' interest of traffic cost minimization. In the following we propose two distributed algorithms that avoid such inefficient updates and allow the system to reach an equilibrium allocation of items from which no ISP has an interest to deviate. Such an allocation is a pure strategy Nash equilibrium of the strategic game $\langle N, (\mathcal{C}_i)_{i \in N}, (C_i)_{i \in N} \rangle$, in which each ISP i aims to minimize its own cost C_i defined in (2).

Definition 1. A cache allocation $\mathcal{C}^* \in \times_{i \in N} \mathcal{C}_i$ is an equilibrium allocation (pure strategy Nash equilibrium) if no single ISP can decrease its cost by deviating from it, that is

$$\forall i \in N, \forall \mathcal{C}_i \in \mathcal{C}_i : C_i(\mathcal{C}_i^*, \mathcal{C}_{-i}^*) \leq C_i(\mathcal{C}_i, \mathcal{C}_{-i}^*) \quad (4)$$

A. Cache-or-Wait (CoW) Algorithm

Example 1 suggests that if one does not allow peering ISPs to update their cache configurations simultaneously, then they would converge to an allocation from which neither of them would have an interest to deviate. In the case of Example 1, such allocations are $(\{1\}, \{2\})$ or $(\{2\}, \{1\})$. Before we describe the Cache-or-Wait (CoW) algorithm, let us recall the notion of an independent set.

Definition 2. We call a set $\mathcal{I} \subseteq N$ an *independent set* of the peering graph \mathcal{G} if it does not contain peering ISPs. Formally

$$\forall i, j \in \mathcal{I}, j \notin \mathcal{N}(i).$$

We denote by \mathcal{J} the set of all the independent sets of the peering graph \mathcal{G} . Consider a sequence of time slots t and a sequence of independent sets $\mathcal{I}_1, \mathcal{I}_2, \dots \in \mathcal{J}$ indexed by t , such that for every time slot $t \geq 1$ and every ISP $i \in N$ there is always a time slot $t' > t$ such that $i \in \mathcal{I}_{t'}$. At each time slot t we allow every ISP $i \in \mathcal{I}_t$ to update the set of its cached content \mathcal{C}_i . ISP $i \in \mathcal{I}_t$ can decide to insert in its cache the items that are requested by one or more of its local users during time slot t but were not cached at the beginning of the time slot. At the same time, ISPs $j \notin \mathcal{I}_t$ are not allowed to update the set of their cached contents. The pseudocode of the CoW algorithm for every time slot $t \geq 1$ is then the following:

-
- Pick \mathcal{I}_t .
 - Allow ISPs $i \in \mathcal{I}_t$ to change their cached items from $\mathcal{C}_i(t-1)$ to $\mathcal{C}_i(t)$,
 - For all $j \notin \mathcal{I}_t$, $\mathcal{C}_j(t) = \mathcal{C}_j(t-1)$.
 - At the end of the time slot inform the ISPs $j \in \mathcal{N}(i)$ about the new cache contents $\mathcal{C}_i(t)$
-

Fig. 1. Pseudo-code of the Cache-or-Wait (CoW) Algorithm

What we are interested in is whether ISPs following the CoW algorithm would reach an equilibrium allocation from which none of them would like to deviate. If CoW reaches such an allocation, then it terminates, and no other cache update will take place. In the following we provide a sufficient condition for CoW to terminate in a finite number of steps. We call the condition *efficiency*, and the condition concerns the changes that an ISP can make to its cache configuration.

Definition 3. Consider the updated cache configuration $\mathcal{C}_i(t)$ of ISP $i \in \mathcal{I}_t$ immediately after time slot t . Define the evicted set as $E_i(t) = \mathcal{C}_i(t-1) \setminus \mathcal{C}_i(t)$ and the inserted set as $I_i(t) = \mathcal{C}_i(t) \setminus \mathcal{C}_i(t-1)$. $\mathcal{C}_i(t)$ is an *efficient update* if for any $o \in I_i(t)$ and any $p \in E_i(t)$

$$C_i^o(\mathcal{C}(t)) + C_i^p(\mathcal{C}(t)) < C_i^o(\mathcal{C}(t-1)) + C_i^p(\mathcal{C}(t-1)) \quad (5)$$

The requirement of efficiency is rather reasonable. Given that the ISPs are profit maximizing entities, it is natural to restrict the changes in the cache configuration to changes that actually lead to lower cost. In order to prove that the *efficiency* condition is sufficient for CoW to converge, we will rely on the *generalized group ordinal potential function* defined as follows.

Definition 4. A function $\Psi : \times_i(\mathcal{C}_i) \rightarrow \mathbb{R}$ is a *generalized group ordinal potential function* for the game $\langle N, (\mathcal{C}_i)_{i \in N}, (C_i)_{i \in N} \rangle$ if the change of Ψ is strictly positive whenever an arbitrary subset $\mathcal{V} \subseteq \mathcal{I} \in \mathcal{J}$ of ISPs decrease their costs by changing their strategies,

$$C_i(C'_i, \mathcal{C}_{-i}) - C_i(C_i, \mathcal{C}_{-i}) > 0, \quad \forall i \in \mathcal{V} \Rightarrow \\ \Psi(\mathcal{C}_{\mathcal{V}}, \mathcal{C}_{-\mathcal{V}}) - \Psi(C'_{\mathcal{V}}, \mathcal{C}_{-\mathcal{V}}) > 0. \quad (6)$$

Observe that if we define every independent set to be a *singleton*, then the group ordinal potential function Ψ is the *generalized ordinal potential function* defined in [17].

We start constructing a generalized group ordinal potential function by defining the cost saving of ISP i for cache allocation C_i as $\Psi_i(\mathcal{C}) = C_i(\emptyset, \mathcal{C}_{-i}) - C_i(C_i, \mathcal{C}_{-i})$. After substituting (2) we obtain

$$\Psi_i(\mathcal{C}) = \alpha_i \sum_{\mathcal{H}_i \cup \mathcal{R}_i} w_i^o + \gamma_i \sum_{\mathcal{O} \setminus \{\mathcal{H}_i \cup \mathcal{R}_i\}} w_i^o - \left[\alpha_i \sum_{C_i \cup \mathcal{H}_i \cup \mathcal{R}_i} w_i^o + \right. \\ \left. + \gamma_i \sum_{\mathcal{O} \setminus \{C_i \cup \mathcal{H}_i \cup \mathcal{R}_i\}} w_i^o \right] = \sum_{o \in C_i \setminus \{\mathcal{H}_i \cup \mathcal{R}_i\}} [\gamma_i - \alpha_i] w_i^o. \quad (7)$$

Note that the value of $\Psi_i(\mathcal{C})$ is not influenced by any item $o \notin C_i$. We are now ready to prove the following.

Theorem 1. *If every ISP performs efficient updates then the function $\Psi : \times_i(\mathcal{C}_i) \rightarrow \mathbb{R}$ defined as $\Psi(\mathcal{C}) = \sum_{i \in N} \Psi_i(\mathcal{C})$ increases strictly upon every update and COW terminates in an equilibrium allocation after a finite number of updates.*

Proof: We will start by showing that an efficient update made by any ISP i in the independent set \mathcal{I} strictly increases Ψ_i and cannot decrease Ψ_j of any ISP $j \neq i$, hence Ψ is a generalized group ordinal potential function for efficient updates. Without loss of generality, consider the efficient update $C_i(t)$ made by ISP $i \in \mathcal{I}_t$ at time slot t . In the following we show that $\Psi_j(C_i(t), \mathcal{C}_{-i}(t-1)) \geq \Psi_j(\mathcal{C}(t-1))$ for all $j \in N$. Observe that from the definition of $\Psi_i(\mathcal{C})$ it follows directly that for ISP i

$$\Psi_i(C_i(t), \mathcal{C}_{-i}(t-1)) > \Psi_i(\mathcal{C}(t-1)).$$

A) Consider $k \notin \mathcal{N}(i)$. Observe that the cost of ISP k is not a function of C_i :

- if $k \notin \mathcal{I}$, ISP k does not make any efficient update at time slot t , thus $\Psi_k(C_i(t), \mathcal{C}_{-i}(t-1)) = \Psi_k(\mathcal{C}(t-1))$;
- if $k \in \mathcal{I}$, $k \neq i$, Ψ_k is not influenced by C_i .

B) Consider $j \in \mathcal{N}(i)$. Consider $o \in I_i(t)$ and $p \in E_i(t)$. From the cost function defined in (1) it follows that $C_i^p(t+1) \geq C_i^p(t)$. Substituting it in the definition of efficient improvement step in (5), it follows that $C_i^o(t) > C_i^o(t+1) \Rightarrow o \notin \mathcal{R}_i(t) \Rightarrow o \notin C_j(t)$, thus $\Psi_j(C_i(t), \mathcal{C}_{-i}(t-1))$ is not affected by item o .

Consider now item p :

- If $p \notin C_j(t)$, then $\Psi_j(C_i(t), \mathcal{C}_{-i}(t-1))$ is not affected by item p .
- If $p \in C_j(t)$, then $\Psi_j(C_i(t), \mathcal{C}_{-i}(t-1)) \geq \Psi_j(\mathcal{C}(t-1))$ (the inequality is strict if $p \notin \{\mathcal{H}_j \cup \mathcal{R}_j(t+1)\}$).

It follows that the function Ψ increases strictly upon every efficient update. Since $\times_i(\mathcal{C}_i)$ is a finite set, Ψ cannot increase indefinitely and COW must terminate in an equilibrium allocation after a finite number of updates. ■

The following corollaries are consequences of Theorem 1

Corollary 1. *In the case of coordinated peering under perfect information there is at least one equilibrium allocation.*

Corollary 2. *If every ISP performs efficient updates then the number of time slots needed to reach an equilibrium is finite with probability 1.*

Thus, a network of ISPs in which only non-peering ISPs perform efficient updates simultaneously at every time slot reaches an equilibrium allocation after a finite number of updates.

B. Cache-no-Wait (CNW) Algorithm

A significant shortcoming of COW is that in slot t it disallows ISPs $j \notin \mathcal{I}_t$ to perform an update. Since the number of independent sets equals at least $\chi(\mathcal{G})$, the chromatic number of the ISP peering graph, an ISP can perform an update on average in every $\chi(\mathcal{G})^{th}$ time slot, in the worst case once every $|N|$ time slots. This restriction would provide little incentive for ISPs to adhere to the algorithm. In the following we therefore investigate what happens if every ISP in the system is allowed to perform an efficient update during every time slot. The pseudo-code of the CNW algorithm for time slot $t \geq 0$ looks as follows.

-
- Every ISP $i \in N$ is allowed to change its cached items from $C_i(t-1)$ to $C_i(t)$.
 - At the end of the time slot ISP i informs the ISPs $j \in \mathcal{N}(i)$ about the new cache contents $C_i(t)$
-

Fig. 2. Pseudo-code of the Cache-no-Wait (CNW) Algorithm

Theorem 2. *If every ISP performs only efficient updates, CNW terminates in an equilibrium allocation with probability 1.*

Proof: Every update of the cache allocation of an ISP is triggered by an interest message sent by a local user. Consider now the arrival of interest messages for item o generated by the local users of ISP i , which has intensity w_i^o . Given that the distribution F_i^o of the inter-arrival times is exponential with parameter w_i^o , there is a non-zero probability $e^{-w_i^o \Delta}$ that item o is not requested during a time slot of length Δ .

Let us consider now a sequence of time slots. For every ISP $i \in \mathcal{I}_t$ there is a positive probability $\epsilon_i(C_i(t-1))$ that the interest messages generated during time slot t are for items that are either cached locally, are cached by a peering ISP or are not popular enough for being cached. If the cache configuration $C_i(t-1)$ minimizes ISP i 's expected future cost with respect to $\mathcal{C}_{-i}(t-1)$, then $\epsilon_i(C_i(t-1)) = 1$. Otherwise, with probability $0 < \epsilon_i(C_i(t-1)) < 1$ ISP i does not update its cache configuration at time slot t , even if in principle its cache configuration could be improved, because no interest message arrives for an item o that could improve its cache

configuration. To summarize, at every time slot $t \geq 1$, $\mathcal{C}_i(t)$ is updated according to the following

- $\mathcal{C}_i(t) = \mathcal{C}_i(t-1)$ w.p. $\epsilon_i(\mathcal{C}_i(t-1))$
- $\mathcal{C}_i(t)$ is an efficient update w.p. $1 - \epsilon_i(\mathcal{C}_i(t-1))$.

In the following we use an argument similar to the one used in [18] in order to prove that from every cache configuration $\mathcal{C}(t)$ there is a path to a Nash equilibrium, despite simultaneous cache updates by peering ISPs.

Consider $\mathcal{C}(t-1)$ at the beginning of time slot t . If $\mathcal{C}(t-1)$ is not a Nash equilibrium, then there is a non-zero probability that during time slot t the ISPs that make an update belong to an independent set \mathcal{I}_t . This probability is lower bounded by

$$\prod_{i \in \mathcal{I}_t} [1 - \epsilon_i(\mathcal{C}_i(t-1))] \cdot \prod_{i \in N \setminus \mathcal{I}_t} \epsilon_i(\mathcal{C}_i(t-1)) > 0,$$

which is the probability that the ISPs updating their cache configuration during time slot t are exactly \mathcal{I}_t . By Theorem 1 we know that if in every time slot it is only ISPs that form an independent set that perform efficient updates, then they reach a Nash equilibrium after a finite number of steps.

Thus there exists a positive probability p and a positive integer T such that, starting from an arbitrary global cache configuration $\mathcal{C}(t)$ at an arbitrary time slot t , the probability that CNW will reach a Nash equilibrium by time slot $t+T$ is at least p . Consequently, the probability that CNW has not reached a Nash equilibrium within a time slot t' is at most $(1-p)^{\frac{t'}{T-1}}$. Observe that $(1-p)^{\frac{t'}{T-1}}$ goes to zero as $t' \rightarrow \infty$, and this proves the theorem. ■

C. Uncoordinated Content-peering

Let us consider now the case of uncoordinated content peering under perfect information. Recall that in this case ISP i does not receive information from ISPs $j \in \mathcal{N}(i)$ about the contents of their caches. As long as an item is cached at ISP i , that is, $o \in \mathcal{C}_i$, the ISP would not forward interest messages to ISPs $j \in \mathcal{N}(i)$, and thus it would not know if o is cached at its neighbors. When ISP i receives an interest message for an item $o \notin \mathcal{C}_i$, it would have to discover whether ISPs $j \in \mathcal{N}(i)$ have item o in their cache by forwarding an interest message to every ISP $j \in \mathcal{N}(i)$. Every discovery of the cache contents of the neighboring ISPs is thus triggered by a cache miss at ISP j . If $o \notin \mathcal{R}_i$, it has to be retrieved over a transit link, and hence ISP i can cache it by evicting an item $p \in \mathcal{C}_i$ if $w_i^p < w_i^o$. Observe that any cache allocation in which no future request can trigger a change is a stable cache allocation under uncoordinated content-peering

Definition 5. \mathcal{C}_i is a *stable cache allocation* for uncoordinated content peering if, $\forall i \in N$ and $\forall o, p \in \mathcal{O}$

$$p \in \mathcal{C}_i, o \notin \{\mathcal{L}_i \cup \mathcal{R}_i\} \Rightarrow w_i^p > w_i^o.$$

For example, the cache allocation where every ISP $i \in N$ is caching the K_i items with highest arrival intensities w_i^o , is a stable allocation for the case of uncoordinated content peering. There can be many other stable allocations, as shown by the following proposition.

Proposition 3. *Every equilibrium allocation under coordinated content-peering is a stable allocation under uncoordinated content-peering.*

Proof: Consider the equilibrium allocation \mathcal{C} under coordinated content-peering. Assume that an interest message for item $o \in \mathcal{O}$ arrives at ISP i . If $o \in \{\mathcal{L}_i \cup \mathcal{R}_i\}$, then ISP i retrieves item o locally or from a peering ISP and its cache configuration does not change. If $o \notin \{\mathcal{L}_i \cup \mathcal{R}_i\}$, then ISP i retrieves item o through a transit link. Let us assume that o is then inserted by ISP i in place of item $p \in \mathcal{C}$, meaning that \mathcal{C} is not a stable allocation under uncoordinated content-peering, and $w_i^o > w_i^p$. Since $o \notin \mathcal{R}_i(t)$, we know from (2) that

$$C_i^o(\mathcal{C}) = \gamma_i w_i^o > C_i^o(\mathcal{C}') = \alpha_i w_i^o \quad (8)$$

Similarly, since $p \in \mathcal{C}_i$ we have that $C_i^p(\mathcal{C}) = \alpha_i w_i^p$. Now consider the following two cases:

- $p \in \mathcal{R}_i(t)$, then $C_i^p(\mathcal{C}') = C_i^p(\mathcal{C}) = \alpha_i w_i^p$.
- $p \notin \mathcal{R}_i(t)$, then $C_i^p(\mathcal{C}') = \gamma_i w_i^p$.

Putting these together we obtain

$$C_i^o(\mathcal{C}') + C_i^p(\mathcal{C}') < C_i^o(\mathcal{C}) + C_i^p(\mathcal{C}), \quad (9)$$

which means that, by Definition 3, \mathcal{C}' is an efficient update. Since ISP i was able to make an efficient update in cache configuration \mathcal{C} , it follows that \mathcal{C} could not have been an equilibrium allocation under coordinated content-peering, which contradicts our initial assumption. This proves the proposition. ■

We can easily show that the converse is not true: the cache allocation $(\{1\}, \{1\})$ in Example 1 is not an equilibrium allocation under coordinated content-peering, but since $w_i^1 > w_i^2$, no ISP will insert item 2 in its cache under uncoordinated peering and thus it is a stable allocation.

Given a stream of interest messages and perfect information, in the following we address whether caching without coordination converges to a stable allocation. For the analysis, we define the *instantaneous download* (ID) assumption, under which after a cache miss the file is instantaneously downloaded into the cache, *before* another interest message could arrive. The ID assumption is common practice in the caching literature [12], [13], and as we show, it allows fast convergence to a stable allocation.

Proposition 4. *Under the ID assumption uncoordinated peering reaches a stable cache allocation after a finite number of cache updates.*

Proof: Consider an arbitrary cache update of ISP $i \in N$. Consider an interest message that arrives for object $o \notin \{\mathcal{L}_i \cup \mathcal{R}_i\}$, and assume that ISP i updates its cache configuration from \mathcal{C}_i to \mathcal{C}'_i inserting item o in place of item p . Recall from the proof of Proposition 3, that \mathcal{C}'_i is an efficient update. The ID assumption guarantees that ISP i is the only ISP that updates its cache configuration in this time instant. Therefore we know from Theorem 1 that the global function $\Psi : \times_i(\mathcal{C}_i) \rightarrow \mathbb{R}$ strictly increases at every cache update. Since $\times_i(\mathcal{C}_i)$ is a finite set, Ψ cannot increase indefinitely and a stable allocation is reached after a finite number of updates. This proves the proposition. ■

With the ID assumption uncoordinated peering converges in a finite number of updates, similar to CoW. Without this assumption the convergence is much slower.

Proposition 5. *Without the ID assumption uncoordinated peering reaches a stable cache allocation with probability one.*

Proof: Given that the arrival processes of the interest messages at different ISPs are independent, there is a positive probability $\epsilon > 0$ that at some point in time ISP i is the only ISP that needs to update its cache, and that it can do so before another interest message arrives. Following the same arguments as in the proof of Theorem 2, it can be shown that the probability of reaching a stable cache allocation after t time units approaches 1 when $t \rightarrow \infty$. ■

We have thus far shown that under perfect information both coordinated and uncoordinated content-peering will lead to equilibrium, alternatively stable cache allocations. We now turn to the case of imperfect information.

IV. THE CASE OF IMPERFECT INFORMATION

Until now we assumed that following a cache miss, when ISP i has to decide whether to cache item o it has a perfect estimate of the arrival intensity w_i^o of every item, and thus it is always able to evict one of the least popular items. In the following we consider that the estimation of the item popularities is imperfect. We consider CoW throughout the section for two reasons. First, under the ID assumption uncoordinated content-peering behaves very similar to CoW. Second, the set of equilibria under CoW coincides with those under CNW.

Under imperfect information the system can not settle in any single equilibrium or stable allocation, unlike in the case of perfect information. Nevertheless, the cache allocations that are most likely to occur are not arbitrary, and in the following we show that it is possible to characterize them.

We model the evolution of the system state (the set of cached items) under imperfect information by a regular perturbed Markov process. Consider first the case of coordinated peering under perfect information and the CoW algorithm. For the following analysis we can restrict ourselves to the case that each independent set \mathcal{I} is a singleton. The evolution of the system state can be modeled by a Markov chain P^0 ; the transition probability between states \mathcal{C} and \mathcal{C}' is non-zero if and only if $\mathcal{C}' = (\mathcal{C}'_i, \mathcal{C}_{-i})$ for some i such that the update \mathcal{C}_i to \mathcal{C}'_i by ISP i is an efficient update. Observe that the Markov chain P^0 is not irreducible, as every equilibrium state is an absorbing state. We refer to this chain as the unperturbed process.

Consider now coordinated peering under imperfect information. Under imperfect information the probability that item o will be evicted and item p inserted even though $w_i^o > w_i^p$ is non-zero as given in (3). The resulting Markov process P^β is a regular perturbed Markov process of P^0 [19], because for every $\beta > 0$ it is irreducible and for every \mathcal{C} and \mathcal{C}' , P^β converges to P^0 at an exponential rate, $\lim_{\beta \rightarrow 0} P_{\mathcal{C}, \mathcal{C}'}^\beta = P_{\mathcal{C}, \mathcal{C}'}^0$. Furthermore, if $P_{\mathcal{C}, \mathcal{C}'}^\beta > 0$ for some $\beta > 0$ then for some $r(\mathcal{C}, \mathcal{C}') \geq 0$

$$\lim_{\beta \rightarrow 0} e^{\frac{1}{\beta} r(\mathcal{C}, \mathcal{C}')} \cdot P_{\mathcal{C}, \mathcal{C}'}^\beta = \epsilon > 0. \quad (10)$$

We refer to $r(\mathcal{C}, \mathcal{C}') \geq 0$ as the resistance of the transition from allocation \mathcal{C} to \mathcal{C}' . The resistance is 0 if there is a transition in the unperturbed Markov process. Since P^β is an irreducible aperiodic finite Markov process, it has a unique stationary distribution for $\beta > 0$. We now recall a result from Young [19].

Lemma 1 (Young [19]). *Let P^β be a regular perturbed Markov process, and let μ^β be the unique stationary distribution of P^β for each $\beta > 0$. Then $\lim_{\beta \rightarrow 0} \mu^{e^{-\frac{1}{\beta}}} = \mu^0$ exists, and μ^0 is a stationary distribution of P^0 . The domain of μ^0 is a non-empty subset of the absorbing states of the unperturbed Markov process.*

By Lemma 1 there is thus a stationary distribution μ^0 of the unperturbed process such that, for small β , the system will likely be in a state in the domain of μ^0 . In the rest of the section we try to infer what cache allocations are most likely to occur in the following scenario. Consider a set of $N = \{1, \dots, |N|\}$ ISPs, and items $\mathcal{O} = \{1, \dots, |\mathcal{O}|\}$. Let $\rho_i(o)$ be the rank in terms of popularity of item o in ISP i , and let \mathcal{T}_i be the set of the K_i items such that $\rho_i(o) \leq K_i$. For a cache allocation \mathcal{C} denote by $h(\mathcal{C})$ the number of items o such that o is cached by an ISP i but $\rho_i(o) > K_i$.

We consider that the items with highest arrival intensity are the same among the different ISPs, and we denote them by the set $\mathcal{T} = \bigcup_i \mathcal{T}_i$. We start by investigating the cache allocations that are most likely to occur in the case of disjoint interests. In this case the K_i items with highest arrival intensity of the ISPs form disjoint sets, namely $\mathcal{T}_i \cap \mathcal{T}_j = \emptyset$, for all $i \neq j \in N$. We will first show the following

Lemma 2. *Let \mathcal{C}^* be the allocation in which every ISP caches its most popular items, namely $\mathcal{C}_i^* = \mathcal{T}_i$. For any absorbing state \mathcal{C}' such that $h(\mathcal{C}') > 2$, there exists an absorbing state \mathcal{C}'' such that $h(\mathcal{C}'') = 2$ and $r(\mathcal{C}^*, \mathcal{C}'') < r(\mathcal{C}^*, \mathcal{C}')$.*

Proof: Let \mathcal{S} be the path with least resistance from \mathcal{C}^* to \mathcal{C}' . Observe that, since $\mathcal{C}_i^* = \mathcal{T}_i$, at least $h(\mathcal{C}')$ mistakes are needed to reach \mathcal{C}' . Denote by i the first ISP that makes a mistake in \mathcal{S} , and by o and q the mistakenly evicted and inserted items, respectively. Since \mathcal{S} is the path with least resistance, there exists $j \in \mathcal{N}(i)$ that makes at least one mistake. Consider the first mistake of ISP j and call p and r the evicted and inserted item, respectively. Observe that $o, p \in \mathcal{T}_i \cup \mathcal{T}_j$. We will now show that these two mistakes are enough to reach the absorbing state \mathcal{C}'' defined as $\mathcal{C}_j'' = \mathcal{C}_j^* \setminus \{p\} \cup \{o\}$, $\mathcal{C}_i'' = \mathcal{C}_i^* \setminus \{o\} \cup \{p\}$, $\mathcal{C}_h'' = \mathcal{C}_h^* \forall h \in N \setminus \{i, j\}$, and hence $r(\mathcal{C}^*, \mathcal{C}'') < r(\mathcal{C}^*, \mathcal{C}')$. Let us start from \mathcal{C}^* and consider the state reached after committing the two mistakes. Observe that, since $q \notin \mathcal{T}_i \cup \mathcal{T}_j$, then $\rho_i(p) < \rho_i(q)$. Furthermore we know that there is no ISP $h \neq j$, such that $p \in \mathcal{C}_h^*$. Hence ISP i can evict q and insert p without making a mistake. If $r = o$ then we reached \mathcal{C}'' . If $r \neq o$ then, following the same argument, ISP j can insert o and evict r without making a mistake, reaching \mathcal{C}'' . ■

We will now use Lemma 2 to prove the following

Proposition 6. *If $\mathcal{T}_i \cap \mathcal{T}_j = \emptyset$ for all $i \neq j \in N$, then $\lim_{\frac{1}{\beta} \rightarrow \infty} P(\mathcal{C}(t) = \mathcal{C}^*) = 1$.*

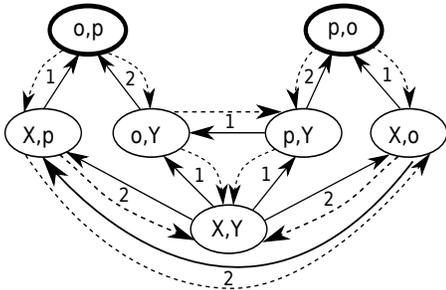


Fig. 3. State transition diagram of the unperturbed Markov process (solid lines). (o, p) and (p, o) are absorbing states in the unperturbed Markov process, but only the equilibrium (o, p) is the domain of μ^0 .

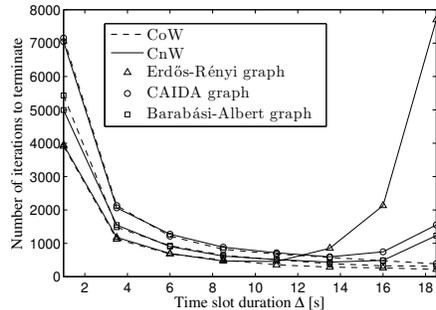


Fig. 4. Average number of iterations needed to reach an equilibrium allocation as a function of the time slot duration Δ for three different peering graphs and algorithms CoW and CNW.

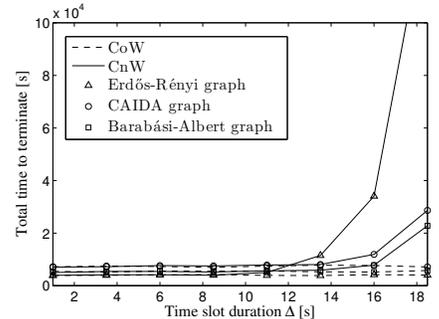


Fig. 5. Average time needed to terminate as a function of the time slot duration Δ for three different peering graphs and algorithms CoW and CNW.

Proof: As a consequence of Lemma 2 it is sufficient to show that for every absorbing state \mathcal{C}'' such that $h(\mathcal{C}'') = 2$, it holds that $r(\mathcal{C}^*, \mathcal{C}'') > r(\mathcal{C}'', \mathcal{C}^*)$. For brevity define \mathcal{C}'' as in the proof of Lemma 2. Assume, w.l.o.g., that in the path with least resistance from \mathcal{C}^* to \mathcal{C}'' , ISP i makes a mistake before ISP j by inserting item $q \notin \mathcal{T}_i \cup \mathcal{T}_j$ in place of item o . Then $r(\mathcal{C}^*, \mathcal{C}'') > w_i^o - w_i^q$. Observe now that, since $q \notin \mathcal{T}_i \cup \mathcal{T}_j$, from the absorbing state \mathcal{C}'' the mistake of ISP i of evicting item p and inserting q , with resistance $w_i^p - w_i^q$, is enough to reach \mathcal{C}^* . Hence $r(\mathcal{C}'', \mathcal{C}^*) \leq w_i^p - w_i^q$. This proves the Proposition. ■

The following illustrates the proof on a simple example.

Example 2. Consider a complete graph and $K_i = 1$. The $|N|$ most popular items are the same in every ISP, but item o has a distinct rank at every ISP. In every equilibrium the $|N|$ most popular items are cached, one at every ISP, and thus there are $|N|!$ equilibria. Fig 3 shows the state transition diagram of the unperturbed Markov process (with solid lines) for the case of two ISPs, $|N| = 2$. The figure only shows the transitions between states. X and Y stand for an arbitrary item other than o and p , and the states (p, Y) and (X, p) ((o, Y) and (X, o)) represent all states in which item p (item o) is cached by ISP 1 and ISP 2, respectively. The dashed lines show transitions due to mistakes that are needed to move from one equilibrium to a state from which both equilibria are reachable (there is a positive probability of reaching it) in the unperturbed process. These transitions only exist in the perturbed Markov process. With perfect information there are two equilibrium allocations, which are the absorbing states (o, p) and (p, o) of the unperturbed process. The two equilibrium allocations are, however, not equally likely to be visited by the perturbed process.

Observe that in the unperturbed process, equilibrium (o, p) is reachable from every allocation except from equilibrium (p, o) . Therefore, in the perturbed process one mistake suffices to leave equilibrium (p, o) and to enter a transient state of the unperturbed process from which both equilibria are reachable in the unperturbed process. It takes, however, two mistakes in close succession to leave equilibrium (o, p) and to enter a transient state of the unperturbed process from which both equilibria are reachable in the unperturbed process. As β decreases, the probability of two successive mistakes decreases exponentially

faster than that of a single mistake, and thus the perturbed process will be almost exclusively in state (o, p) , thus $\mathcal{C}^* = (o, p)$.

A similar reasoning can be used to get insight into the evolution of the system state in the case that the ranking of the items is the same among all ISPs, namely $\mathcal{T}_i = \mathcal{T}_j$ for all $i, j \in N$. As an example, we show the following.

Proposition 7. If the arrival intensity w_i^o for an item o for which $\rho_i(o) \leq K_i$ increases at ISP i , then $\lim_{\beta \rightarrow 0} P(o \in \mathcal{C}_i(t))$ increases.

Proof: Consider the state transition diagram of the perturbed Markov process P^β . For a state \mathcal{C} for which $o \in \mathcal{C}_i$, the transition probability that corresponds to ISP i mistakenly evicting o decreases. For a state \mathcal{C} for which $o \notin \mathcal{C}_i$, the transition probability to the states \mathcal{C}' for which $o \in \mathcal{C}'_i$ increases, and the transition probability to other states decreases. Reconciling these changes with the global balance equation for the set of states $\{\mathcal{C} | o \in \mathcal{C}_i\}$ proves the proposition. ■

The impact of the number of peers of an ISP and that of the amount of storage K_i can be analyzed similarly, but we omit the analysis due to lack of space.

V. NUMERICAL RESULTS

In the following we show simulation results to illustrate the analytical results of Sections III and IV for CoW and CNW.

A. Perfect Information

Figures 4 and 5 show the average number of iterations and the average time the algorithms CoW and CNW need to terminate as a function of the time slot duration Δ , respectively. We report results for three different peering graphs. The CAIDA graph is based on the Internet AS-level peering topology in the CAIDA dataset [20]. The dataset contains 36878 ASes and 103485 transit and peering links between ASes as identified in [21]. The CAIDA graph is the largest connected component of peering ASes in the data set, and consists of 616 ISPs with measured average node degree of 9.66. The Erdős-Rényi (ER) and Barabási-Albert (BA) random graphs have the same number of vertices and the same average node degree as the CAIDA graph. For the CoW algorithm, we used the Welsh-Powell algorithm to find a coloring [22] of the peering graph. We used $\alpha_i = 1$, $\gamma_i = 10$ and cache capacity $K_i = 10$ at every ISP.

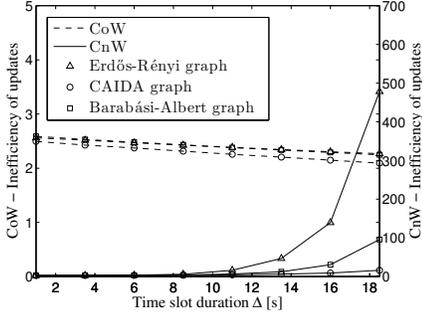


Fig. 6. Average inefficiency as a function of the time slot duration Δ for three different peering graphs and algorithms CoW and CNW.

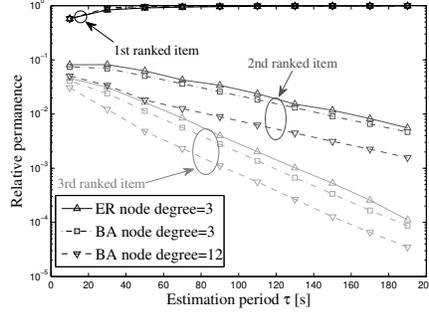


Fig. 7. Relative permanence of the three items with highest arrival intensity in the ISPs' caches, as a function of the intensity estimation interval τ , for three different random peering graphs.

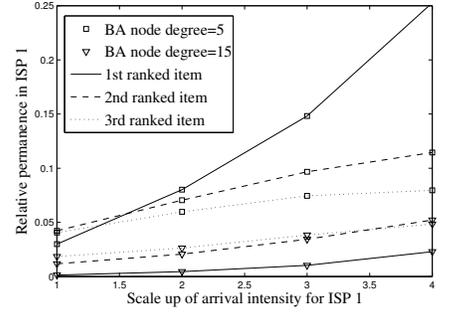


Fig. 8. Relative permanence of the three items with highest arrival intensity in ISP 1's cache, as a function of the scaling factor at ISP 1, for two Barabási-Albert peering graphs with different average node degree.

Each ISP receives interest messages for $|\mathcal{O}| = 3000$ items. The arrival intensities w_i^o follow Zipf's law with exponent 1, and for all $i \in N$ it holds $\sum_{o \in \mathcal{O}} w_i^o = 1$. Each data point in the figures is the average of the results obtained from 40 simulations.

Figure 4 shows that the number of iterations the CoW algorithm needs to reach an equilibrium allocation monotonically decreases with the time slot length. The longer the time slots, the more interest messages the ISPs receive within a time slot. This enables the ISPs to insert more highly popular objects per iteration. Furthermore, since only ISPs in an independent set can make updates at each iteration, simultaneous cache updates like the ones shown in Example 1 cannot occur. Consistently, the total time needed for the CoW algorithm to converge, shown in Figure 5, remains constant independent of the slot length Δ .

The CNW algorithm exhibits significantly different behavior for long time slots, as the number of iterations needed to terminate increases compared to the CoW algorithm. This happens because using the CNW algorithm a higher number of arrivals per time slot leads to a higher number of simultaneous updates, which disturb convergence. Figure 4 shows that simultaneous updates are most likely to occur in ER graphs. In BA graphs simultaneous updates would occur mainly among the few nodes with high degree, and since most ISPs have low node degree, the CNW algorithm would converge faster than on ER graphs. For the same reason, for small time slots when simultaneous updates are unlikely to occur, both the CoW and CNW algorithms perform best on the Erdős-Rényi random graph. From Figure 5 we notice that, as expected, the time for the CNW algorithm to terminate starts to increase with high values of the slot length. This increase is fast for the ER graph due to the higher occurrence of simultaneous updates, as we discussed above.

Figure 6 shows the number of items inserted in cache (potentially several times) for the two algorithms until termination divided by the minimum number of items needed to be inserted to reach the same equilibrium. We refer to this quantity as the *inefficiency of updates*. While the inefficiency of the CoW algorithm decreases slowly with the time slot length, that of the CNW algorithm shows a fast increase for high values of Δ , in particular for the ER and the BA graphs, which can be attributed to the simultaneous updates under CNW. These results show that although CNW would be more appealing as it allows ISPs

to update their cache contents all the time, CoW terminates significantly faster and is more efficient.

B. Imperfect Information

In the following we show results for the case when the estimation of the items' arrival intensities is imperfect. We consider that every ISP estimates the arrival intensities of the items by counting the number of arrivals under a period of τ seconds. As in the case of imperfect information the CoW algorithm would never terminate, we collected the statistics on the permanence of the various items in the cache of each ISP over 10^5 time slots. We considered 50 ISPs and a time slot of 70 seconds, which in the case of perfect information would guarantee a fast termination of the CoW algorithm. We first validate Proposition 6 for the case of $K_i = 1$, hence we consider that the item with the highest arrival intensity is different at every ISP. Figure 7 shows the average relative permanence in the ISPs' caches of the three items with highest arrival intensity, as a function of the estimation interval τ , for three random peering graphs. The results show that the probability of caching the item with highest arrival intensity approaches 1 when τ increases, and thus validate Proposition 6. Furthermore we observe that the probability of caching items with lower arrival intensities decreases exponentially with τ .

In the next scenario we start from the setting described in Proposition 7, where the ranking of the items' intensities is the same among all ISPs. We scale the arrival intensity w_1^o of every item o at ISP 1 by the same factor, while keeping the intensities at the other ISPs constant. Figure 8 shows the average relative permanence in ISP 1's cache of the three items with highest arrival intensities as a function of w_1 . The results confirm that a higher w_1 leads to a higher relative permanence in the ISP's cache of the items with highest arrival intensity. Concerning the influence of the peering graph, the figure shows a constantly lower permanence of the best items for the BA graph with higher average node degree. This is due to that with a higher number of peering links the probability that the best items are in a peering ISP's cache gets higher.

VI. RELATED WORK

There is a large variety of cache eviction policies from Least recently used (LRU) to the recent Adaptive replacement

cache [15]. Most analytical work on the performance of cache eviction policies for stand-alone caches focused on the LRU policy [23], [24], [13]. An iterative algorithm for calculating the cache hit rate was proposed in [23], closed-form asymptotic results were provided for particular popularity distributions in [24] and recently in [13]. These works considered stand-alone caches.

The cache hit rate for cache hierarchies was investigated in the context of web caches and content-centric networks [25], [26], [27], [28]. General topologies were considered for content-centric networks [4], [12], [6]. An iterative algorithm to approximate the cache miss rate in a network of caches was proposed in [12]. The authors in [4] considered various network topology-aware policies to improve the overall cache hit rate in a network of caches. In [6] the authors probabilistic caching to increase the cache hit rate in a network of caches. These works consider that the caches route requests irrespective of the associated traffic costs, and assume a single network operator with a single performance objective. In our work we account for the profit maximizing behavior of individual network operators and model the resulting interaction between caches.

Replication for content delivery in a hierarchy of caches was considered recently in [29], [30]. The authors in [29] considered a centralized algorithm for content placement, while distributed algorithms were analyzed in [30]. A game theoretical analysis of equilibria and convergence for the case of replication on an arbitrary topology was provided in [31]. Opposed to replication, we consider an arbitrary topology of caches, and we consider that caches do not follow an algorithm engineered for good global performance but they follow their individual interests.

To the best of our knowledge ours is the first work that considers a network of selfish caches, including the effects of evictions, and provides a game-theoretical analysis of the resulting cache allocations.

VII. CONCLUSION

We proposed a model of the interactions between the caches managed by peering ASes in a content-centric network. We used the model to investigate whether peering ASes need to coordinate in order to achieve stable and efficient cache allocations in the case of content-level peering. We showed that irrespective of whether the ISPs coordinate, the cache allocations of the ISPs engaged in content-level peering will reach a stable state. However, in order for the resulting stable cache configurations to be efficient in terms of cost, the ASes would have to periodically exchange information about the content of their caches. If fast convergence to a stable allocation is important too then synchronization is needed to avoid simultaneous cache evictions by peering ISPs. Furthermore, we showed that if the content popularity estimates are inaccurate, content-peering is likely to lead to cost efficient cache allocations, and we gave insight into the structure of the most likely cache allocations.

REFERENCES

- [1] T. Koponen, M. Chawla, B.-G. Chun, A. Ermolinskiy, K. H. Kim, S. Shenker, and I. Stoica, "A data-oriented (and beyond) network architecture," in *Proc. of ACM SIGCOMM*, vol. 37, no. 4, 2007, pp. 181–192.
- [2] C. Dannewitz, "NetInf: An Information-Centric Design for the Future Internet," in *Proc. of GI/IG KuVS Work. on The Future Internet*, 2009.
- [3] V. Jacobson, D. K. Smetters, J. D. Thornton, M. F. Plass, N. H. Briggs, and R. L. Braynard, "Networking named content," in *Proc. of ACM CoNEXT*, 2009.
- [4] D. Rossi and G. Rossini, "On sizing CCN content stores by exploiting topological information," in *Proc. of IEEE Infocom, NOMEN Workshop*, 2012, pp. 280–285.
- [5] E. J. Rosensweig and J. Kurose, "Breadcrumbs: Efficient, Best-Effort Content Location in Cache Networks," in *Proc. of IEEE Infocom*, 2009, pp. 2631–2635.
- [6] I. Psaras, W. K. Chai, and G. Pavlou, "Probabilistic In-Network Caching for Information-Centric Networks," in *ICN workshop*, 2012, pp. 1–6.
- [7] P. Faratin, D. Clark, P. Gilmore, S. Bauer, A. Berger, and W. Lehr, "Complexity of Internet Interconnections : Technology , Incentives and Implications for Policy," in *Proc. of Telecommunications Policy Research Conference*, 2007, pp. 1–31.
- [8] Z. M. Mao, R. Govindan, G. Varghese, and R. H. Katz, "Route Flap Damping Exacerbates Internet Routing Convergence," in *Proc. of ACM SIGCOMM*, vol. 32, no. 4, 2002, p. 221.
- [9] S. Agarwal, C.-N. Chuah, S. Bhattacharyya, and C. Diot, "The impact of BGP dynamics on intra-domain traffic," in *Proc. of ACM SIGMETRICS*, vol. 32, no. 1, 2004, p. 319.
- [10] R. Sami, M. Schapira, and A. Zohar, "Searching for Stability in Interdomain Routing," in *Proc. of IEEE Infocom*, 2009, pp. 549–557.
- [11] L. G. L. Gao and J. Rexford, "Stable Internet routing without global coordination," *IEEE/ACM Trans. Netw.*, vol. 9, no. 6, pp. 681–692, 2001.
- [12] E. J. Rosensweig, J. Kurose, and D. Towsley, "Approximate Models for General Cache Networks," in *Proc. of IEEE INFOCOM*, 2010, pp. 1–9.
- [13] C. Fricker, P. Robert, and J. Roberts, "A versatile and accurate approximation for LRU cache performance," in *Proc. of the 24th International Teletraffic Congress*, 2012.
- [14] E. G. Coffman Jr. and P. J. Denning, *Operating Systems Theory*, 1973.
- [15] N. Megiddo and D. Modha, "ARC: A Self-Tuning, Low Overhead Replacement Cache," in *Proc. of FAST*, 2003, pp. 115 – 130.
- [16] J. G. Skellam, "The frequency distribution of the difference between two Poisson variates belonging to different populations." *Journal Of The Royal Statistical Society*, vol. 109, no. 3, p. 296, 1946.
- [17] D. Monderer and L. S. Shapley, "Potential games," *Games and Economic Behavior*, vol. 14, pp. 124–143, 1996.
- [18] H. P. Young, *Strategic learning and its limits*, 2004.
- [19] —, "The evolution of conventions," *Econometrica: Journal of the Econometric Society*, vol. 61, no. 1, pp. 57–84, 1993.
- [20] "CAIDA. Automated Autonomous System (AS) ranking." [Online]. Available: <http://as-rank.caida.org/data/>
- [21] X. Dimitropoulos, D. Krioukov, M. Fomenkov, B. Huffaker, Y. Hyun, K. Claffy, and G. Riley, "AS relationships: inference and validation," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 37, pp. 29–40, 2007.
- [22] D. J. A. Welsh and M. B. Powell, "An upper bound for the chromatic number of a graph and its application to timetabling problems," *The Computer Journal*, vol. 10, no. 1, pp. 85–86, 1967.
- [23] A. Dan and D. Towsley, "An approximate analysis of the LRU and FIFO buffer replacement schemes," in *Proc. of ACM SIGMETRICS*, vol. 18, no. 1, 1990, pp. 143–152.
- [24] P. R. Jelenković, "Asymptotic approximation of the move-to-front search cost distribution and least-recently used caching fault probabilities," *The Annals of Applied Probability*, vol. 9, no. 2, pp. 430–464, 1999.
- [25] M. Busari and C. Williamson, "Simulation Evaluation of a Heterogeneous Web Proxy Caching Hierarchy," in *Proc. of MASCOTS*, 2001, p. 379.
- [26] H. Che, Z. Wang, and Y. Tung, "Analysis and design of hierarchical Web caching systems," in *Proc. of IEEE Infocom*, vol. 3, 2001, pp. 1416–1424.
- [27] C. Williamson, "On filter effects in web caching hierarchies," *ACM Trans. Int. Tech.*, vol. 2, no. 1, pp. 47–77, 2002.
- [28] I. Psaras, R. G. Clegg, R. Landa, W. K. Chai, and G. Pavlou, "Modelling and evaluation of CCN-caching trees," in *Proc. of IFIP Networking*, 2011, pp. 78–91.
- [29] M. Korupolu and M. Dahlin, "Coordinated placement and replacement for large-scale distributed caches," *IEEE Trans. Knowl. Data Eng.*, vol. 14, no. 6, pp. 1317–1329, 2002.
- [30] S. Borst, V. Gupta, and A. Walid, "Distributed Caching Algorithms for Content Distribution Networks," in *Proc. of IEEE Infocom*, 2010, pp. 1–9.
- [31] V. Pacifici and G. Dán, "Selfish content replication on graphs," in *Proc. of the 23rd International Teletraffic Congress*, 2011.