Coordinated Selfish Distributed Caching for Peering Content-centric Networks

Valentino Pacifici and György Dán

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 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. We show that content-level peering leads to stable cache configurations, both with and without coordination. However, peering ISPs that coordinate to avoid simultaneous updates converge to a stable configuration more efficiently. 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.**

Index Terms—**Content-centric networks, cache networks, autonomous caches, content peering, stable content allocations.**

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], [7].

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 [8], 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 [9], [10], the number of updates needed for BGP convergence [11], and general conditions for cycle-free IP routes [12].

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 cache network optimized for local performance. In lack of a central authority capable of enforcing a globally optimal allocation of contents to caches, it will be the interaction between cache networks that will determine the global cache allocation. 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 in terms of the 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 charged less than 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.

In this paper we address this question 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 efficiency of the caching decisions 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

Manuscript submitted for review March 3, 2015; revised 21 October 2015.
Valentino Pacifici and György Dán are with the Laboratory for Communication Networks, School of Electrical Engineering, and the ACCESS Linnaeus Centre, KTH Royal Institute of Technology. Osquldas väg 10, 10044 Stockholm, Sweden. E-mail: {pacifici,gyuri}@kth.se.
The work was partly funded by the Swedish Research Council through projects 2010:581 and 621-2014-635.

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 [13], [14], 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) [15], [13], [14] 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 $\mathcal{C}_i \in \mathfrak{C}_i = \{\mathcal{C} \subset \mathcal{O} : |\mathcal{C}| = K_i\}$, 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, and by β_i the unit cost of retrieving an item from a peering ISP $j \in \mathcal{N}(i)$. The traffic on the transit link is charged by volume with unit cost γ_i, and we make the reasonable assumption that $\gamma_i > \beta_i \geq \alpha_i$. A particular case of interest is when retrieving an item from a peering cache is not more costly than retrieving it locally, i.e., $\beta_i = \alpha_i$ for all $i \in N$. We refer to this case as *free peering*. The model of *free peering* is motivated by that, once a peering link has been established between two ISPs, there is no additional cost for traffic.

B. Content-peering

We consider that time is divided into time slots, and peering ISPs *synchronously exchange information* about the contents of their summary caches *periodically*, at the end of every time slot. 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 its local cache. Otherwise, before ISP i would forward the interest message to its transit provider, it tries to leverage its neighbors' caches by consulting 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

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

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

$$C_i(\mathcal{C}_i, \mathcal{C}_{-i}) = \sum_{o \in \mathcal{O}} C_i^o(\mathcal{C}_i, \mathcal{C}_{-i}) \tag{2}$$

$$= \alpha_i \sum_{\mathcal{L}_i} w_i^o + \beta_i \sum_{\mathcal{R}_i \setminus \mathcal{L}_i} w_i^o + \gamma_i \sum_{\mathcal{O} \setminus \{\mathcal{L}_i \cup \mathcal{R}_i\}} w_i^o, \tag{3}$$

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 locally is obtained from a peering ISP or 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 [16] for a survey of some recent algorithms).

We model the decision whether to evict item $p \in \mathcal{C}_i$ based on the comparison of the cost C_i incurred by ISP i when caching item o in place of item $p \in \mathcal{C}_i$.

ISP i caches item $o \notin \mathcal{C}_i$ in place of item $p \in \mathcal{C}_i$ if

$$C_i(\mathcal{C}_i \setminus \{p\} \cup \{o\}, \mathcal{C}_{-i}) < C_i(\mathcal{C}_i, \mathcal{C}_{-i}), \tag{4}$$

which is equivalent to

$$C_i^o(\varnothing, \mathcal{C}_{-i}) - C_i^o(\{o\}, \mathcal{C}_{-i}) > C_i^p(\varnothing, \mathcal{C}_{-i}) - C_i^p(\{p\}, \mathcal{C}_{-i}).$$

Let $\Psi_i^q(\mathcal{C}_{-i}) \triangleq C_i^q(\varnothing, \mathcal{C}_{-i}) - C_i^q(\{q\}, \mathcal{C}_{-i})$ be the cost saving of ISP i for caching item $q \in \mathcal{O}$. We consider that, in order to make a caching decision, ISP i calculates the cost savings $\Psi_i^o(\mathcal{C}_{-i})$ and $\Psi_i^p(\mathcal{C}_{-i})$ based on its estimates $\overline{w}_i^o$ and $\overline{w}_i^p$ of the arrival intensities w_i^o and w_i^p for the item o to be cached and for the item p in the cache, respectively. We consider two cases with respect to the accuracy of the arrival intensity estimates.

Perfect information: Under perfect information $\overline{w}_i^o = w_i^o$ $\forall o \in \mathcal{O}$.

Imperfect information: Under imperfect information we consider that the probability of misestimation decreases exponentially with the difference in cost savings, that is, for $\Psi_i^o(\mathcal{C}_{-i}) > \Psi_i^p(\mathcal{C}_{-i})$ we have

$$P(\overline{\Psi}_i^o(\mathcal{C}_{-i}) < \overline{\Psi}_i^p(\mathcal{C}_{-i})) \propto \epsilon e^{-\frac{1}{\nu}(\Psi_i^o(\mathcal{C}_{-i}) - \Psi_i^p(\mathcal{C}_{-i}))} \quad (5)$$

As an example, consider an ISP i and a content allocation $\mathcal{C}$ such that $\mathcal{R}_i(\mathcal{C}) = \varnothing$, then (5) would have the form

$$P(\overline{\Psi}_i^o(\mathcal{C}_{-i}) < \overline{\Psi}_i^p(\mathcal{C}_{-i})) \propto \epsilon e^{-\frac{1}{\nu}(w_i^o[\gamma_i - \alpha_i] - w_i^p[\gamma_i - \alpha_i])} = \epsilon e^{-\frac{1}{\nu}[\gamma_i - \alpha_i](w_i^o - w_i^p)}.$$

This method of modeling the misestimation probability 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 [14]. Under a perfect LFU policy, if we denote the interval over which the request frequencies are calculated by τ, then the estimate $\overline{w}_i^p$ follows a Poisson distribution with parameter $w_i^p\tau$. The difference $k = \overline{w}_i^o\tau - \overline{w}_i^p\tau$ of two estimates thus follows the Skellam distribution [17] 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 $I_{|k|}(.)$ 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.

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 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 $\mathcal{C}_1 = \mathcal{C}_2 = \{1\}$ and the cost for retrieving an item from a peer is small enough, i.e. $\beta_i \gtrsim \alpha_i$, 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 $C_i' = \beta_i \left(\frac{w_i^{h_i} + w_i^{l_i}}{2}\right) + \gamma_i \left(\frac{w_i^{h_i} + w_i^{l_i}}{2}\right) > C_i$.*

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. Such an allocation corresponds to a pure strategy Nash equilibrium of the strategic game $\Gamma =< N, (\mathfrak{C}_i)_{i \in N}, (C_i)_{i \in N} >$, in which each ISP i aims to minimize its own cost C_i defined in (3).

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

$$\forall i \in N, \forall \mathcal{C}_i \in \mathfrak{C}_i : C_i(\mathcal{C}_i^*, \mathcal{C}_{-i}^*) \leq C_i(\mathcal{C}_i, \mathcal{C}_{-i}^*) \quad (6)$$

The strategic game Γ was extensively studied in [18], the following result follows from [18, Theorem 1],

Theorem 1. *In the case of perfect information there is at least one equilibrium allocation.*

In the following we propose three distributed algorithms that avoid the inefficient updates shown in Example 1 and allow the system to reach an equilibrium allocation of items from which no ISP has an interest to deviate.

A. Asynchronous (Async) Algorithm

Example 1 suggests that if the ISPs coordinate so that they do not 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 $(\{2\}, \{1\})$. This intuition provides the rationale for the Async algorithm.

We start the description of the algorithm with the following definition.

Definition 2. A sequence $N_t \subseteq N$, $t = 1, \ldots$ of sets of ISPs is a complete sequence, if for all time slots t and each ISP $i \in N$ there exists a time slot $t' > t$ such that $i \in N_{t'}$.

In the Async algorithm N_t is a singleton, and at each time slot t, the algorithm allows the unique ISP $i_t \in N_t$ to update the set of its cached content $\mathcal{C}_{i_t}$. ISP 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, the rest of the ISPs $N \setminus \{i_t\}$ are not allowed to update the set of their cached contents. The pseudocode of the Async algorithm for every time slot $t \geq 1$ is then the following:

What we are interested in is whether ISPs following the Async algorithm would reach an equilibrium allocation from

Async Algorithm

```
INPUT: arbitrary cache allocation C(0)
t ← 1
At time slot t do
    Allow ISP i_t ∈ N_t to change its cached items
    from C_{i_t}(t − 1) to C_{i_t}(t)
    for all j ∈ N \ {i_t} do C_j(t) = C_j(t − 1) end
    At the end of the time slot inform the ISPs j ∈ 𝒩(i_t)
    about the new cache contents C_{i_t}(t)
    t ← t + 1
end
```

Fig. 1. Pseudo-code of the Asynchronous (Async) Algorithm

which none of them would like to deviate. If the ASYNC algorithm reaches such an allocation, then it terminates, and no other cache update will take place.

In the following we provide a sufficient condition for the ASYNC algorithm to reach an equilibrium allocation. We call the condition *efficiency*, and the condition concerns the changes that each ISP $i \in N$ can make to its cache configuration.

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

$$C^o_{i_t}(\mathcal{C}(t)) + C^p_{i_t}(\mathcal{C}(t)) < C^o_{i_t}(\mathcal{C}(t-1)) + C^p_{i_t}(\mathcal{C}(t-1)) \quad (7)$$

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 rational changes, i.e., changes that actually lead to lower cost. The concept of efficient update is similar to the concept of better reply in learning in strategic games.

Definition 4. The cache configuration $\mathcal{C}_{i_t}(t)$ is a *better reply* for ISP i_t in the strategic game Γ if $C_{i_t}(\mathcal{C}(t)) < C_{i_t}(\mathcal{C}(t-1))$.

It follows from Definition 3 and Definition 4 that any efficient update is a better reply in the strategic game Γ, although the converse is not true, i.e., some evictions and insertions in a better reply might not be efficient.

We model the evolution of the system state (the set of cache allocations) under the ASYNC algorithm by a Markov chain P^A; the transition probability $P(X_{t+1} = \mathcal{C}'|X_t = \mathcal{C})$ between states $X_t = \mathcal{C}$ and $X_{t+1} = \mathcal{C}' \neq \mathcal{C}$ at time slot t is non-zero if and only if there exists exactly one ISP $i \in N$ such that $\mathcal{C}'_i \neq \mathcal{C}_i$ and the update $\mathcal{C}_i$ to $\mathcal{C}'_i$ by ISP i is an efficient update. In that case, the probability $P(X_{t+1} = \mathcal{C}'|X_t = \mathcal{C})$ is directly proportional to the probability that ISP i receives the interest messages necessary to update its allocation from $\mathcal{C}_i$ to $\mathcal{C}'_i$ during time slot t. Given that the distribution F^o_i of the inter-arrival times is exponential with parameter w^o_i,

$$P(X_{t+1} = \mathcal{C}'|X_t = \mathcal{C}) \propto \prod_{o \in \mathcal{C}'_i \setminus \mathcal{C}_i} \left[1 - e^{-w^o_i \Delta_t}\right].$$

Observe that the Markov chain P^A is not irreducible, as every equilibrium state is an absorbing state.

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

Proof. We prove the theorem by showing that P^A is an absorbing Markov chain whose absorbing states are the equilibrium allocations defined in (6), therefore the probability that the ASYNC algorithm reaches an equilibrium allocation is 1.

Call $\mathfrak{C}^*$ the set of absorbing states of P^A. Observe that $\mathfrak{C}^*$ corresponds to the set of pure strategy Nash Equilibria of the strategic game $\Gamma =< N, (\mathfrak{C}_i)_{i \in N}, (C_i)_{i \in N} >$ that was shown to be weakly acyclic under best replies in [18]. Weak acyclicity implies that from any state $\mathcal{C}^0 \notin \mathfrak{C}^*$, there exists a finite sequence of cache allocations $\mathcal{C}^0, \mathcal{C}^1, ...\mathcal{C}^{K-1}, \mathcal{C}^K$, such that *1)* $\mathcal{C}^K$ is an equilibrium allocation and *2)* for each $k \in \{1, 2, .., K\}$ there exists exactly one ISP $i^k \in N$ s.t. $\mathcal{C}^k_{i^k} \neq \mathcal{C}^{k-1}_{i^k}$ and $\mathcal{C}^k_{i^k} = \arg\min_{\mathcal{C}_{i^k} \in \mathfrak{C}_{i^k}} C_{i^k}(\mathcal{C}_{i^k}, \mathcal{C}^{k-1}_{-i^k})$. It is easy to see that the update from $\mathcal{C}^{k-1}_{i^k}$ to $\mathcal{C}^k_{i^k}$ is an efficient update of ISP i^k. In the following we prove that, for each $k \in \{1, 2, .., K\}$, state $\mathcal{C}^k$ is accessible from state $\mathcal{C}^{k-1}$ in P^A.

Assume w.l.o.g, that at time slot t process P^A is at state $X_t = \mathcal{C}^{k-1}$. Call t' the smallest time slot such that $t' > t$ and $i_{t'} = i^k$. For each time slot u such that $u \geq t$ and $u < t'$, consider the arrival of interest messages for item o generated by the local users of ISP i_u, with intensity $w^o_{i_u}$. As every update of the cache allocation of ISP i_u is triggered by an interest message sent by a local user, and given that the distribution $F^o_{i_u}$ of the inter-arrival times is exponential with parameter $w^o_{i_u}$, there is a non-zero probability $e^{-w^o_{i_u} \Delta_u}$ that item o is not requested during time slot u of length Δ_u.
It follows that, starting from state $X_t = \mathcal{C}^{k-1}$, the probability that process P^A reaches $X_{t'-1} = \mathcal{C}^{k-1}$ is at least

$$P(X_{t'-1}{=}\, \mathcal{C}^{k-1}|X_t{=}\, \mathcal{C}^{k-1}) \geq \prod_{u=t}^{t'-1} \left[\prod_{o \notin \mathcal{C}_{i_u}} e^{-w^o_{i_u} \Delta_u}\right] > 0.$$

Hence, the probability $P(\mathcal{C}^k|\mathcal{C}^{k-1})$ to reach state $\mathcal{C}^k$ from $\mathcal{C}^{k-1}$ under P^A is at least

$$\prod_{u=t}^{t'-1} \left[\prod_{o \notin \mathcal{C}_{i_u}} e^{-w^o_{i_u} \Delta_u}\right] \cdot \prod_{o \in \mathcal{C}^k_{i^k} \setminus \mathcal{C}^{k-1}_{i^k}} \left[1 - e^{-w^o_{i^k} \Delta_t}\right] > 0. \quad (8)$$

The second term of (8) is a lower bound on the probability that during time slot t', ISP i^k receives the interest messages necessary to update its cache allocation from $\mathcal{C}^{k-1}_{i^k}$ to $\mathcal{C}^k_{i^k}$. It follows that P^0 is an absorbing Markov chain, which proves the theorem. □

B. Cache-or-Wait (COW) Algorithm

A significant shortcoming of the ASYNC algorithm is that in slot t it disallows any ISP $j \in N \setminus \{i_t\}$ to perform an update. As a consequence, one ISP can perform an update on average every $|N|$ time slots. This restriction would provide little incentive for ISPs to adhere to the algorithm. In the following we investigate the effects of relaxing the requirement of strict coordination by allowing multiple ISPs to perform efficient updates during the same time slot.

Example 1 suggests that the oscillating behavior of the cache content is a consequence of the simultaneous cache updates by *neighboring* ISPs. The Cache-or-Wait (COW) algorithm only allows *non-neighboring* ISPs to perform simultaneous updates. Before we describe the COW algorithm, let us recall the notion of an independent set.

Definition 5. 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 $\mathfrak{I}$ the set of all the independent sets of the peering graph $\mathcal{G}$. Consider a sequence of time slots t and a complete sequence of independent sets $\mathcal{I}_1, \mathcal{I}_2, \ldots \in \mathfrak{I}$ indexed by t. At each time slot t we allow every ISP $i \in \mathcal{I}_t$ to update the set of its cached content $\mathcal{C}_i$. 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:

CoW Algorithm

INPUT: arbitrary cache allocation $\mathcal{C}(0)$
$t \leftarrow 1$
At time slot t **do**
  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 $j \notin \mathcal{I}_t$ **do** $\mathcal{C}_j(t) = \mathcal{C}_j(t-1)$ **end**
  At the end of the time slot inform the ISPs $j \in \mathcal{N}(i)$ about the new cache contents $\mathcal{C}_i(t)$
  $t \leftarrow t+1$
end

Fig. 2. Pseudo-code of the Cache-or-Wait (CoW) Algorithm

Theorem 3. *If every ISP performs only efficient updates, then the* CoW *algorithm terminates in an equilibrium allocation with probability 1.*

Proof. The proof follows the same arguments as the proof of Theorem 2. We model the evolution of the system state under the CoW algorithm by a Markov chain P^W. P^W has the same state space as P^A but differs in terms of transition probabilities. In P^W, the transition probability between states $\mathcal{C}$ and $\mathcal{C}' \neq \mathcal{C}$ is non-zero if and only if there exists $\mathcal{I} \in \mathfrak{I}$ such that for every ISP i for which $\mathcal{C}'_i \neq \mathcal{C}_i$ we have $i \in \mathcal{I}$, and the update from $\mathcal{C}_i$ to $\mathcal{C}'_i$ is an efficient update. In the following we calculate a lower bound on the probability $P(\mathcal{C}^k|\mathcal{C}^{k-1})$ under P^W, for each $k \in \{1, 2, .., K\}$ in the finite sequence of cache allocations $\mathcal{C}^0, \mathcal{C}^1, \ldots \mathcal{C}^{K-1}$.

Assume that, at time slot t, process P^W is at state $X_t = \mathcal{C}^{k-1}$. Call i^k the ISP s.t. $\mathcal{C}^k_{i^k} \neq \mathcal{C}^{k-1}_{i^k}$ and t' the smallest time slot such that $t' > t$ and $i^k \in \mathcal{I}_{t'}$. The probability that process P^W will be at state $X_{t'-1} = \mathcal{C}^{k-1}$ at the beginning of time slot t' is lower bounded by the probability that no ISP $j \in \{\mathcal{I}_u | u \geq t, u \leq t'-1\}$ receives an interest message that triggers an update, that is

$$P(X_{t'-1} = \mathcal{C}^{k-1} | X_t = \mathcal{C}^{k-1}) \geq \prod_{u=t}^{t'-1} \prod_{j \in \mathcal{I}_u} \left[\prod_{o \notin \mathcal{C}^k_j} e^{-w^o_j \Delta_u} \right] > 0.$$

Similarly, we can calculate a lower bound on the probability of the transition from state $\mathcal{C}^{k-1}$ to state $\mathcal{C}^k$ during time slot t' as

$$P(X_{t'} = \mathcal{C}^k | X_{t'-1} = \mathcal{C}^{k-1}) \geq \prod_{j \in \mathcal{I}_{t'} \setminus \{i^k\}} \left[\prod_{o \notin \mathcal{C}_j} e^{-w^o_j \Delta_{t'}} \right] \cdot \prod_{o \in \mathcal{C}^k_{i^k} \setminus \mathcal{C}^{k-1}_{i^k}} \left[1 - e^{-w^o_{i^k} \Delta_{t'}} \right] > 0.$$

The first term of the product is a lower bound on the probability that no ISP $j \in \mathcal{I}_{t'}$ other than i^k receives an interest message that triggers an update at time slot t'. The second term was introduced in the proof of Theorem 2.

Finally, we can express a lower bound on the probability $P(\mathcal{C}^k|\mathcal{C}^{k-1})$ as

$$P(\mathcal{C}^k|\mathcal{C}^{k-1}) \geq P(X_{t'-1} = \mathcal{C}^{k-1} | X_t = \mathcal{C}^{k-1}) \cdot P(X_{t'} = \mathcal{C}^k | X_{t'-1} = \mathcal{C}^{k-1}) > 0.$$

This proves the theorem. □

The following corollary is a consequence of Theorem 3

Corollary 1. *If every ISP performs efficient updates, then the number of* time slots *needed by the* CoW *algorithm to reach an equilibrium allocation is finite with probability 1, and thus the number of efficient updates is finite with probability 1.*

In the following we prove a stronger result on the number of *efficient updates* required to reach an equilibrium allocation in the *free peering* case. Recall that, in the *free peering* case, $\alpha_i = \beta_i$ for all $i \in N$.

Theorem 4. *In the* free peering *case, if every ISP performs efficient updates then the* CoW *algorithm terminates in an equilibrium allocation after a finite number of efficient updates.*

Proof. We will prove the theorem by showing that there exists a global function $\Psi : \times_i(\mathfrak{C}_i) \to \mathbb{R}$ that strictly increases at every efficient update made by any ISP i following the CoW algorithm. We define $\Psi(\mathcal{C}) = \sum_{i \in N} \Psi_i(\mathcal{C})$, where $\Psi_i(\mathcal{C})$ is the cost saving of ISP i for allocation $\mathcal{C} = (\mathcal{C}_i, \mathcal{C}_{-i})$,

$$\Psi_i(\mathcal{C}) = \sum_{o \in \mathcal{C}_i} \Psi^o_i(\mathcal{C}_{-i}) = \sum_{o \in \mathcal{C}_i} \left(C^o_i(\varnothing, \mathcal{C}_{-i}) - C^o_i(\{o\}, \mathcal{C}_{-i}) \right).$$

Without loss of generality, consider the efficient update $\mathcal{C}_i(t)$ made by ISP $i \in \mathcal{I}_t$ at time slot t. In the following we show that $\Psi_j(\mathcal{C}_i(t), \mathcal{C}_{-i}(t-1)) \geq \Psi_j(\mathcal{C}(t-1))$ for all $j \in N$. Observe that it follows directly from the definition of $\Psi_i(\mathcal{C})$ that for ISP i

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

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

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

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 $C^p_i(t+1) \geq C^p_i(t)$. Substituting it in the definition of efficient improvement step in (7), it follows that $C^o_i(t) > C^o_i(t+1) \Rightarrow o \notin \mathcal{R}_i(t) \Rightarrow o \notin \mathcal{C}_j(t)$, thus $\Psi_j(\mathcal{C}_i(t), \mathcal{C}_{-i}(t-1))$ is not affected by item o.
Consider now item p:

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

Therefore, the function Ψ increases strictly upon every efficient update. Since $\times_i(\mathfrak{C}_i)$ is a finite set, Ψ cannot increase indefinitely and the CoW algorithm must terminate in an equilibrium allocation after a finite number of efficient updates.

Note that, in game theoretical terminology, the function Ψ is a *generalized ordinal potential function* for the strategic game Γ. $\square$

Thus, a network of ISPs, in which only non-peering ISPs perform efficient updates during the same time slot, eventually reaches an equilibrium allocation. 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 up to every $\chi(\mathcal{G})^{th}$ time slots. In the worst case, for complete peering graphs, for which $\chi(\mathcal{G}) = |N|$, the CoW algorithm would be equivalent to the ASYNC algorithm, hence the ISPs would have little incentive to adhere to the CoW algorithm.

C. Cache-no-Wait (CNW) Algorithm

In the following we investigate what happens if we allow every ISP $i \in N$ in the system to update the set of its cached content $\mathcal{C}_i$ during every time slot. The pseudo-code of the CNW algorithm for time slots $t \geq 0$ is shown in Figure 3.

```
CNW Algorithm
INPUT: arbitrary cache allocation C(0)
t ← 1
At time slot t do
    Every ISP i ∈ 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 ∈ 𝒩(i) about the new cache contents C_i(t)
    t ← t + 1
end
```

Fig. 3. Pseudo-code of the Cache-no-Wait (CNW) Algorithm

Using the same arguments as in the proofs of Theorems 2 and 3 we can prove the following.

Theorem 5. *If every ISP performs only efficient updates,* CNW *terminates in an equilibrium allocation with probability 1.*

Proof. Call P^N the Markov chain that models the evolution of the system state under the CNW algorithm. In P^N the transition probability between states $\mathcal{C}$ and $\mathcal{C}' \neq \mathcal{C}$ is non-zero if and only if, for every $i \in N$ such that $\mathcal{C}'_i \neq \mathcal{C}_i$, the update $\mathcal{C}_i$ to $\mathcal{C}'_i$ by ISP i is an efficient update. The probability $P(\mathcal{C}^k|\mathcal{C}^{k-1})$ under P^N, for each $k \in \{1, 2, .., K\}$ is at least

$$\prod_{o \in I_i^k} \left[1 - e^{-w_i^o \Delta_t}\right] \cdot \prod_{j \in N \setminus \{i\}} \prod_{o \notin \mathcal{C}_j} e^{-w_j^o \Delta_t} > 0,$$

where $I_i^k = \mathcal{C}_i^k \setminus \mathcal{C}_i^{k-1}$. It follows that P^0 is an absorbing Markov chain, which proves the theorem. $\square$

We have thus far shown that under perfect information content-level peering will eventually lead to stable cache allocations, independently of whether the ISPs coordinate. In the case of *free peering* we could provide a stronger result, as ISPs that coordinate are guaranteed to reach a stable allocation after a finite number of updates. We will later, in Section V, investigate using simulations how the convergence speeds differ depending on the frequency of coordination. 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 items that yield the lowest cost saving. In the following we consider that the estimation of the item popularities is imperfect. To ease the analysis we consider the ASYNC algorithm throughout the section as the set of equilibria under the ASYNC algorithm coincides with those under the CoW and CNW algorithms.

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 in some cases, it is possible to characterize them.

Call P^ε the Markov process that models the system under imperfect information, where $\varepsilon = e^{-\frac{1}{\nu}}$ is a scalar parameter that indicates the overall level of noise. Let us recall the following definition from [19].

Definition 6. A Markov process P^ε is a regular perturbation of the Markov process P^A if ε takes on all values in some interval $(0, a]$, and the following conditions hold for all $\mathcal{C}, \mathcal{C}' \in \times_{i \in N} \mathfrak{C}_i$:

1) P^ε is aperiodic and irreducible for all $\varepsilon \in (0, a]$,
2) $\lim_{\varepsilon \to 0} P^\varepsilon_{\mathcal{C},\mathcal{C}'} = P^A_{\mathcal{C},\mathcal{C}'}$,
3) $P^\varepsilon_{\mathcal{C},\mathcal{C}'} > 0$ for some ε implies $\exists r(\mathcal{C}, \mathcal{C}') \geq 0$ such that $0 < \lim_{\varepsilon \to 0} \varepsilon^{-r(\mathcal{C},\mathcal{C}')} \cdot P^\varepsilon_{\mathcal{C},\mathcal{C}'} < \infty$.

We can now prove the following.

Theorem 6. *The Markov process P^ε is a regular perturbation of the Markov process P^A.*

Proof. 1) It follows from (5) that under imperfect information the probability that item o will be evicted and item p inserted even though $\Psi_i^o > \Psi_i^p$ is non-zero, therefore for every $\varepsilon > 0$, P^ε is irreducible. Furthermore, it follows from the distribution F_i^o of the inter-arrival times that $P(X_{t+1} = \mathcal{C}|X_t = \mathcal{C}) > 0$, $\forall \mathcal{C} \in \times_i(\mathfrak{C}_i)$, therefore P^ε is aperiodic.
2) From (5), for every $\mathcal{C}$ and $\mathcal{C}'$, P^ε converges to P^A at an exponential rate, $\lim_{\varepsilon \to 0} P^\varepsilon_{\mathcal{C},\mathcal{C}'} = P^A_{\mathcal{C},\mathcal{C}'}$.
3) For all $\mathcal{C}, \mathcal{C}' \in \times_{i \in N} \mathfrak{C}_i$, if $P^\varepsilon_{\mathcal{C},\mathcal{C}'} > 0$ then $\exists i \in N$ such that $\mathcal{C}_i \neq \mathcal{C}'_i$. We partition the inserted set $I = \mathcal{C}'_i \setminus \mathcal{C}_i$ in two sets, R and W, such that R is the set of content items that satisfy condition (7), and W is the set of objects mistakenly inserted. Call f an order of arrivals of the interest messages for the items in I, i.e. f is an ordering of the items in I. We define E_f as the set of items evicted upon the insertion of the items in W. It follows from (5) that the probability $P^\varepsilon(E_f \to W|f)$ of inserting the items in W in place of the items in E_f, given a particular order f of arrivals is

$$P^\varepsilon(E_f \to W|f) = \epsilon^{|W|} e^{-\frac{1}{\nu}\left(\sum_{o \in W} \Psi_i^o(\mathcal{C}_{-i}) - \sum_{p \in E_f} \Psi_i^p(\mathcal{C}_{-i})\right)}, \tag{9}$$

which does not depend on the order of eviction of the objects in E_f. We can then express $P^{\varepsilon}_{\mathcal{C},\mathcal{C}'}$ in the form

$$P^{\varepsilon}_{\mathcal{C},\mathcal{C}'} \propto \prod_{o\in I}\left[1-e^{-w_i^o\Delta}\right]\sum_f P(f)P^{\varepsilon}(E_f\rightarrow W|f). \quad (10)$$

Observe that the probability $P(f)$ that the interest messages arrive in a particular order f is independent from the level of noise ε. Therefore, as $\varepsilon\rightarrow 0$, (10) is dominated by the term $P^{\varepsilon}(E_f\rightarrow W|f)$ with the largest exponent. By defining $r(\mathcal{C},\mathcal{C}')\triangleq \min_f\left(\sum_{o\in W}\Psi_i^o(\mathcal{C}_{-i})-\sum_{p\in E_f}\Psi_i^p(\mathcal{C}_{-i})\right)>0$, we have

$$0<\lim_{\varepsilon\rightarrow 0} e^{\frac{1}{\nu}r(\mathcal{C},\mathcal{C}')}\cdot P^{\varepsilon}_{\mathcal{C},\mathcal{C}'}<\infty,$$

which proves the theorem. □

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 (i.e., $W=\varnothing$). Since P^{ε} is an irreducible aperiodic finite Markov process, it has a unique stationary distribution for $\varepsilon>0$. We now recall a result from Young [19].

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

By Lemma 1 there is thus a stationary distribution μ^A of the unperturbed process such that, for small ε, the system will likely be in a state in the domain of μ^A. As the support of the stationary distribution μ^A is a subset of the recurrent communication classes of P^A, for small ε the perturbed process P^{ε} is likely to be in a particular subset of the equilibrium states. In the rest of the section we characterize the cache allocations that correspond to the equilibrium states that are most likely to be visited, for two scenarios.

A. The free peering *case*

We start by considering the *free peering* case, i.e. $\alpha_i=\beta_i$, for all $i\in N$. Observe that, at every ISP $i\in N$, the cost saving for every item $o\in\mathcal{R}_i(\mathcal{C})$ is $\Psi_i^o(\mathcal{C}_{-i})=w_i^o\left[\beta_i-\alpha_i\right]=0$. In other words, in the *free peering* case, no ISP i will ever insert any item o that is already cached by any of its peering ISPs $\mathcal{N}(i)$.

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 $\mathcal{T}_i\cap\mathcal{T}_j=\varnothing$, for all $i\neq j\in N$. We will first show the following

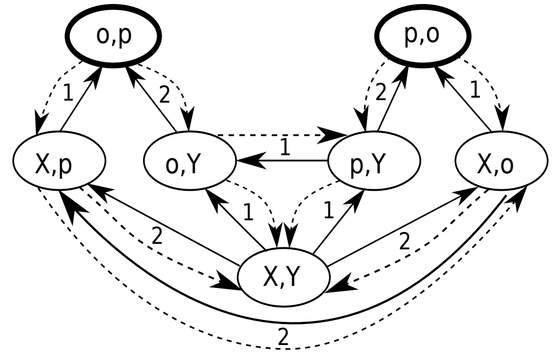

Fig. 4. State transition diagram of the unperturbed Markov process (solid lines). (o,p) and (p,o) are absorbing states in the unperturbed Markov process, between the two equilibria. but only the equilibrium (o,p) is the domain of μ^A.

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 $q\notin\mathcal{T}_i\cup\mathcal{T}_j$, thus $\rho_i(p)<\rho_i(q)$. 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 7. *If $\mathcal{T}_i\cap\mathcal{T}_j=\varnothing$ for all $i\neq j\in N$, then* $\lim_{\varepsilon\rightarrow 0}P(\mathcal{C}(t)=\mathcal{C}^*)=1$.

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 4 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 the level of noise ε 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 8. *If the arrival intensity w_i^o for an item o for which $\rho_i(o) \leq K_i$ increases at ISP i, then $\lim_{\varepsilon \to 0} P(o \in \mathcal{C}_i(t))$ increases.*

Proof. Consider the state transition diagram of the perturbed Markov process P^ε. 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. We omit the analysis for brevity, and turn to the general case instead.

B. The general case

In the following we consider the general case, when $\gamma_i > \beta_i \geq \alpha_i$. We show that the results of the previous section may not hold and show that Proposition 7 does not hold in general. We start by considering a scenario similar to the one described in Example 2, when the K_i items with highest arrival intensity of the ISPs form disjoint sets.

Example 3. *Consider a complete peering graph of 4 ISPs and $K_i = 1\, \forall i \in N$. Let the most popular items of ISPs 1, 2, 3, and 4 be a, o, p and b, respectively. Thus, the allocation in which every ISP caches its most popular item is $\mathcal{C}^* = (a, o, p, b)$. Furthermore, let us assume that the following inequalities hold*

$$w_1^a [\beta_1 - \alpha_1] > w_2^q [\gamma_1 - \alpha_1], \ \forall q \in \mathcal{O} \setminus \{a\}, \quad (11)$$

$$w_2^p [\gamma_2 - \alpha_2] > w_2^r [\gamma_2 - \alpha_2] > w_2^o [\beta_2 - \alpha_2], \quad (12)$$

$$w_3^o [\gamma_3 - \alpha_3] > w_3^s [\gamma_3 - \alpha_3] > w_3^p [\beta_3 - \alpha_3], \quad (13)$$

$$w_4^b [\beta_4 - \alpha_4] > w_4^q [\gamma_4 - \alpha_4], \ \forall q \in \mathcal{O} \setminus \{b\}. \quad (14)$$

Observe that (11)-(14) imply that the allocation $\mathcal{C}'' = (a, p, o, b)$ is an absorbing state, and $h(\mathcal{C}'') = 2$. Furthermore, $\mathcal{C}^*$ and $\mathcal{C}''$ are the only absorbing states.

We are ready to prove the following

Proposition 9. *If $\mathcal{T}_i \cap \mathcal{T}_j = \varnothing$ for all $i \neq j \in N$, then there might exist two or more stochastically stable absorbing states, i.e. $\lim_{\varepsilon \to 0} P(\mathcal{C}(t) = \mathcal{C}^*) < 1$.*

Proof. We use Example 3 to prove the proposition. In particular, we show that for some values of w_1^o and w_4^p, the absorbing states $\mathcal{C}^*$ and $\mathcal{C}''$ have the same stochastic potential.
Consider the following two transitions: ISP 1 inserts item o in place of item a, and ISP 4 inserts item p in place of item b. We refer to the two transitions as (TI) and (T2), respectively. We start by showing that there exists a path from $\mathcal{C}^*$ to $\mathcal{C}''$ in which the transitions (T1) and (T2) are the only transitions with positive resistance. Starting from $\mathcal{C}^*$, after transitions (T1) and (T2), the process reaches allocation (o, o, p, p). It follows from (11) and (14) that reaching $\mathcal{C}''$ does not require additional mistakes

$$(o, o, p, p) \underset{2}{\rightarrow} (o, r, p, p) \underset{3}{\rightarrow} (o, r, s, p) \underset{1}{\rightarrow} (a, r, s, p)$$
$$\underset{4}{\rightarrow} (a, r, s, b) \underset{2}{\rightarrow} (a, p, s, b) \underset{3}{\rightarrow} (\boldsymbol{a}, \boldsymbol{p}, \boldsymbol{o}, \boldsymbol{b}).$$

Similarly, there exists a path from $\mathcal{C}''$ to $\mathcal{C}^*$ in which the transitions (T1) and (T2) are the only transitions with positive resistance. Starting from $\mathcal{C}''$, after transitions (T1) and (T2), the process reaches allocation (o, p, o, p) and then reaches allocation $\mathcal{C}^*$ with no additional mistakes

$$(o, p, o, p) \underset{2}{\rightarrow} (o, r, o, p) \underset{3}{\rightarrow} (o, r, s, p) \underset{1}{\rightarrow} (a, r, s, p)$$
$$\underset{4}{\rightarrow} (a, r, s, b) \underset{2}{\rightarrow} (a, o, s, b) \underset{3}{\rightarrow} (\boldsymbol{a}, \boldsymbol{o}, \boldsymbol{p}, \boldsymbol{b}).$$

Since the peering graph is complete, $\mathcal{R}_1(\mathcal{C}^*) = \mathcal{R}_1(\mathcal{C}'') = \{\boldsymbol{o}, \boldsymbol{p}, b\}$ and $\mathcal{R}_4(\mathcal{C}^*) = \mathcal{R}_4(\mathcal{C}'') = \{\boldsymbol{o}, \boldsymbol{p}, a\}$. It follows that, starting from $\mathcal{C}^*$ or $\mathcal{C}''$, the resistances of transitions (T1) and (T2) are the same,

$$w_1^a [\gamma_1 - \alpha_1] - w_1^o [\beta_1 - \alpha_1] + w_4^b [\gamma_4 - \alpha_4] - w_4^p [\beta_4 - \alpha_4].$$

As the expression above does not depend on the average arrival intensities of interest messages at ISPs 2 and 3, there exist w_1^o and w_4^p such that the two paths described above are the

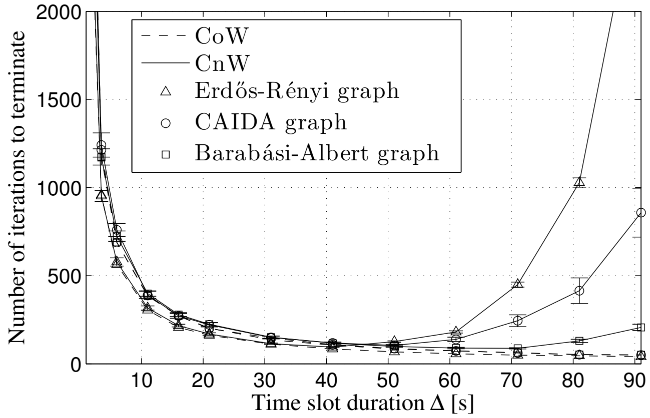

Fig. 5. Average number of iterations needed to reach an equilibrium allocation as a function of the time slot duration Δ for three different peering graphs and algorithms CoW and CnW. Results for $\beta_i = 1.5$.

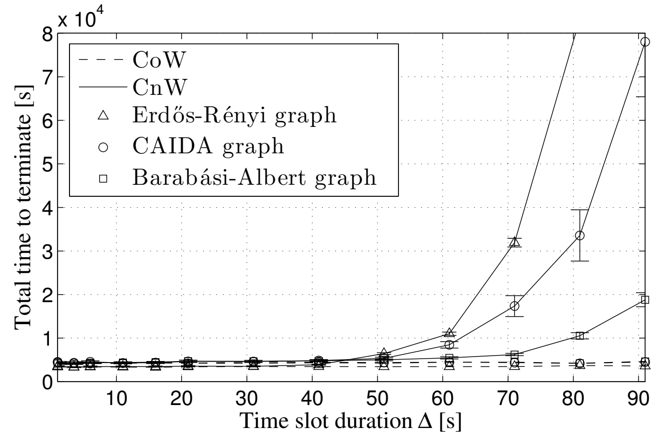

Fig. 6. Average time needed to terminate as a function of the time slot duration Δ for three different peering graphs and algorithms CoW and CnW. Results for $\beta_i = 1.5$.

paths with least resistance from $\mathcal{C}^*$ to $\mathcal{C}''$ and from $\mathcal{C}''$ to $\mathcal{C}^*$, respectively. It follows that the allocations $\mathcal{C}^*$ and $\mathcal{C}''$ have the same stochastic potential, that is $r(\mathcal{C}^*, \mathcal{C}'') = r(\mathcal{C}'', \mathcal{C}^*)$. This proves the proposition. $\square$

It follows from Proposition 9 that if peering is not free, two or more cache allocations might be almost equally likely to emerge, even when the ISPs have disjoint interest.

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 5 and 6 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 vertexes 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 $\alpha_i = 1$, $\beta_i = 1.5$, $\gamma_i = 10$ and cache capacity $K_i = 10$ at every ISP.
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 100 simulations, and the error bars show the 95% confidence intervals. We omit the confidence intervals when they are within 5% of the averages.

Figure 5 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 6, 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 5 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 ER random graph. From Figure 6 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 7 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.

An important question is how the number of simultaneous updates that occur under CnW is affected by the peering cost β_i. Figure 8 shows the number of iterations needed to

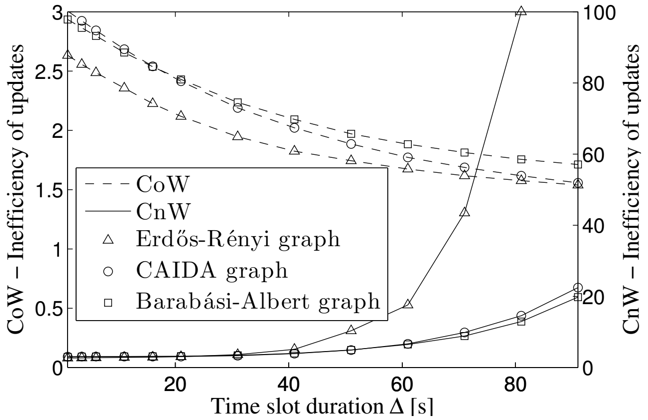

Fig. 7. Average inefficiency as a function of the time slot duration Δ for three different peering graphs and algorithms CoW and CnW. Results for $\beta_i = 1.5$.

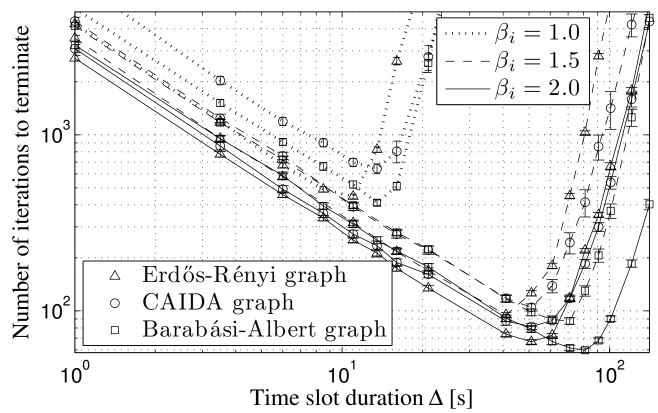

Fig. 8. Average number of iterations needed to reach an equilibrium allocation as a function of the time slot duration Δ for algorithm CnW. Results for three different peering graphs and peering costs.

reach an equilibrium allocation under the CnW algorithm as a function of the time slot duration Δ. In the figure we show the results for peering costs $\beta_i \in \{1.0, 1.5, 2.0\}$. The results for $\beta_i = 1.5$ are the same as shown in Figure 5. The figure shows that the number of iterations decays as a power of the time slot duration, as long as convergence is not disturbed by simultaneous updates. At the same time, once simultaneous updates become frequent, increasing the time slot duration leads to a fast increase of the number of iterations. In the *free peering* case, i.e. $\beta_i = 1 (= \alpha_i)$, the number of iterations needed to reach an equilibrium allocation increases for significantly smaller time slot durations compared to the $\beta_i = 1.5$ and $\beta_i = 2.0$ cases. This happens because any item $o \in \mathcal{C}_i$ such that $o \in \mathcal{C}_j$ and $j \in \mathcal{N}(i)$ can potentially be evicted by both ISPs i and j, i.e. $\Psi_i^o(\mathcal{C}_{-i}) = \Psi_j^o(\mathcal{C}_{-j}) = 0$. This is not the case when $\alpha_i < \beta_i$, as higher peering costs correspond to higher cost savings for the items that are available at peering ISPs. Therefore, higher peering costs cause a lower number of simultaneous updates for equal time slot duration.

These results show that although CnW would be more appealing as it allows ISPs to update their cache contents all the time, CoW might be necessary when the information exchange happens infrequently or when the unit cost of peering traffic is close to the unit cost of retrieving an item from a local cache.

B. Imperfect Information

In the following we show results for the case when the estimation of the items' arrival intensities is imperfect under *free peering*. 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 7 for the case of $K_i = 1$, hence we consider that the item with the highest arrival intensity is different at every ISP.

Figure 9 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 7. 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 8, 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 10 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 [16]. Most analytical work on the performance of cache eviction policies for stand-alone caches focused on the LRU policy [23], [24], [14]. 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 [14]. 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], [13], [6]. An iterative algorithm

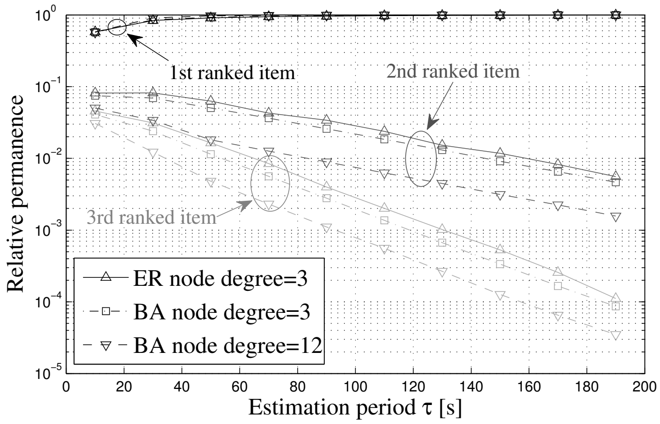

Fig. 9. Relative permanence of the three items with highest arrival intensity in the ISPs' caches, as a function of the intensity estimation interval τ, for three different random peering graphs.

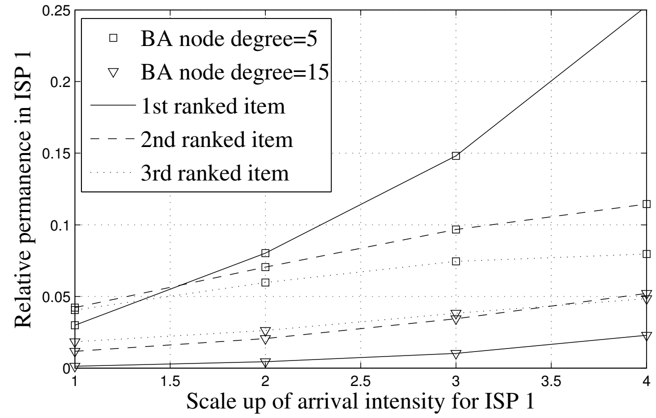

Fig. 10. Relative permanence of the three items with highest arrival intensity in ISP 1's cache, as a function of the scaling factor at ISP 1, for two Barabási-Albert peering graphs with different average node degree.

to approximate the cache miss rate in a network of caches was proposed in [13]. The authors in [4] considered various network topology-aware policies to improve the overall cache hit rate in a network of caches. In [6], [7] the authors use probabilistic caching to increase the cache hit rate and the fairness among content flows in a network of caches. The authors in [29] analyzed the impact of the initial system state on the selection of the system's steady-state. These works consider that the caches route requests irrespective of the associated traffic costs. [30] shows that considering the traffic costs of a network operator leads to cache allocations that are suboptimal in terms of hit rate. Common to the aforementioned works is the assumption of 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 [31], [32]. The authors in [31] considered a centralized algorithm for content placement, while distributed algorithms were analyzed in [32]. A game theoretical analysis of distributed content replication was provided in [33], [34], [18], [35]. The authors in [33] showed that the Price of Anarchy on a complete graph is unbounded and it depends on the unit costs to retrieve content items. [34] showed that, the social optimum may not be a Nash equilibrium even on a complete graph topology. In [18], [35] the authors investigated the existence of Nash equilibria and the convergence properties for the case of replication on an arbitrary graph topology, while [36] showed that a Nash equilibrium may not exist for certain graph topologies and access costs, and proposed a distributed algorithm for finding ex-post individually rational content allocations. 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.

Closest to this work is [37], which considered a network of selfish caches, including the effects of evictions, and provided a game-theoretical analysis of the resulting cache allocations. This paper extends the results in [37] by relaxing the assumption of free content-peering and considering the more general case when retreiving content from within the ISP is not more costly than retrieving it from peering ISPs.

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 how the level of coordination influences the ability of peering ASs 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. If fast convergence to a stable allocation is important too then coordination is needed to avoid simultaneous cache evictions by peering ISPs. Furthermore, we gave insight into the structure of the most likely cache allocations for the case when the content popularity estimates are inaccurate. We showed that, in the general case, various cache allocations may be almost equally likely to emerge. However, if peering traffic is free, content-peering is likely to lead to the cache allocation that is most efficient.

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/TG 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] ——, "In-Network Cache Management and Resource Allocation for Information-Centric Networks," *IEEE Trans. Parallel Distrib. Syst.*, vol. 25, no. 11, pp. 2920 – 2931, 2014.

[8] 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.

[9] 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.

[10] 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.

[11] R. Sami, M. Schapira, and A. Zohar, "Searching for Stability in Interdomain Routing," in *Proc. of IEEE INFOCOM*, 2009, pp. 549–557.

[12] 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.

[13] E. J. Rosensweig, J. Kurose, and D. Towsley, "Approximate Models for General Cache Networks," in *Proc. of IEEE INFOCOM*, 2010, pp. 1–9.

[14] C. Fricker, P. Robert, and J. Roberts, "A versatile and accurate approximation for LRU cache performance," in *Proc. of the 24th International Teletraffic Congress (ITC)*, 2012, pp. 1–8.

[15] E. G. Coffman Jr. and P. J. Denning, *Operating Systems Theory*. Prentice Hall, 1973.

[16] N. Megiddo and D. Modha, "ARC: A Self-Tuning, Low Overhead Replacement Cache," in *Proc. of USENIX File & Storage Technologies Conference (FAST)*, 2003, pp. 115 – 130.

[17] 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.

[18] V. Pacifici and G. Dán, "Convergence in Player-Specific Graphical Resource Allocation Games," *IEEE J. Sel. Areas Commun.*, vol. 30, no. 11, pp. 2190–2199, 2012.

[19] H. P. Young, "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," *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*, 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] E. J. Rosensweig, D. S. Menasche, and J. Kurose, "On the Steady-State of Cache Networks," in *Proc. of IEEE INFOCOM*, 2013, pp. 887–895.

[30] A. Araldo, M. Mangili, F. Martignon, and D. Rossi, "Cost-aware caching: optimizing cache provisioning and object placement in ICN," in *Proc. of IEEE GLOBECOM*, 2014, pp. 1108 – 1113.

[31] M. R. 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.

[32] S. Borst, V. Gupta, and A. Walid, "Distributed Caching Algorithms for Content Distribution Networks," in *Proc. of IEEE INFOCOM*, 2010, pp. 1478–1486.

[33] G. Pollatos, O. Telelis, and V. Zissimopoulos, "On the social cost of distributed selfish content replication," in *Proc. of IFIP Networking*, 2008, pp. 195–206.

[34] E. Jaho, M. Karaliopoulos, and I. Stavrakakis, "Social similarity favors cooperation: the distributed content replication case," *IEEE Trans. Parallel Distrib. Syst.*, vol. 24, no. 3, pp. 601–613, 2013.

[35] V. Pacifici and G. Dán, "Selfish Content Replication on Graphs," in *Proc. of the 23rd International Teletraffic Congress*, 2011, pp. 119–126.

[36] ——, "Distributed Algorithms for Content Allocation in Interconnected Content Distribution Networks," in *Proc. of IEEE INFOCOM*, 2015, pp. 2362–2370.

[37] ——, "Content-peering Dynamics of Autonomous Caches in a Content-centric Network," in *Proc. of IEEE INFOCOM*, 2013, pp. 1079 – 1087.