Optimal Cache Budget Distribution for Hierarchical ICN Networks

Alireza Montazeri
Department of Computer Science
University of Saskatchewan
Saskatoon, Canada
Email: alm164@mail.usask.ca

Dwight Makaroff
Department of Computer Science
University of Saskatchewan
Saskatoon, Canada
Email:makaroff@cs.usask.ca

***Abstract*—Caching facilities can be deployed by all or some of the ICN nodes on the path of delivering data items from a content source to users. However, some inconsistent conclusions have been made from different studies regarding the benefits of in-network caching. To investigate the benefits of in-network caching, we propose an analytical model that optimally distributes a total cache budget among the nodes of a given ICN network in an environment that does not follow the Independent Reference Model (IRM).**

The cache budget distribution problem is studied with respect to optimizing system-centric and user-centric metrics, using a small number of synthetic and realistic topologies as case studies. Our findings reveal the benefits of in-network caching as well as the optimal distribution of the cache budget in ICNs with respect to our selected objective function. Although the efficiency of in-network caching on user-centric metrics strongly depends on topologies and the strength of temporal locality, in-network caching is very helpful in optimizing the ISP-centric metrics for all network and traffic settings.

I. Introduction

Hierarchical caches have attracted renewed interest since they are fundamental to the efficient operation of Information Centric Networks (ICNs). By storing data items close to users, the cost of retrieval is reduced (server and network bandwidth from the ISP's point of view). Caching data items close to users also reduces latency resulting in better quality of experience (user's point of view). The performance of such in-network caching depends in part on cache replacement and replication/placement algorithms.

Cache replacement algorithms (e.g. LRU, k-LRU [1] and hybrid caching [2]) choose what data items to evict from the cache when the cache is full. Cache replication/placement algorithms, (e.g. LCE and LCD) on the other hand, determine which nodes should cache a copy of a given data item.

In ICNs, a request for data item i from a user is forwarded to the source hosting i. If ICN node l on this path has a copy of i, l makes a copy of i and forwards that to the user. Upon the arrival of i at intermediate ICN node k on the path from l to the user, k may cache the data item. ICN node k can decide whether to cache the item or not. In Leave Copy Everywhere (LCE) replication mechanism for example, node k always stores data items, but results in many redundant copies of a data item in the network.

On the other hand, ICN nodes may collaborate with the other nodes on the delivery path to replicate items to optimize retrieval distance. Leave Copy Down (LCD) [3], Move Copy Down (MCD) [4], ProbCache [5] and age-based replication/placement algorithms [6] are examples of this concept. In LCD, for example, a request for item i hits at the cache of ICN node k. A copy of data item i on the delivery path from k is only cached at the node that follows k on this path. More requests for i place copies at ICN nodes closer to users.

Caching facilities can be deployed by all [7] or some [8] of the nodes on the delivery path. Some studies have investigated the efficiency of ICN caching, both empirically [9], [10] and analytically [5], which have inconsistent results. For example, Danzig *et al.* [11] and Rossini *et al.* [12] believe that in-network caching can be more effective than only caching at the edge of the network. Fayazbakhsh *et al.* [9] and Psaras *et al.* [13], on the other hand, believe caching closer to the network edge brings more benefits compared to the benefits brought by deploying cache at intermediate nodes. Furthermore, Chai *et al.* show that selecting only some of the ICN nodes on the delivery path increases profit further [8]. This suggests that the ICN literature still lacks an empirical and analytical deep understanding of the benefits brought by in-network caching.

We propose an analytical model that optimally distributes a total cache budget of C among the nodes of ICN networks under non-IRM environment. The cache budget distribution is studied regarding optimizing the following metrics: 1) system-centric metrics (e.g. total miss ratio representing server load), user-centric metrics (e.g. average hop distance representing latency), and 3) a combination of the above.

The rest of this paper is organized as follows: Related work is covered in Section II. A mathematical expression for the optimal cache distribution is given in Section III. Section IV investigates the optimal cache distribution among the nodes of an ICN for LRU cache replacement and LCE cache replication algorithms for various topologies and network metrics. Finally, Section V concludes this paper.

II. Related Work

A. Modelling Caching Algorithms Under IRM

One distinguishing component of ICN nodes is in-network caching. These connected ICN nodes then construct a hierar-

chy of caches for distribution/storage of content items from particular publishing sources. Performance evaluation in a large hierarchy of caches through simulations is extremely costly. Garetto *et al.*, for example, found that investigating the caching performance in an ICN with 1365 nodes through simulations needs substantial memory, high CPU usage and long time to enter into steady state [1]. Modelling is much less computationally expensive and does not depend on the number of requests in the simulation.

The most accurate approximation to calculate the hit probability of data item i in a LRU cache under IRM was originally proposed by Fagin [14] and further explored by Che *et al.* [15]. With the notations in Table I, they define $\tau_{k,i}$ as the *characteristic time approximation* of item i at cache k; the time before C_k distinct data items (not including data item i) are requested at node k. They also assume $\tau_{k,i}$ is independent of i; this property is confirmed by Fricker *et al.* [16] with a Zipf popularity distribution.

Having τ_k as $\tau_{k,i}$ independent of i, data item i is in cache k at time t if and only if less than τ_k has elapsed since the last request for item i at node k. Under a Poisson arrival process assumption, the time-average probability $P^{in}_{k,i}$ that data item i with arrival rate $\lambda_{k,i}$ is in cache k is given by $P^{in}_{k,i}(\tau_k) = 1 - e^{-\lambda_{k,i}\tau_k}$. Assuming $\sum_{j=1}^{N} P^{in}_{k,j}(\tau_k) = C_k$, Che *et al.* compute the corresponding τ_k. As an immediate consequence of the Poisson Arrivals See Time Averages (PASTA) property for Poisson arrivals, $P^{in}_{k,i}$ also represents the hit probability $P^{hit}_{k,i}$ [1]. Since τ_k is a function of C_k, $P^{in}_{k,i}$ and $P^{hit}_{k,i}$ are then shown by $P^{in}_{k,i}(C_k)$ and $P^{hit}_{k,i}(C_k)$ respectively.

B. Modelling Caching Algorithms Under non-IRM

The IRM ignores temporal and geographical localities in request sequences. Garetto *et al.* consider a 2-stage hyper-exponential to apply temporal locality to users' requests [1], [17]. They assume the intensities of renewal processes are modulated by a Zipf distribution. Then, they define the rates of exponential stages as $\lambda^1_{k,i} = \lambda_{k,i}z$ and $\lambda^2_{k,i} = \lambda_{k,i}z^{-1}$, in which parameter z applies a temporal locality into requests arrival process. The CDF of inter-request times of data item i at node k is then calculated as $F_{k,i}(t) = 1 - \gamma e^{-\lambda^1_{k,i}t} - (1-\gamma)e^{-\lambda^2_{k,i}t}$, in which $\gamma = z/(z+1)$. Thus, the average request rate $\lambda_{k,i}$ is then given by $\lambda_{k,i} = 1/\big(\int_0^\infty \big(1 - F_{k,i}(t)\big)dt\big)$.

They extend Che *et al.*'s approximation [15] to model LRU renewal traffic. They argue that under a general request process $P^{in}_{k,i}(C_k)$ and $P^{hit}_{k,i}(C_k)$ are not equal since PASTA no longer holds. Consequently, to compute $P^{in}_{k,i}(C_k)$, they consider that data item i is in cache k at time t if and only if the last request for i arrived at node k in $[t-\tau_k, t)$. As a result, $P^{in}_{k,i}(C_k) = \widehat{F}_{k,i}(\tau_k)$, in which $\widehat{F}_{k,i}(t) = \lambda_{k,i}\int_0^t (1 - F_{k,i}(\theta))d\theta$. On the other hand, when computing $P^{hit}_{k,i}(C_k)$, Garetto *et al.* condition on the fact that a request arrives at time t. Thus, the probability that the previous request for i arrived to cache k in $[t-\tau_k, t)$ is equal to the probability that the last inter-request time is not larger than τ_k. Therefore, $P^{hit}_{k,i}(C_k) = F_{k,i}(\tau_k)$.

In addition to temporal locality, data items may have geographically differential popularity. Large-scale systems must satisfy smaller heterogeneous user communities having different interests. Studies on the local request frequency distribution show power-law properties [18], [19]. In addition, other studies conclude that the global frequency of data item requests also have power-law distributions [20]. In particular, Zink *et al.* observe weak correlation between global and local request frequency [21].

C. Modelling a Cache Hierarchy

We then need to model the arrival rate of users' requests at the intermediate ICN nodes. The models proposed by Rosensweig *et al.* [22], Carofiglio *et al.* [23] and Dabirmoghaddam *et al.* [24] rely on the independence assumption among caches, assuming that requests arriving at each cache satisfy the IRM assumptions. For intermediate ICN node k, assuming $\lambda^e_{k,i}$ is the exogenous request rate of item i at k and R_k is the set of all k's neighbouring ICN nodes from which k may receive a request for i, Rosensweig *et al.* find $\lambda_{k,i} = \lambda^e_{k,i} + \sum_{u\in R_k}\lambda'_{u,i}$ and $\lambda'_{u,i} = \lambda_{u,i}\big(1 - P^{hit}_{u,i}(C_u)\big)$ [22]. Rosensweig *et al.* identified the main potential sources of prediction error that appears between the simulation results and their model: the violation of the IRM (or Poisson) assumption on the miss streams of LRU caches at intermediate ICN nodes [22]. Despite this prediction error, we use Rosensweig's model for the request arrival process at intermediate caches due to its simplicity.

TABLE I: Notations

N	number of data items
$p_{k,i}$	item i's popularity at ICN node k
$\lambda_{k,i}$	arrival rate of data item i at ICN node k
$\lambda'_{k,i}$	miss rate of data item i at ICN node k
λ_k	arrival rate at ICN node k
λ'_k	the overall miss rate of data items at ICN node k
$d_{k,i}$	average distance to retrieve data i from edge node k
τ_k	characteristic time approximation at ICN node k
z	temporal locality parameter
k^i	i'th immediate parent of k, e.g. k^0 and k^1 are node k itself and k's immediate parent respectively
C_k	size of cache allocated to ICN node k
L_k	set of all children of ICN node k
R	set of all ICN nodes
E	set of all ICN edge nodes
C	the overall cache budge; $C = \sum_{\forall k\in R} C_k$
D	the depth of a overlay ICN tree

III. Model and Assumptions

In the rest of this paper, we assume the following:

1) There is one origin server/producer s for all data items,
2) ICN constructs an overlay tree consisting of all ICN nodes, rooted at the source node, for the content delivery,
3) Users are only connected to edge ICN nodes, and intermediate ICN nodes receive only endogenous traffic (i.e. $\lambda_{k,i} = \sum_{u\in L_k}\lambda'_{u,i} \quad \forall k \in R - E$),
4) Garetto's model [1], [17] is used to apply temporal locality ($\lambda_{k,i} = 1/\big(\int_0^\infty \big(1 - F_{k,i}(t)\big)dt\big) \quad \forall k \in E$),

5) Users' requests have geographical locality [25],
6) LRU and LCE are the cache replacement and placement algorithms, respectively.

The goal is the optimal cache budget distribution among the caches of the network considering the following metrics:

- Distance/Latency: data items should be cached as close as possible to the user. Distance is measured based on the number of hops, which is strongly correlated with network latency. A full study of network protocol issues and bandwidth is beyond the scope of this work.
- Miss ratio: A lower overall miss ratio in an ICN network imposes a smaller load on the source node.

Having these two metrics, we optimize the following:

$$\underset{C_k \forall k \in R}{\text{minimize}} \begin{cases} \sum_{\forall k \in E} d_k(C_k) \\ \frac{1}{\sum_{\forall k \in R} \lambda_k} \sum_{\forall k \in R} \lambda'_k(C_k) \end{cases} \tag{1}$$

$$\text{subject to} \begin{cases} \sum_{\forall k \in R} C_k = C \\ C_k \geq 0 \quad \forall k, \end{cases} \tag{2}$$

in which $\lambda'_k(C_k)$ is given by

$$\lambda'_k(C_k) = \sum_{i=1}^{N} \lambda_{k,i} \big(1 - P^{hit}_{k,i}(C_k)\big) \quad \forall k, \tag{3}$$

and $d_k(C_k)$ is the average distance for retrieving data items from node k, given by

$$d_k(C_k) = \sum_{i=1}^{N} \lambda_{k,i} d_{k,i}(C_k) \quad \forall k, \tag{4}$$

where $d_{k,i}(C_k)$ as the average distance to retrieve data i from edge node k is calculated as

$$d_{k,i}(C_k) = \sum_{l=0}^{k^l \neq s} \prod_{j=0}^{l} \big(1 - P^{hit}_{k^j,i}(C_{k^j})\big). \tag{5}$$

Now, the two objective functions are combined as follows:

$$\underset{C_k \forall k \in R}{\text{minimize}} \Big\{ (1-w) \frac{\sum_{\forall k \in E} d_k(C_k)}{D} + w \frac{\sum_{\forall k \in R} \lambda'_k(C_k)}{\sum_{\forall k \in R} \lambda_k} \Big\}, \tag{6}$$

in which D, the depth of hierarchical ICN tree, normalizes the first objective and w is the weighting coefficient. Note that (6) is a non-linear optimization problem since $d_k(C_k)$ and $\lambda'_k(C_k)$ are both exponential functions of C_k.

IV. Numeric Results

In this section, we report on results of allocating cache according to 2 policies: Network Overall Caching (NOC) and Edge-Only Caching (EOC). In the former, the cache budge is distributed optimally among all nodes in the ICN tree. In the latter, however, the cache budget is optimally distributed only among the edge nodes of the ICN tree. The following two metrics are considered to compare NOC against EOC:

- Relative benefit of NOC over EOC in terms of load on the server: assuming h_{EOC} and h_{NOC} as the average load on the server in EOC and NOC respectively, this metric would be $\frac{h_{EOC}-h_{NOC}}{h_{EOC}}$. Server load is calculated as $\lambda_s = \sum_{\forall k \in L_s} \lambda'_k(C_k)$.
- Relative benefit of NOC over EOC in terms of distance: assuming d_{EOC} and d_{NOC} as the average distance in EOC and NOC respectively, this metric would be $\frac{d_{EOC}-d_{NOC}}{d_{EOC}}$. The average distance to access data items is calculated by the upper equation in (1).

In previous work [25], we proposed and developed an algorithm to generate requests with geographical locality that has Zipf properties in each region and combines to form a Zipf distribution in the global region. The cache budget distribution optimization is solved for the following three scenarios:

- geographical locality and strong temporal locality: to apply the temporal locality, $z = 10$ is chosen for the second order hyper-exponential process.
- geographical locality and weak temporal locality: to apply the temporal locality, $z = 2$ is chosen for the second order hyper-exponential process.
- geographical locality and no temporal locality: having $z = 1$ for the second order hyper-exponential process implies no temporal locality.

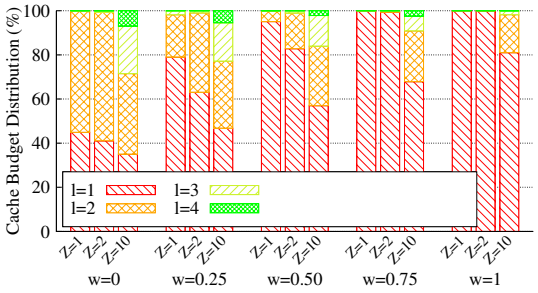
The request probability for item i at edge node k ($p_{k,i}$), follows a Zipf distribution with α and $\lambda_{k,i} = \lambda_k p_{k,i}$. We solve the optimization problem (6) with the constraints in (2) in Matlab for $N = 10000$, the global request rate $\lambda = \sum_{\forall k \in E} \lambda_k = 4$, and various values of C, α and topologies. The optimal solution for each case is a set of C_k for all ICN nodes ($\forall k \in R$). The optimal C_k are then used to calculate h_{EOC}, h_{NOC}, d_{EOC} and d_{NOC}.

The analytical findings for realistic topologies in Section IV-B, are validated by simulations using ccnSim [26]. The optimal C_k values obtained are used as the cache size of ICN nodes in simulations. Then, based on the simulation output (e.g. average distance/server load), the relative benefit of NOC over EOC is computed. The simulation results are the average of five simulations runs.

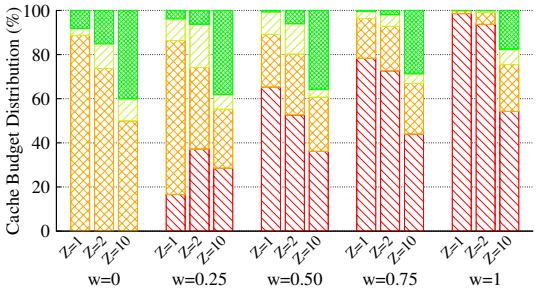
A. Tree Topologies

In this section, the optimal cache budget distribution for a ternary tree with depth of 4 and different values of Zipf parameter α and C is solved: $l = 1$ shows the root of the tree and largest value of l corresponds to the level of edge nodes. Figure 1 shows the cache distribution among different levels on this tree for different values of α. For $\alpha = 0.8$ (Figure 1a), as w moves from 0 to 1, (more weight is given to miss ratio), a large fraction of cache budget is distributed among high level intermediate nodes. For example, 45% of the cache budget is allocated to the root of the tree for $z = 1$ and $w = 0$. The fraction of cache budget allocated to the root however, increases to 100% for $z = 1$ and $w = 1$.

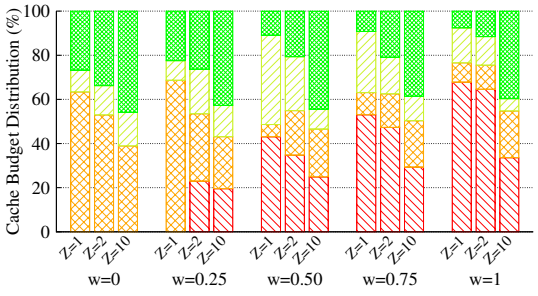
The other point from Figure 1 is that, as stronger temporal locality is applied ($z = 10$), allocated cache budget among lower level caches increases. In case of $\alpha = 0.8$ for instance, 95% and 5% of the cache budget is allocated to first level and second levels of the tree respectively for $w = 0.5$ and $z = 1$.



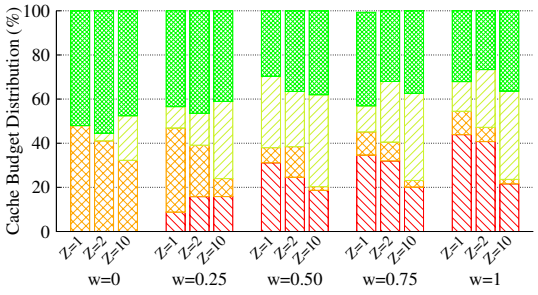
(a) $\alpha = 0.8$



(b) $\alpha = 1.0$



(c) $\alpha = 1.2$



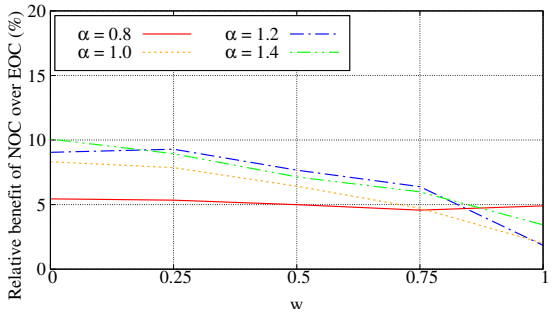
(d) $\alpha = 1.4$

Fig. 1: Optimal Cache Budget Distribution, $C = 1000$

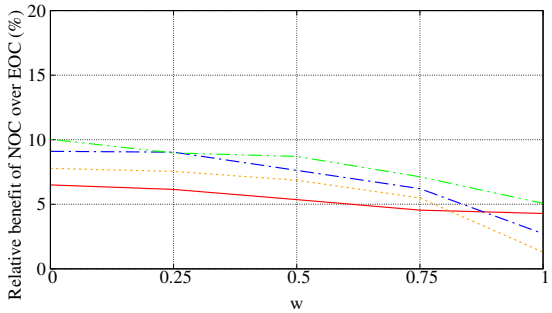
The cache distribution for the levels changes to 57%, 27%, 14% and 2% for $w = 0.5$ and $z = 10$. With strong temporal locality, caching items at the lower level ICN nodes results in shorter latency as well as smaller load on the server. This trend for w and z is also seen for other values of α.

Comparing Figures 1a, 1b, 1c and 1d with each other illustrates the effect of α on the optimal distribution of the cache budget. As α increases, a larger fraction of the cache budget is scattered among lower level caches. When $w = 0.50$ and $z = 1$, no cache is allocated to the two lowest levels of the tree for $\alpha = 0.8$. The third and fourth levels however, consume more than 60% of the cache budget together for $\alpha = 1.4$.

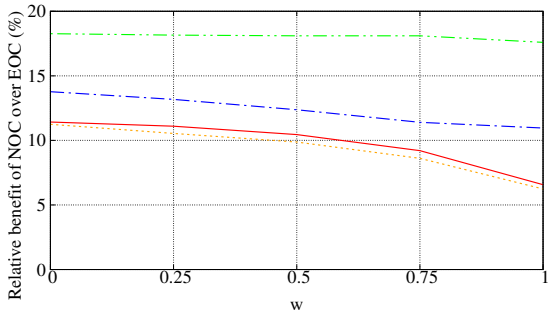
Figure 2 represents the relative benefit of optimal in-network caching over edge caching for average latency. Figure 2a shows that the in-network caching optimization for $z = 1$ and $w = 0$ decreases average distance by 5.5%, 8.3%, 9% and 10% when α equals to 0.8, 1, 1.2 and 1.4. This benefit rises as temporal locality gets stronger; for $z = 10$ and $w = 0$, in-network caching optimization decreases the average distance by 11.4%, 11.3%, 13.7% and 18.3% when α equals to 0.8, 1.0, 1.2 and 1.4, respectively. Figure 2 also shows that the benefit of in-network caching on distance decreases as w increases. For $w = 1.0$, a high fraction of the cache budget is allocated to the ICN nodes at higher level. However, the performance of in-network caching does not get worse than edge-only caching as the relative benefit of in-network caching over edge-only caching is positive for all values of z and α.



(a) $z = 1$



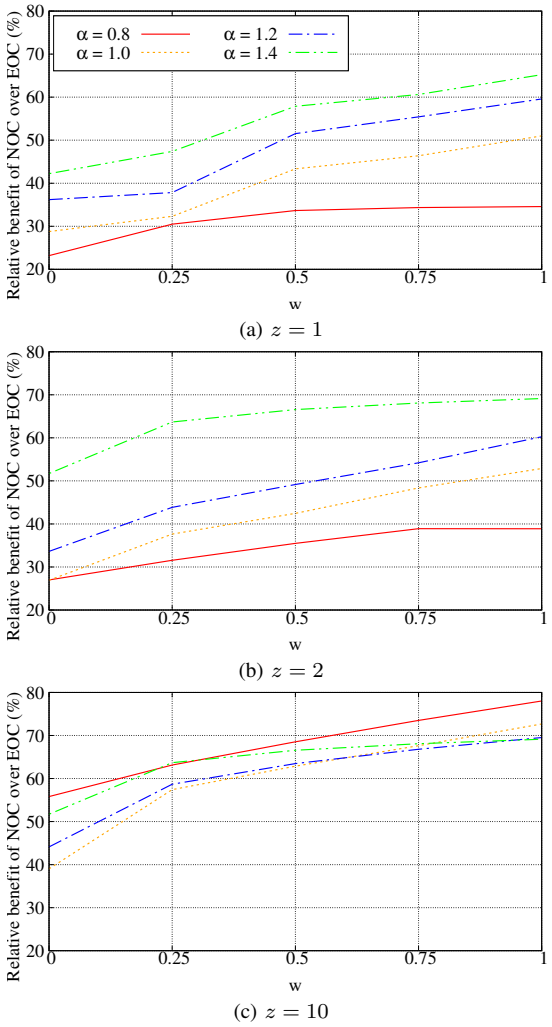
(b) $z = 2$



(c) $z = 10$

Fig. 2: NOC vs. EOC, hops, $C = 1000$

Figure 3 depicts the influence of cache distribution optimization on server load. Lower miss ratios reduce server load.



(a) $z = 1$

(b) $z = 2$

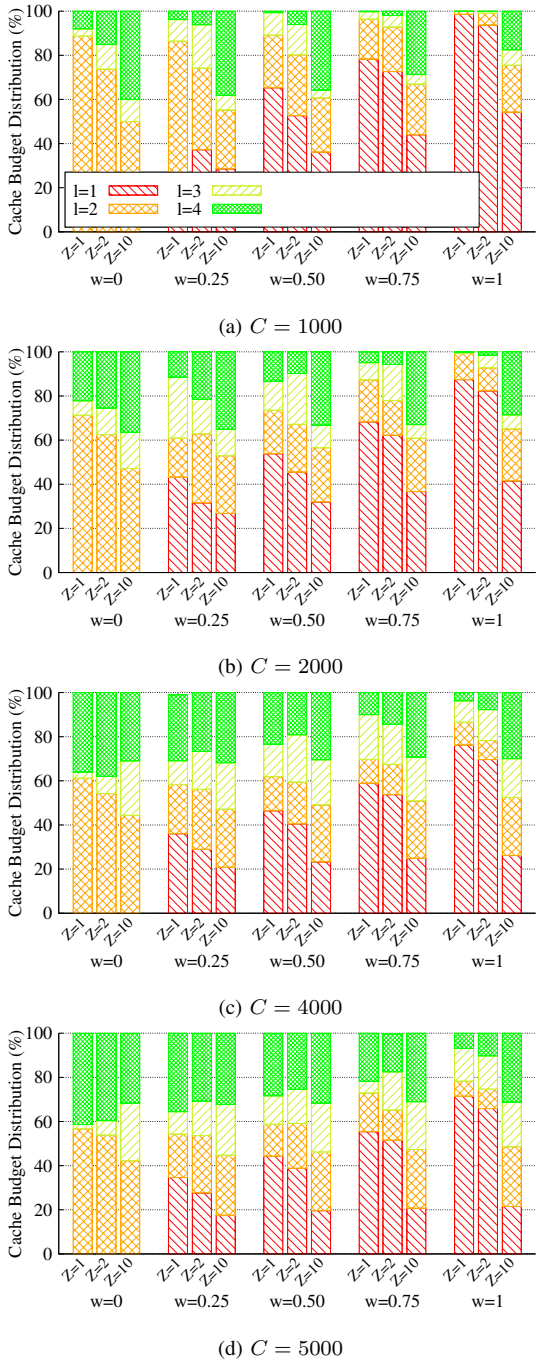
(c) $z = 10$

Fig. 3: NOC vs. EOC, average server load, $C = 1000$

The figure shows that the optimal in-network caching in the tree provides at least 23.2% less load on the server ($z = 1$, $\alpha = 0.8$ and $w = 0$). This benefit goes up close to 80% for strong temporal locality of $z = 10$ and $\alpha = 0.8$, $w = 1.0$.

Having $\alpha = 1.0$ and an equal weight $w = 0.5$, the optimal in-network caching decreases the distance by 6.4%, 6.9% and 9.9% for $z = 1$, $z = 2$ and $z = 10$ respectively. The optimal in-network caching however, decreases the average load on the server by 43%, 42% and 62.9% for $z = 1$, $z = 2$ and $z = 10$. While in-network caching may have a small influence on the average distance (up to 9.9%), network caching can be very effective on minimizing the load on the server (at least 42%).

Figure 4 represents the influence of C on optimal in-network caching. Similar to Figure 1, a larger value of w allocates a larger proportion of C to caches at intermediate levels; in addition, stronger temporal locality requires more cache budget at lower levels of the tree. In Figure 4, optimal in-network caching distributes a larger fraction of C among lower levels as C increases. When $z = 2$ and $w = 1$, more than 90% of the cache budget is allocated to the first level of the tree when $C = 1000$. The fraction of cache budget allocated to the first level decreases to less than 70% when $C = 5000$.



(a) $C = 1000$

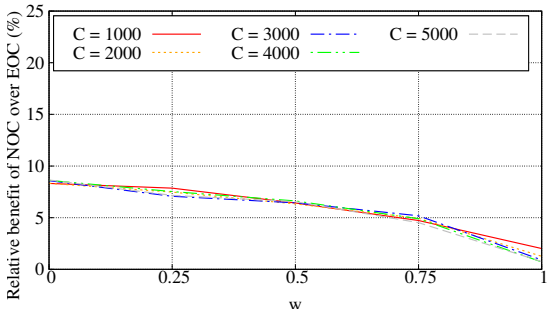
(b) $C = 2000$

(c) $C = 4000$

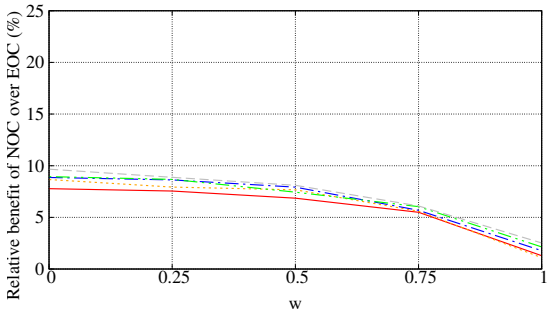
(d) $C = 5000$

Fig. 4: Optimal Cache Budget Distribution, $\alpha = 1.0$

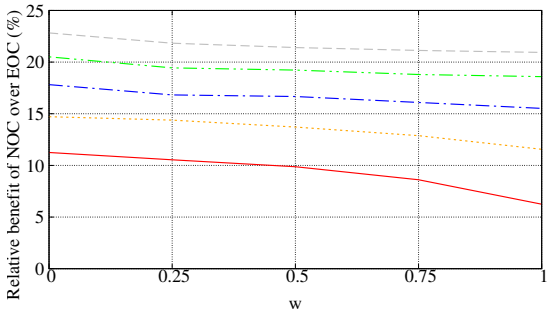
Figure 5 shows the benefits of optimal in-network caching on average distance for different values of z. Strong temporal locality causes a large difference in relative benefit of in-network caching for different cache budgets. Having $w = 0.5$ and $z = 10$, in-network caching reduces distance by 22% when $C = 5000$ compared to when $C = 1000$ (10%). This difference becomes negligible when $z = 1$. A similar behaviour is also seen for overall server load in Figure 6.



(a) $z = 1$

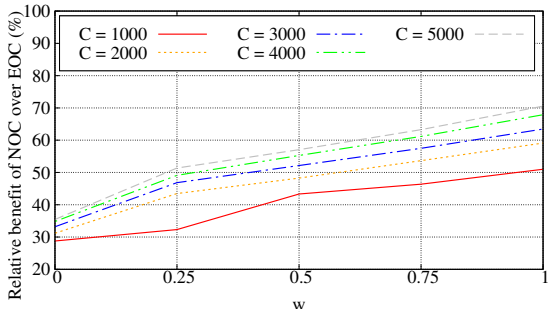


(b) $z = 2$

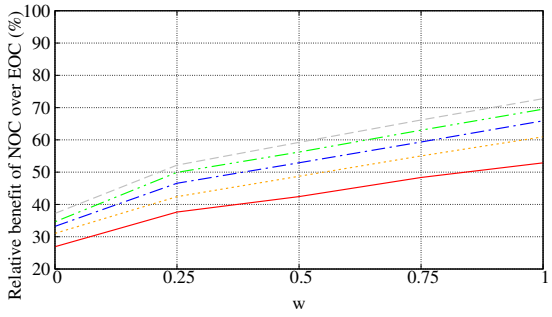


(c) $z = 10$

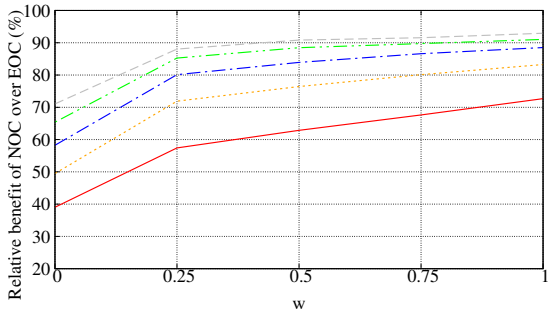
Fig. 5: NOC vs. EOC, average number of hops, $\alpha = 1.0$



(a) $z = 1$



(b) $z = 2$



(c) $z = 10$

Fig. 6: NOC vs. EOC, average server load, $\alpha = 1.0$

B. Realistic Topologies

Table II summarizes a number of realistic topologies used in previous works (e.g. [27]). They vary in depth and intermediate node degree.

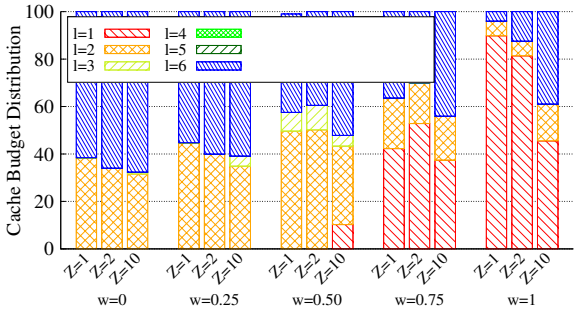
TABLE II: Specification of topologies.

	specifications				
name	*inter-nodes*	*edge-nodes*	*depth*	*max degree*	*average degree*
Level3	5	41	5	29	9.00
Dtelecom	7	61	4	52	9.57
Tiger	12	10	5	4	1.75
Geant	12	10	6	4	1.75

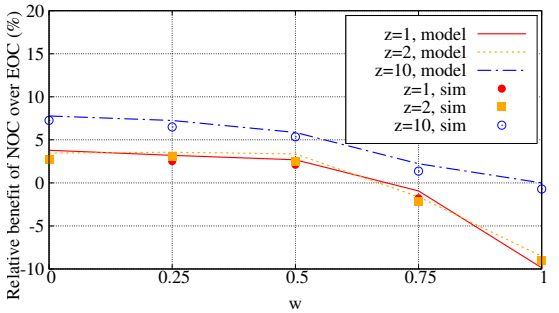
Figure 7a shows that optimal in-network caching in the Geant topology may not allocate any cache to some intermediate nodes, consistent with Chai *et al.* [8]. Minimizing the distance ($w = 0$) for example, allocates more than 60% of C to the edge nodes of the topology; no cache would be allocated to any other levels, except level 2 that takes the remainder of the cache budget. Similar to the synthetic topology, a larger fraction of C is allocated to higher levels as w gets closer to 1; however, no cache budget is distributed to levels 3, 4 and 5 for all temporal locality values when $w = 1$. Caching at some levels of Geant topology brings no performance benefit.

Figure 7b depicts the influence of w on superiority of in-network caching over edge-only caching for average distance. The cache sizes obtained by (6) are used in the ccnSim simulations and the relative benefits match very closely with the analytical predictions. The maximum benefit brought by in-network caching is less than 10% when $w = 0$. In-network caching fails to be effective on minimizing the distance as w gets greater than 0.75. Figure 7c on the other hand illustrates the influence of w on superiority of in-network caching over edge-only caching for overall load on the server. This figure shows that the in-network caching provides the network with at least 15% less load on the server when $w = 0$. This goes up to at least 40% when $w = 1$. Choosing w here depends on how relevant the reduced latency is perceived by users. For instance, if less than 10% shorter distance has no influence of users' experience, $w = 0.75$ can be selected to guarantee at least 30% less server load.

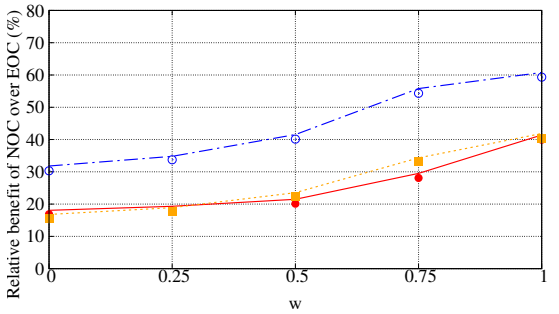
As in Section IV-A, an increase in w allocates more cache



(a) Cache distribution



(b) Average distance



(c) Average load on server

Fig. 7: Geant cache distribution, $\alpha = 1.0, C = 1000$

to higher levels of the distribution tree and lower level nodes obtain more cache with stronger temporal locality. Similarly, some tree levels receive no cache allocation. Figure 8 depicts a similar distribution for Tiger to that for Geant. The results for distance/server load are not shown, due to space limitations.

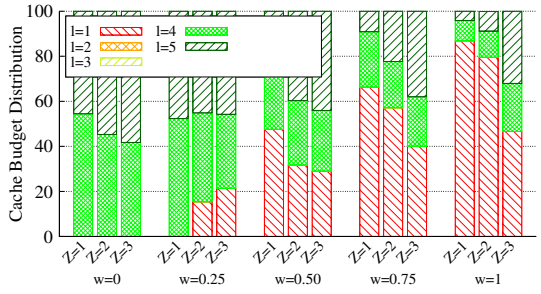
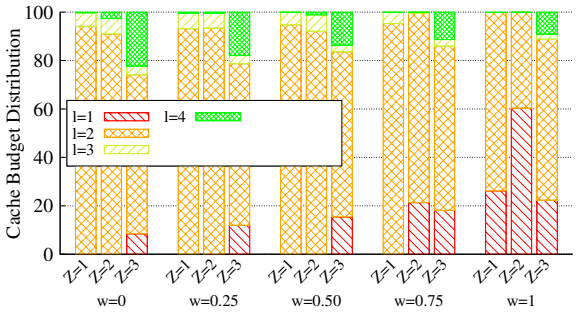
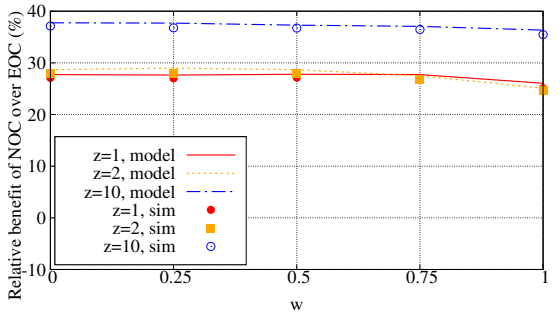


Fig. 8: Tiger distribution, $\alpha = 1.0, C = 1000$

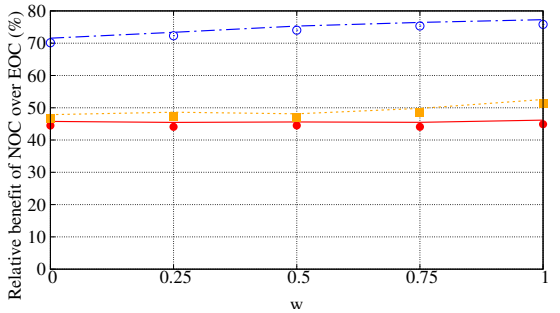
Dtelecom (Figure 9) and Level3 (Figure 10) topologies show different properties. Table II shows how they are different from Geant and Tiger. There are more edge nodes in these topologies; in addition, there are intermediate nodes with high degree. For the Dtelecom topology, Figure 9a shows that a larger fraction of C is allocated to level 2 where the nodes with higher degree are located; although the fraction of C that is distributed among edge nodes increases as temporal locality gets stronger. Figures 9b and 9c illustrate a benefit of in-



(a) Cache distribution



(b) Average distance



(c) Average load on server

Fig. 9: Dtelecom cache distribution, $\alpha = 1.0, C = 1000$

network caching of at least 25% and 45% for average retrieval distance and load on the server respectively. The benefit of optimal in-network caching is constant over w. The reason is the allocation of a larger fraction of C to one level that has ICN nodes with high degree. Having $w = 0.5$ and $z = 10$ results in 36% shorter retrieval distance as well as 75% less load on the server in case of optimal in-network caching. The same story is true for Level3, except that the node with higher degree is located at third level and its degree is much smaller than the degree of the similar node in Dtelecom. The results of influence of optimal caching on distance/server load for Level3 are not shown, due to space limitations.

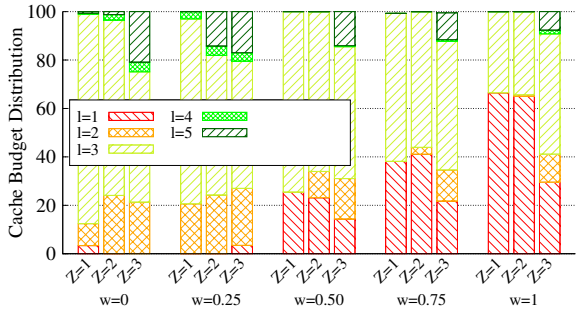


Fig. 10: Level3 cache distribution, $\alpha = 1.0, C = 1000$

V. Conclusion and Future Work

This paper modelled the distribution of users' requests in a non-IRM environment in the network as an optimization problem taking metrics from users' and ISP's point of view into account. The solution for this problem depicts an optimal distribution of cache budget C among the nodes in ICN networks. Studying various settings for Zipf parameter for the distribution of users' requests (α), strength of temporal locality (z), total cache budgets (C) and topologies shows

- Stronger temporal locality causes a larger distribution of total cache budget among edge nodes.
- As total cache budget of C decreases, fraction of C that is allocated to edge nodes shrinks.
- The benefit of in-network caching on distance strongly depends on topologies and temporal locality. The relative benefit of up to 10% is observed for $z = \{1, 2\}$ in the tree, Geant and Tiger topologies. On the other hand, an optimal in-network caching in Level3 and Dtelecom topologies ends in at least 20% shorter distance.
- In-network caching is very helpful in decreasing the overall miss ratio for all settings. A lower overall miss ratio in the system results in forwarding less traffic out of the local ICN network (inter-network traffic) as well as lower server load.

While the findings of this paper are based on LCE and LRU, the optimal cache distribution for other cache replacement algorithms (e.g. k-LRU [1] and LRU(m) [28]) and cache replication algorithms (e.g. LCD [3]) is part of future work.

References

[1] M. Garetto, E. Leonardi, and V. Martina, "A unified approach to the performance analysis of caching systems," *ACM Transactions on Modeling and Performance Evaluation of Computing Systems*, vol. 1, no. 3, pp. 12:1–12:28, May 2016.

[2] A. Kulkarni and A. Seetharam, "Exploiting Correlations in Request Streams: A Case for Hybrid Caching in Cache Networks," in *IEEE LCN*, Chicago, IL, Oct. 2018, pp. 1–9.

[3] N. Laoutaris, H. Che, and I. Stavrakakis, "The LCD interconnection of LRU caches and its analysis," *Performance Evaluation*, vol. 63, no. 7, pp. 609–634, Jul. 2006.

[4] N. Laoutaris, S. Syntila, and I. Stavrakakis, "Meta algorithms for hierarchical web caches," in *IEEE ICPCC*, Phoenix, AZ, Apr. 2004, pp. 445–452.

[5] I. Psaras, W. Chai, and G. Pavlou, "Probabilistic in-network caching for Information-Centric Networks," in *ACM Workshop on Information-Centric Networking*, Helsinki, Finland, Aug. 2012, pp. 55–60.

[6] Z. Ming, M. Xu, and D. Wang, "Age-based cooperative caching in Information-Centric Networking," in *ICCCN*, Shanghai, China, Aug. 2014, pp. 1–8.

[7] V. Jacobsen, D. Smetters, J. Thornton, M. Plass, N. Briggs, and R. Braynard, "Networking Named Content," in *Emerging Networking Experiments and Technologies*, Rome, Italy, Dec. 2009, pp. 1–12.

[8] W. K. Chai, D. He, I. Psaras, and G. Pavlou, "Cache "less for more" in Information-Centric Networks," in *IFIP Networking*, Prague, Czech Republic, May 2012, pp. 27–40.

[9] S. K. Fayazbakhsh, Y. Lin, A. Tootoonchian, A. Ghodsi, T. Koponen, B. Maggs, K. Ng, V. Sekar, and S. Shenker, "Less pain, most of the gain: incrementally deployable ICN," *SIGCOMM Computer Communication Review*, vol. 43, no. 4, pp. 147–158, Aug. 2013.

[10] G. Tyson, S. Kaune, S. Miles, Y. El-khatib, A. Mauthe, and A. Taweel, "A Trace-Driven Analysis of Caching in Content-Centric Networks," in *IEEE ICCC*, Munich, Germany, Jul. 2012, pp. 1–7.

[11] P. B. Danzig, R. S. Hall, and M. F. Schwartz, "A case for caching file objects inside internetworks," in *ACM SIGCOMM*, San Francisco, CA, Oct. 1993, pp. 239–248.

[12] G. Rossini and D. Rossi, "Coupling caching and forwarding: benefits, analysis, and implementation," in *ACM ICN*, Paris, France, Sep. 2014, pp. 127–136.

[13] I. Psaras, R. G. Clegg, R. Landa, W. K. Chai, and G. Pavlou, "Modelling and evaluation of CCN-caching trees," in *IFIP Networking*, Valencia, Spain, May 2011, pp. 78–91.

[14] R. Fagin, "Asymptotic miss ratios over independent references," *Journal of Computer and System Sciences*, vol. 14, no. 2, pp. 222 – 250, Apr. 1977.

[15] H. Che, Y. Tung, and Z. Wang, "Hierarchical web caching systems: modeling, design and experimental results," *IEEE Journal on Selected Areas in Communications*, vol. 20, no. 7, pp. 1305–1314, Sep. 2002.

[16] C. Fricker, P. Robert, and J. Roberts, "A versatile and accurate approximation for LRU cache performance," in *ITC*, Anaheim, CA, Sep. 2012, pp. 1–8.

[17] M. Garetto, E. Leonardi, and S. Traverso, "Efficient analysis of caching strategies under dynamic content popularity," in *IEEE INFOCOM*, Hong Kong, Apr. 2015, pp. 2263–2271.

[18] P. Gill, M. Arlitt, Z. Li, and A. Mahanti, "Youtube traffic characterization: a view from the edge," in *ACM IMC*, San Diego, CA, Oct. 2007, pp. 15–28.

[19] A. Mahanti, C. Williamson, N. Carlsson, M. Arlitt, and A. Mahanti, "Characterizing the file hosting ecosystem: a view from the edge," *Performance Evaluation*, pp. 1085 – 1102, Nov. 2011.

[20] M. Saxena, U. Sharan, and S. Fahmy, "Analyzing video services in web 2.0: a global perspective," in *NOSSDAV*, Braunschweig, Germany, May 2008, pp. 39–44.

[21] M. Zink, K. Suh, Y. Gu, and J. Kurose, "Watch global, cache local: YouTube network traffic at a campus network: measurements and implications," in *MMCN*, vol. 6818, San Jose, CA, Jan. 2008, pp. 681 805–1–13.

[22] E. J. Rosensweig, J. Kurose, and D. Towsley, "Approximate models for general cache networks," in *IEEE INFOCOM*, San Diego, CA, Mar. 2010, pp. 1–9.

[23] G. Carofiglio, M. Gallo, L. Muscariello, and D. Perino, "Modeling data transfer in Content-centric Networking," in *ITC*, San Francisco, CA, Sep. 2011, pp. 111–118.

[24] A. Dabirmoghaddam, M. Barijough, and J. Garcia-Luna-Aceves, "Understanding optimal caching and opportunistic caching at "the edge" of Information-Centric Networks," in *ACM ICN*, Paris, France, Sep. 2014, pp. 47–56.

[25] A. Montazeri and D. Makaroff, "Geographically-distinct request patterns for caching in Information-Centric Networks," in *IEEE LCN*, Singapore, Oct. 2017, pp. 579–582.

[26] R. Chiocchetti, D. Rossi, and G. Rossini, "ccnsim: An highly scalable CCN simulator," in *IEEE ICC*, Budapest, Hungary, Jun. 2013, pp. 2309–2314.

[27] C. Bernardini, T. Silverston, and O. Festor, "A comparison of caching strategies for Content-Centric Networking," in *Proceedings of the IEEE Global Communications Conference*, San Diego, CA, Dec. 2015, pp. 1–6.

[28] N. Gast and B. V. Houdt, "Asymptotically exact TTL-approximations of the cache replacement algorithms LRU(m) and h-LRU," in *ITC*, vol. 01, Würzburg, Germany, Sep. 2016, pp. 157–165.