ARPN Journal of Engineering and Applied Sciences
©2006-2016 Asian Research Publishing Network (ARPN). All rights reserved.
www.arpnjournals.com

CACHE-SKIP APPROACH FOR INFORMATION-CENTRIC NETWORK

Ibrahim Abdullahi[1, 2] and Suki Arif[1]
[1]InterNetWorks Research Laboratory, School of Computing, Universiti Utara Malaysia, UUM Sintok, Kedah, Malaysia
[2]Departmental of Mathematics/Computer Science, Ibrahim Badamasi Babangida University, Lapai, Niger State, Nigeria
ibrojay01@ibbu.edu.ng

ABSTRACT

Several ICN cache deployment and management techniques have since been using the Web management techniques to manage information sharing and better cache-hit ratio. Leave Copy Down, Leave Copy Everywhere and Probabilistic cache managements have gained more attention. However, with Leave Copy Everywhere being the initial design specification in ICN proposal, several research issues of content manageability have posed a threat of particularly content and path redundancy. This paper presents an extensive simulation analysis of the popular cache management techniques by subjecting the concepts into different network topologies to investigate the prospect of extending and proposing a new form of cache management in ICN known as Cache-skip. Cache-skip use the consciousness of time of request, network size and Time Since Birth (TSB) and Time Since Inception (TSI) to carefully dedicate the positions of caching to benefits hit rates and less network stress as a form to efficiently utilize the bandwidth and enhance hits.

Keywords: information-centric networks, cache deployments, cache-skip, cache management, on-path, off-path.

INTRODUCTION

Current practice on the Internet has been overtaken by multimedia data, smart gadget connectivity and time consciousness [1] [2] [3]. Users are becoming more demanding in terms of ubiquitously engaging their devices for updates, processing and dissemination of information. Internet of Things (IoT), Internet of Everything (IoE) and the content-aware network are the promises of the future Internet. Information-centric network (ICN) [4] [5], promised to use the name ahead of the traditional addressessing) Internet. With ICNs' added advantage of the unique data structures of content stores (CS), pending interest table (PIT) and the forwarding information base (FIB) in some ICN approaches (particularly NDN), caching is presumed as the driving force to achieving the content-centrism for the future.

Caching in ICN is defined in different forms, strategies and concepts. Among the current submitted cache management practices are the Leave Copy Down (LCD) [5] [6], Leave Copy Everywhere (LCE), Move Copy Down (MCD) and Probabilistic cache (ProbCache) [7]. Currently, as researches prevail thereby ICN gaining more acceptance and popularity, several studies identified the need to extend the LCD caching strategy [8]. LCD with its high cache-hit benefits users (usually referred to subscriber) due to the close (proxy) deployment of the needed data (interest) as seen in NDN.

However, new cache strategies promised better performance to the initial LCD in ICN [5]. These strategies are the LCE, MCD, ProbCache, AssignCache, Cooperating cache, coordinating cache etc. as will be presented and described in the subsequent sections. With high opportunities and benefits in these strategies, bandwidth usability is managed better due to ICN form of caching a content in a node (router) every time a data or interest crossed a node. A user (Clients1 and 2) benefits from ICN through gaining access to data or information with lower hop counts and crosses once a neighbor node has earlier demanded for that data [9]. See Figure-1.

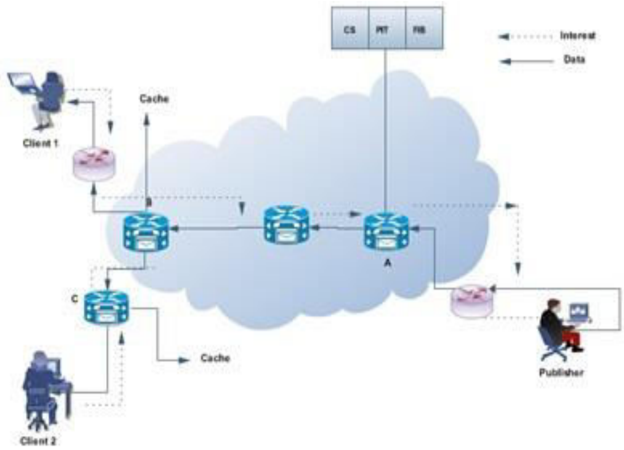

Figure-1. ICN overview.

Convincingly, ICN cache strategies LCD, LCE, and MCD and ProbCache [6] are beneficial in their advantages of availing information to subscribers. However, challenges are envisaged as *content* and *path redundancy* occurs when large chunks are flooded on all route-path routers. Once the LCD is in practice, high tendencies of leaving copy down would result in having to cache data that will have less or no relevance to a node stowed data. Excessive LCD will therefore consume paths and memory due to the limitation of cache sizes and blocks. In LCD, contents are dropped close to the publisher nodes once a request is served. Consequently, the data could as well be a factor of consuming the available storage of a node that acts as an intermediary for interests or packet data delivery path.

In another study by Psaras *et al.* [9], *path redundancy* was also identified as a challenge envisioned when using the LCD or LCE strategies. *Path redundancy* could also be defined as the exhaustion of memory path or

VOL. 11, NO. 5, MARCH 2016 ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences
©2006-2016 Asian Research Publishing Network (ARPN). All rights reserved.
www.arpnjournals.com

nodes due to the LCE and LCD cache deployment strategies in ICN. A node holding a chunk spaced data that have not been evicted, results into dropping of other data (requests) on that path. These are seen as the challenges that motivates our proposed *cache-skip deployment* strategy to mitigate the problems of *content* and *path redundancy* in ICN. *Content redundancy* is the resulting cache problem that could arise as a result of data ubiquity in a network. Excessive lodging of the chunks results in out-of memory condition for other contents and interest moving on-path of the network. Content and data represent the information gotten from a publisher while interest or requested-data mean the same. The use of data and content chunk shall be referred interchangeably to define a resulting data from a publisher while interest and requested-data shall also be used interchangeably meaning a request from the user (subscriber).

CACHING STRATEGIES IN ICN

Caching in ICN can be defined based on their deployment form, managing algorithm, placement and replacement. In this study we shall be covering the caching management techniques. Recent literatures [9] [10] [11] [12] has shown that the LCD, LCE and ProbCache gained more popularity among other cache management techniques.

Leave Copy Everywhere (LCE)

LCE is a cache management technique that has been used in the traditional web management and deployment of the current Internet [6]. In LCE, once a network N exists in an ordered hierarchy, and a request is placed by a subscriber (User).

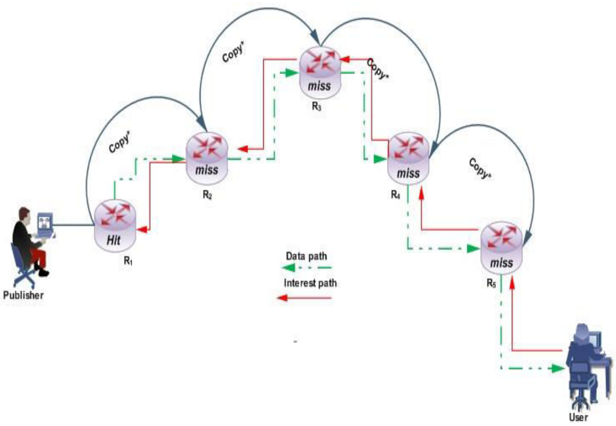

Figure-2. Leave copy everywhere.

The request for the data is then advertised or publish into the network using the routing and forwarding algorithm in the action. As depicted in Figure 2, the Publisher sends out the requested-data via channels and nodes serving as its neighbor. In LCE, request keep crossing nodes until a cache is *hit* (obtained data), the operation of searching for data in a node without being granted is termed as *cache-miss*. From Figure-2, LCE, the router with identification R_1 holds the needed data. Using this cache management technique copies chunk of the data into all routers, on-path as a replica. This is however seen as a criteria for improving cache-hit, but accumulating redundant data, waste of bandwidth and path delay.

Leave Copy Down (LCD)

In LCD, the management of the cache is entirely different as the cache command aims at carefully avoiding lot of cache redundancy but suffers from hit network *stress*. Stress is the distance that a node need to cover before caching a data or communicating with another router node. In LCD, a cache-hit router (cache found), drops a copy of the data chunk to its immediate neighbor node. This benefit subsequent request for such data by having to visit the cached node in lieu of traversing all to the main source. Figure-3.; below depicts the operation and cache manageability in LCD.

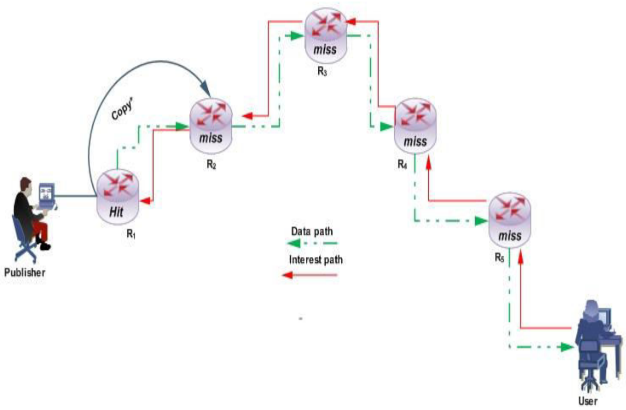

Figure-3. Leave copy down.

Router R_5 - R_1 shows the cache relationship between the content requester and the publisher.

Move Copy Down (MCD)

Move Copy Down (MCD) aims at predictory minimizing the amount of replica data chunk on nodes. However, MCD is considered a fast option in data accessing due to its operation of uniquely deleting the copied data from a local repository. Once a request is sent into the network, a path search is initiated with all nodes searched. A *miss* is recorded every time a node is visited and the data is not found while a cache-hit is returned when the content is obtained. In all aforementioned cache managements, it is understood that every preceding node knows the information of its neighbor. Node cognizant played the role that enables the MCD, LCD and LCE engage in the copying. Weakness in MCD is thus obtained when several requests are placed just immediately in a neighbor node bearing little stress to the local replica.

ARPN Journal of Engineering and Applied Sciences
©2006-2016 Asian Research Publishing Network (ARPN). All rights reserved.

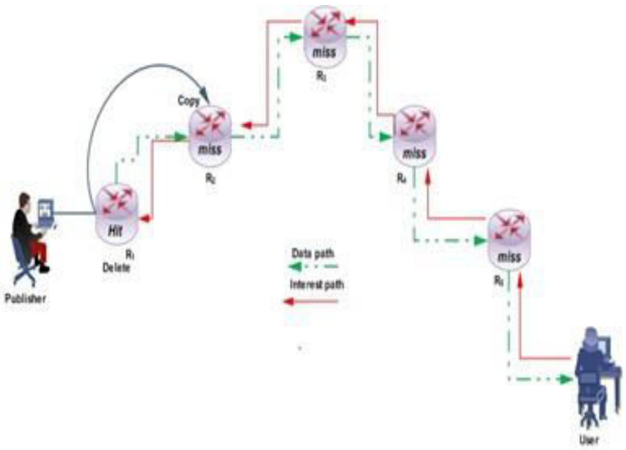

Figure-4. Move copy down.

MCD therefore deletes the copy in the local replica without interfering with the main origin. Figure-4 explains the relationship in MCD and *cache-hits*.

Probabilistic Cache (ProbCache)

As the name implies, ProbCache is a special functional cache management technique that use little or no cognizant to the preceding node. It is usually thought of as an extension of LCD. In ProbCache, cache routers are requesting data as the usual requests in other techniques. A *Miss* in ProbCache is defined as in LCD, LCDE and MCD. Once a hit is recorded, the cache node copy of the data chunk is moved and copied into a successive node on-path with its distinction in placement using the appropriate probability estimation. It is thus worthy to know that the nodes are copied in an (*l-1*) manner as described by Loutaras *et al.*, and Psaras *et al.*, [6] [7]. ProbCache as depicted on Figure 5, exhibits cache operation by placing a copy with the probability *p* and thus not satisfying the (*1-p*) *q* node. This is assumed to be fair as compared to the LCD, and LCE. In ProbCache, a node could be probabilistically selected which may in-turn become a *proxy* to the user. A good cache-hit is always feasable when nodes have to traverse less with better stress to a content-cache.

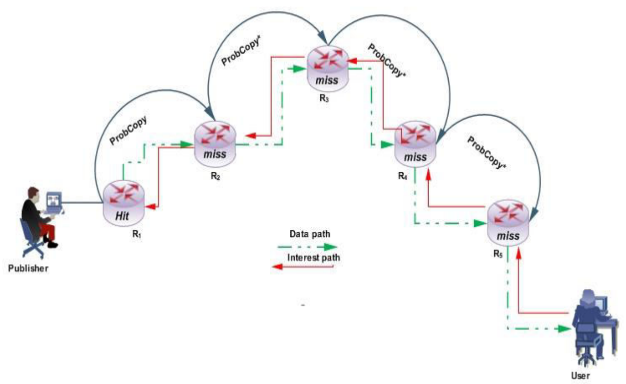

Figure-5. ProbCache.

Figure-5 shows the possibility of assuming a *cache-hit* after copying a content. The form of Probcache ends up having only a probabilistic cache copy on a node. In Figure 5, the ProbCopy* assumes an immediate hit only for diagram illustration.

STRATEGIES COMPARISON IN ICN

This section presents the computational and simulation comparisons of LCD, LCE and ProbCache. For easier analysis, the paper focuses on the three popular strategies. In the results obtained from simulation, ProbCache is seen as a leading strategy using the same parameters and network topologies. In the analysis, cache-hit ratio of the studied cache managements were carefully observed in two (2) popular ICN topologies. Abilene and Duetsche-Telecom network topologies were used for the comparison and analysis.

For the simulation comparisons of LCD, LCE and ProbCache, the following parameters provided in Table-1 were used. For further details refer to [12].

Table-1. Simulation parameters.

Traffic Generation information	
Number of users	3980
Number of files	10,000
Mapping Algorithm	Random
Time- limit	86,400
Social graph	Facebook
Number of communities	5
Cache replacement	LRU

Abilene topology cases

Abilene network topology: Sometimes referred to Intetnet2 network topology. Abilene topology was designed with the sole object of transferring large size data across nodes. Its building was ideal to test and compare our simulation parameters to record the cache-hits and observe the efficacy of our cache managements. Abilene heterogenous nature depicts eleven (11) station sites as nodes. Its hierarchical structure provides the flexible neighbor composition needed to test ICN cache deployment. This forms the bases of our topology selection to test.

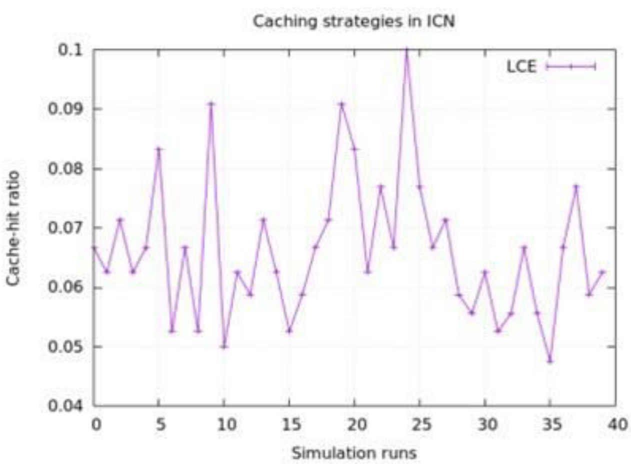

Figure-6. LCE Abilene.

VOL. 11, NO. 5, MARCH 2016 ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences
©2006-2016 Asian Research Publishing Network (ARPN). All rights reserved.
www.arpnjournals.com

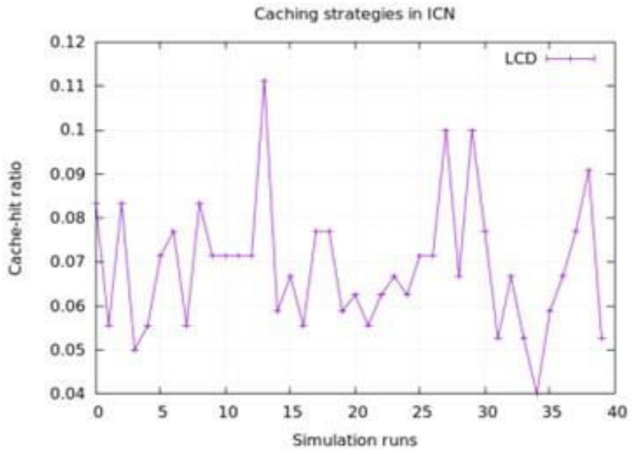

Figure-7. LCD Abilene.

In Figure-6, we present the results obtained from our simulations using the socialccnSim [13]. The results obtained shows the cache-hit ratio of the content cached. On Abilene topology using Table-1 parameters. It was observed that the hit-ratio was good as noticed in the graph with a cache-hit ratio hitting the 100 mark at the 25 run of the simulation. The correlation of the runs of simulation was not far from granting a user request almost at good runs between 20-25. A declination observed in the simulation is attributed to the exhaustion of router (path) space earlier mentioned as path redundancy. LCE cache requested data on all router making contents reside redundantly on nodes. This shows a waste of bandwidth with a trade-off of a cache-hit. Comparing with the LCD, the cache-hits differ in their form of acquiring a requested data on the Abilene topology.

From the results obtained through simulation, the LCD performs better in terms of hit-rate as compared to the LCE. However, it hits the high mark at simulation 13 run. It shoots high to 0.11 cache-hit ratio due to its advantage of caching at the immediate nodes without necessarily deleting its preceding node. However, in LCD, the stress was high, but a trade-off of bandwidth utilization was obtained against hit-ratio. It is thus right to say a lesser cache redundancy was obtained.

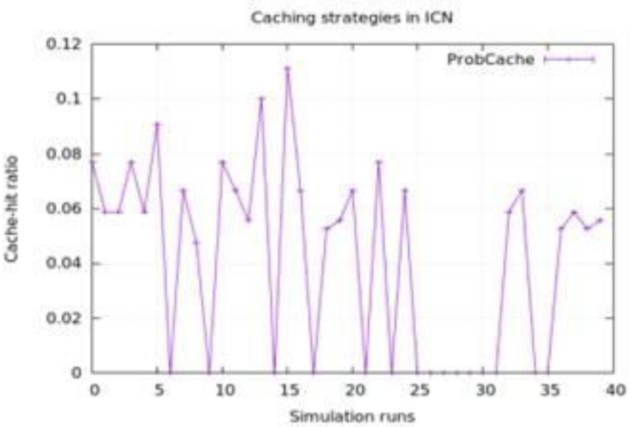

Figure-8. ProbCache Abilene.

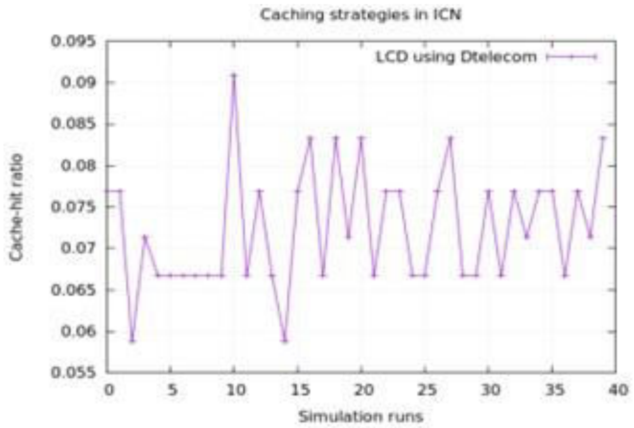

Figure-9. LCD Dtelecom.

ProbCache has been seen through the simulation to be more effective in terms of redundancy in path and content. Its only negative transformation was observed between runs 25-31. It was noticed that no cache-hit was obtained as the runs and demand for the data increased. This shows the probability of the content being cached was $(1-p) = q$. However, it is fair to conclude that the trade-off was high in terms of keeping the path less congested, better stress probabilistically against LCD. In LCE, the ProbCache won in terms of avoiding all routers/nodes consumed. ProbCache can thus be extended to improve stress, and cache-hit ratio through *proxy* positioning.

Deutsche-Telecom topology cases

To test fully the heterogeneity in the ordering of nodes and stations, we used the D-telecom topology to investigate the cache-hits and stress on the network of content. We found out through the simulation that the values of the stress were quite different compared to what we obtained in Abilene topology. Using the same network traffic parameters as presented in Table-1 the following results were obtained.

From Figure-9, the D-Telecom cache-hit was almost the same as the hit ratio was observed to be more at the 0.072 between the 2-3, 5-6, 14-15, 20-22 and continues in that order by increment. However, the highest hit ratio was noticed this time around on the 33 run simulation mark on the x-axis. This has proven that the amount of data flooded is high and as such affecting the hit ratio. Consequently, in LCD run on D-telecom, the results appear to be the opposite as more hit were constant or related at about 0.077.

As previously mentioned, the LCD saves the memory waste and implies a hit on D-telecom due to the heterogeneous ordering of the topology.

Conclusively, it is fair to note that LCD saves bandwidth and recorded a good hit using the D-telecom topology.

From Figure-11, ProbCache seems the most flexible cache management technique and suited to a D-telecom architectural structure of a network. The

VOL. 11, NO. 5, MARCH 2016 ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences
©2006-2016 Asian Research Publishing Network (ARPN). All rights reserved.

parameters on Table-1 were also used for the results obtained. It is worthy to mention that once a data has been found (cache-hit), probabilistically copies of the data is replicated as caches in the next availing spaced nodes. This thus, increase the chances of a data gaining better hits and likely proxy positioning to the subscriber.

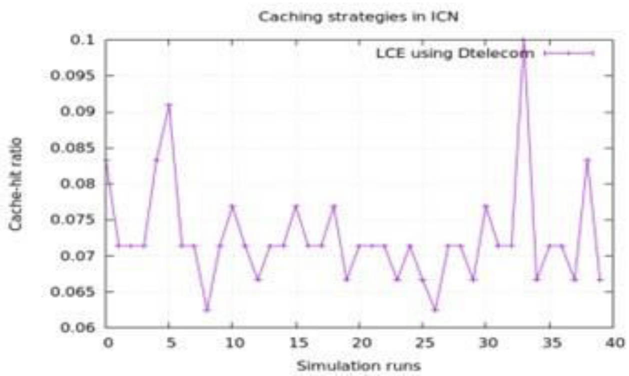

Figure-10. LCE Dtelecom.

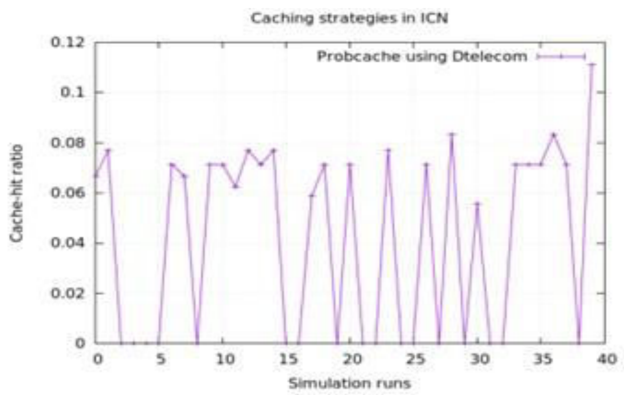

Figure-11. ProbCache Dtel.

Results summary

Simulation results have given the bases for an extension of the ProbCache to enhance data availability with lesser path and content redundancy. In ICN, the yearn advantage is to avail data almost at all nodes on the path. But the excessive requests of data driven activities on the Internet posed a threat to the traffic in upward and downward streaming as submitted by cisco [1-4].

CACHE-SKIP CACHING MANAGEMENT

Cache-skip is a technique aimed at extending the possibility of caching in a more flexible form. As opposed to the earlier cache managements in section 2, the skip cache will function in an LCD to the first node in succession to the point of hit. On the LCD practice in the cache-hit node, a length calculation usually referred to as *Time Since Inception (TSI)* and *Time Since Birth (TSB)* is used to probabilistically select the mid-position routers on the network. For detail explanation on TSI and TBI refer to Psaras *et al* [7]. The next section presents our proposed equations and the description of the network size extension by introducing new variables in relation to time, distance and position.

Model formulation

Cache model in ICN using the skip technique, is a method of using the graph theory to create positions and node clusters.

Assume a network as a graph S, with Vertices V and Edges E, thus given as:

$$H = (V, E) \tag{1}$$

where $V = \{v_1, v_2, v_3, ..., v_n\}$ represents the vertices routers) in the network.

$$\text{Edges } E = \{E_1, E_2, E_3, ..., E_m\} \tag{2}$$

such that

$$Size(H) = \Sigma_{Ei \in} E|Ei| \tag{3}$$

according to Gallo *et al.*, [13].

The definition and representation of the inner matrix $|Ei|$ presents the relationship of the nodes. Such that $E \subseteq V$, where E represents the nodes/routers and V represents the connector/paths.

such that $\forall E, \exists V,$

$$\text{Where } (E_1, E_2, E_3, \dots, E_n) + (V_1, V_2, V_3, \dots, V_m) = \omega_i \tag{4}$$

$$\omega_i = \sum_{i=1}^{n} (E + V) \dots (5)$$

the resulting network is therefore = $\omega_i, \omega_j, \omega_k \dots \omega_m$

such that $i \neq j \neq k;$

$$Size(H*) \xrightarrow{yields} \Sigma Ei \in E|Ei| + \theta \tag{5}$$

where θ defines the change in time of data request.

The resulting network relationship is then presented as

$$\text{Cache-skip} = \sum E_i \in E|\omega_i| \tag{6}$$

$i = 1, 2, 3, \dots, k$

Cache-skip description

Adopting the Abilene-like topology due to its tree heirachy relationships between nodes, we present the operation of the skip as follows:

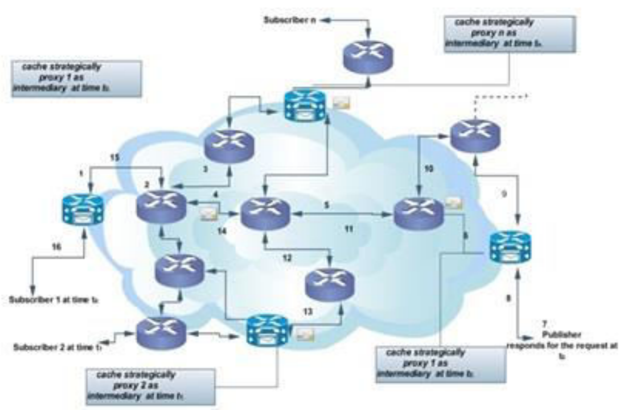

Figure-12. Cache-skip strategy.

©2006-2016 Asian Research Publishing Network (ARPN). All rights reserved.

The operation is started by a user referred to as subscriber 1 sending an interest at time t_0. The *TSI* and *TBI* are then calculated as we represent the operation as cache strategically on Figure-12. Once a cache is hit, a copy is immediately deposited on the next neighbor router as a proxy to the subscriber.

The benefitting advantage of the cache-skip is to mitigate the cache and path redundancy on-path of the network. A mid-calculation is thus computed using the *TSI*. Once a request is issued by subscriber 2, at time t_1, a new route computation is formed and obtained as shown on equation (5) and (6). The concept continues in this fashion until the whole data is strategically moved in and out of the nodes depending on the adopted placement and replacement algorithm.

CONCLUSIONS

This paper proposed a new cache management technique for ICN to reduce the mount of redundant data flooded into the network. Mathematical description of the relationships between different parameters was used. Cache path redundancy has been seen as a little ICN cache hurdle that need to be carefully managed. The formula presented and described could thus be used to simulate and compare the results with the already simulated results in socialccnSim for future evaluation.

AKNOWLEDGEMENTS

The authors wish to thank the Ministry of Education, Malaysia for funding this study under the Long Term Research Grant Scheme (LRGS/bu/2012/UUM/Teknologi Komunikasi dan Infomasi).

REFERENCES

[1] Abdullahi, S. Arif and S. Hassan. 2015. Ubiquitous Shift with Information Centric Network Caching Using Fog Computing. Computational Intelligence in Information Systems. 331: 327-335.

[2] Click, "Internet Time Line," 2015.

[3] Cisco, "Cisco Visual Networking Index:Global Mobile Data Traffic Forecast Update, 2013-2018," 2014.

[4] B. Ahlgren, C. Dannewitz, C. Imbrenda, D. Kutscher and B. Ohlman. 2012. A survey of information-centric networking. Communications Magazine, IEEE. 50(7): 26-36, July.

[5] V. Jacobson, D. K. Smetters, J. D. Thornton, M. Plass, N. Briggs and R. Braynard. 2009. Networking Named Content. Proceedings of the 5th International Conference on Emerging Networking Experiments and Technologies.

[6] N. Laoutaris, S. Syntila and I. Stavrakakis. 2004. Meta algorithms for hierarchical Web caches. Performance, Computing, and Communications, IEEE International Conference.

[7] Psaras, W. K. Chai and G. Pavlou. 2012. Probabilistic In-network Caching for Information-centric Networks. Proceedings of the Second Edition of the ICN Workshop on Information-centric Networking, New York, NY, USA.

[8] Abdullahi, S. Arif and S. Hassan. 2015. Survey on caching approaches in Information Centric Networking. Journal of Network and Computer Applications. 56: 48-59.

[9] Psaras, W. Chai and G. Pavlou. 2014. In-Network Cache Management and Resource Allocation for Information-Centric Networks. Parallel and Distributed Systems IEEE Transactions. 1:1.

[10] Abdullahi, S. Hassan and S. Arif. 2015. Prospective use of Bloom Filter and Muxing for Information Centric Network Caching. ARPN Journal of Engineering and Applied Sciences. 10(3): 1169-1177.

[11] C. Fang, R. Yu, T. Huang, J. Liu and J. Liu. 2015. A Survey of Green Information-Centric Networking: Research Issues and Challenges. Communications Surveys Tutorials IEEE: 1-1.

[12] G. Xylomenos, C. Ververidis, V. Siris, N. Fotiou, C. Tsilopoulos, X. Vasilakos, K. Katsaros and G. Polyzos. 2014. A Survey of Information-Centric Networking Research. Communications Surveys Tutorials, IEEE. 16(2): 1024-1049.

[13] B. Cesar, Social CCN Sim is a CCN simulator, 2014.

[14] G. Gallo, G. Longo, S. Pallottino and S. Nguyen. 1993. Directed Hypergraphs and Applications. Discrete Appl. Math. 42(2-3): 177-201.