VOL. 10, NO. 3, FEBRUARY 2015 ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences

www.arpnjournals.com

PROSPECTIVE USE OF BLOOM FILTER AND MUXING FOR INFORMATION CENTRIC NETWORK CACHING

Ibrahim Abdullahi[1, 2], Suhaidi Hassan[1] and Suki Arif[1]
[1]InterNetworks Research Laboratory, School of Computing. Universiti Utara Malaysia. UUM Sintok Kedah, Malaysia
[2]Department of Mathematics/Computer Science, Ibrahim Badamasi Babangida University, Lapai. Niger State, Nigeria
E-Mail: ibrahim@internetworks.my

ABSTRACT

Information dissemination as the main objective of setting up the Internet has seen different folds to improve its course. Information centric networking (ICN) has been introduced with the aim of curtailing some future challenges posed at the traditional Internet in the nearest future. ICN advantage of caching chunks of information on-path and off-path of the network stands the paradigm out as an alternative shift from host centric network to name centric. ICN caching approach can thereby significantly reduce amount of times a host is visited. Bloom Filter with its advantage of fast searching and false positivity characteristics are seen as form of message retrieval practice to improve interest serving on the network. This paper analyzed the advantages of vending and adopting Bloom Filters and Muxing as research directions to minimize excessive bandwidth consumption, lesser delays, prompt delivery of information, higher throughput and the ability to share information from troubled stations. Concepts are proposed and wider algorithms are pointed out to increase the overall ICN framework as related to caching and other network services.

Keywords: information centric network, caching, content centric network, bloom filters, muxing, scheduling.

INTRODUCTION

ICN has been developing with various architectures and forms of design. These include Content Centric Networking (CCN) and Named Data Network (NDN) (Jacobson, et al. 2009), Data Object Network (DONA) (Tyson, et al. 2013), Network of Information (NetInf) (Ahlgren, et al. 2008) (Dannewitz, Herlich and Karl 2012) among others. However, for the purpose of this paper, the concentration will be more on information retrieval, forwarding and caching.

Researches have presented quite a number of dividends of adopting ICN as future Internet. The ICN concept is best described as a neighbor aid communication through subscription of interest and the publishing of the required interest as information forwarding (Jacobson, et al. 2009). Jacobson, et al. proves that the information referred to by names stands the advantage of lesser host visitation.

ICN has been envisioned as the future Internet. However, some open research issues exist particularly in naming, routing, scalability, mobility, scheduling, security and caching. Suggested works have submitted the potential use of flat naming, hierarchical and distributed hash naming schemes (Jacobson, et al. 2009).

Therefore, ICN as a future replacement of objects referral by name serves better advantages than IP addresses. Ciscos report on the huge amount of information on the Internet and mostly multimedia has thus necessitated the need for subscribing into Internet of names and object (Jacobson, Smetters, James D., & Michael F., 2009). With the properties of the neighbor node caching a chunk through the *Leave Copy Down* (Psaras, Chai and Pavlou, 2013); it makes ICN more beneficial to reduce bandwidth consumption by caching along the paths. Once a router or node has requested for information through subscription of interest, the interest is looked up on the entire network or on some parts of the network using the shortest path to the source of the interest.

The major challenge here becomes that the interest is directly granted via retrieval from the special store known as Content Store (CS). Once the interest matches the information in the store, it is then served. Consequently, if the information is not hit (found), the interest is placed in the ICN Pending Interest Table (PIT). Additionally, on the event that the interest is met, Forwarding Information Base (FIB) forwards the interest through the route of communication as the interest comes (see Figure-1).

Researches conducted on caching are yet to conclusively convince the optimal caching practice that best suit the ICN paradigm. Therefore, the need to propose a fast agglomerated concept of Bloom Filters (Bloom, 1970) is not off the track. Bloom Filters advantage of fast searching will necessitate its inclusion into ICN caching. When interest are sent out by subscribers of object using names as submitted by Jacobson *et al.*(2009) and Ahlgren, Dannewitz, *et al*. (2012), fast look-up are needed especially on highly loaded network. This would improve the overall throughput and bandwidth of the entire network activities. Psaras et al., suggested that a reserved memory chunk is needed for the farthest node so as to minimize redundancy and improve further request grants (Psaras, Chai and Pavlou, 2013).

Cache placement on the network has also seen lots of active researches such as the proxy, active, reverse proxy, adaptive caching architectures (Nagaraj, 2004),((Psaras, Chai and Pavlou 2013), (Psaras, Chai and Pavlou 2012), (Wang, et al. 2013) and (Wang, Zhang and Bensaou 2013)). Without a proper caching in place, ICN cannot be achievable.

ARPN Journal of Engineering and Applied Sciences

www.arpnjournals.com

For the purpose of this paper, proxy caching shall be used along Bloom Filter technique to enhance the lookup. A Longer Job First (LJF) scheduling algorithm (Abdullahi and Junaidu 2013) is proposed to enhance and increase the overall number of caches on-path as the interest is served. Studies have tried to identify the problems of replacing the object contents cached on the nodes. However, searching a long shrubby network will demand a fast approach to return the results of the subscribed object.

The next section of the paper will be discussing the motivation and related works on caching in ICN. Section 3 shall introduce the major components of the ICN architecture while Section 4 shall suggest the possible researchable areas. Bloom Filter properties and cache placement shall be covered in Section 5. Section 6 will present a proposed algorithm while Section 7 concludes the paper.

MOTIVATION AND RELATED WORK

Caching is the main concept that underlines the ICN technology. Additionally, when caches are mentioned, due to its lower space management nature, the need to have an acceptable placement of frequently requested information posed a necessity. ICN fashions in a way that every router node that has information past, caches what passes? In the event that a request is sent and the spaces become exhausted, the placement must take place in this respect. ((Psaras, Chai, & Pavlou, 2013), (Wang, Zhang, & Bensaou, 2013), (Zhang, Wen, Xie, & Yu, 2013)

ProbCache as suggested in (Psaras, Chai and Pavlou 2013) uses the probabilistic algorithm for content caching. The results obtained demonstrated the use of a length of the network and nodes, path capacity and monitoring. The results through simulation show that it is vital for contents to leave a space for path caching on the network and also to multiplex contents from server to clients. The placement policy used was Least Recently Used (LRU) for objects and content cahing. Using the LRU as demonstrated in the paper, proved to have reduced the redundancy of objects in the store and improved hit ratio. Part of the achievements of the work submitted the ideal number of hop crosses were lesser after the first initiation of the publish-subscribe.

In another study by (Wang, Li, Tyson, Uhlig, & Xie, 2013) in optimal cache allocation for CCN, cache allocation problem was addressed. The question that motivated this research was also captured as "how storage should be distributed? And where exactly should cache capacity be placed?" (Wang, et al. 2013). The conclusion from the work proves that heterogeneous allocation out-performs homogenous cache allocation. However, it was noted that a mix of the placement policy serves the cache practice better. Most popularly used placement policies known as Least Frequently Used (LFU) and cache everything as studied in (Jacobson et al., 2009) yeild good results. Additionally, LFU with fixed caching probability also provided several results of importance.

Wang et al. in (Wang, Zhang and Bensaou 2013) and (Zhang, et al. 2013) studied caching in the Intra-Autonomous System (AS) cooperative caching. The study proposed solution to redundancy request on path. The work made an inclusion of the autonomous neighbor nodes serving similar interest to another by collaborating. It was gathered that the work solved the problem of eliminating redundancy interest by the AS serving each other's request. It was completely opposite of the work done by (Psaras, Chai and Pavlou 2012), (Wang, Li, Tyson, Uhlig, & Xie, 2013), and (Zhang, Wen, Xie, & Yu, 2013).

Motivation in this paper is a quest of adding a faster look up in the network which could as well make ICN architecture faster in request delivery. This will also aid in minimizing excessive delays before a Pending Interest Table (PIT) respond. In order to address these issues, Bloom Filters introduced by Burton H. Bloom in 1970 for space management would be suggested. Bloom Filter technique (Bloom, 1970) aims to minimize the delay making this contribution entirely different from the aforementioned in other studies. Whereas Multiplexing (Muxing) concepts of circuits will categorize similar requests of interest coming from different nodes. By so doing, reducing individual request sending into the networks independently. This will definitely minimize loads and transportation costs on the network.

OVERVIEW OF ICN

The ICN proposed by Jacobson et al. in 2009. Also described in (Ahlgren, D'Ambrosio, et al. 2008) and (Ahlgren, Dannewitz, et al. 2012) aims at pulling information from the neighbor nodes. This however reduced the general inter-communication of frequent posting of information. Redundancy of information is better minimized through caching. Thus saving a chunk or entire piece of what is important or most recently used or Recently Frequently Used (RFU) information.

Figure 1 describes how ICN works. Interests and data objects are subscribed and fed to the nodes through a neighbor channel. A user (Subscriber 1) starts by initializing a request known as interest at action 1. The interest is recorded into the content store and served immediately if the interests exist. This interest is forwarded via the same route and path where the interest came from using the special forwarding property in the Forwarding Information Base (FIB). A record of such interest is kept for an entry record in the PIT once the interest is not served thereby looking up the network to pull the object. At point (a) on the figure, a publisher at the other end publishes some objects into the network. Point (b) shows how the information is forwarded to a cache-enable router node. Thus, saving the object in its store for other interest from different sites to feed and pull.

The operation continues in this fashion using names as discussed in the following section of the paper. From Figure-1, object from the required interest is first pulled at action point (d). This is only achievable as a result of the caching properties in router S1.

VOL. 10, NO. 3, FEBRUARY 2015 ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences

www.arpnjournals.com

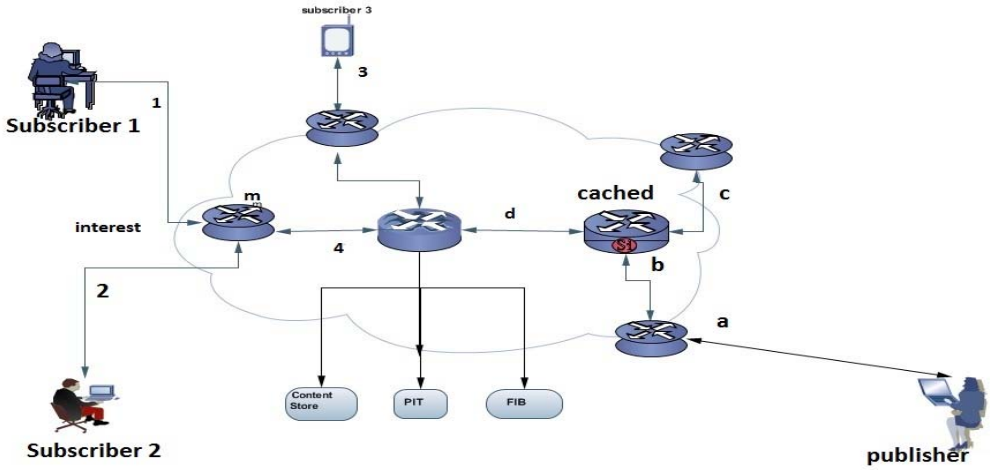

Figure-1. ICN architecture.

However, like many other paradigms, this concept has its challenging points yet to be fully tackled. Amongst them are:

1. Naming

Information is required to be referred to by names in lieu of the IP addressing as in the traditional Internet. Hierarchical, flat and distributed hash functions have been proposed and used (Jacobson, et al. 2009), (Tyson, et al. 2013) (Zhang, et al. 2013), (Kubiatowicz, et al. 2000). Naming in ICN defines how a subscriber (requester) of Interest sends out interests/objects. Once a name of an interest is posted on the network, the name resolution function handles the next set of direction providing a match. It differs from the traditional Domain Name System (DNS) server and the Dynamic Host Configuration Protocol (DHCP) through some features. ICN dream sounds promising as the hurdles of sending interests to farther distances will definitely be minimized through the cache node ability.

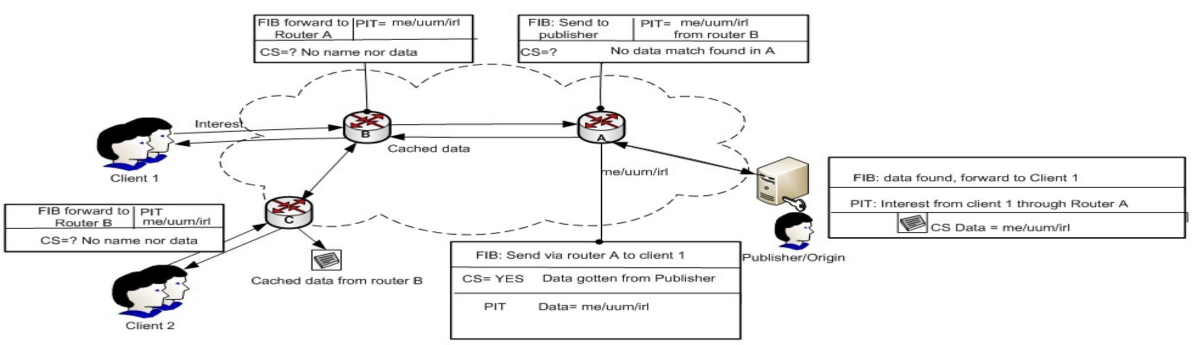

Figure-2. Naming and resolution in ICN.

1. Name resolution

Interests are sent out onto the network using names. As soon as the interests are delivered, cache-enabled nodes respond to the interest. This is used in all popular ICN architectures (NDN, NetInf, DONA, etc.) except the Publish/Subscribe Internet Routing Paradigm (PSRIP). In PSRIP also known as the Publish-Subscribe Technology (PURSUIT) of the FP7 EU Project 2010, name resolution is usually settled on the special server-like station known as *Rendezvous Node* (RN). When subscriptions are posed by the requester, it is channeled into the local RN which in turn communicates to find a match in the Global RN. For other architectures such as CCN, DONA, NetInf, distributed hash functions are proposed to make the matching. An acceptable standard is yet to be reached on the matter for the most suitable form of matching and resolving names in ICN.

From Figure-2, Client 1 sends out an interest using the flat scheme naming, this name is forwarded to router A as depicted on the figure. Once a match is found through resolution algorithms, the interest is served. From the Figure-2, Router B had to use the FIB to forward the interest so as to obtain the object *me/uum.irl* published by the origin. It is clear to note that Client 2 would not need to travel down the lane as router B had previously cached the data requested for.

2. Mobility

Challenging as it may seem, mobility is the ability of serving nodes that consist the network or the ability to handle nodes going on and off the network (Tyson, et al. 2013). As submitted earlier that mobile and multimedia information is becoming ubiquitous, mobility needs to be properly handled. ICN technique, through the idea of the neighbor serving action, tends to solve the mobility issues without disrupting the application layer. Once a request has been served and cached on a node, regardless of who subscribed the information earlier, next demand by different subscriber on the same interest are usually served. This however is seen as one of the advantages of vending into the Internet of names of ICN since the DNS and DHCP does not need to be contacted. However, the issue of mobility is also open to research mostly when the cache contents are done off-path.

2. Security

ICN ability to forward cached information requires adequate security feature. Careful responses to interests are coordinated with some security functionality. Just like the traditional Internet, Secure Socket Layer (SSL) that checks the accurate matching of sent and receiving information is needed. When interests are published, the ICN concept gets the match of the senders name and path. If the interest is found in the content store, then the forwarding node through the FIB creates the routing for its delivery. Some security features have already been proposed such as unique key distribution, stale timing, digital signing, encryption, binding, hash key distribution among others (Ahlgren, Dannewitz, et al. 2012), (Ahlgren, D'Ambrosio, et al. 2008), (Jacobson, et al. 2009). To gaurantee the trust worthiness, confidentiality, authenticity and integrity of information objects, information-centric security measures are required.

3. Scheduling

Scheduling defines the form and order to which processes are assigned functions and operations (Abdullahi and Junaidu 2013). In ICN, one will notice that all information as interest and objects need to be forwarded after discovery. This along several forms of information transport has yielded suggestions of scheduling schemes such as the Least Frequently Used (LFU) (Wang, et al. 2013) and First Come First Served (FCFS) as describe in (Jacobson, et al. 2009) to avoid duplicates of interest in the PIT. The proper and acceptable means of scheduling is therefore needed to drive the ICN concept to optimum functioning. From the study by Psaras, Chai and Pavlou in 2012, redundancy and bandwidth could be minimized if the interests are first served to the farthest requested. It is logical that if the subscribers are linear in nature, once the farthest node is served using the Longer Job First (LJF) (Abdullahi and Junaidu 2013), on path nodes and route can therefore cache to use the LCD submitted by (Psaras, Chai and Pavlou 2013), (Psaras, Chai and Pavlou 2012) and (Jacobson, et al. 2009).

The aforemention points are vital and thus require further research to make ICN standardized.

RESEARCH ISSUES

ICN caching with its wide advantages of neighbor providing information as objects and interests is seen to be challenged by some issues. One of the top most among many issues is caching (Ahlgren, Dannewitz, Imbrenda, Kutscher, & Ohlman, 2012). Caching provides the chunk in the network or outside the network with the help of an additional registration or routing information. The issues affecting caching include:

1) Placement policies
2) Replacement policies
3) Fast scan and search
4) Cache device or site positioning

For the purpose of this paper, few of the Placement policies will be mentioned as related to other studies. Popular contents and objects are cached and provided when demanded. Direct mapping of information objects could be refered to as Cache First Object (CFO).

1. Placement policies in ICN are yet to present the optimal and the suitable time for object and interest to be cached. The advantage of CFO is to increase the hit ratio by automatically serving the interest as fast as possible from the content store. The disadvantage of CFO could be seen when the network grows spontaneously. Since cache sizes are always limited, it is therefore necesssary to vend into researching an optimal placement policy in ICN.
2. Replacement policy defines the order of saving and discarding object out of the memory. Least Frequently Used (LFU) has served one of the top most repacement policies. It amounts and counts based on a predictory history of refering an object. In ICN, LFU could as well be computed and predicted from the amounts of interests served in the PIT. Random replacements and Adaptive replacement policies have

also been used in several studies to create random space in the memory and balance between the LFU and MFU respectively.

3. Quick scan within the network is very important as when interest are served. The advantage of proxy cache architecture provides the fastest result. In proxy positioning and cache architecture, the caching is done at the edge in order to closely place to the subscriber (Nagaraj, 2004). When the network grows large, it is logically conclusive that cache nodes placed in the middle (on-path) would serve the best result. For this reason, the Bloom Filter (Bloom, 1970) with its advantage properties of array and probability makes searching and interest serving easier.
4. Cache device and positioning presents the form of caching either on-path or off-path using additional information and devices. Multiplexing as studied in (Huang & Leung, 2008) unarguably saves space, bandwidth and excess visitation to the producer station. It would be most beneficial to ICN to adopt the multiplexing of same request types and multiplex (*mux*) it as one to the PIT. This will reduce the amount of energy needed to send the interests in pieces. Figure-3 presents the cache-enable router (node) placement thereby *muxing* three similar interests from different subscribers.

PROSPECTIVE USE OF BLOOM FILTERS AND MULTIPLEXING (MUXING) IN ICN

Bloom filters

Bloom Filters (Bloom, 1970) is introduced by Bloom H. Burton in 1970 have been used to solve various searching and architectural organisational problems. With Bloom Filters advantage of lesser error through the usage of its hash function and probability that an element or entity does not belong to a set would make interest searching faster. Bloom Filters can be incorporated in the search of Pending Interest Table (PIT). In recent times Distributed Hash Tables (DHT) were used to resolve names and provide a fast look up. Bloom Filters will be demonstrated in the remaining sections of the paper.

Properties of bloom filter

Some of the properties of Bloom Filters include:

1) Fast look-ups during search
2) Returns false positive
3) Easily identify a member of a set or not

For the purpose of ICN, Bloom Filters make a fast search and a look-up by utilizing the probability functions that exist in the filter as pointed out by Crainiceanu in (Crainiceanu, 2013). Multiplexing will be used to test the grouping and membership of interests in the PIT and CS. When the same interests are sent, the properties of the multiplexing will combine the interest as one to lesser the streaming and save bandwidth. Figure 3 presents the pictorial representation of the proposed research direction using the multiplexing for similar group interest sending and Bloom Filters to aid in fast look-ups.

From the Figure, users 1-3 (Subscribers) send out interests to the network. Once the interests get to the first node (router), the content store makes a look-up to see if such interest is stored. On the event that the interest is not found, the interest is then placed in the special pending queue known as the Pending Interest Table (PIT).

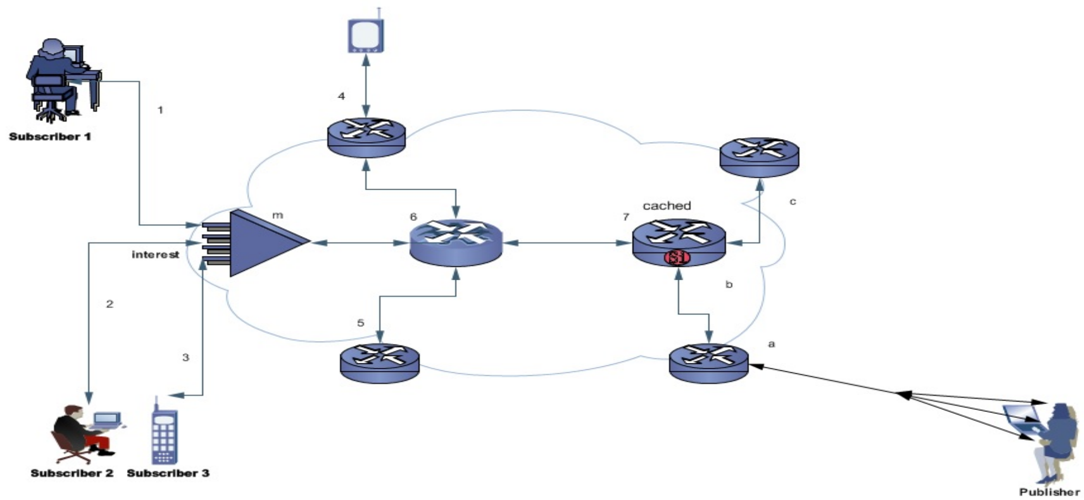

Figure-3. Multiplexing interest.

Subscriber interests are *muxed* from the first point of sending. The multiplexing algorithm work in the way of combining the same interests to minimize the sending and transport costs. This operation helps to minimize the overall bandwidth consumption, lower latency and overall immediate interest look up.

Multiplexing (*muxing*)

ICN adopting a technique of multiplexing interests is open to research of improving the overall message and information dissemination. From Figure-3, the first operation is initialized by Subscriber 1 with the action number 1. A router node ***m*** with the Bloom Filter and multiplexing algorithm is added to the network. The idea of using the Bloom Filter is to enhance the searching and make name resolution easier. Once the interest is placed in the PIT, FIB forwards the interest to the neighbor node with the intention of getting a feedback. Consequently, Subscribers 2 and 3 also demanded for the same data or object by Actions 2 and 3 respectively. Their individual interest is look-up by the filter to confirm if the interests belong to the same set.

The publisher on the other end, send out data objects into the network through action a. Once the interest are forwarded based on the ICN concepts, the interest are removed from the PIT to pave way and space for new interest. As submitted by Psara et al., the case of redundancy needs to be mitigated. Therefore, according to study on coordinated caching in (Psaras, Chai, & Pavlou, 2013) and (Psaras, Chai, & Pavlou, 2012), farther look-ups or interests needs to be served first before the closer nodes. When farther nodes are visited for the first time, there would be no need scheduling on the energy and time consumption to closer nodes.

Scenario test

Consider an example of a populated ICN network with considerable size Z, with U as a set of information such that U={information}, Publisher P with data set elements X. Each element X has a unique interest *i* such that:

$i = 1,2,3,\ldots,n$; that is used to satisfy the client request Y for S as Subscriber. Mapping and matching interest between X_i and Y_i represents the Publisher and Subscriber respectively. Such that:

ICN U= {Information, *i*}, P= $\{X_1, X_2, X_3,\ldots, X_n\}$,

S= $\{Y_1, Y_2, Y_3, \ldots, Y_n\}$. Another Set of routers R are used as the medium of interconnection between the nodes and stations as R = $\{R_1, R_2, R_3, \ldots, R_n\}$.

To apply the fast look up, Bloom Filter (BF) is introduced into the network. When interests are posted by Subscribers as request, the first node R looks into the CS for a match. The name of the object is forwarded to the PIT. The advantage of adding BF here is to serve as a catalyst for efficient result finding after name resolution. As a potential of handling similarities of interest, *Muxing* (Multiplexing and demultiplexing) interest are served in one route. This proves lesser bandwidth consumption, time to retrieve and present object is lessened. *Muxing* is possible when the interests from several Subscribers are ***AND*** or intersected. An intersection of a set A is the result of two or more sets having the same elements appearing in set B and C. Example:

B ***AND*** C = A … (1)

Where B={ 1,2,3,4} and C ={3,2,4,}

Therefore, A={ 2,3,4}.

Using the above result in ICN concepts, similar interests are therefore ***'AND'*** and served as multiplexed.

The benefit of introducing BF as discussed above is to grant fast results of a set member element (interest) or not. Autonomous stations (AS) from Figure 4, is visited to grant a request. All pending interests placed in individual nodes are searched and added to the list. When the result becomes negative, the interest is followed up in another node or station. From Figure 4, Subscriber 1 sends out interest *i*. as Y_i in action 1. It was observed that an intersection of interest exists between Subscribers 1-3. This necessitated the ***mux*** operation of the interests.

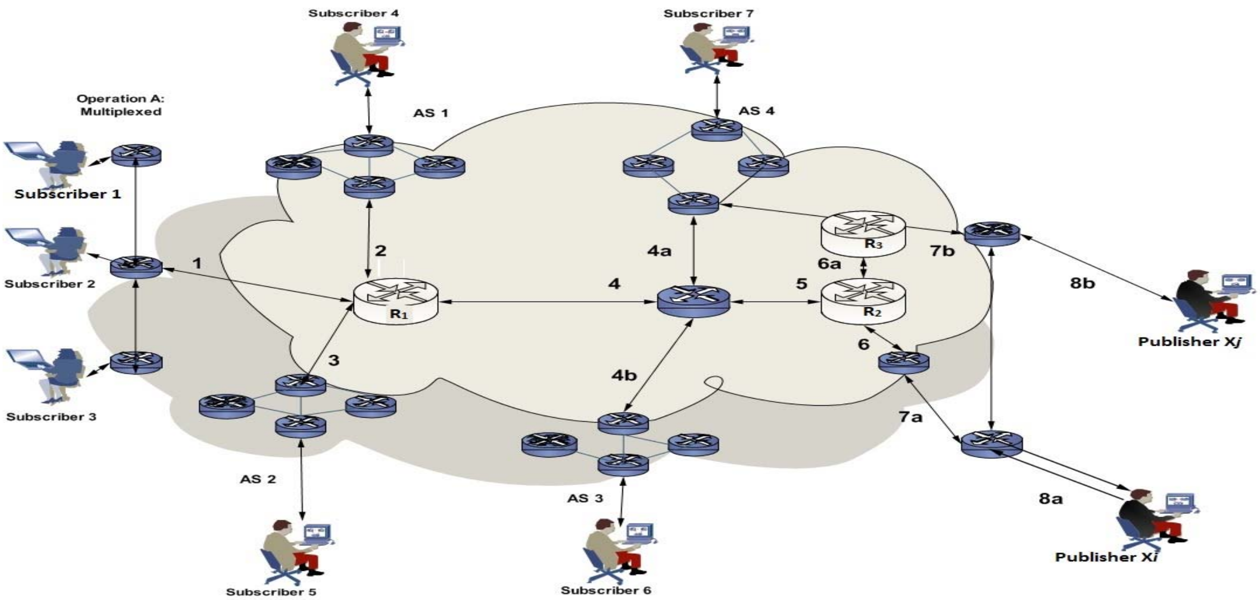

Figure-4. Fast Look-up using Bloom and Multiplexing.

Bloom Filters are in-cooperated into the autonomous stations to provide a union of the interests in all stations. This makes the interest searching a lot more frequent, faster and forwarded easily. From Figure 4, action 1 as interest from subscribers checks the router (R_1) station on-path. R_1 is cache enabled to reduce similar interest serving. When Subscriber 1-3 represented as Y_1-Y_3 sends out interests, action 2 is performed thereby looking unto the PIT to grant the interest. However, AS 2 demands some information *i* using the FIB. Assuming publisher X_i sends out a match of the requested information *i* , eight or more actions will be needed to benefit from the long routed network. Therefore, according to (Psaras, Chai, & Pavlou, 2013) the longer interest require more traveling parts for delivery, a Longer Job First (LJF) scheduling algorithm (Abdullahi & Junaidu, 2013) needs to be used to access the interest/object source. When interest/objects are requested by Subscriber 1-3, the resulting objects will be cached on Router R_2 and R_3 on a reverse action 6 and 6a respectively.

The potential use of introducing BF thus reduced the interaction and interest look-ups within each AS. Intra-nodes communications at AS are therefore used to agglomerate the results of available interests in each station of AS 1, AS 2, AS 3 and AS 4. Intra-cache (Wang, Zhang, & Bensaou, 2013) in AS makes the filter results easily obtainable and saving the excessive bandwidth that would have been used individually using inter AS searches. The concepts of caching on-path of R_1, R_2 and R_3 minimizes the cache object redundancy which are always seen as a major challenge in caching concepts (Psaras, Chai, & Pavlou, 2013), (Wang, Zhang, & Bensaou, 2013). Researches have been made in different studies to avoid caching objects with less popularity, or least recently used items. Caching objects that do not need frequent request consumes the bandwidth and the cache memory spaces. Blooming interest is thus seen as a predictor solution to the problems of redundancy of objects on-path.

Scenario description

Suppose a populated network is set as Z:

1) Subscriber S request an interest Y_i, such that *i* is the unique identity of the interest
2) *i* is placed on the PIT of the first node
3) **Look-up** from the first Router node R_1 to get a match using Bloom Filters
4) **serve** the interest *i* from the Subscriber Y_i
5) **else** move interest to the next router node
6) Check to find the similar interests by intersecting
7) **Combine** the interest (*mux*) as one to save space and bandwidth

8) Check from the neighbor AS, the results in Bloom Filters
9) **Look** into the entire network for the interest
10) Publisher X_i sends an interest *i* , into the network
11) **Cache** *i*, for the subsequent similar request on the first router node (R*i*)
12) **Send** resulting interest/object via the same route
13) **Cache** results for next request

Advantages of using Bloom Filters and Muxing

Prospective inclusion of the Bloom Filter and muxing practice on the ICN will yield better services in terms of message dissemination. Fast lookups are initiated intra node while the searching of a member element (object) belonging to an autonomous site is done between stations.
Some of the advantages envisioned in the ICN include the following;

1) Overall network latency is improved
2) Delay jitter is lower than non-Bloom Filter and mux ICN architectures
3) Objects and interest finding is much faster
4) High inter-node communications to and different stations are minimized
5) Object redundancy is mitigated
6) Improved cache management through selected node caching
7) Easily scalable
8) Enhanced mobility due to the bloom element membership
9) Better security
10) Easy name resolution through Bloom Filter probability functions
11) Higher throughput by AS specification

CONCLUSION

Bloom Filter and muxing are key concepts of data structures, architectures, and information system topics of great advantage. In this paper, the envisioned ICN as an optional paradigm to the traditional Internet was discussed. However, the main idea of ICN is channeled through fast delivery of information on the Internet using the mid-neighbor as an option. Caching therefore is the building block of the ICN architecture. Research open issues always remain in caching through the idea of fast lookups, what to cache? And how to place and replace the elements with lesser redundancy and effective bandwidth usage. So Bloom Filter as proposed in this paper is to aid in minimizing high linear searches that add high latency to the network. Categorizing and grouping interests in AS are the contribution in order to improve caching. Muxing as described is used to collate similar interest of objects thereby reducing the amounts of time and memory spaces available. Once interests are intersected, it becomes lesser and easier to save on the PIT for forwarding.

Therefore, ICN caching and message dissemination can be improved using Bloom Filter technology. The future works can further be pointed out by identifying the optimal sizes of the filters in each AS. Another issue that was not fully considered was the differences in interest from other stations without muxing. All of these issues are still open to further studies and testing. The paper thus concludes by exposing the potentials of Bloom Filters and muxing to improve caching in ICN.

ACKNOWLEDGEMENTS

This research is one of the Malaysian Government funded projects under Ministry of Higher Education (MOHE) Long-Term Research Grant (LRGS) with reference number LRGS/TD/2011/UKM/ICT/02.

REFERENCES

Abdullahi I. and Junaidu S. (2013, August). Empirical Framework to Mitigate Problems in Longer Job First Scheduling Algorithm (LJF+CBT). International Journal of Computer Applications, 75, 9-14.

Ahlgren B., D'Ambrosio M., Marchisio M., Marsh I., Dannewitz C., Ohlman B., et al. (2008). Design Consideration for a Network of Information. ACM CoNEXT Conference, CoNEXT'08 (pp. 66:1-66:6). NY, USA: ACM.

Ahlgren B., Dannewitz C., Imbrenda C., Kutscher D. and Ohlman B. (2012). A Survey of Information-Centric networking. IEEE Communication magazine, Special issues on Information-Centric Networking, 1-24.

Bloom B. (1970, July). Space/Time trade offs in hash coding with allowable errors. Communication, ACM, 13, 422-426.

Crainiceanu A. (2013). Bloofi: A Hierarchical Bloom Filter Index with Applications to Distributed Data Provenance. Proceedings of the 2Nd International Workshop on Cloud Intelligence (pp. 4:1--4:8). New York, NY, USA: ACM.

Dannewitz C., Herlich M. and Karl H. (2012). OpennetInf-prototyping an Information-centric architecture. Local Computing Workshops (ICN Workshops) (pp. 1061-1069). IEEE.

FP7 EU Project, P. (2010). Retrieved August 20, 2014, from http://www.fp7-pursuit.eu/

Huang C. E. and Leung C. (2008). Downlink mixed-traffic scheduling with packet division multiplexing. Proceeding of the 3rd ACM workshop on performance monitoring and measurement of heterogeneous wireless and wired networks (pp. 165-172). New York, USA: ACM.

Jacobson V., Smetters D. K., James D. T. and Michael F., P. (2009). Networking Named Content. ACM CoNEXT '09, 1-12.

Kubiatowicz J., Bindel D., Chen Y., Czerwinski S., Eaton P., Geels D., et al. (2000). Oceanstore: An architecture for global-scale persistent storage,. SIGPLAN, 35, 190-201.

Nagaraj S. (2004). Web Caching and its application. London: Kluwer Academic Publishers.

Psaras I., Chai K. W. and Pavlou G. (2013). In-Network Cache Management and Resource Allocation for Information-Centric Networks. IEEE Transactions on parallel and sitributed systems, 1-11.

Psaras I., Chai W. K. and Pavlou G. (2012). Probablistic in-network caching for Information-centric networks. Proceeding of the Second Edition of the ICN workshop on Information Centric Network, ICN"12 (pp. 55-60). ACM.

Tyson G., Sastry N., Cuevas I. R. and Mauthe A. (2013, December). A Survey of Mobility in Information-Centric Networks. ACM Communication, 56, 90-98.

Wang J., Zhang J. abd Bensaou B. (2013). Intra-AS cooperative caching for content-centric networks. Proceedings of 3rd ACM SIGCOM Workshop on Information-centric networking, ICN'13 (pp. 61-66). New York: ACM.

Wang Y., Li Z., Tyson S., Uhlig S. and Xie G. (2013). Optimal cache allocation for content-centric networking. IEEE International Conference on Network Protocols.

Zhang H., Wen Y., Xie H. and Yu D. (2013). DHT Platforms in Distributed Hash Table. (Springer, Ed.) Springer Briefs in Computer Science, 23-38.