What Benefits Does NDN Have in Supporting Mobility

Xiaoke Jiang, Jun Bi, You Wang
Institute for Network Sciences and Cyberspace, Tsinghua University
Department of Computer Science and Technology, Tsinghua University
Tsinghua National Laboratory for Information Science and Technology
Email:{jiangxk10, wangyou10}@mails.tsinghua.edu.cn, junbi@tsinghua.edu.cn

***Abstract*—Inspired by forwarding hint and previous IP mobility solutions, we adopt forwarding hint, which intent to solve scalability problem, to support producer mobility. In this paper, we point out how those new elements, such as cache, content-oriented security, content-centric data transmission, benefit mobility. We implement a prototype to analyze the benefits that NDN has in supporting mobility. Our analysis and evaluation conclude that during mapping updating delay following mobility, only popular contents can get benefits from caching while unpopular contents gain little. What's more, only if the content is able to be accessed by partial consumers, caching would amplify the benefits and extend receivers to the rest of consumers, which has significant contribution during routing convergence or mapping updating delay after mobility happens.**

***Index Terms*—NDN, CCN, Mobility**

I. Introduction

Named Data Networking (NDN) is a promising future network architecture. As its name implies, every piece of data, rather than host, is named in NDN, which makes Content the first-class citizen of the network. NDN has changed the network architecture in four perspective: 1) Network layer uses the names from the application directly, therefore, name-based routing, content cache and multicast are naturally supported on network layer; 2) NDN secures data directly instead of container or channel; 3) NDN adopts a pull schema only to transmit data; 4) Forwarding plane of NDN is stateful, compared to that of IP.

There are two kinds of datagrams in NDN: Interest and Data. Application, which requires specific data, is called consumer. And application which could originally provides the requested data is called producer. Consumer sends an Interest containing the name of the requested data into the network; NDN routers employ name-based routing to forward Interest and keep state of those forwarded Interest by PIT (Pending Interest Table); The Interest stops until it arrives at producer, or an immediate router which has cached the requested data before. Then the Data follows the reverse path of Interest and backs to the consumer.

Anyway, NDN moves to Content-Centric data transmission from Host-Centric. Applications request what they want from the network, which resembles a unified content bucket instead of end-to-end pipeline.

The advantages of the the new architecture are significant, it decouples requests and responses in both space and time [1], which brings some remarkable benefits to support mobility, for example:

- Security is decoupled from channel and location [2]: Today's IP mobility solutions take a lot of efforts on security, such as Return Routability Procedure in Mobile IPv6[3], key exchange in HIP[4]. But in NDN, security is no longer associated with channel and location, mobility does not lead any security challenge, that is why NDN does not require extra support or patch to guarantee the security in mobility.
- Consumer Mobility is easily supported: Here consumer mobility refers to consumer moves to new location. After a consumer moves to new location, re-expressing the unsatisfied Interests could cover the mobility. Since mobile devices, such as smart phones, Persona Assistant Devices (PADs) serve the role of consumer most of the time, mobility problem is erased greatly.
- Cache provides contents during routing convergence or mapping updating delay. Consumers may also get replies from cache in the network, no matter the producer is mobile or not. When producer mobility happens, the cache can provide requested contents, which partially support producer mobility.
- Mobility between different separate branches of networking, such as 3/4G, WiFi, Delay-Tolerant Network (DTN), Mobile Ad-Hoc Network (MANET): mobility between different branches of networking is quite common scenario. NDN focus on contents, channel change is covered by routing and forwarding abstraction[2], thus, the mobility between different branches of networking does not need special mechanism to handle.

Consumer mobility is easily solved in NDN, however, producer mobility is still remained obstacle. After producer moves to a new location, routers should be capable of forwarding the corresponding Interests to the latest and correct link. In most cases, we reply on routing system update and convergence. It is obvious that routing system surely suffer from scalability and convergence problem, since namespace of name is much larger than that of IP address.

Lixia Zhang et al. introduced forwarding hint into NDN[5] to solve routing scalability. In their approach, the ISP prefix is carried in a new filed, called forwarding hint in the Interest packet. Application names in the Interest and Data are intact and are visible to routers, allowing Interest to be satisfied. The forwarding hint is used in routing table lookup when Interest packets are being processed by routers. It is a hint provided by then end host to routers about where to forward the Interest. The forwarding hint is nothing more but an indication of where the requested content may reside[5]. There is a mapping system to translate name to forwarding hint.

Inspired by forwarding hint and previous IP mobility solutions, we adopt forwarding hint to solve producer mobility. We implement a prototype on NS-3[6], furthermore, we present an analysis on the benefits that NDN has in supporting mobility, and influenced factors in the new scenario. Our analysis concludes that only popular contents can get benefits from caching while unpopular contents gain nothing; and during the mapping system updating delay, only if the content is accessible by consumers, caching would amplify the benefits and extend receivers to other "blind" consumers.

This paper is organized as follows:In Section II we introduce the forwarding hint based solution. The we analyze and evaluate the benefits of NDN in supporting mobility in Section III and IV, respectively. In Section V, we give and brief introduction on NDN and its mobility research. It's quite short due to the space limitation. Finally, Section VI concludes the whole paper.

II. Methodology

A. Forwarding Hint to support Producer Mobility

As we can conclude method to solve producer mobility, we get the idea of forwarding hint[5], where forwarding hint is used to solve NDN routing scalability. In this approach, the ISP prefix is carried in a new filed, called forwarding hint, or forwarding alias, in the Interest packet. Application name is still the main identifier of Interest and Data. Routers look up Data in the CS with application name, then look up PIT entry with application name, and use the application name as its first choice to forward Interest. However, if all previous operations do not get a hit, the forwarding hint is used to look up in Forwarding Information Base (FIB) when Interest packet are being processed. It is a hint provided by then end host to routers about where to forward the Interest. Lixia Zhang et al. defined forwarding hint as nothing more but a hint of where the requested content may reside.

Here we extend the namespace of forwarding hint from ISP to any name prefix of content producer. For example, AT&T provides Internet service for a company, say dropbox. AT&T can announce its forwarding hint /att to Default Free Zone (DFZ), while the dropbox can only announce its forwarding hint /att/dropbox to AT&T's network. A producer located inside the dropbox network could announce name prefix in dropbox's local network. And routers could forward them correctly. For example, a name prefix /skype/user/john announced inside dropbox's network with the forwarding hint /att/dropbox, it is forwarded based on the forwarding hint prefix /att in DFZ, and /att/dropbox is used inside AT&T. Finally, when the Interest enters dropbox's network, the name prefix is used directly.

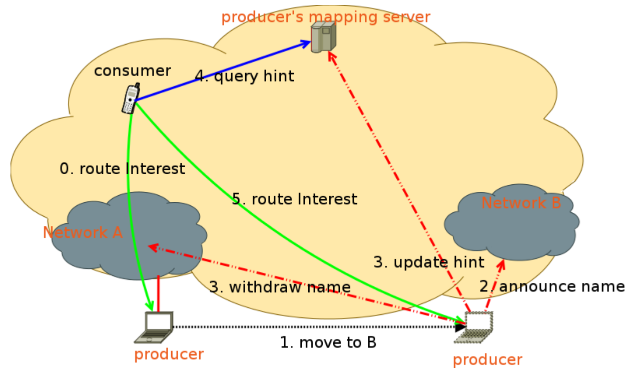


Fig. 1: Interaction Process

B. DNS-like Mapping System

Since there are two identifiers in Interest, there must be a mapping system to do the translation between content name and forwarding hint. Since DNS is most successful mapping system, here we use a DNS-like mapping system. Note that the mapping servers is not maintained by ISPs. The mapping servers is owned by application provider company, such as dropbox and skype, etc. And it is probably located in another network, as shown in Figure 1.

After a producer moves to a new network, it should announce its name in the new network, withdraw its name from the previous network, and update its forwarding hint in mapping system. Therefore, consumers can query the hint, and Interest can be routed to the right network, as shown in Figure 1.

For the whole process, announcement, withdrawal and updating lead to delay in practice, which has significant influence to the performance.

III. Benefits That NDN has

After adopting forwarding hint to solve mobility problem, we step further in this paper. We try to analyze the benefits that has in support producer mobility.

A. Popularity differences of contents

Different contents have different popularity, therefore, cache hit ratio is also different. After mobility, if Interest is satisfied by cached copy, then consumer does not influenced. To some extend, cache save mobility. However, Interests are not 100 percents satisfied by cache. It's more likelihood that popular content could gain more benefits.

B. Applications Differences

Different applications need different services. For example, some applications needs reliable transmission, therefore, consumers keep re-issuing the unsatisfied Interests to request contents. On the other hand, some applications just keep asking for latest contents, even if there is packet loss. The different behaviors lead to different cache hit ratio and gain different benefits.

C. Mapping Updating Delay

After producer moves to a new location and before it updates its record in mapping system, all consumers cannot get the correct forwarding hint. We call this *blind period*.

There is a special period when some consumer get the latest forwarding hint while some get the answer from cache which is not correct. We call this period *partial blind*. Compared to IP, partial blind period in NDN is much more special. In partial blind period, some consumers are able to introduce the requested contents to the network, and those blind consumers could also to get contents from cache.

IV. Evaluation

A. Implementation and Simulation Scenario

We have implemented a prototype on ns-3. We add forwarding hint field into Interest packet and implement a mapping system. The underlying layer protocol (Layer-2) can encode and decode the new field. The application name $\rightarrow$ forwarding hint mapping system is implemented as surveyed in [7].

We also implement a content requester which requests content according to a Zipf-like distribution [8]. The probability of each consumer to request the i*th* content is P(i). P(i) follows the Pareto-Zipf distribution :

$$P(i) = \frac{\Omega}{(i+q)^{\alpha}}$$

Ω is normalization constant, making $\sum_{i=0}^{9999} P(i) = 1$. Here α and q are 0.7, $i \subset [0, 9999]$

The process of simulation is divided into stages. At first, all cache is empty and all consumers know the exact mapping of all the contents, so all requests can get data replies and fill the cache space, which is named *Normal Stage*. Then, we make one of the content producers move to new location, however, none of the consumers is aware this location, which is named *Blind Stage*. In order to evaluate mapping update delay, there may also exist *Partial Blind Stage*, in which *some* consumers know the new location while others don't; Finally, the mapping system finishes updating, leading to a new stable state with redistributed content in cache, which is named *Redistributed Stage*.

Each consumer requests the contents according to the requesting probability above. We treat 2000 continuous request sequences as one *"turn"*. Each stage has 10 turns, except for the partial blind stage, which may include several sub-stages depending on the speed of mapping convergence. We compared results using different cache sizes and cache updating policies. Default cache size is 100 contents, which is 1% of all the contents, and cache updating policy is Least Recently Used (LRU).

We use a real intra-AS network as the topology for simulation, as shown in Figure 2, which includes 21 routers numbered from 1 to 21. There are 5 content producers who connect with routers 2, 6, 10, 12 and 16 respectively. The mobile, a moving producer, connects with router 12. There is one content consumer connecting each router, which is representative of all the end hosts connecting to that router. There are 10000 contents numbered from 0 to 9999. Contents are distributed equally among the producers.

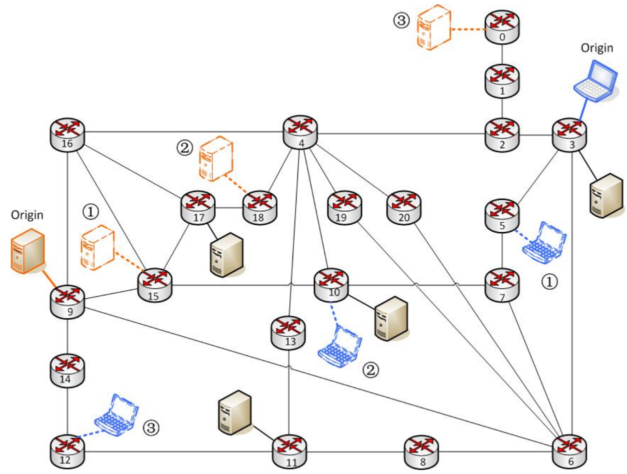


Fig. 2: A Real intra-AS Topology Used in Analysis: The yellow desktop serves the producer mobile, while the blue laptop serves consumer mobile. They move from origin place to different locations ①, ②, ③ step by step. The locations are different distance form origin place in the topology.

We use Caching Ratio (CR), which donates to ratio of Interests satisfied by cached Data every turn to measure the performance. In our scenario, CR can infer number of satisfied Interests in blind stage, since no Interests is routed to the right place during blind stage; and in partial blind stage, CR's increase represents the magnification benefits of NDN architecture.

The stages of mobility which can be easily identified by changing of caching ratio. Before the mobility happens, we assume the network is in a stable state, which means caching ratio fluctuates in a small range. After mobility happens, mapping updating delays follow, during this stage none or some consumers are aware of the new location of the mobile. We call it blind or partial blind stage, in which requests are forwarded to the outdated location.

Furthermore, we explore the benefits gained from NDN architecture. This depends on a lot of factors, such as content popularities, moving distance, caching size and updating policy. What's more, the benefits are also influenced by application traits, such as retransmission, on-demand, or instant.

B. Popularity of Moving Contents

Line 1 (Hottest) shown in Figure 3a represents the contents from number 1 to 336, and

$$\sum_{i=0}^{336} p(i) = 0.333$$

Line 2 (Middle) represents the contents from 337 to 2274, whose accumulative total of requesting probability is also 0.333. Line 3 (Coldest) represents the remaining contents.

Moving contents of Line 1 (Hottest) shown in Figure 3b represents contents numbered from 0 to 336, and Line 2

(Middle) represents contents from 5000 to 5336, while Line 3 (Coldest) represents contents from 9663 to 9999.

If the contents are hot enough, they can meet a lot of requests only with cache. What's more, this phenomenon can last for a long time. We extend the blind stage several times longer to find that Hottest-CRoMC stay near 60% for the hottest data (Top 3.36%). It seems the network (cache) works as a storage of the hot contents.

However, for the unpopular contents, cache helps little when consumers are in the blind stage. aCRoMC drops gradually until it reaches zero.

In the blind stage, a request cannot retrieve moving contents from content producer, and the copy of those contents in the cache may be replaced by background requests without the contents being retrieved. Thus, average Caching Ratio (aCR) of those moving contents will drop as shown. aCR reaches stability soon, meaning that some requests can be met though the original content is not aware by the network, even if the mapping system delay is really long (which may be caused by breakdown of host or long distance mobility).

Hottest Contents have an enormous amount of influence to the aCR because of their occupation of majority cache storage. Our detailed data shows that the hottest 1% contents fill out 60% cache space of all the network, and hottest 10% content fill out 94%.

We can also find that, in the blind stage, the cache distribution suffers the most dramatic changes. That's why dots of blind stage *cover* those of Normal and Redistributed Stage as shown in Figure 3c: content distribution in Blind stage is most out-of-order, due to the influence of mobility.

Cache serves the hot contents as storage, which greatly remits the negative influences of a mobility event, while unpopular contents can't be so lucky. For those unpopular contents, aCR is very low and most of request must reach to the original content location, which is just the same as IP schema. In this case, NDN adds little help to their mobility.

C. Application

As analyzed in [9], we classify information traffic types based on two characteristics: a)reliable vs. unreliable transfer and b) realtime vs. on-demand delivery

1) Reliability: Generally, retransmission is the most important solution to implement reliable transmission. Retransmission, means re-issue in NDN. In our simulation, if the first request fails, re-issue will continue until the content returns, or the re-issue trial reaches 3 times. 4 lines in Figure 4 cover the simulation process with and without re-issue, but the lines of two groups are highly overlapped. This means retransmission has little impact on the mobile scene. This simulation works under ideal conditions, which rules out the failure of network layer operation and focus on the influence of mobility.

In this case, retransmission adds little improvement for mobility. If the cache can't help the first request, it can't help the other re-issue, under producer mobility scenario. However retransmission works well under the consumer mobility scenario.

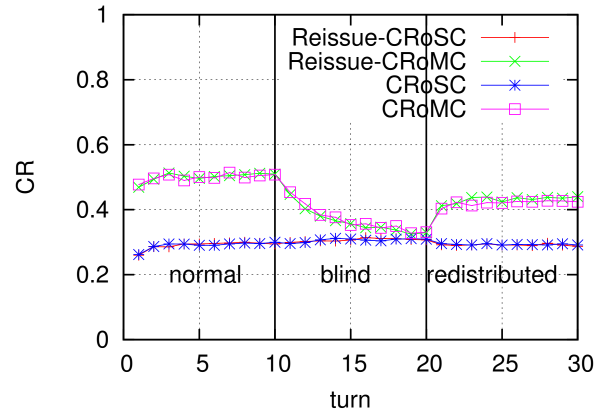


Fig. 4: Different Applications. no retransmission vs. retransmission. Reissue means application reissue Interest when privous Interest timeouts. The whole moving event is separated into there stages: normal, blind and redistribution.

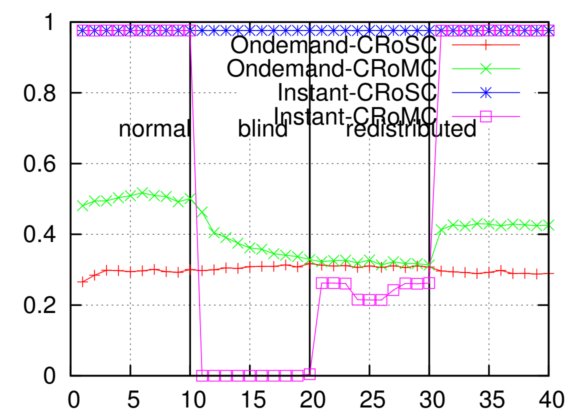
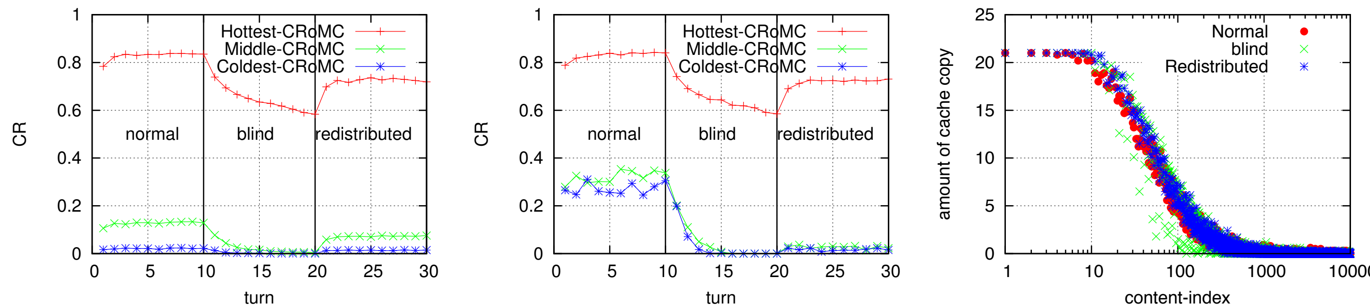


Fig. 5: Different Applications. Real-time vs. On-demand applications. Ondemand donates to applications like ftp, web, which request data follows the zipf-like distribution. Instant donates to applications like video conference, request newest data. The whole moving event is separated into three stages: normal, blind and redistribution.

2) Real-time vs. On-demand: The other characteristics are realtime and on-demand. Realtime application requests do not follow the probability distribution above but issue the newest data every time. For example, all consumers, which play roles of participators of a video conference request the newest data at the same time. aCR is extremely high, meaning that most of the participators can't retrieve contents from cache, which highly benefits realtime application.

However, the blind stage is a disaster. In which case, the newest contents hasn't been cached, and consumers are sending the request to the out-date location, as shown in Figure 5. This can be highly improved only if one consumer updates the location information successfully, as shown in partial blind stage. In this case, only the consumer connecting to the same router with the mobile will update its mapping system, but 25% of the requests retrieve contents from the cache.



(a) Contents Are Classified by Cumulative Probability (b) Contents Are classified by Amount (c) Cached Contents Distribution in Different Stages

Fig. 3: Content Popularity: To be short, we use some abbreviation. aCRoMC denotes to "average Caching Ratio of Moving Contents". X-axis label "turn" donates to number of sent Interests, every 2000 Interests are treated as one turn.In figure (a), the sum request probability of hottest contents are 33.3% ($i \subset [0, 336]$), so does middle contents and cold contents. ($i \subset [337, 2274]$ and $i \subset [2275, 9999]$ respectively). In figure (b), The sum amount of hottest contents is 337 ($i \subset [0, 336]$), so does middle contents and coldest contents. ($i \subset [5000, 5336]$ and $i \subset [9663, 9999]$ respectively)

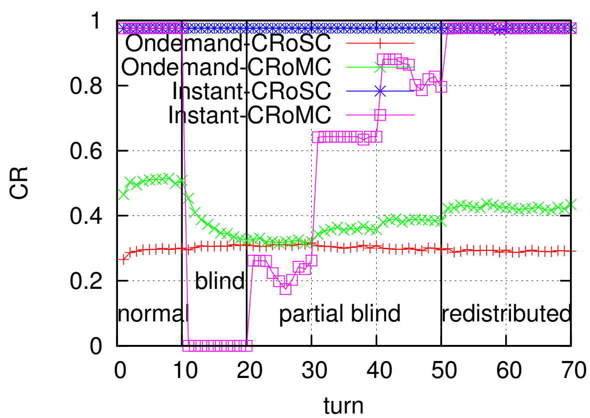


Fig. 6: Mapping Updating Delay. The whole moving event is extended to four stages: normal, blind, partial blind and redistribution. Nodes become aware of the new location one by one in partial blind stage, which is a continuous process.

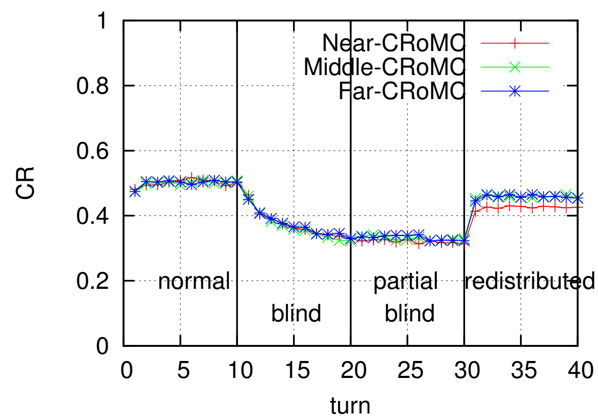


Fig. 7: Distance and Partial blind. Near donates to that producer moves from origin place to ① in Figure 2; Middle donates to Location ② and Far donates to ③

D. Mapping Update Delay

When a producer moves to the new location, mapping system must update its location information. To handle all the request from the whole Internet, the mapping system must be implemented with a distributed method. Thus, there is a problem of updating delay. Some consumers get the new location, who are called *"Smart Nodes"*, while some others are not. There are more smart nodes over time until all nodes are smart; then, the updating finishes successfully.

We divide the sub-stage of updating convergence by distance from the mobile's new location to consumer's location. The first sub-stage is totally blind; then, the consumers under the same router with the mobile (at new location) receives the correct mapping result. Then, consumers cover 1 hop distance to the mobile, then 2 hops, until all consumers are covered.

In the blind stage, aCRoMC will drop and then increase step by step while updating convergence move towards to the end, until we reach the stable stage of the new topology no matter what kind of the application.

E. Moving Distance and Cache Settings

As shown in Figure 7, moving range has no influence to aCR in the blind stage. aCR drops in the blind stage and finally, reaches the new stable state in the redistribution stage. The new aCR depends on the new topology, and not the distance between the mobile moves.

We also explore the influence of cache settings, including cache updating policy and cache size as shown in Figure 8. Because of different cache settings, the specific value of CRoMC will be different, but we conclude that they all show the same features. Our detailed comparisons also prove this.

V. Related Work

A. NDN

In this section we introduce the NDN background and the mobility solutions in IP network.

Content-Centric Networking (CCN) [10] or Named Data Networking (NDN) [11] adopts a pull model when retrieving contents and completely abandons the way of the traditional end-to-end model. Interests for contents are sent by data

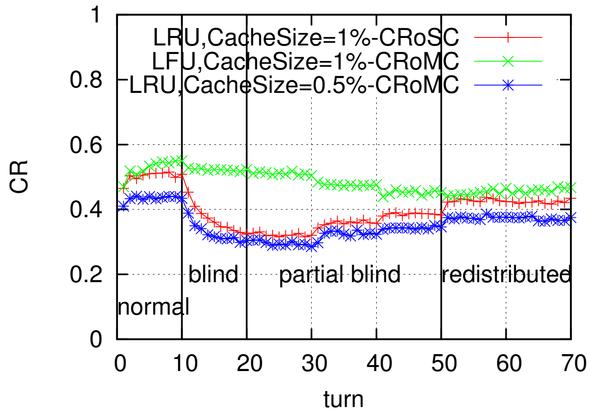


Fig. 8: Influences of Caching Replacement Policies and Cache Size. CacheSize is measured by number of contents. LRU (Least Recently Used) and LFU (Least Frequently Used) donate to caching replacement policies.

consumers and routed by content names within them. Some Interests may get corresponding Data in the CS of immediate router, whilst others will finally reach the producer.

[2] presents a new perspective on mobility support in NDN, especially the separate branches of networking research, such as DTN, MANET, etc. This paper argues that NDN helps to integrate different networks into mobility support. And when the channel changes (a new kind of mobility), data transmission keeps working.

B. Mobility Solution

Ravindran et al[12]. handled mobility in an NDN network seamlessly through a control plane that enables mobile node's mobility irrespective of it being a consumer, producer or both, and proposed three cross-layer network-assisted seamless mobility schemes which leverage the features of NDN including location/identity split, in-network caching, and publish/subscribe paradigm.

Tyson et al[13]. explored the world of ICN, looking at how a shift from host-centric to information-centric design principles could support greater mobility in the future Internet, and discussing a number of challenges of ICN, especially the ability to handle the scalability challenges of increasing numbers of content items and providers.

VI. Conclusion

In this paper, we point out the advantages of the new elements, such as cache, content-oriented security, content-centric data transmission, in supporting mobility. Then we adopts forwarding hint to support producer mobility in NDN, and furthermore analyze the benefits that NDN has in supporting mobility. Our conclusions include:

- Popular contents gain remarkable benefits from NDN architecture. Under our experiment environment (topology, cache size, cache updating policy and requesting probability), 30% requests of the hottest 3.36% contents can get what they want during blind stage. However, unpopular contents get little.
- During updating delay of mapping system, only if the content is accessible by consumers, cache system would amplify the benefits and extend receivers to those blind consumers.
- Producer mobility has different influences to different applications. Re-expressing Interest makes little sense for producer mobility. Furthermore, real-time application is easier to be affected by mobility than on-demand application.

VII. Acknowledgement

Supported by National Science Foundation of China under Grant 61073172, Program for New Century Excellent Talents in University, and National Basic Research Program ("973" Program) of China under Grant 2009CB320501

References

[1] A. Ghodsi, S. Shenker, T. Koponen, A. Singla, B. Raghavan, and J. Wilcox. Information-centric networking: seeing the forest for the trees. In *Proceedings of the 10th ACM Workshop on Hot Topics in Networks*, page 1. ACM, 2011.

[2] Zhenkai Zhu, Alexander Afanasyev, and Lixia Zhang. A new perspective on mobility support. http://named-data.net/wp-content/uploads/TRmobility.pdf, 2013.

[3] David Johnson, Charles Perkins, Jari Arkko, et al. Mobility support in ipv6, 2004.

[4] Robert Moskowitz. Host identity protocol architecture. 2012.

[5] Alexander Afanasyev, Cheng Yi, Lan Wang, Beichuan Zhang, and Lixia Zhang. Scaling ndn routing. Technical report, NDN Technical Report NDN-0004, 2012.

[6] ns-3. http://www.nsnam.org/.

[7] L. Zhang, R. Wakikawa, and Z. Zhu. Support mobility in the global internet. In *proceedings of the 1st ACM workshop on Mobile internet through cellular networks*, pages 1–6. ACM, 2009.

[8] L. Breslau, P. Cao, L. Fan, G. Phillips, and S. Shenker. Web caching and zipf-like distributions: Evidence and implications. In *INFOCOM'99. Eighteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, volume 1, pages 126–134. IEEE, 1999.

[9] C. Tsilopoulos and G. Xylomenos. Supporting diverse traffic types in information centric networks. In *Proceedings of the ACM SIGCOMM workshop on Information-centric networking*, pages 13–18. ACM, 2011.

[10] V. Jacobson, D.K. Smetters, J.D. Thornton, M.F. Plass, N.H. Briggs, and R.L. Braynard. Networking named content. In *Proceedings of the 5th International Conference on Emerging Networking Experiments and Technologies*. ACM, 2009.

[11] L. Zhang, D. Estrin, J. Burke, V. Jacobson, J.D. Thornton, D.K. Smetters, B. Zhang, G. Tsudik, D. Massey, C. Papadopoulos, et al. Named data networking (ndn) project. Technical report, Tech. report ndn-0001, PARC, 2010.

[12] Ravishankar Ravindran, Samantha Lo, Xinwen Zhang, and Guoqiang Wang. Supporting seamless mobility in named data networking. In *Communications (ICC), 2012 IEEE International Conference on*, pages 5854–5869. IEEE, 2012.

[13] Gareth Tyson, Nishanth Sastry, Ruben Cuevas, Ivica Rimac, and Andreas Mauthe. A survey of mobility in information-centric networks. *Communications of the ACM*, 56(12):90–98, 2013.