Journal of Physics: Conference Series

PAPER • **OPEN ACCESS**

Multi-attribute Probability Caching Algorithm in Named Data Network

To cite this article: Li Zhang and Qiang Zhang 2020 *J. Phys.: Conf. Ser.* **1570** 012010

Multi-attribute Probability Caching Algorithm in Named Data Network

Li Zhang[a], Qiang Zhang *

Department of Informatics Beijing University of Technology Beijing, China

[a] zl_hlj@126.com

[*]438563477@qq.com

ABSTRACT: Since entering the new century, Internet technology has developed rapidly and has become an important infrastructure in today's society. In recent years, with the popularization of optical fiber applications, the network bandwidth has been greatly improved, and the growth rate of network traffic has also shown explosive growth. The traditional TCP/IP network has exposed many serious problems in the face of increasing traffic. In particular, the issues of scalability, reliability, security, flexibility and mobility are more prominent. In order to solve the shortcomings of the existing Internet architecture, the current domestic and international schemes mainly include "improvement" and "revolution". Improvement is the gradual improvement of the deficiencies of the existing Internet system. However, this method cannot essentially solve these deficiencies, and it is difficult to meet the needs of future network development. The revolution is to redesign the information-centric network as the Internet system structure, completely changing the existing Internet system in terms of scalability, security and mobility, in order to adapt to the needs of future social development. Named-Data Networking (NDN) in many information-centric network architectures is a network architecture with excellent research and development. This article focuses on the research of NDN network caching technology. In terms of caching decisions, a multi-attribute probabilistic caching algorithm is proposed, which can effectively reduce the cache of redundant data in routing nodes in NDN networks and improve the utilization of routing node cache space, reduce network delay, increase the cache hit rate of routing nodes, and improve the overall performance of the NDN network.

CCS Concepts:• Future network→Information center network • Named data network→Cache optimization.

1. INTRODUCTION

NDN is an excellent system architecture in information-centric network construction. There are one or more routing nodes between producers and consumers in the NDN network. The routing nodes are responsible for the forwarding of Interest packets [1], the backtracking of Data packets and Data caching. Each routing node contains three tables, namely the Pending Interest Table (PIT) [2,3], Forwarding Information Base (FIB) [4] and Content Store (CS) [5]. The PIT records the upstream information sent by all Interest packets. When the Interest packet reaches the routing node, the PIT table is checked. If the Interest information exists in the PIT table, the source port of the Interest is recorded. If there is no Interest information, the Interest table entry is created. The table information traces the Data packet correctly. When the Data packet traces back from the corresponding port, the PIT entry matching the data is deleted. FIB stores the <prefix, interface list> ancestor [6], uses the longest matching query of the value of the Interest name attribute, and forwards the Interest packet from the potentially required Data port. The CS is similar to the memory cache in the IP router, and caches the data traced back to the routing node. The NDN network is an information-centric network, and the data cached in the routing nodes can realize the reuse of the same data. Reasonable use of caching strategies can greatly improve network performance, reduce network latency, save the use of routing node cache space, and effectively relieve network bandwidth pressure. The LCE (Leave Copy Everywhere) [7] caching strategy is the default caching strategy currently used in NDN networks. The core of this caching strategy is that each routing node will cache the backtracked Data. At present, for the research on improving the performance of NDN network through design cache optimization, the more classic cache optimization strategies are LCD (Leave Cache Down) [8], MCD (Move Cache Down), and Prob (Probability Cache) [9]. LCD cache strategy is that when an Interest request hits in a node's CS table, the next hop of the node will cache the data. The MCD cache strategy is based on the LCD. The difference is that MCD will delete the data of the node Prob is a random probability caching strategy, that is, when the Data packet traces back to each routing node, whether caching occurs is determined by the set probability. The selection of different caching strategies in the NDN network has different effects on the performance of the NDN network. Although the current default LCE caching strategy of the NDN network is simple, this strategy caches Data at each routing node, causing the same Data redundant cache to waste cache resources in different routing nodes. This caching strategy will also cause the data cached in the routing node, and the speed of cache replacement will be accelerated, so that important Data cannot be kept in the routing node for a long time. The LCD and MCD algorithm is that when Interest hits Data at this node, it will only cache Data at the next routing node. The advantage of this design is that it can reduce the cache of the same redundant Data at the routing node compared to LCE, but it is the same in the actual production environment. Data resources are usually requested by multiple users. This kind of caching strategy cannot cache data resources in the routing node that is closest to the user in time, resulting in the need to obtain data from a farther data source when requesting the same data again. Network latency. The Prob caching strategy sets a fixed caching probability for data at each routing node. This caching strategy is relatively simple and does not consider factors such as distinguishing the attributes of the data produced by the producer. For example, shared resources such as videos, pictures, and documents need to be cached, but non-shared resources such as chat information do not need to be cached. In this paper, a multi-attribute probabilistic cache algorithm (MAPC) is proposed. This cache optimization algorithm effectively reduces the redundant storage of the same Data by different routing nodes in the NDN network, saves the use of routing node caches, and reduces network latency, improve NDN network performance. For the design of this caching strategy, it is necessary to produce the attributes of the data itself from the producer, the weight attributes and data packets that the routing node occupies in the NDN network are cached as close as possible to the attributes of the consumer routing node, and the dynamic probability is determined according to these three attributes, Whether the data is cached at the routing node.

2. Research on Multi-attribute Probability Caching Algorithm

2.1 Degree Centrality

Degree centrality originally originated from the study of social networks. The famous British anthropologist Brown put forward the concept of "social networks" in the study of the composition of social structural relationships. Since the 1930s, a large number of anthropologists and sociologists have continuously studied and improved the concept of "social network", and produced a large number of research theories. In 1979, C. Freeman put forward the concept of "degree centrality" for the first time in the study of "The Concept of Newness in Social Networks". Degree centrality is widely used in graph theory and network analysis. In a network, degree centrality is a key indicator to determine whether the nodes in the network are important. In the network, the more the number of other network nodes connected to a network node, the more important this network node is, and the greater its degree of centrality. As shown in Figure 1, the degree centrality of node v3 in the network is greater than all other nodes, and v3 node is the most important node in the network.

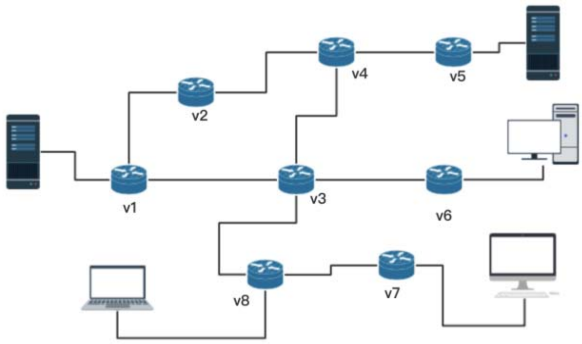

Figure 1 network node centrality

2.2 *The Design Idea of Multi-attribute Probability Caching Algorithm*

When the routing node in the NDN network caches data, it can cache according to the attributes of the data itself. In the actual production environment, the data produced by producers can be marked as different attributes. In this article, we divide the data produced by producers into two types: non-shareable attribute data and shareable attribute data. Non-shareable attribute data means that such data will only be requested once by the unique consumer in the NDN network, so caching this attribute data is a waste of cache resources of routing nodes in the network, such as mail and instant messaging , Voice calls, etc. Shareable attribute data means that such data generated by producers in the NDN network will be repeatedly requested by consumers, such as streaming media broadcasts, pictures, and videos. In the NDN network, the routing node caches the sharable attribute data, so that the consumer node can obtain the request data from the nearest routing node, which can improve the network response speed and reduce the use of network traffic.

This article classifies the producer's production data attributes as part of the multi-attribute probabilistic caching algorithm, that is, the producer adds the Status field to the data packet when producing the data packet. When the data in the data packet is shareable attribute data, the Status The value is set to 1; when the data in the data packet is unshareable attribute data, the value of Status is set to 0. Whenever the data packet traces back to the routing node, the routing node obtains the Status value in the data packet and judges its value. When the value is 0, the data carried by the data packet is not cached (that is, the cache probability of the data is 0). When the value is 1, whether the data carried by the data packet is cached is determined by the subsequent probability.

The NDN network is a mesh topology. The number of routers connected to each routing node in the network is random. The routing node can receive the interest request sent by the routing node to which it is connected. After the routing node receives the interest request, if there is no cache hit, check whether there is an entry for the interest request in the PIT table, if not, create the entry and record the port number of the source of the interest request; if If there is, it will check whether the source port number exists in the interest table entry, if there is, the interest request is discarded, and if it does not exist, the port number is added to the interest table entry. Within a certain time range, routing nodes in the NDN network will receive interest requests sent from different ports, which is known from the "degree centrality" principle. For a routing node in the NDN network, if the sum of the number of all interest table entry ports in the routing node PIT is larger, it means that the centrality of the routing node is greater, and the node is more easily accessed. Increasing the probability of the node's cached data is beneficial to improve the cache hit rate of other routing nodes in the NDN network to access the node. In this paper, each routing node is set with the attribute of node centrality probability. When a data packet is traced back to the routing node, the node centrality probability is used as an attribute of whether the data packet carries data cache. The greater the probability of node centrality, the greater the probability of data carried in the cached data packet of the routing node. In this paper, the attribute DegreeProb of the node centrality is defined to indicate the characteristic that the data packet has different caching probability due to different "degree centrality" of routing nodes. The probability formula of node centrality is defined as: DegreeProb = PortCount / (NodeCount + PortCount), where PortCount is the sum of the number of all interest table entries in the routing node PIT, and NodeCount is the sum of the total number of routing nodes in the NDN network.

Any consumer node in NDN will have a certain probability to request the data produced by the producer. After the interest packet sent by the consumer hits the data at the data source node, the data packet will be traced back to the consumer node along the reverse path of the interest packet. If a data packet is traced back to an intermediate routing node, the probability that the data carried in the data packet will be cached increases as the number of data packets passing through the routing node increases, then the data carried by the data packet will have a greater probability It is cached near the consumer node. When the consumer node requests the data again, it can obtain the data from the vicinity of its cached routing node, which reduces the transmission distance of the data packet in the NDN and improves the network response speed. In this paper, an attribute field of hops is added to the data packets produced by the producer, which is used to record the number of routing nodes that the data packet passes after leaving the producer node. Each time a data packet passes a routing node, the hop value in the data packet increases by one. When the hop value carried in the data packet is larger, the probability that the data packet is cached at the routing node is greater. In this paper, the attribute DistanceProb of the node distance probability is defined to indicate the characteristic that the packet has a different probability of being cached at the routing node due to different carrying hop values. The node distance probability formula is defined as: DistanceProb = HopCount / (HopCount + Constant) where HopCount is the hop value of the data packet passing through the routing node, Constant value is a positive integer, and its size value can be adjusted according to the actual network needs (generally the two most Half the distance of the far routing node, because this can ensure that DistanceProb becomes evenly larger according to the increment of the hop count).

In summary, the multi-attribute probabilistic caching algorithm designed in this paper considers the attributes of the producer's production data itself, the weight attributes and traceback data occupied by the routing nodes in the NDN network, and caches the attributes of the routing nodes near the consumers as much as possible. According to these three The aspect probability dynamically determines whether the data is cached at the routing nodes it passes through during the backtracking process. The implementation process of the multi-attribute probabilistic caching strategy is that when the data packet reaches the routing node, the Status value is first determined. If the Status value is 0 (both attribute data that is not shareable, the probability of occurrence of the cache is 0) the data does not cache, if the Status value Whether the data carried in the data packet is cached at 1 is determined by the attribute probability AttributeProb. The formula is defined as: AttributeProb = (1- (DegreeProb + DistanceProb)) where DegreeProb is the node centrality probability and DistanceProb is the node distance probability. The multi-attribute probabilistic cache algorithm is the same as the Prob algorithm, and the network administrator can set the threshold Q according to the actual situation of the network when it is used. If the probability AttributeProb value is less than Q, the routing node caches the data carried in the data packet,

Journal of Physics: Conference Series **1570** (2020) 012010 doi:10.1088/1742-6596/1570/1/012010

if the probability value is greater than Q, the data carried in the data packet is not cached, neither (0, Q) caches data, (Q, 1) does not cache data.

2.3 Data structure design

The design and implementation of the multi-attribute probabilistic caching algorithm requires the following data structure, that is, the attribute field value of the producer's production data, the value of the hop field that the data packet reaches the routing node, and the total number of port numbers of all interest table entries in the routing node PIT. . In order to obtain the above data information, the data structure needs to be designed as follows.

Packet structure design： The design and implementation of the multi-attribute probabilistic cache algorithm requires the addition of two attribute fields on the basis of the original data packet, namely Status and HopCount, as shown in Figure 2. The value of the Status attribute is used to mark whether the data carried in the data package produced by the producer is shared data, when the Status value is 1, it is shared data, and when the value is 0, it is non-shared data. The value of the HopCount attribute is used to record the number of routing nodes that have passed through during the backtracking of the data packet, and its initial value is 0. The HopCount value is increased by one every time a routing node passes.

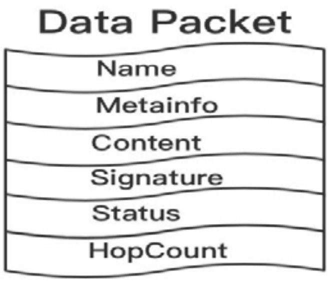

Figure 2 Data Packet Structure

Data structure design of attribute probability value： The design and implementation of the multi-attribute probability caching algorithm also needs to implement the following data structure, and its class diagram is shown in Figure 3. The PortCount field is used to count the total number of ports of all interest table entries of the routing node PIT. The node centrality probability field DegreeProb is used to record the calculated node centrality probability value, and the node distance probability field DistanceProb is used to record the calculated node distance probability value. , The attribute probability field AttributeProb is used to record the calculated attribute probability value.

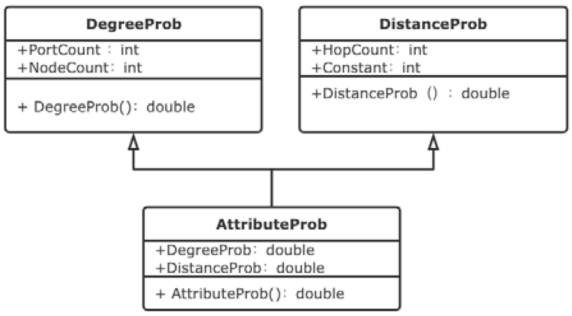

Figure 3 probability class design

3. Algorithm Simulation Experiment and Analysis

3.1 Basic setting of simulation experiment

The experiment of this subject is completed under Linux system, the system version is Ubuntu16.04, and the ndnSIM [10] simulation simulation platform uses version 2.2. The simulation experiment department belongs to a 30 * 30 grid network topology. A total of 900 nodes are set in the entire NDN network, and one node is randomly selected as a producer and 30 nodes are randomly selected as consumers. The upper limit of transmission bandwidth between routing nodes is 10Mbps, and the simulated distance between routing nodes is 1km. All consumers send 300 interest packets to the producer every second for data requests, and the cache space size of the routing node is set to 100. According to the design principle of the MAPC algorithm proposed above, we set the constant value NodeCount in the Prob = PortCount / (NodeCount + PortCount) in the MAPC algorithm to 900, and set the Prob cache algorithm and the MAPC cache algorithm The probability threshold is set to 0.5, and also the node distance probability formula: DistanceProb = HopCount / (HopCount + Constant) Constant value Constant is set to 30. The simulation experiment network topology is shown in Figure 4.

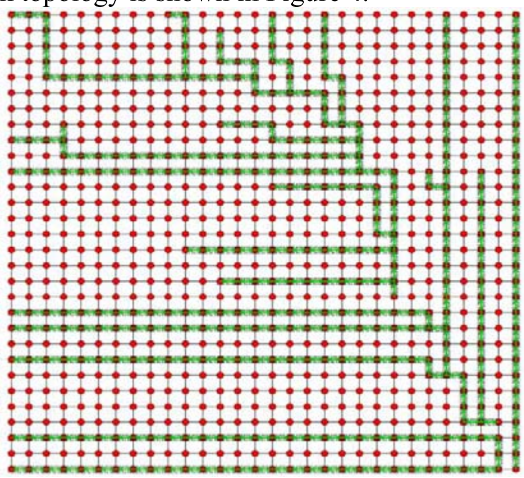

Figure 4. experiment process and network topology

3.2 Algorithm Experiment Evaluation Index

In this experiment, we will select various experimental data as the NDN network performance evaluation index. It is verified that the MAPC algorithm has more advantages in NDN caching than the NDN default LCE algorithm and Prob algorithm. In this experimental data analysis, we will select the experimental data of cache data volume, network delay, and cache hit rate as NDN network performance evaluation indicators.

As shown in Figure 5, the content shown in this figure is the change in the cache data volume of all routing nodes with the simulation experiment time in the NDN network. It can be seen from the figure that NDN's default LCE cache algorithm and Prob cache algorithm. After 200s, as the simulation time increases, the amount of cached data in the network has no longer increased. This means that the cache space of the routing node participating in communication has been exhausted. When new data that needs to be cached arrives at the routing node, the data can only be cached by way of cache replacement. Frequent cache replacement will not only consume the routing node Computing resources, but also reduces the data residence time in the cache, reducing the cache hit rate when other consumer nodes get the data.

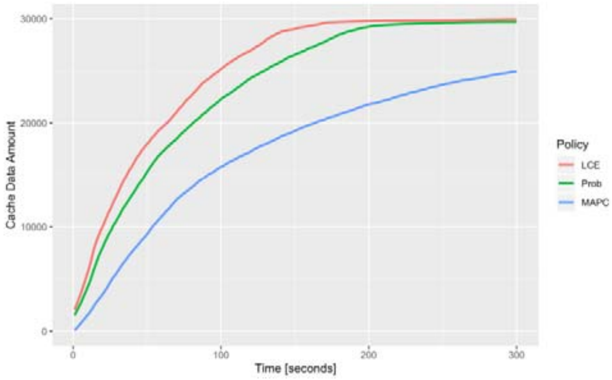

Figure 5 comparison of cache data volume with simulation time

As shown in Figure 6, the content represented in the figure is that, in an NDN network using three different algorithms respectively, its FullDelay value changes with simulation time. It can be seen from the figure that the MAPC caching algorithm has a FullDelay value lower than that of the other two caching algorithms for most of the entire simulation experiment. This is of great significance to the overall performance improvement of the NDN network.

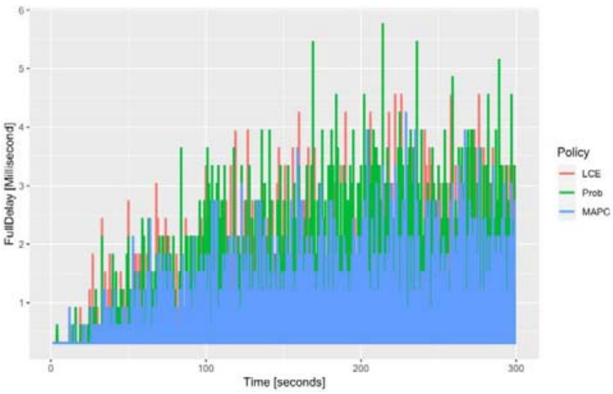

Figure 6 comparison of network delay with simulation time

As shown in Figure 7, the content shown in the figure is in the NDN network, which uses three different cache calculations, and its cache hit rate changes with the simulation time. It can be seen from the figure that as the experimental simulation time increases, the cache hit rate of the NDN network using the MAPC cache algorithm is gradually greater than that of the NDN network using the other two cache algorithms. This shows that as the simulation time increases, the cache space of routing nodes in the NDN network using the LCE and Prob cache algorithms is gradually exhausted, and cache replacement occurs. A large amount of the same redundant data is cached in the network, reducing the diversity of data cached by routing nodes in the network, thereby reducing the cache hit rate of intermediate routing nodes.

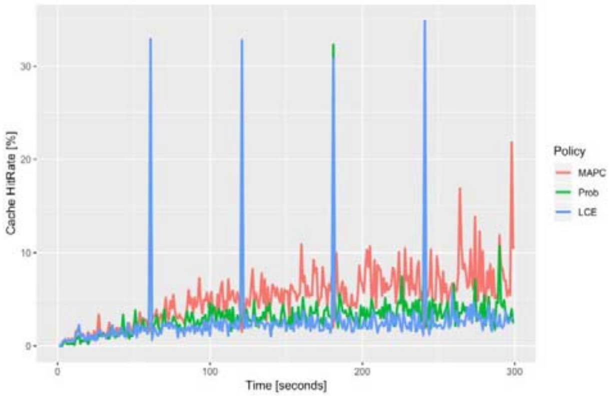

Figure 7 comparison of cache hit rate with simulation time

4. Conclusion and Future Works

The article proposes a multi-attribute probability caching algorithm. Through the algorithm, when the intermediate routing node receives the traceback data packet in the NDN network, it determines the data attribute carried in the data packet. The routing node only provides a certain cache probability for the shareable attribute data. The cache probability is based on the routing node's degree The probability of node centrality calculated by "centrality" and the probability of node distance calculated based on the number of hops carried in the data packet are jointly determined. The cache decision algorithm can effectively reduce the cache of redundant data in the NDN network, improve the utilization of the routing node's cache space, reduce the network delay, and increase the routing node's cache hit rate, thereby achieving the effect of improving the overall performance of the NDN network. Although the caching algorithm has achieved certain results, it is not perfect, and there is still room for optimization and research. It is mainly reflected in the following aspects. This article only defines the attributes of producer production data as two types: sharable attribute data and non-sharable attribute data. This definition is relatively rough, and it can further divide the shareable attribute data produced by the producer in detail and give a certain probability cache weight. In this way, the cache resources of routing nodes in the NDN network can be used more economically.

ACKNOWLEDGMENTS

Our thanks to ACM SIGCHI for allowing us to modify templates they had developed.

REFERENCES

[1] Xiaoke Jiang, Jun Bi. Interest Set Mechanism to Improve the Transport of Named Data Networking[J].ACM SIGCOMM,2013:515-516.
[2] Mai HL, Nguyen NT, Doyen G, et al. On the Readiness of NDN for a Secure Deployment: The Case of Pending Interest Table. In: Ifip International Conference on Autonomous Infrastructure, 2016.
[3] Dai H, Liu B, Yan C, et al. On Pending Interest Table in Named Data Networking. In: Eighth Acm/ieee Symposium on Architectures for Networking & Communications Systems, 2012.
[4] Zhuo L, Liu K, Liu D, et al. Hybrid wireless networks with FIB-based Named Data Networking [J]. Eurasip Journal on Wireless Communications & Networking, 2017,2017(1): 54.
[5] Rezazad M, Tay YC. A cache miss equation for partitioning an NDN content store. In: Asian Internet Engineering Conference, 2013.
[6] Named Data NetworkingProject.[Online].http://www.named-data.net.

[7] Laoutaris N, Syntila S, Stavrakakis I. Meta algorithms for hierarchical Web caches. In: Proc. of the 2004 IEEE Int'l Performance,Computing and Communications Conf. 2004. 445−452. [doi: 10.1109/PCCC.2004.1395054].
[8] Laoutaris N, Che H, Stavrakakis I. The LCD interconnection of LRU caches and its analysis. Performance Evaluation, 2006,63(7):609−634. [doi: 10.1016/j.peva.2005.05.003].
[9] Laoutaris N, Syntila S, Stavrakakis I. Meta algorithms for hierarchical Web caches. In: IEEE International Conference on Performance, 2005.
[10] Zhang L, Zhang L, Zhang L. On the Evolution of ndnSIM: an Open-Source Simulator for NDN Experimentation[M]. ACM, 2017.