Summary Cache of IoT Data Using ICN

Hayate Sasaki* and Noriaki Kamiyama†
*Graduate School of Engineering, Fukuoka University, Fukuoka 814-0180, Japan
Email: td202004@cis.fukuoka-u.ac.jp
† Faculty of Engineering, Fukuoka University, Fukuoka 814-0180, Japan
Email: kamiyama@fukuoka-u.ac.jp

***Abstract*—Internet of Things (IoT), which measures various environmental data such as temperature with sensor devices installed in various places, collects the measured data in a cloud data center, analyzes the data, and obtains useful information, is attracting wide attention. Obtaining the average value and standard deviation of sensor information of target items existing in a specific area is one of the expected services of IoT. Although the amount of data transmitted by each sensor is small, it is necessary to collect data from many sensors existing in the target area. Therefore, the processing load of routers such as name resolution and data forwarding will increase, so the information centric networking (ICN), which transfers data directly using the name of data, has attracted wide attention as a new network architecture efficiently delivering IoT data. In this paper, we propose to cache the summary data, i.e., the average and standard deviation, of IoT data at routers and reuse them for other requests. By computer simulations, we show that the average total number of hops of data transferred for each request is reduced by about 70% to 99%, and the average number of memory accesses at each router by about 50% to 90%, compared to the case without using the cached summary data.**

I. Introduction

With the spread of the Internet of Things (IoT), everything is connected to the Internet and new services are born. When the IoT devices that is the target of data acquisition are fixed, and the IoT services periodically and continuously collect data from the fixed IoT devices, packet transmission using the ID of end host as a locator, i.e., the Internet, is effective. However, when data is collected from all IoT devices that satisfy a certain condition, packet transmission using the ID of end host as a locator is inefficient. In other words, on the Internet, we need to give each data in advance an address according to the existing location and register the correspondence between the address and the name of the data in the DNS (domain name system). However, in the IoT, data is continuously generated from a huge number of IoT devices, so allocating addresses and providing the correspondence between the address and name of data is difficult. In addition, IoT devices have the limitation of lifetime because battery replacement is difficult after being installed. Although the power consumption of the IoT devices can be suppressed by turning them off when data is not transmitted, IoT services cannot acquire data if they are turned off.

These problems can be solved by using the information-centric networking (ICN). By using the ICN to transfer IoT data, the IoT application can request the desired data by directly using the name of sensing data without specifying the target IoT devices. Also, in the ICN, routers can cache data, and IoT services can obtain data from router caches, so the IoT devices does not need to be constantly connected to the network. Therefore, transferring IoT data using the ICN is drawing attention [2].

One type of expected IoT service is acquiring the average and standard deviation of sensed data within a specific area. In this service, we need to collect sensed data from all IoT devices existing in the target area requested by the user, and the processing load required for the network increases rapidly as the target area expands. On the other hand, in the ICN, routers can cache IoT data and deliver the IoT data from the router caches. Therefore, when transmitting the IoT data using the ICN, we can cache the summary data, e.g., the sum of data calculated for a specific area and the sum of squares, at routers and reuse them for subsequent service requests. Therefore, in this paper, to reduce the number of data transmissions and the number of memory accesses at routers, we propose to cache the sum, square-sum, and number of samples of IoT data in various target area as the summary data at routers and reuse them for subsequent service requests. The proposed caching method can apply for various types of IoT data take continuous or discreate values for which the average and standard deviation can be derived. Therefore, the proposed caching method is not application specific. As far as the authors know, no caching method of caching summary data has been proposed, so the idea of caching summary data depending on the area of data is novel. The proposed method dynamically calculating and caching summary data for various areas, so caching summary data is not equivalent to the simple TTL based caching. The contributions of this paper are summarized below.

- We propose a mechanism to cache the sum, square-sum, and number of samples of IoT data in various areas at routers as the summary data and reuse them to calculate the average and standard deviation of IoT data in wider areas to reduce the amount of data transmitted in the network as well as the number of memory accesses at routers.
- Through the computer simulations assuming the square area in Chiyoda-ku, Tokyo, Japan as the entire target area, we show that the average total number of hops of data transferred for each request is reduced by about 70% to 99%, and the average number of memory accesses at each router by about 50% to 90%, compared to the case without cache.

Section II summarizes the related works, and we describe the assumptions in Section III. In Section IV, we describe the proposed method and present the numerical results in Section V. Finally, we summarize this manuscript in Section VI.

978-3-903176-32-4 © 2021 IFIP

II. Related Works

Various methods for transmitting IoT data using the ICN have been proposed. In the ICN, data is transmitted on the path where an Interest packet is transmitted in the reverse direction by storing the input face number in the PIT (pending interest table) of each router on the path. Therefore, Gundogan et al. proposed to hold the PIT entry of routers on the route for a long time [5]. The CP (content proxies) is provided, and each CP is allocated a set of responsible prefixes and holds those contents. When publisher publishes data, the relay router repeats publish toward the corresponding CP. On the other hand, when the subscriber subscribes, the Interest packet is transferred to the CP. Also, Arshad et al. proposed a naming method that combines a hierarchical name and a flat name in an environment where IoT devices are placed in a hierarchical organization such as a university [1]. Bouk et al. proposed a naming method that combines a hierarchical type and a flat type when the ICN is used for road and vehicle communication [3]. Dong et al. proposed a new format of Interest message for requesting IoT data and an aggregation function of FIB (forward information base) entry [4].

In addition, Kurihara et al. proposed to construct a router FIB using a trie with pairs of position name represented by Z notation and data name in the COPSS which is the NDN publish and subscribe system [6]. This method used a quaternary number based on the Z notation representing the position for the destination name, and each digit corresponds to the element of the hierarchical name, so this method can transmit Interest packets based on the position and aggregate FIB entries at routers. Furthermore, the authors also proposed a naming method for position-based transfer using the Z notation [7]. There is also a comparative evaluation of the performance in IoT data transfer under various cache control methods of the ICN using an actual IoT devices [10]. Because a freshness was important in IoT data, Zhang et al. classified network nodes into three layers, i.e., root, middle, and edge, set a threshold for each layer, and cached data only when the elapsed time from data generation was less than the threshold [11]. However, it has not been studied so far to cache the summary data, i.e, the average and standard deviation of IoT data, at routers and reuse the summary data for other requests.

III. Assumed Conditions

As a type of ICN, we assume the NDN (named data networking) [9] in this manuscript. We assume an IoT service that acquires the average value and standard deviation of the specified type of data, e.g., temperature, pressure, and precipitation, in a target area, and the pull-type delivery in which data is transmitted to users when users request the data. The area ID and data type are given as the attributes of the sensing data, and the Z notation is used as the name of the area. The Z notation divides the target square area into four areas as shown in Fig. 1, and it assigns a quaternary number from 0 to 3 to each divided area. Furthermore, each divided area is further divided into four parts, and quaternary numbers are similarly assigned. For example, each minimum area that has been divided into four times twice can be represented by a 2-digit quaternary number such as 1/1 and 3/1. By repeating this process of dividing into four layers hierarchically, each digit Z_i can indicate the position of each layer i, and the area you want to specify can be represented by a combination of quaternary numbers $\boldsymbol{Z}_k = (Z_1, Z_2, \cdots, Z_k)$ with an arbitrary number of k digits. Prefixes of Interest are the Z notation of the target area. FIB entries are looked up by a combination of quaternary numbers, so the FIB size can be effectively suppressed [6].

As shown in Fig. 2, one gateway (GW) is provided in each minimum area where all digits are specified in the Z notation, and each GW is connected to the edge router with the shortest distance from the GW. The IoT device intermittently connects with the GW in the smallest area where the IoT device exists, and the GW stores the received data which IoT devices transmit. Each router has cache memory which can store the summary IoT data. Freshness is important for many types of IoT data, so the TTL (time to live) is set for each IoT data, and the data that exceeds the TTL is deleted from the cache memory of routers. It is assumed that the storage capacity of cache memory of routers is sufficient to store just summary data of small size, and data stored at routers are not removed except when exceeding the TTL. Between GWs, the network topology is constructed in a tree structure of K layers, and service users are connected to routers at the lowest layer.

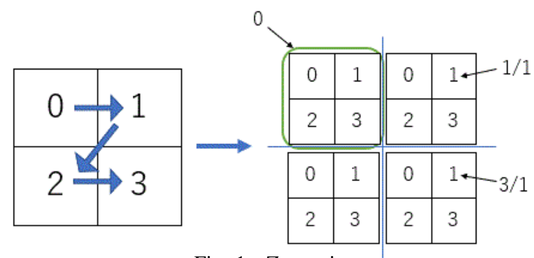

Fig. 1. Z notation

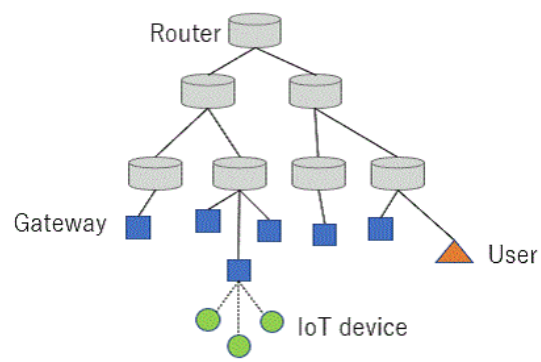

Fig. 2. Network topology assumed

IV. Summary Caching of IoT Data

Each router caches the sum of received data, the square-sum of received data, and the number of sample devices in any target area of Z notation. Let $S_1(\boldsymbol{Z}_k)$ be the sum of data of any data type in the target area written in arbitrary Z notation $\boldsymbol{Z}_k$ of k digits. Moreover, let $S_2(\boldsymbol{Z}_k)$ and $C(\boldsymbol{Z}_k)$ denote the square-sum of data and the number of sample devices, respectively. To efficiently calculate the standard deviation of IoT data, keeping the squared sum of data at each router is effective. Because each digit of $\boldsymbol{Z}_k$ can take an integer between zero and three, they are obtained by $S_1(\boldsymbol{Z}_k) = \sum_{j=0}^{3} S_1(\boldsymbol{Z}_k, j)$, $S_2(\boldsymbol{Z}_k) = \sum_{j=0}^{3} S_2(\boldsymbol{Z}_k, j)$, and $C(\boldsymbol{Z}_k) = \sum_{j=0}^{3} C(\boldsymbol{Z}_k, j)$, where $\boldsymbol{Z}_k, j$ means $\boldsymbol{Z}_{k+1}$ in which the top k digits agree with $\boldsymbol{Z}_k$, and the last digit is j. Using these variables, the average value $\mu(\boldsymbol{Z}_k)$ and the standard deviation $\sigma(\boldsymbol{Z}_k)$ of the target area

$\boldsymbol{Z}_k$ are obtained by $\mu(\boldsymbol{Z}_k) = S_1(\boldsymbol{Z}_k)/C(\boldsymbol{Z}_k)$ and $\sigma(\boldsymbol{Z}_k) = \sqrt{S_2(\boldsymbol{Z}_k)/C(\boldsymbol{Z}_k) - \{\mu(\boldsymbol{Z}_k)\}^2}$.

The service user sends the Interest packet with specifying the target area as prefix, the earliest acquisition time, and the data type. The router that receives the Interest packet looks up the FIB and transfers the Interest packet to all the neighboring routers that match the requested prefix so that the Interest packet is transferred to all the GWs included in the requested target area. However, if each router caches the summary data of all or part of the requested target area, the summary data for that part is delivered from the router to the requesting GW without forwarding the Interest packet.

When the Interest packet arrives at the GW included in the request target area, the GW calculates the sum and square-sum of the data and the number of sample devices in the minimum area which satisfy the conditions of the time range and the data type. The GW sends the summary data calculated to the requesting user on the path which the Interest packet travels in the reverse direction. Each router that receives the summary data transfers the summary data to the neighboring router according to the PIT. However, if all the data in the requested target area is included in the obtained data, the received data is cached at the same time. As a result, when all the data in any Z notation area is cached, the summary data that aggregates them is newly calculated and cached.

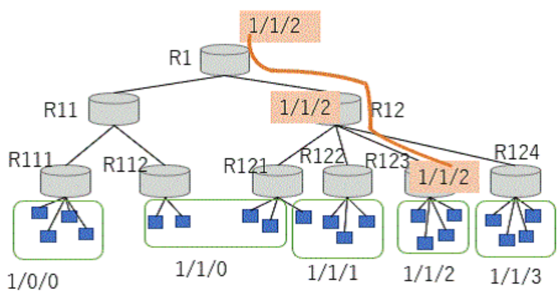

Fig. 3. Case that user requests IoT data for area (1/1/2)

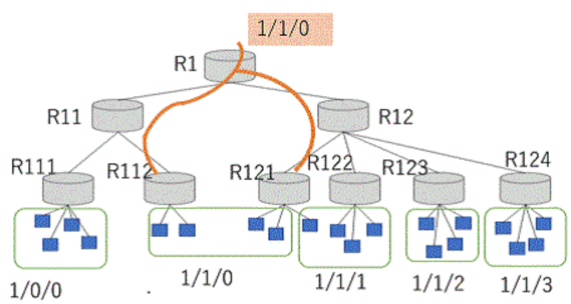

Fig. 4. Case that user requests IoT data for area (1/1/0)

An example of Interest and data transmission is shown in Figs. 3, 4, and 5. For example, as shown in Fig. 3, when a user requests IoT data for the area with prefix of (1/1/2), the summary data (1/1/2) is collected at routers R1, R12, and R123, and the summary data of area (1/1/2) is cached at these three routers. As shown in Fig. 4, when a user requests data for the area with prefix (1/1/0), the IoT data (1/1/0) are obtained from both R112 and R121, and the entire data of (1/1/0) is collected at only router R1. Therefore, although the IoT data for (1/1/0) is transferred at R112 and R121, the summary data (1/1/0) is cached only at router R1 where all the summary data of (1/1/0) is collected. Routers R112 and R121 do not cache the received data because it is incomplete data of (1/1/0).

Moreover, as shown in Fig. 5, let us consider the case that a user requests IoT data for the area with prefix (1/1) when router R1 caches summary data (1/1/0), and R12 caches summary data (1/1/2). Summary data (1/1/0) and (1/1/2) which are part of data (1/1) are sent from routers R1 and R12, so only summary data (1/1/1) and summary data (1/1/3) need to be obtained from the GWs. Interest packets are transferred to routers R121, R122, and R124 that accommodate the GWs corresponding to the target areas (1/1/1) and (1/1/3). After receiving summary data (1/1/1), (1/1/2), and (1/1/3), router R1 creates summary data (1/1) and stores summary data (1/1) at its cache memory.

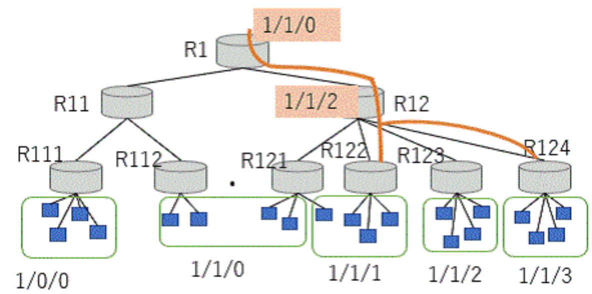

Fig. 5. Case that user requests IoT data for area (1/1)

V. Performance Evaluation

A. Simulation Condition

We evaluated the effect of the proposed method using a computer simulator developed by ourselves using C language. From the population density map [8] of Tokyo, a square area with 16 km on one side of the coordinates centering on Chiyoda Ward was cut out as the evaluation area. The process of dividing the evaluation area into four sub areas in the east-west and north-south directions was repeated six times. Therefore, the minimum area was a square area with one side of 250 m, and the evaluation area was divided into 64×64 minimum areas in the east-west and north-south directions. This means that the evaluation area was divided into $2^6 = 64$ minimum square areas. One GW was provided in each minimum square area, and we set the number of IoT devices in each minimum square area to the value proportional to the population ratio of each minimum square area. User requests were generated according to the Poisson process with the average rate of 1 per second from a GW randomly selected with the probability proportional to the population ratio of minimum square areas. For each user request, we randomly selected the digit number r of Z notation according to the Zipf distribution $p(r) = c/(7-r)^\theta$ in the range from $1 \leq r \leq 6$ where c was a normalized random number, and the value of each digit was randomly selected within the range of 0 to 3.

Routers were placed at the positions of the 58 telephone-exchange buildings of NTT located in the evaluation area. The router topology was a tree type with a depth of three. One router of the highest layer 1 was assigned to the telephone-exchange building in Otemachi, and three routers of layer 2 were assigned to the three telephone-exchange buildings located near the three stations that covered the evaluation area evenly. 12 routers of layer 3 were assigned to 12 telephone-exchange buildings that covered the evaluation area evenly as well, and 42 routers of layer 4 were assigned in the remaining

42 telephone-exchange buildings. Except the router of layer 1, we connected each router to the nearest router in the layer one level above, and each GW is connected to the closest router of layer 4. A trie was used for the FIB and cache data structures of routers to reduce the time complexity of the search process. Each node of the trie had pointers from 0 to 3 other trie nodes so that we can specify any area represented by Z notations. Interest packets are transferred to adjacent router using the area ID expressed by Z notation, so the output face number of the adjacent router was set to each leaf node of the trie. In the cache, a trie was also constructed in the same way as the FIB, and each node of the trie had no output face number but a binary variable for judging whether to have cached data.

B. Average Total Hop Length

The average total hop length η is defined as the average of η_f, the sum of the flow hop lengths delivereing data packets in the unit of minimum square area that constructs the entire data transmitted for user request f. For example, when the target area of user request f is (1/0/1/2/0), this user requests data from four minimum square areas, (1/0/1/2/0/0), (1/0/1/2/0/1), (1/0/1/2/0/2), and (1/0/1/2/0/3), so η_f is the sum of flow hop length of delivering data of these four minimum square areas. We divided time into time slots with length of 100 seconds. We show the results when r was randomly selected within the range of $1 \leq r \leq 6$ with the probability according to the Zipf distribution $p(z) = c/(7-r)^{\theta}$.

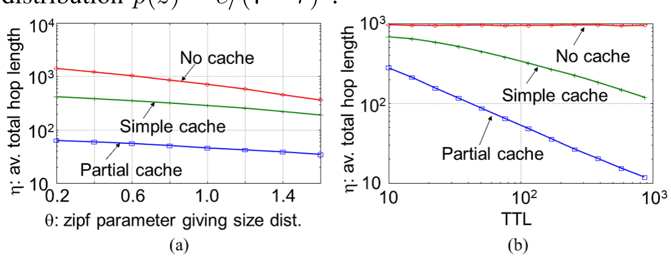

Fig. 6. Average total hop length η against (a) Zipf parameter giving size distribution of requested area θ and (b) TTL of cached data

The average total hop length η was plotted against θ in Fig. 6(a) when setting the TTL to 100 seconds. The results of three methods were shown in the figure: the case without using caches, denoted as *No cache*, the case caching at the transit routers in units of the minimum square area, denoted as *Simple cache*, and the proposed method, denoted as *Partial cache*. As θ increased, the ratio of request with the target areas of small size increased, so the amount of data transmitted decreased, and η decreased. However, in the proposed partial cache method and the simple cache method, the effect of caches decreased as θ increased, so the reduction degree of η with increasing θ was small compared with that of the no cache. The proposed method significantly reduced the average total hop length η in the entire range of θ compared with the other two methods.

Figure 6(b) plots η against the TTL when setting $\theta = 0.7$. As the TTL increased, the proposed method and the simple cache method increased the amount of cached summary data, so η decreased. The proposed method significantly reduced the total hop length in the entire range of TTL compared with the other two methods. We confirmed that the proposed method can reduce the average total number of hops of transferred data in one delivery by about 70% to 99% compared with the case without using caches.

VI. Conclusion

One of the expected services of the IoT is a service that obtains the average and standard deviation of sensor information of target items existing in a specific target area. Although each sensor sends a small amount of data, it is necessary to collect data from many sensors existing in the target area, so the network load caused by name resolution and packet transfer at routers significantly increases. Therefore, the ICN, which transfers data by directly using the name of the data, is attracting a wide attention as a new network architecture that avoids the overhead of name resolution and collects the data required by the network. However, the processing load of routers will be still high because we need to transmit IoT data from a huge number of IoT devices. In this paper, to reduce the amount of traffic in the network, we proposed to cache and reuse the calculation results of the average and standard deviation of IoT data at routers as the summary data. By computer simulation, we evaluated the average total number of hops of delivery flows of IoT data and the average number of memory accesses required at each router to look up the FIBs and cache memories. As a result, we clarified that the proposed method reduced the average total number of flow hop lengths for each user request by about 70% to 99% and the average memory access count by about 50% to 90% compared with the case without caches. As the size of the target area increases, the number of minimum areas increases, and this increases the maximum number of digits in the z notation. We will investigate the increase of size of FIB/cache trees at each router as the future work.

Acknowledgements

This work was supported by JSPS KAKENHI Grant Number 18K11283.

References

[1] S. Arshad, B. Shahzaad, M. A. Azam, J. Loo, S. H. Ahmed, and S. Aslam, Hierarchical and Flat-Based Hybrid Naming Scheme in Content-Centric Networks of Things, IEEE Internet of Things Journal, Vol. 5, No. 2, pp. 1070-1080, Apr. 2018

[2] S. Arshad, M. A. Azam, M. H. Rehmani, and J. Loo, Recent Advances in Information-Centric Networking-Based Internet of Things (ICN-IoT), IEEE Internet of Things Journal, Vol. 6, No. 2, pp. 2128-2158, Apr. 2019

[3] S. H. Bouk, S. H. Ahmed, and D. Kim, Hierarchical and hash based naming with Compact Trie name management scheme for Vehicular Content Centric Networks, Elsevier Computer Communications, 71, pp. 73-83, 2015

[4] L. Dong and G. Wang, Consumer Oriented IoT Data Discovery and Retrieval in Information-Centric Networks, IEEE PIMRC 2017

[5] C. Gundogan, P. Kietzmann, T. C. Schmidt, and M. Wahlisch, HoPP: Robust and Resilient Publish-Subscribe for an Information-Centric Internet of Things, IEEE LCN 2018

[6] Y. Kurihara, Y. Koizumi, T. Hasegawa, and M. Arumaithurai, Location-Based Forwarding with Multi-Destinations in NDN Networks, IEICE Trans. Communications, E102-B, 9, pp.1822-1831, Sep. 2019

[7] Y. Kurihara, Y. Koizumi, and T. Hasegawa, Compact Data Structures for Location-based Forwarding in NDN Networks, ICN-SRA 2018

[8] Kusano, The Characteristics of Population, Households, and Dwelling Houses within Densely Populated Zones in the Twenty-three Wards of Tokyo, Theory and Applications of GIS, Vol. 23, No. 2, pp.17-22, 2015

[9] NSF Named Data Networking project. [Online]. Available: http://www.named-data.net/

[10] J. Pfender, A. Valera, and W. K. G. Seah, Performance Comparison of Caching Strategies for Information-Centric IoT, ACM ICN 2018

[11] Z. Zhang, C. Lung, I. Lambadaris, and M. St-Hilaire, IoT Data Lifetime-Based Cooperative Caching Scheme for ICN-IoT Networks, IEEE ICC 2018