Unlike personal computers for current web browsing, IoT devices such as sensors smart meters, and actuators are communication equipment with many resource constraints. For example, many of these devices use low capacity batteries without replacing it for several years; thus, highly efficient power saving capability is required for them. They also have only a limited amount of memory, which results in a lightweight software implementation footprint for communication. This means that we are unable to use widely adopted technologies for the Web, such as Hypertext Transfer Protocol (HTTP) [1], as they are. Since HTTP protocol creates rather large length of messages, which consumes a large amount of power and memory space.

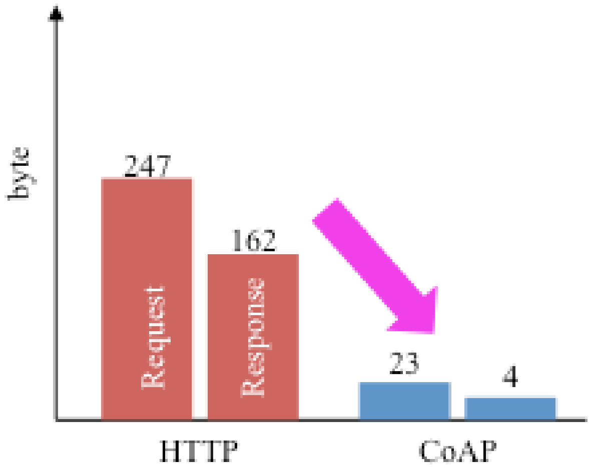


Figure 1(a): Message size (excluding layer 3 and 4 overhead) in byte

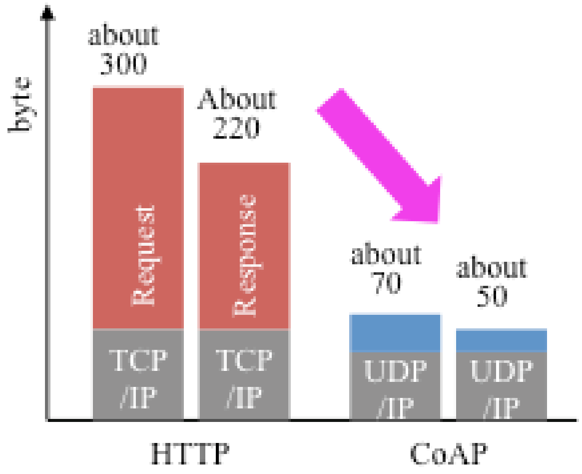


Figure 1(b): Message size (including layer 3 and 4 overhead) in byte

We can see packet aggregation method in the area of encoding voice for transmitting over packet-based networks [4][5][6]. When we transmit real-time voice or sound over packet-based network, it is inefficient to carry an element of sound per a packet. For example, in Voice over IP (VoIP), we use RTP over UDP as its transport layer protocol [4]. Even if we pick up layer 3 and layer 4, UDP has 8 bytes and IP has 20 bytes headers, respectively. On the other hand, when the sound is encoded by PCM [7], it adopts 8kHz sampling and 8bit quantization for digitizing the sound. This means it generate 1 byte of date in every 125 microseconds. When we generate a packet every 125 microseconds, it means transport one byte of data with 28 bytes of overhead even if we think about layer 3 and layer 4. This is very inefficient. Therefore, we aggregate several elements of sound into a packet. If we aggregate 160 sound elements, that means we only need to send a packet per 20 milliseconds. This aggregation highly reduces the overhead and achieves very efficient transmission.

When making this aggregation, we can easily recognize that we add additional delays to the sound element except the last one. If we consider end-to-end communication, we have to minimize the maximum delay to avoid deteriorating real-time property. Therefore, our requirements are reducing the overhead and minimizing the delay; these are two opposite directions of requirements and we have to balance them. For Asynchronous Transfer Mode (ATM) Voice and Telephony over ATM (VTOA), it introduces short cell. Small cells are aggregated into a regular 48-byte ATM cell. To minimize the delay, VTOA specification [5][6] allows small-cell based switching if intermediate ATM switch support a disaggregation and re-aggregation function.

2.2 High-speed wireless network

High-speed wireless networking is the second area for aggregation technique. Physical layer speed of IEEE 802.1n [8] is up to 600Mbps. Its specification, however, requires inter-frame gap and ack-based transmission for successive frames. Therefore, there is a waiting time for sending a successive frame. Thus, in order to increase the transmission efficiency, it is effective to reduce the number of frame and the specification adopts frame aggregation scheme.

2.3 Communication protocols for IoT area network and aggregation techniques

Sensors and smart meters are typical IoT devices that have limited battery capacity and processing capabilities. When we use these devices for IoT communication, we need to minimize packet size and number of transactions. Therefore, lightweight communication protocols such as 6LoWPAN-HC [9] and CoAP [2] are used. These protocols generate short packets compared to those of HTTP, which adopts a generally very long text message for its payload. On the other hand, the CoAP uses binary encoding for its payload for conveying control messages for devices with poor resources. CoAP dramatically reduces packet size. The CoAP has a fixed 4-byte header. Other fields are non-mandatory options so that the minimum packet size is only 4 bytes; this results in very small header overhead. The CoAP architecture consists of clients, servers, and proxies. A client sends requests for data retrieval, renewal, and removal. A server responds to a message from a client, and a proxy relays the CoAP messages. With 6LoWPAN-HC, the IP header is compressed in accordance with IEEE 802.15.4 [10]. This enables a lightweight protocol that adapts to the IEEE802.15.4 frame with 127 bytes. In order to reduce the number of transactions, aggregation techniques have been investigated [11][12]. Upon reception of multiple packets, those techniques combine individual packets and generate a single packet rather than transmit individual packets as they are. Merging multiple packets increases the effectiveness of

of data, such as temperature or power usage, is assumed so that it can add or take the average for those data in this example. If we allow various types of data, devices, and interconnects of a large number of such devices, the above technique incurs difficulties. A wide area network, such as the Internet, has such characteristics and also must handle various communication patterns and routing paths. For example, the above technique only takes into account one-way aggregation. If we want to aggregate packets at some part of a wide area network but do not want to for some other parts, we cannot differentiate both cases if we use the above technique. Another problem is that for a local sensor network, we can assume a unique single destination. Sensors usually send data to a single sink in a local network. We then can assume that the final destination can perform some complicated processes on the aggregated data. This means we only need a simple operation for intermediate nodes. On the other hand, a wide area network has nodes with different capabilities. We cannot assume a single destination node for the Internet. Therefore, if we perform a specific operation in a node, it affects the overall network and negatively affects a large amount of data flowing through the Internet. Furthermore, in a local network, irreversible operation is an option. For example, if we only need the average value of an area, we can aggregate sensor data by taking their average. For a wide area network, we usually do not assume that a network changes the data content. Therefore, we cannot adopt such irreversible operation for a wide area network.

3 Characteristics and Requirements for Wide Area Network

In this section, we assume that quite a few short packets traverse through the wide area Internet. These short packets include local aggregated packets discussed in the previous section or the original short packets produced by sensor devices. We do not consider the individual results of aggregation done in local area networks and only consider the case in which there are an extremely large number of short packets flowing in a wide area network

The uni-directional aggregation technique discussed in the previous section works well for uniform and fixed local sensor networks. As we mentioned in the previous section, it does not function in a wide area network in which there are many destinations, routes, local networks, and a large amount of information (Figure 3). With this diversity in wide area networks, the following aggregation requirements need to be met.

1. Multi-directionality

To allow various combinations of devices and applications, we need to aggregate packets with different destinations and disaggregate them.

2. Bi-directionality

Bi-directional communication is required to support message transmissions both from IoT devices to a server and from a server to IoT devices. This makes it possible to enable remote management for sensors and feedback control for actuators.

3. Selectivity

Some applications require real-time transmission. Uniform aggregation will increase delay and negatively affect those applications. Therefore, we have to aggregate packets selectively by taking into considering individual types of applications.

We can meet requirement 3 by explicitly indicating allow or deny by using a flag in a packet. According to the flag, an intermediate node can aggregate packets if it is allowed. A node transmits a message that provides a way to prohibit aggregation along the path to the destination. This makes it possible to control the behavior of nodes depending on the characteristics of the message or requirements from the application that uses those data. This flag also prevents unnecessary aggregation.

To meet requirement 4, we need the following procedure. A node that must transfer aggregated packets checks if the next hop node has the capability to process the aggregated packets. If the next hop nodes support such capability, the initial node aggregates the packets and transfers them to the next node. A node declares its supporting capabilities of aggregation and disaggregation, then a source node can check if an intermediate node has restoration capability.

By implementing these four requirements, we can achieve non-unidirectional and complex aggregation and partial aggregation in a wide area network.

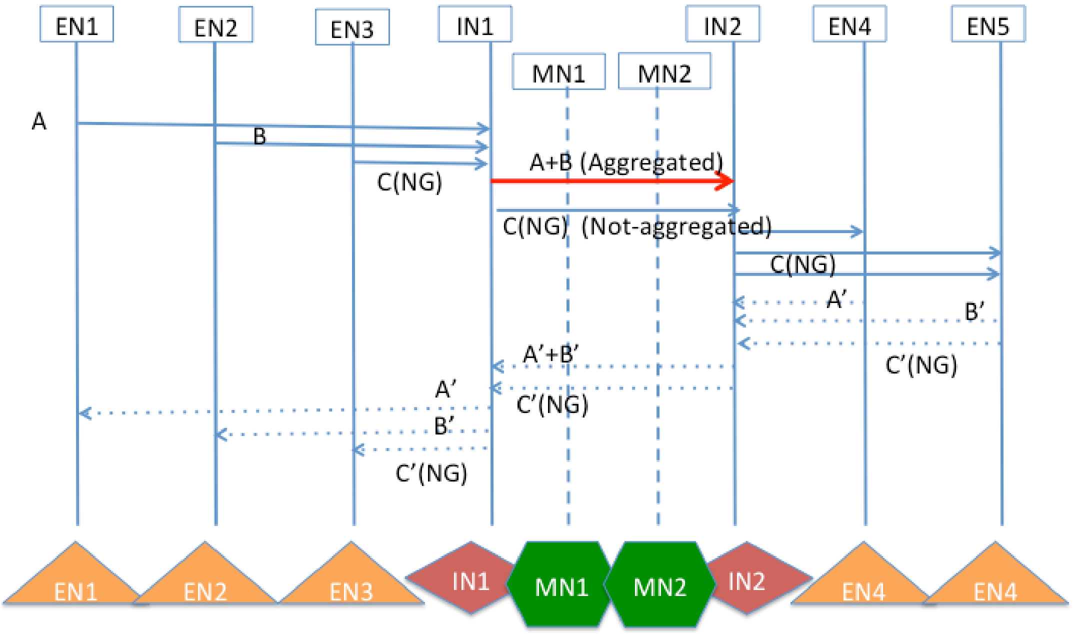


Figure 4: Example of packet aggregation flow

We illustrate our packet aggregation scheme in a wide area network with the above capabilities in Figure 4. The origin nodes (EN1, 2, and 3) send packets. Examples of origin nodes are the egress equipment of local area networks such as home gateways or IoT gateways. Intermediate nodes IN1 and 3, which correspond to the proxy in the application layer, aggregate packets received in a fixed time interval or disaggregate them. Intermediate nodes IN1 and 2 usually correspond to edge routers in the Internet. The IN1 node bundles and aggregates short packets with the same destination address or destination servers in the same autonomous system (AS) domain. There is another kind of intermediate node (MN1 and 2). These nodes do not support aggrega-

4.2 Aggregation and disaggregation procedures

We explain our message aggregation process as follows. We first identify the target group of messages. More specifically, we determine among which receiving messages we aggregate or among which messages in queue we aggregate. This identification is based on destination IP addresses or a next-hop router addresses. Then, we start aggregation process. We concatenate multiple packets and create a single packet at the node. Finally, the node transmits the aggregated packet.

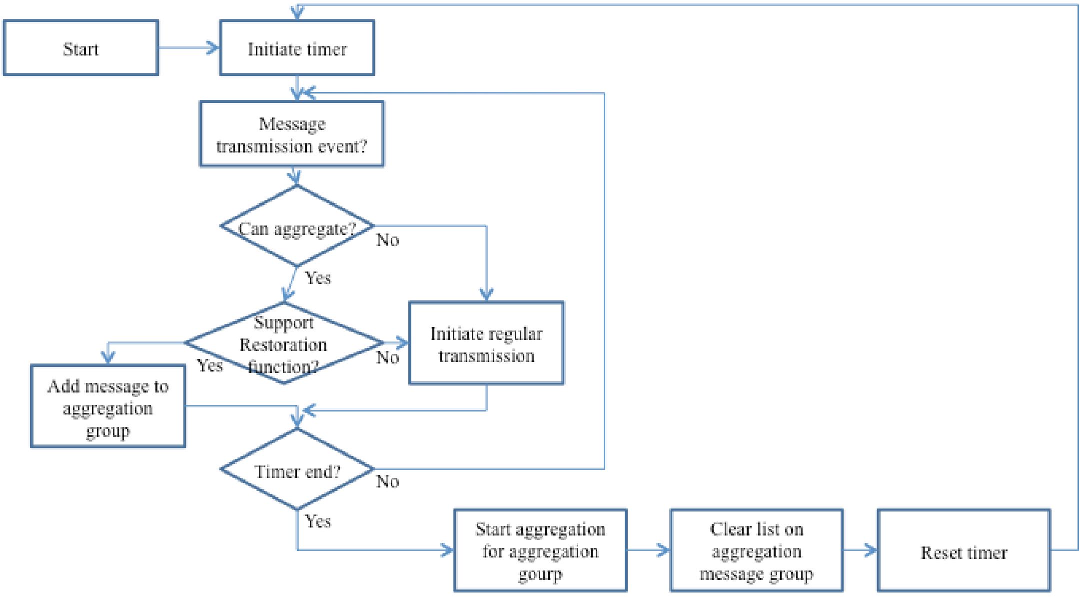


Figure 6: Flow for identifying messages for aggregation

We next explain the procedure for queuing the received packet for aggregation or disaggregation (Figure 6). When a node start running, it initiates a timer for aggregation. If the node supports restoration function, it disaggregates messages that it receives and puts the resorted message for the processing queue. The node repeats the above process until the aggregation timer expires. We here make it possible to multiple aggregations by putting once disaggregated packets to a processing queue for the sake of re-aggregation with other messages. When the timer for aggregation expires, we rest the timer and initiate a process for identifying messages for aggregation from the processing queue. Then restart the timer for receiving next messages. We repeat this process at the node.

We next explain the procedure for identifying aggregation messages from the processing queue (Figure 7). Right after the beginning of the process, the process picks up a message from the processing queue. It checks the destination of the message and if the destination is the node itself, it does not need to transfer the packet so the node invokes the regular process for receiving a message. If the message needs to transfer another node or a destination, we check if it has a flag for prohibit aggregation. If the message is allowed aggregation, the node checks whether the

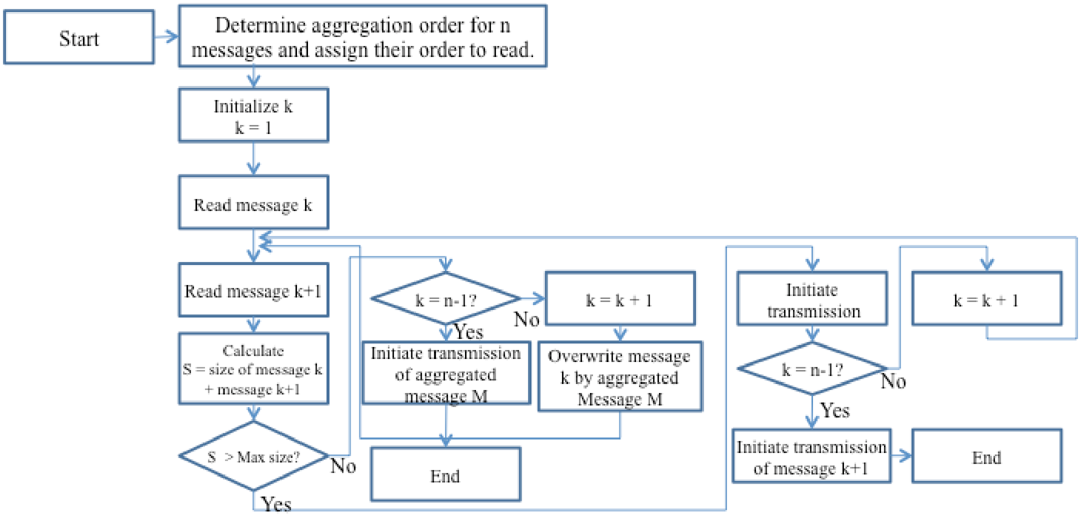


Figure 8: Message aggregation process

4.3 Identifying router capabilities for aggregation and disaggregation

To determine if a node can send an aggregated message to the next-hop or destination, the node needs to know if the next-hop node can restore the individual aggregated message. In other words, we have to identify the next-hop node that has the restoration function. There are many methods for achieving this. For this purpose, we can use the same communication protocol with the IoT message. We can also use other communication protocols. Or, we can adopt a Publish/Subscribe scheme for this [14]. In Figures 9 and 10, we show our CoAP-based implementation for inquiry, response, and notify messages for indicating the support of the restoration function. Note that we include text-based message for ease of understanding in these figures, but the CoAP is a binary encoded protocol so the actual CoAP payload does not show any text message. Figure 8 shows an inquiry message for the support of the disaggregation function.

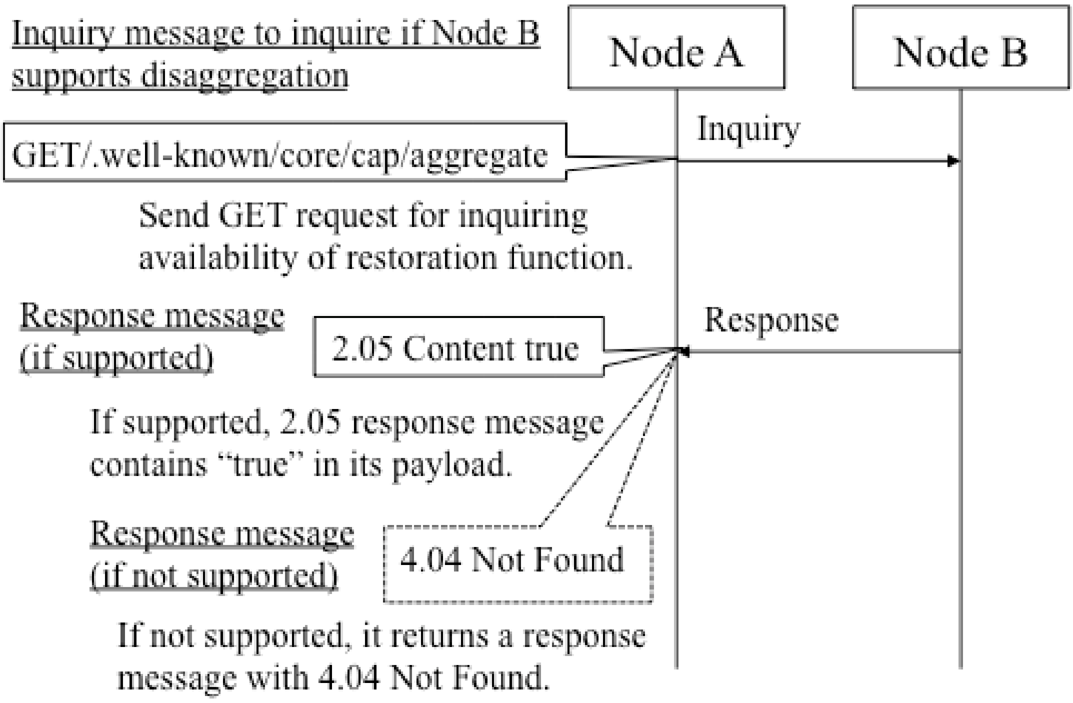


Figure 9: Inquiry and responses using CoAP

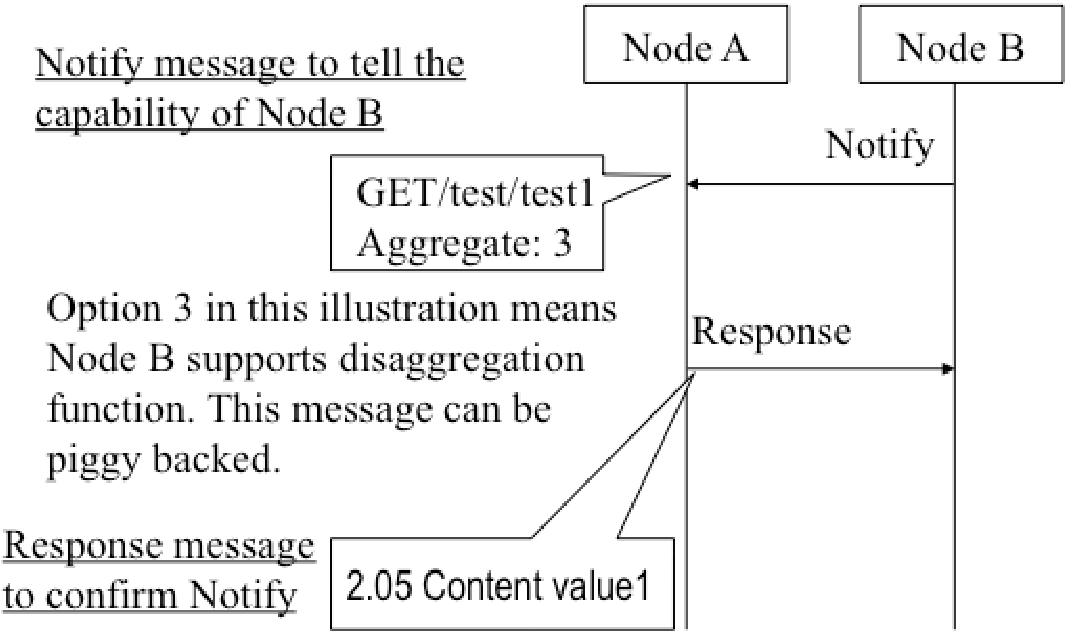
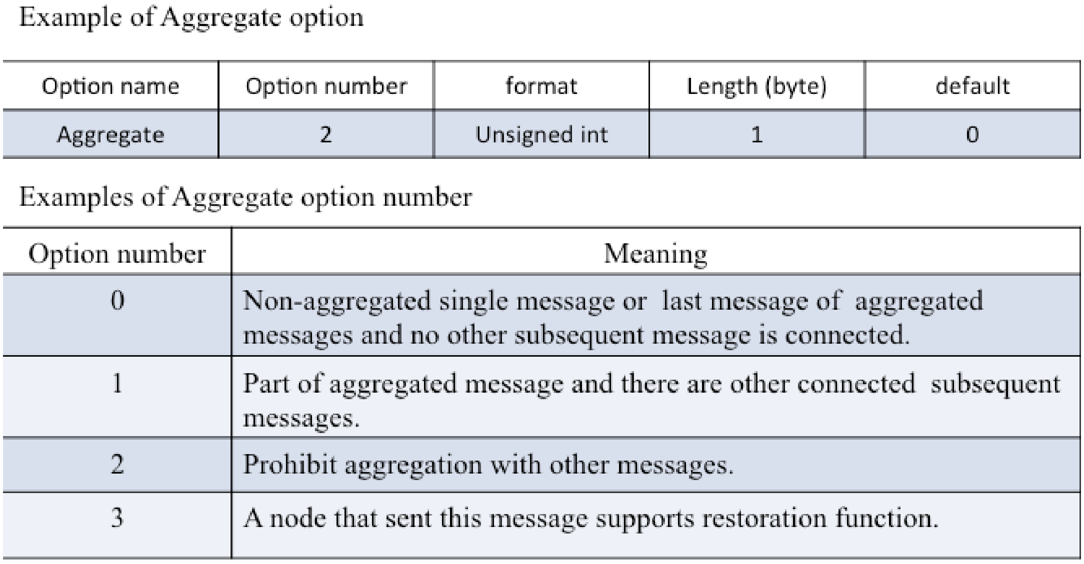


Figure 10: Notify using CoAP

The source node sends this message to the adjacent or further ahead node if the recipient of this message supports the disaggregation function. This message exchange occurs independent of the user data transfer; it is sent periodically or at the time of route-information renewal. Using this message exchange, Node A can gather up-to-date information of the capabilities of its nearby nodes. Figure 9 shows a notify message. A node uses this message to advertise its supporting capabilities to adjacent nodes. If a node changes its supporting capability depending on its load, it can use this notification message.

Table 1: Description of CoAP aggregate option for message aggregation, restoration, prohibition of aggregation



Example of Aggregate option

Option name	Option number	format	Length (byte)	default
Aggregate	2	Unsigned int	1	0

Examples of Aggregate option number

Option number	Meaning
0	Non-aggregated single message or last message of aggregated messages and no other subsequent message is connected.
1	Part of aggregated message and there are other connected subsequent messages.
2	Prohibit aggregation with other messages.
3	A node that sent this message supports restoration function.

We can use either the in-band or out-of-band method for this information exchange. Table 1 shows a case in which the in-band method relays on the CoAP. We define the aggregate option as a CoAP option. With a value specified by this option, we represent the supporting capability. If we uniquely define the value of this option within a network management domain, we can use the CoAP without converting it to another protocol in a wide area network. Since we use the CoAP for carrying IoT data traffic, CoAP-based in-band message exchange means we can piggyback messages such as inquiry and notify on the data traffic.

5 Evaluation on Power Consumption

In this section, we evaluate how the aggregation scheme reduces the power consumption of the entire network. Our objective does not aim to reduce single router power consumption. As we can easily recognize, adding a new function at edge router increase the power consumption. However, our strategy is to cut the total power consumption for the entire network. In [15], they measure power consumptions for commercial routers. Their findings include that the smaller the packet size is, the larger it consumes energy. Thus we can expect that if we use lager packet size in wide-area network, we might be able to compensate the energy increase incurred by aggregation function at edge routers. In order to validate this expectation, we rely on the model established in [16]. They model the power consumption of a router by that of measuring software router in detail. They analyze which part of router consumes energy by what kind of relationships. They identified that the total energy consumption is calculated as the summation of (1) CPU usage, (2) Memory usage, (3) Interface usage, and (4) energy usage during idle status. They empirically derived several key parameters and create a total energy consumption model as a function of packet forwarding rate. They create their model for analyzing caching function but their model can be used our case as well. This is because although their scope is for evaluation of caching function in Information Centric Network but we can utilize their model for our packet aggregation function as both of them need to process packets above layer 4. By comparing their model, we adopt almost all values of parameters in [16] except calculations for clock cycles of aggregation and forwarding rate. We use 7120 for their summation of F_1, 0 for F_2, and 9612 for their summation of F_3 in their equation (5).

We assume a simple 4-node model for our evaluation; two edge routers and two core routers. For core routers without aggregation function, we assign 2039 for F_3 and set all 0 for F_1 and F_2 to model that it does not perform aggregation function. We assume 4 CPU cores for our routers.

We assume the size of IoT short packet as 46 byte and aggregated frame size is 1500 byte. Therefore, about 30 short packets can be aggregated into a packet. By this assumption, we vary the input rate of the packet from 100Mbit/s to 10Gbit/s. Note that we use the software router for our both edge and core routers as we anticipated massive deployment of Network Function Virtualization (NFV) and software routers will widely be used in core network as well. The result is shown in Figure 11.

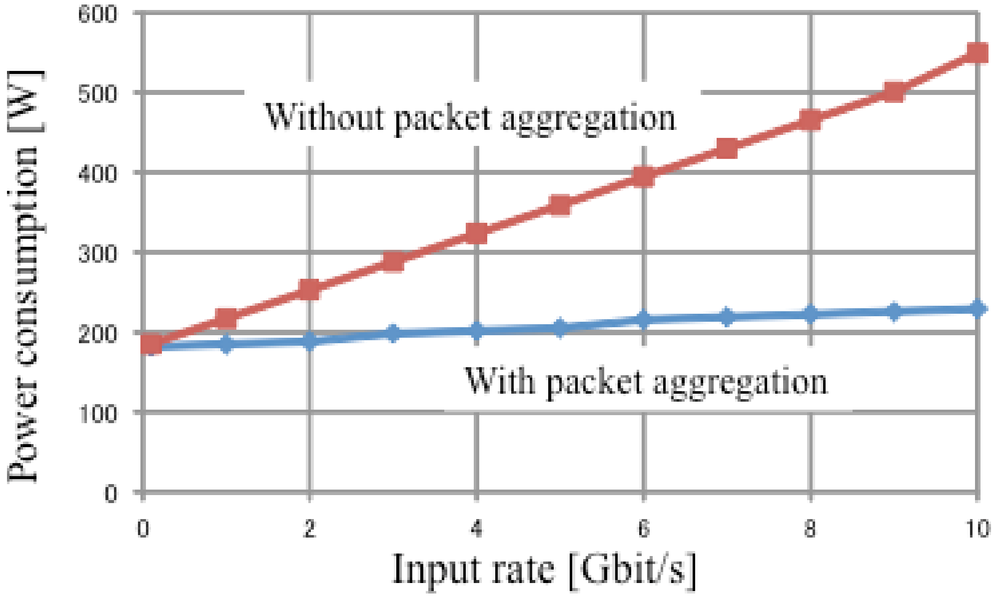


Figure 11: Comparison of power consumption with and without aggregation

As we can see from this figure, for small input rate, the difference for out reduction is small but the more increase the input rate, the more difference we observe in power consumptions. Although our evaluation is only one simple limited case and actual network configuration is more complex, we can see how packet aggregation can save the energy of the entire network.

By observing this result, we also are able to point out that workload reduction within wide-area network is closely related to power consumption of the entire network.

6 Related works

There are numerous works considering packet aggregation technique. Even by limiting to their area on IoT networking, many of them are focusing on wireless sensor networking. The authors of [11] discuss gateway placement problems for wires adhoc networks to optimize delay performance. In [12], they discuss data aggregation using anycast technique and design a protocol without maintaining explicit network structure. [17] discusses performance improvement by packet aggregation technique between a single segment of IEEE 802ac. Those works are focusing on performance improvement for wireless networking from various aspects but not considering end-to-end IoT networking.

In [18], they try to formulate packet aggregation technique using queueing model and consider how to minimize the system utilization. They modeled single entry network and mainly consider delay performance and system utilization.

The most relating work with us is [19]. They focused on end-to-end network and especially targeting IoT applications with small size packet to improve the performance of wide-area networks. Their objective is, however, to minimize the end-to-end delay of each packet to support real-time IoT applications. We agree with them that there exist some such use cases for IoT applications but in many cases, we believe IoT applications creating small packets are non-real-time one.

Our contribution of this paper is that we formulate a network architecture supporting end-to-end IoT network filled with small packets to a wide-area network. Since small packet is harmful for router performance, our objective is to mitigate the degradation of network performance and thus energy consumption of the wide-area network. We then realize this architecture using CoAP-based implementation and showed the algorithm for aggregation and disaggregation. We also evaluate how packet aggregation reduces the power consumption in wide-area network.

7 Conclusion

We proposed a packet aggregation scheme and described the requirements and an implementation for applying it to a wide area network. With our proposed aggregation scheme, we can reduce the burden on the routers in core network in the wide area Internet due to the huge amount of short packets.

From the architectural viewpoint, our scheme creates overlay networks. For small packets aggregated into a larger packet, they cannot perceive the existence of intermediate core routers. Therefore, there are overlay networks on the wide area Internet. Aggregation and Disaggregation points are the nodes for these overlay networks. We anticipate the emergence of a huge number of IoT devices connecting to the Internet. By constructing overlay networks, we not only can reduce the packet processing load in a router but also can create a logical network over the Internet. We can define this logical network based on the types of information created by devices. Different types of information have different requirements for network performance, such as delay and loss. If we can aggregate appropriately, we can create multiple logical networks with different characteristics. If we further extend the logical network to implement the meaning of sensor information, we can achieve networking based on the meaning or value of information [20].

We implemented our proposed scheme by extending CoAP as a type of transport layer. This approach treats CoAP as the end-to-end convergence layer. Therefore, our approach maintains end-to-end transparency at the transport layer level. It will be an appropriate solution for service providers who want to directly control their remote IoT devices. When we extend this approach for more general information centric networking with multiple service providers sharing the same IoT devices, consideration on the appropriate layers is left for a further study item from the viewpoint of logical networking by using aggregation.

One might think adding many rich capability overloading edge routers. We, however, thinks that recent edge routers can enhance their capability by adding daughter boards with CPUs or NPUs and thus additional functions can be supported by those daughter board without degrading their forwarding capability. In near future, we further anticipate massive deployments of Network Function Virtualization (NFV) technology for router. Therefore, many routers will be virtualized and located in datacenters supported by scalable server technologies. Of course this increase the power consumption at edge routers but on the other hand, as we showed in section 5, we can reduce the entire network energy consumption.

References

[1] R. Fielding, J. Gettys, J. Mogul, H. Frystyk, L. Masinter, P. Leach, and T. Berners-Lee, “Hypertext Transfer Protocol -- HTTP/1.1,” RFC2616, IETF, 1999.

[2] Z. Shelby, K. Hartke, and C. Bormann, “The Constrained Application Protocol (CoAP),” RFC7252, IETF, 2014.

[3] R. Ishibashi, T. Ohba, and A. Koike, “A packet aggregation method in wide area M2M/IoT networks,” Proc. IEICE General Conference, B-7-54, 2013, pp. 198.

[4] H. Schulzrinne, S. Casner, R. Frederick, and V. Jacobson, “ RTP: A Transport Protocol for Real-Time Applications,” IETF RFC3550, 2003.

[5] ITU-T, “B-ISDN ATM Adaptation Layer specification: Type 2 AAL,” Recommendation I.363.2, September 1997.

[6] ITU-T, “Segmentation and Reassembly Service Specific Convergence Sublayer for the AAL type 2,” Recommendation I.366.1, June 1998.

[7] ITU-T, “Pulse code modulation (PCM) of voice frequencies,” Recommendation G.711, November 1988.

[8] IEEE, “Local and metropolitan area networks-- Specific requirements-- Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications Amendment 5: Enhancements for Higher Throughput, ” 802.11n, IEEE Standard for Information technology, 2009.

[9] J. Hui (Editor), and P. Thubert, “Compression Format for IPv6 Datagrams over IEEE 802.15.4-Based Networks,” RFC6282, IETF, 2011.

[10] IEEE Computer Society, “IEEE Std. 802.15, 4-2006,” 2006.

[11] J. L. Wong, R. Jafari, and M. Potkonjak, “Gateway placement for latency and energy efficient data aggregation,” 29th Annual IEEE International Conference on Local Computer Networks, 2004, pp. 490– 497.

[12] K.W. Fan, S. Liu, and P. Sinha, “Structure-Free Data Aggregation in Sensor Networks,” IEEE Transactions on Mobile Computing, Vol. 6, Issue 8, 2007, pp. 929 – 942.

[13] E. Mlaih, and S. A. Aly, “Secure Hop-by-Hop Aggregation of End-to-End Concealed Data in Wireless Sensor Network,” Proc. 2nd IEEE MCN'08 workshop, Phoenix, AZ, 2008.

[14] P.T. Eugster, P.A. Felber, R. Guerraoui, and A.M. Kermarrec, “The many faces of publish/subscribe,” ACM Computer Survey, vol.35, no.2, 2003, pp.114-131.

[15] J. Chabarec, J. Sommers, P. Bardord, C. Estan, D. Tsiang, and S. Wright, “Power Awareness in Network Design and Routing” Proc. of The 27th Conf. on Computer Communications, 2008, DOI: 10.1109/INFOCOM.2008.93.

[16] T. Hasegawa, Y. Nakai, K. Ohsugi, J. Takemasa, K. Koizumi, and I. Pasaras, “Empirically Modeling How a Multicore Software ICN Router and an ICN Network Consume Power,” Proc. of the 1st ACM Conf. on Information-Centric Networking, 2014, pp. 157-166.

[17] B. Bellalta, J. Barcelo, D. Staehle, A. Vinel, and M. Oliver, “On Performance of Packet Aggregation in IEEE 802.11ac MU-MIMO WLANs,” IEEE Communications Letters, Vol. 16, Issue 10, 2012, pp. 1588-1591.

[18] J.H. Hong, and K. Sohraby, “On Modeling, Analysis, and Optimization of Packet Aggregation Systems,” IEEE Trans. on Communications, Vol. 58, No. 2, 2010, pp.660-668.

[19] A. Sawabe, K. Tsukamoto, and Y. Oie, “QoS-aware Packet Chunking Schemes for M2M Cloud Services” 28th Int’l Conf. on Advanced Information Networking and Applications Workshops, 2014, pp. 166-173.

[20] V. Jacobson, D. K. Smetters, J. D. Thornton, M. F. Plass, N. H. Briggs, and R. L. Braynard, “Networking named content,” Proc. of the 5th Int’l Conf. on Emerging networking experiments and technologies, 2009.