J Inf Process Syst, Vol.15, No.2, pp.440~448, April 2019
https://doi.org/10.3745/JIPS.03.0113

ISSN 1976-913X (Print)
ISSN 2092-805X (Electronic)

JOURNAL OF INFORMATION PROCESSING SYSTEMS JIPS

Scalable Service Placement in the Fog Computing Environment for the IoT-Based Smart City

Jonghwa Choi* and Sanghyun Ahn**

Abstract

The Internet of Things (IoT) is one of the main enablers for situation awareness needed in accomplishing smart cities. IoT devices, especially for monitoring purposes, have stringent timing requirements which may not be met by cloud computing. This deficiency of cloud computing can be overcome by fog computing for which fog nodes are placed close to IoT devices. Because of low capabilities of fog nodes compared to cloud data centers, fog nodes may not be deployed with all the services required by IoT devices. Thus, in this article, we focus on the issue of fog service placement and present the recent research trends in this issue. Most of the literature on fog service placement deals with determining an appropriate fog node satisfying the various requirements like delay from the perspective of one or more service requests. In this article, we aim to effectively place fog services in accordance with the pre-obtained service demands, which may have been collected during the prior time interval, instead of on-demand service placement for one or more service requests. The concept of the logical fog network is newly presented for the sake of the scalability of fog service placement in a large-scale smart city. The logical fog network is formed in a tree topology rooted at the cloud data center. Based on the logical fog network, a service placement approach is proposed so that services can be placed on fog nodes in a resource-effective way.

Keywords

Fog Computing, Internet of Things (IoT), Service Management, Service Placement, Service Provisioning

1. Introduction

Nowadays, the Internet of Things (IoT) is attracting many researchers because it allows various types of devices such as smartphones, cars and refrigerators to be attached to the Internet via wired and wireless communication links. IoT allows to sense and monitor the target environment, to collect and analyze the sensed information and to actuate based on the feedback information from the analysis. Many IoT applications have stringent timing requirements which may not be met by cloud computing because of long-distance message transmissions through the network between the cloud data center and IoT devices. Hence, the concept of fog computing has been introduced by CISCO in 2012 [1]. Fog nodes are located near to IoT devices (or end devices) and in between IoT devices and the cloud data center which is the last resort of providing services to IoT devices.

In fog computing, fog nodes located in the proximity of IoT devices provide services to them in lieu of

Manuscript received February 28, 2019; accepted March 23, 2019.
Corresponding Author: Sanghyun Ahn (ahn@uos.ac.kr)
* Dept. of Computer Science and Engineering, University of Seoul, Seoul, Korea (David13@uos.ac.kr, ahn@uos.ac.kr)

the cloud data center without causing long latency by executing the lightweight virtualized service images allocated from the cloud data center. But, due to the limited capacity (CPU, storage, etc.) of fog nodes, only a subset of services can be placed on each fog node (i.e., it is infeasible to place all the services on each fog node).

In recent years, service management has become one of the main issues of fog computing because the optimal selection of a fog node is important from the performance-wise aspect of executing a service (or an application). In the literature, service placement, provisioning, deployment and allocation are mentioned most frequently in line with fog service management. In [2], service placement is defined as a service management decision about where a service is to be executed. Service placement, allocation, deployment and provisioning are interchangeably used in the literature. Service offloading is newly introduced in [2] where service offloading is differentiated from service placement in that service offloading moves (or offloads) an already-placed service to one or more other fog nodes for the purpose of performance enhancement.

In most of the literature on service placement, the service placement problem is formulated as an optimization problem with an objective and several constraints, and proved to be NP-hard along with a proposed heuristic approach. In general, the problem space is constrained in placing a service on demand on one or more fog nodes satisfying the given service requirements, such as capacity, computing, latency, etc., for the execution of the service. Placing services on fog nodes in the per-service request-based (i.e., on demand) manner may result in improper resource usage of fog nodes due to redundantly or non-optimally placed services. If we provision fog nodes with services based on the pre-obtained service demands, the resources of fog nodes can be effectively utilized. Up to our knowledge, this issue has not been addressed so far. For the service placement based on the pre-obtained service demands, IoT devices are required to send service requests to the cloud data center as shown in Fig. 1. Then, the cloud data center decides the right fog node for the execution of the service and places the corresponding service on the fog node if the fog node has not been already deployed with the service.

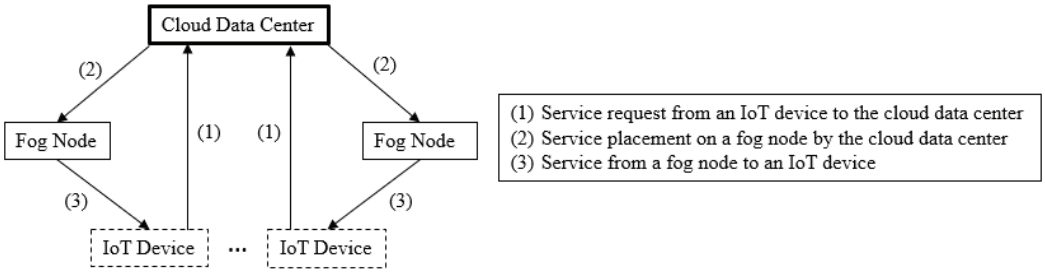


Fig. 1. The procedure of handling the service requests of IoT devices.

In placing fog services, we need to consider various factors like the resource usage and the locations of fog nodes and the latency and resource requirements and the locations of service requesters (i.e., IoT devices), which is too complex to get an optimal solution in polynomial time. In this article, we focus on the issue of placing services on fog nodes and go through the recent literature relevant to the issue. Furthermore, we propose a fog computing infrastructure, the logical fog network, that can satisfy the requirements of IoT services in a scalable way with considering the characteristics of the large-scale smart city. The logical fog network is formed in a tree topology to reflect the hierarchical nature of the

administrative districts of a smart city. As a realistic example, we construct the logical fog network specifically formed for Seoul, the capital city of Korea. Moreover, we propose a service placement mechanism based on the logical fog network satisfying the service requirements of IoT devices and the resource capacities of fog nodes. The proposed service placement mechanism targets to minimize the number of services placed on fog nodes in order to optimize the resource utilization of fog nodes. From the simulation-based performance evaluation, we show that our service placement mechanism based on the logical fog network can accommodate more service requests of IoT devices than the per-service request-based mechanism.

The rest of the article is organized as follows. In Section 2, we describe the recent research trends in service placement in the fog computing environment. In Section 3, we introduce the concept of the logical fog network for the scalability of service provisioning in the large-scale smart city environment and propose a service placement approach based on the logical fog network. The performance of the proposed service placement mechanism is evaluated in Section 4. Finally, Section 5 concludes this article.

2. Research Trends in Fog Service Placement

The issue of service placement or provisioning has been studied by many researchers for the fog computing environment with various objectives and constraints.

In [3], the authors focused on the application (service) provisioning problem in the fog-cloud environment from a network perspective to guarantee the quality of service (QoS) of application data streams in terms of transmission delay and bandwidth for each application. In [3], application provisioning is defined to find the host node (i.e., fog node) and data routing for a specific application. The authors formulate the issue as the single application provisioning (SAP) and the multi-application provisioning (MAP) problems. SAP determines the path to the fog node satisfying the bandwidth and delay requirements of data traffics from end devices for one application. MAP is the extension of SAP for multiple applications from the perspective of network link capacity (bandwidth) usage.

In [4], the deployment of multi-component application in the fog hierarchy is defined as the components deployment problem (CDP) which determines candidate fog nodes to be deployed with the components of an application according to application specific QoS requirements of end devices on latency and bandwidth, software and hardware capabilities of fog nodes and business policies. They targeted to deploy large scale applications composed of multiple components that can be independently deployable and work together in the fog infrastructure, and proved that the CDP problem is NP-hard by the reduction from the subgraph isomorphism problem (SIP). The CDP problem is to find the optimal deployment of a single application based on the service requests of end devices in a centralized manner.

Both [3] and [4] determine the fog nodes to be provisioned with services in a centralized way for a given set of service requests with taking into consideration of each individual fog node, not of the entire fog nodes, from the perspective of resource usage. The consideration on the capacity of each fog node is just for checking the relevant constraint, but not for optimizing the overall resource usage of fog nodes. The latter is more appropriate for enhancing the scalability of the fog infrastructure in terms of resource usage.

Related to the resource provision of fog nodes, Skarlat et al. [5,6] proposed a fog computing architecture composed of fog colonies each of which consists of one fog orchestration control node and multiple fog

cells. The fog orchestration control node is a fog cell with extended functionalities like managing the fog cells in the same fog colony and the other connected control nodes. The fog cell is the software components running on a fog node. The fog orchestration control node keeps all the services in its service registry, located in the low-cost abundant storage unit, and deploys the corresponding service to the fog cell covering the service requester on demand. In the scheme, all the services are kept in the storage of the fog orchestration control node and deployed on fog nodes per service request by the control node. This will incur overwhelming communication overhead and delay in deploying services on demand from fog orchestration control nodes to fog cells.

In [2], the authors dealt with the issue of service offloading for the execution of complex IoT services in the fog computing environment. They pointed out that complex IoT services may not be executed on a single fog node because of the limited capacity of fog nodes and service offloading can be the solution to overcome this deficiency. Service offloading is to offload service executions from a fog node to another set of fog nodes with considering the resources of the cloud data center and fog nodes. For service offloading of complex IoT services, service atomization and parallel resource allocation were introduced.

In [7], a lightweight framework FOGPLAN for QoS-aware dynamic provisioning for fog services is proposed. The authors formulated an optimization problem and proposed two greedy algorithms. In FOGPLAN, services are dynamically deployed on or released from fog nodes to minimize the resource cost and satisfy the latency and the QoS constraints.

Mahmud et al. [8] proposed a latency-aware application module placement in the fog computing environment and formulated a linear programming problem that minimizes the number of computationally active fog nodes. In their scheme, the computational component of a fog node can be turned off if all the deployed application modules of the fog node are relocated to the other fog nodes for further execution.

The mechanisms mentioned in [7,8] are along the same lines in that the release of fog nodes [7] and the turning-off of fog nodes by service relocation [8] can have the same effect of minimizing the resource usage of fog nodes by reducing active fog nodes.

In [9], the authors proposed a cost-efficient availability guaranteed deployment of IoT services for the network function virtualization (NFV)-enabled fog computing environment. In their scheme, the improvement potential of virtual network functions (VNFs) is measured for the availability of service function chains (SFCs). Moreover, VNF redundancy deployment mechanisms are presented for the protection of SFCs.

In [10], the authors proposed a four-tier combined fog-cloud architecture on which the service allocation problem is formulated as an integer linear programming problem. The objective of the problem is to minimize the total delay of providing services.

In [11], the microservice architecture is considered, where applications are composed of cascades of coupled microservice modules and formulate a mixed integer nonlinear program problem for the fog resource allocation from the perspective of the edge-infrastructure owner. The objective is to deploy the containerized fog applications so that the owner revenue is maximized with satisfying the requirements of applications. As a heuristic algorithm, the authors proposed an iterative greedy algorithm for application placement.

As we have seen so far, in the literature on fog service placement, the objective function and the constraints are determined by considering various aspects of the fog computing environment, such as application (or service) architecture, fog node architecture, fog-cloud architecture, network conditions,

network technology, revenue of the fog infrastructure owner, etc. We can classify the service placement mechanisms into three categories; application-perspective, fog resource-perspective and network-perspective. Service placement mechanisms in the application-perspective category consider the application architecture like a multi-component application running on multiple fog nodes in a distributed way. A service placement mechanism belongs to the fog resource-perspective category if it considers the resource management of fog nodes specifically for service execution. Service placement mechanisms in the network-perspective category consider network conditions like link capacity and paths between fog nodes and IoT devices—e.g., the application-perspective category [2,4,11], the fog resource-perspective category [5-8], and the network-perspective category [3,9,10]. In recent several years, the fog service placement issue has been tackled by many researchers from various aspects, but none has considered the overall resource usage of fog nodes. In this article, we focus on how to place services on fog nodes so that the overall resource usage of fog nodes can be optimized.

3. Logical Fog Network and Service Placement

Fog nodes are deployed based on a specific deployment strategy with considering the end devices requesting services and various environmental conditions, etc. The locations of fog nodes can be anywhere in the given network, determined by the deployment strategy which is out of the scope of this article. In general, the given physical network topology of the fog computing environment is an irregular mesh which increases the computational complexity of determining the optimal fog nodes to be provisioned with services. For the sake of scalability, we propose to form a logical tree topology of fog nodes rooted at the cloud data center and name this as the logical fog network, resulting in the reduced computational complexity.

We assume only one cloud data center that has the information on the logical fog network such as the location and the capacity of each fog node. The logical fog network can be easily applied to the case of multiple data centers. From the perspective of service placement, the cloud data center can be assumed to have unlimited capacity so that any requirements, except for latency, of end devices can be accommodated; that is, if there exist any service requests not accommodated by any fog nodes, the cloud data center can cover those service requests.

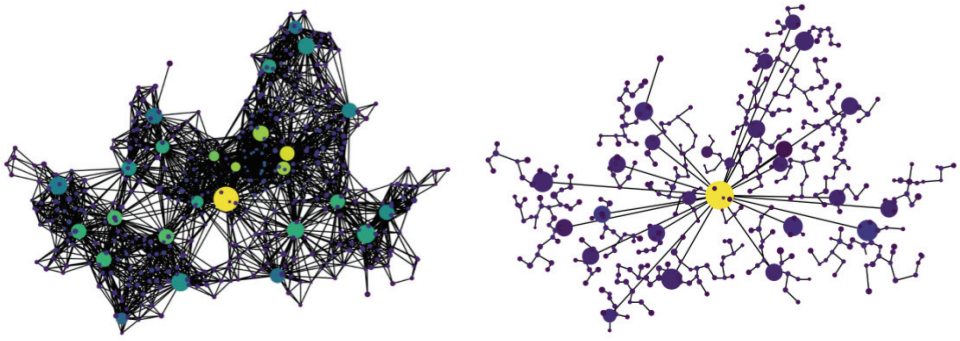
The cloud data center is the root of the logical fog network and the rest of the fog nodes are included in the logical fog network in the decreasing order of the capacity. At first, a given number (i.e., a decision parameter) of fog nodes with larger capacities are included in the logical fog network as the child fog nodes of the cloud data center. Then, for each child fog node of the cloud data center, its path delay to the cloud data center is modified to a value (i.e., a decision parameter) less than its original delay. After that, each fog node not in the logical fog network is included in the logical fog network one by one via the minimum delay path to the cloud data center so that the latency requirement of each IoT device can be satisfied.

The capacity is used as the criterion of selecting the child fog nodes of the cloud data center because a fog node with more capacity can have more services deployed and higher possibility of serving more requests of end devices. Through the above-described procedure of the logical fog network construction, we can obtain a logical fog network that is appropriate for less services provisioned thanks to the tree

topology and for lower delays from end devices to fog nodes thanks to the shortest path branches. The reason for adjusting the delay of the path between the cloud data center to each of its child fog nodes is to achieve the load balancing effect by making high capacity fog nodes share the burden of computing load with the cloud data center.

Fig. 2(a) is an example physical fog network designed for Seoul and Fig. 2(b) is the logical fog network constructed by the above-mentioned construction procedure for the physical fog network in Fig. 2(a). The physical fog network in Fig. 2(a) is built by locating one cloud data center (the largest dot in the figure) at the geographic center of Seoul and 449 fog nodes according to the two-level administrative districts of Seoul. The capacity of a fog node is determined proportional to the number of households of the corresponding administrative district. In Fig. 2, the dots except for the largest dot (i.e., the cloud data center) are fog nodes and the size of a dot implies the capacity of the corresponding fog node. The weight of an edge between two fog nodes is determined proportional to the physical distance between them.

Based on the logical fog network, now is the time to consider how to place services on fog nodes. For service provisioning, we assume that the amount of service demands is collected during a priori time interval for each service type. With given the service demand information, the fog nodes are checked from the highest to the lowest level of the logical fog network. The reason for considering fog nodes based on the level is that higher level fog nodes have higher possibility of serving more IoT devices than lower level fog nodes. For each fog node, the amount of demands is used as the criterion of choosing services to be placed on the fog node because more demanded services tend to serve more IoT devices.



(a) (b)

Fig. 2. An example fog network for Seoul: (a) a physical network with fog nodes in a mesh topology and (b) a logical fog network in a tree topology for the physical network of (a).

4. Performance Evaluation

For the performance analysis, we have carried out simulations using Python and the NetworkX package. As for the simulation environment, the network in Fig. 2 is used and 1,000 service types are set to be requested by IoT devices. We adopt a long-tailed distribution of service demands; that is, a small set of the service types is heavily requested and the rest of the service types are not frequently requested. We adopt a long-tailed distribution because a set of popular services are heavily used in the real world. For the simulations, the long-tailed distribution of service demands in Fig. 3 is used. In Fig. 3, about 10% of 1,000 service types are heavily requested by the end devices.

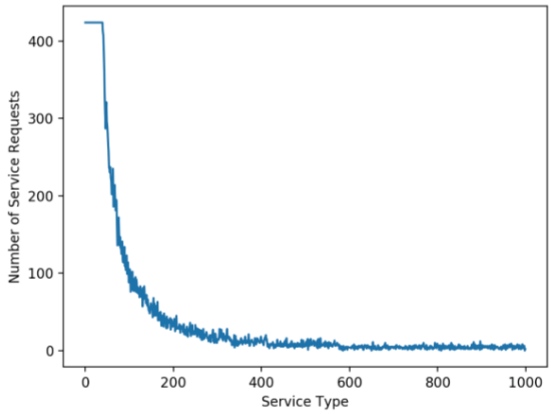


Fig. 3. A long-tailed-distribution of service demands of 1,000 service types.

For the sake of simplicity, we assume that the resource requirement of each service type is the same (i.e., each service request requires 1 unit of fog resources) and the delay requirement of each service type is different. In order to see performance differences more clearly, we limit the maximum resource capacity of the cloud data center to 300, 500 and 750. We compare our logical fog network-based service placement mechanism with the per-service request-based mechanism. In the per-service request-based service placement mechanism, each service request is served on demand by the fog node which is the closest to the service requesting IoT device with satisfying the resource and the delay requirements of the service type. Because the computational complexity of finding the right fog node (satisfying the given requirements) in the given mesh network is high, we design the per-service request-based mechanism to limit the search space to the 2-hop neighboring fog nodes of the service requester. If no right fog node is found in the 2-hop search space, the per-service request-based mechanism looks for the fog nodes on the shortest path from the service requester to the root node (i.e., the cloud data center).

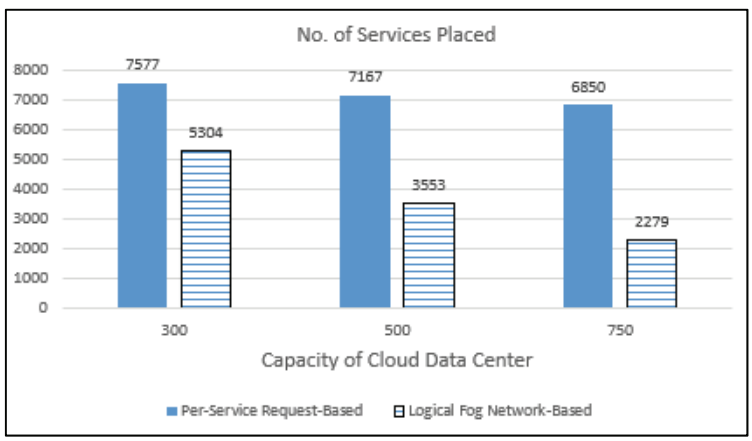


Fig. 4. The number of services placed on fog nodes for various cloud data center capacities.

Fig. 4 shows the total number of services placed on the fog nodes for 42,210 service requests, which indicates the efficiency of resource utilization of fog nodes. We can observe that our logical fog network-based mechanism significantly outperforms (almost a third to two thirds of) the per-service request-based

mechanism in all cases. That is, our mechanism requires less services placed on the fog nodes (i.e., less resources of the fog nodes) in accommodating 42,210 service requests compared with the per-service request-based mechanism. We have varied the maximum resource capacity of the cloud data center in order to see the performance of our mechanism in various scenarios. As the maximum capacity of the cloud data center increases, the number of services placed on fog nodes decreases in both mechanisms because the cloud data center is more capable of accommodating service requests (in other words, fog nodes are less deployed with services because they have less chances of serving the given requests).

5. Conclusion

Smart cities are being realized with the help of various IT technologies such as IoT, AI, big data, cloud computing, etc. We can easily anticipate that there will be enormous IoT devices deployed in a smart city, which may not be effectively served by cloud computing because of inevitable physical delay from the cloud data center to IoT devices. Thus, fog computing becomes an attractive alternative to cloud computing and the service placement issue has been studied by many researchers for the optimized operation of fog computing. We have gone through recently proposed service placement mechanisms with various objectives and constraints, and proposed a scalable fog computing infrastructure, the logical fog network, taking into consideration of the hierarchical administrative district structure of the smart city. In designing the logical fog network, we aimed to achieve load balancing among fog nodes with satisfying the service requirements posed on fog nodes. Also, we proposed a service placement mechanism based on the logical fog network and the pre-obtained service demands. From the simulation results, we observed the effectiveness of our mechanism in terms of the resource usage of fog nodes. Service placement based on the pre-obtained service demands can enhance the overall resource usage of fog nodes, but a service migration strategy is required for gradual service positioning in the fog infrastructure and left for further study.

Acknowledgement

This work was supported by the 2018 Research Fund of the University of Seoul.

References

[1] F. Bonomi, R. Milito, J. Zhu, and A. Addepalli, "Fog computing and its role in the Internet of Things," in *Proceedings of the first edition of the MCC Workshop on Mobile Cloud Computing*, Helsinki, Finland, 2012, pp. 13-16.

[2] V. B. Souza, X. Masip-Bruin, E. Marin-Tordera, S. Sanchez-Lopez, J. Garcia, G. J. Ren, A. Jukan, and A. J. Ferrer, "Towards a proper service placement in combined Fog-to-Cloud (F2C) architectures," *Future Generation Computer Systems*, vol. 87, pp. 1-15, 2018.

[3] R. Yu, G. Xue, and X. Zhang, "Application provisioning in FOG computing-enabled Internet-of-Things: a network perspective," in *Proceedings of IEEE Conference on Computer Communications (INFOCOM)*, Honolulu, HI, 2018, pp. 783-791.

[4] A. Brogi and S. Forti, "QoS-aware deployment of IoT applications through the fog," *IEEE Internet of Things Journal*, vol. 4, no. 5, pp. 1185-1192, 2017.

[5] O. Skarlat, M. Nardelli, S. Schulte, and S. Dustdar, "Towards QoS-aware fog service placement," in *Proceedings of 2017 IEEE 1st international conference on Fog and Edge Computing (ICFEC)*, Madrid, Spain, 2017, pp. 89-96.

[6] O. Skarlat, M. Nardelli, S. Schulte, M. Borkowski, and P. Leitner, "Optimized IoT service placement in the fog," *Service Oriented Computing and Applications*, vol. 11, no. 4, pp. 427-443, 2017.

[7] A. Yousefpour, A. Patil, G. Ishigaki, I. Kim, X. Wang, H. C. Cankaya, Q. Zhang, W. Xie, and J. P. Jue, "FogPlan: a lightweight QoS-aware dynamic fog service provisioning framework," *IEEE Internet of Things Journal*, 2019. http://doi.org/10.1109/JIOT.2019.2896311.

[8] R. Mahmud, K. Ramamohanarao, and R. Buyya, "Latency-aware application module management for fog computing environments," *ACM Transactions on Internet Technology (TOIT)*, vol. 19, no. 1, article no. 9, 2018.

[9] N. T. Dinh and Y. Kim, "An efficient availability guaranteed deployment scheme for IoT service chains over fog-core cloud networks," *Sensors*, vol. 18, no. 11, article no. 3970, 2018.

[10] V. B. C. Souza, W. Ramirez, X. Masip-Bruin, E. Marin-Tordera, G. Ren, and G. Tashakor, "Handling service allocation in combined fog-cloud scenarios, in *Proceedings of 2016 IEEE International Conference on Communications (ICC)*, Kuala Lumpur, Malaysia, 2016, pp. 1-5.

[11] F. Faticanti, F. De Pellegrini, D. Siracusa, D. Santoro, and S. Cretti, "Cutting throughput on the edge: app-aware placement in fog computing," 2018 [Online]. Available: https://arxiv.org/abs/1810.04442.

Jonghwa Choi https://orcid.org/0000-0001-7345-6517

He is currently in the master's degree program at the University of Seoul. His research interests include wireless networks and computing such as mobile ad hoc networks, vehicular communication networks, IoT, wireless sensor networks, fog computing, etc.

Sanghyun Ahn https://orcid.org/0000-0001-7640-4480

She received B.S. and M.S. degrees in Department of Computer Engineering from Seoul National University in 1986 and 1988, respectively, and Ph.D. degree in Department of Computer and Information Science from University of Minnesota in 1993. She has been with University of Seoul as a Professor since 1998 and, before that, served as an Assistant Professor at Sejong University since 1994. Her research areas include wireless and wired networks such as Internet, mobile ad hoc networks, vehicular communication networks, wireless sensor networks, IoT, etc.