Green Data Centers for Cloud-Assisted Mobile Ad-Hoc Networks in 5G

Nguyen Dinh Han, Yonghwa Chung and Minho Jo

Abstract

Cloud-assisted mobile ad-hoc networks are expected to be so popular in future fifth generation (5G) mobile network because the significantly faster performance of 5G communications enables clouds to provide realistic services. On the other hand, the energy consumption problem will become serious due to the highly increased cloud computing speed. Mobile communication link loss occurs when a mobile device leaves the network. Link loss can cause serious energy consumption when the lost link is directly connected to a cloud data server, which is required to do excessive searching and link routing transactions. None of the existing normal methods have solved the problem. We show how this problem can be solved using our proposed mechanism. The experimental results demonstrate that our proposed mechanism saves nine times more energy than normal methods.

Energy savings are now a mandatory demand of most aspects of information and communications [1]. Today, scientists are asked to create not only powerful communication systems but systems that consume less energy. This request has a significant impact on the design of new wireless communication systems [2]. It also requires improvement in existing ones.

Mobile ad-hoc networks born in the 1970s are a type of wireless network that allows us to exchange data in a very convenient way. Although limited in power and capability, mobile ad-hoc networks have proven themselves as convenient infrastructure-free communication tools. Supporting mobility, allowing better connectivity and performance when moving between different networks, they promise an important contribution to the future evolution of the Internet [3].

Recently, cloud computing has been widely recognized by both industry and academia. Advances in cloud computing have made it possible to provide infrastructure, platform, and software as services for users from any computer/device with an Internet connection. Leveraging cloud computing in mobile ad-hoc networks gives birth to rich applications [4]. In particular, for future fifth generation (5G) mobile ad-hoc networks, the study of cloud-assisted service computing can be a hotter topic than ever before.

The combination of cloud computing with mobile ad-hoc networks in 5G obviously creates new powerful networks by giving additional advantages, significantly reduced computing, and longer lasting mobility in the network, respectively. These advantages have facilitated the quick growth of richer applications and services, both of which require the removal of heavy computing from the mobile device, and so the removal allows the mobile device to have longer lasting mobility [4]. However, we have faced a heavy energy consumption problem in cloud computing if a cloud provides data services. In addition, the extremely higher aggregate data rates and the much lower latencies required by 5G will introduce more energy-consuming technologies (e.g. massive MIMO, native support for M2M communication) [2]. The rapidly-growing number of richer applications in cloud-assisted mobile ad-hoc networks will make the energy consumption problem worse. It is an urgent need of both hardware and software solutions to make sure that those networks are more energy efficient, or "greener."

In this article we make an issue of excessive energy consumption in a cloud by the existing normal operational methods when link loss occurs in cloud data servers. Because a mobile device can often leave a mobile ad-hoc network connected directly to a cloud, the link between the cloud data server and the mobile device will often be lost. If a link toward a cloud data server located between the cloud and the mobile ad-hoc network is lost, another cloud data server connected directly to the mobile ad-hoc network should run searching and link routing transactions to find the location of the lost cloud data server. The searching and routing transactions cause excessive energy consumption. Due to the frequency of link loss in mobile ad-hoc networks [4, 5], this problem is critical. Thus, we propose a mechanism to solve excessive energy consumption in cloud data centers. Our proposed mechanism does not require re-searching and re-link routing for the lost cloud data server by way of updating link loss information in a content map and sending the location of queried data kept in the content map. Therefore, additional searching and routing transactions are not required to find the location of the lost cloud data server. Finally, we show the results of experimental work to demonstrate significant energy savings by our proposed mechanism compared to the existing normal method. We also discuss the analysis of the experimental results and future research topics.

Nguyen Dinh Han is with Hung Yen University of Technology and Education.

Yonghwa Chung and Minho Jo (minhojo@korea.ac.kr) are with Korea University.

This research was supported by a Korea University grant.

Related Work

Increasing attention has been paid to energy efficiency in information and communications. In essence, the problem of maximizing the energy efficiency in this area and its derivations are optimization problems. Hence, even though existing solutions are efficient under some limited conditions, there is always room for new contributions. The summary of general energy savings techniques for current and future Internet configurations with different approaches is given in [3]. The specific solutions for mobile cloud computing are introduced in [4–8].

In 2010 B. Andreas *et al.* [1] studied the energy efficiency of future information and communication technologies (ICT), and they raised an emergency call for the development of green ICT-based smart solutions. According to the authors' claims, to attain truly green computing services in a system, a comprehensive approach toward energy efficiency must be established. This involves all system layers and aspects, including both servers and services. Regarding data centers, networks, protocols, and applications as parts of a system, the authors affirmed the feasibility of the development of energy-aware mechanisms to reduce energy consumed by services in that system. Indeed, mechanisms of this sort have been created for several systems, as mentioned below.

In 2013, a framework called *HetNet* was introduced to support mobile cloud computing applications [5]. Offloading techniques developed within *HetNet* allow mobile multimedia applications to rely upon cloud services in order to reduce the energy consumption of mobile devices. Indeed, the offloading techniques enable mobile devices to offload their tasks to resourceful servers in the cloud. Furthermore, if a mobile device has multiple interfaces, the minimum energy consumed by its offloading services can be obtained using the technique given in [9]. Also in 2013, an energy saving mechanism was developed for cloud servers. An energy consumption, flow scheduler, and resource allocation strategy were applied as a middleware scheduling scheme in an integrated manner [8]. This scheme is designed to map service requests to the most appropriate virtual machine cloud servers, according to a specific goal function, for the minimum energy cost and waiting time. The role of energy saving mechanisms for mobile devices and cloud servers, as pointed out in [7], is indispensable to MCC systems.

A specific system (e.g. [10, 11]) and applications may require particular energy saving mechanisms. In [10], the effectiveness of WiFi Direct technology has been examined and evaluated for device-to-device communications. Since energy saving mechanisms in current WiFi networks is insufficient, two new protocols have been developed for WiFi Direct: the *Opportunistic Power Save* protocol and the *Notice of Absence* protocol. Similarly, the effectiveness of a new intelligent prefetching technique for mobile video streaming applications is also verified in [11]. Although the evaluation of energy efficiency is not provided, the prefetching technique may help reduce energy consumption in network transportation because it takes into account link quality. Other energy-saving mechanisms for vehicular ad-hoc networks can be found in [12].

The fifth generation mobile ad-hoc network with more energy-consuming technologies is coming [2]. As we mentioned in the introduction, the problem of link loss in mobile ad-hoc environments is critical in energy savings. Although the above energy-saving mechanisms regarded link loss more or less as a substantial challenge, there are no solutions to deal with that challenge in the mentioned mechanisms. This can lead to a significant drop in service performance and, in turn, an increase in energy consumption. Therefore, it motivates us to complete this work.

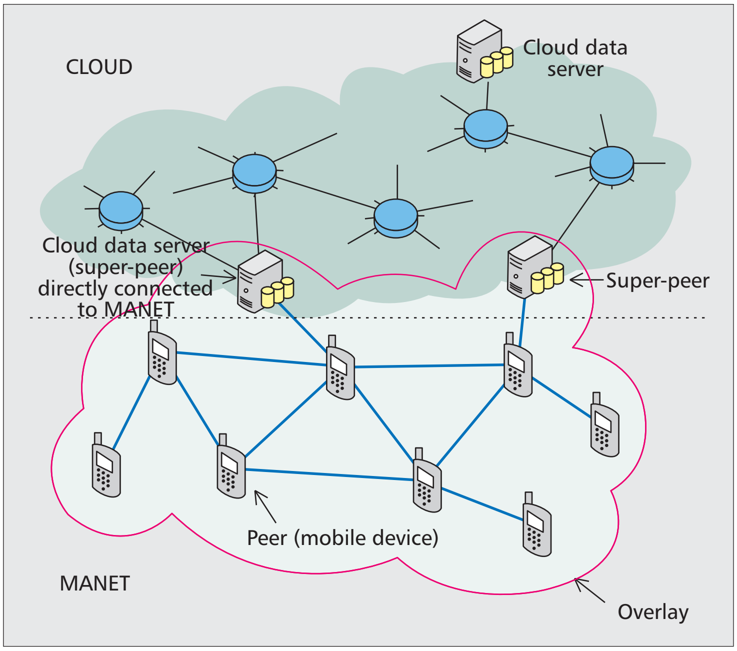

Figure 1. Illustration of the system architecture.

Problem Definition

System Architecture

We consider a cloud-assisted mobile ad-hoc network (MANET) in 5G that is designed to support multimedia and data services running on mobile devices. The network consists of a cloud and a mobile ad-hoc network. The cloud is formed by interconnected data servers. The mobile ad-hoc network is a dynamic set of mobile devices (computers, smartphones, sensors, etc.) that communicate with each other over wireless links. The advantages of the mobile cloud network (e.g. support for richer applications, saving on mobile device energy consumption) for multimedia services have been studied in [4–7]. In this work, we investigate aspects of energy efficiency when using cloud-assisted mobile ad-hoc networks for data services. In reality, data searching (or searching, for short) is most important in cloud-assisted mobile ad-hoc networks for multimedia and data services. Therefore, we focus on developing the new mechanisms of two transactions aimed at energy savings.

We use the concept of an *overlay network* as a service delivery platform in our cloud-assisted mobile ad-hoc network (or just the network, for short). Naturally, the overlay organization reflects a logical view of the network. Thus, using the concept, we can focus on the operation of network transactions regardless of the complexity and chaos of the underlying physical network (refer to [13] for details).

Our cloud-assisted mobile ad-hoc network, which we refer to as the Peer-to-Peer (P2P) network *overlay* (or simply an *overlay*), is a collection of mobile devices called peers and cloud data servers that are adjacent to a MANET (see Fig. 1). Furthermore, we will refer to cloud data servers as *super-peers* when they are logically connected to peers in a MANET, in order to send queried data or relay them from a cloud data server inside the cloud. If a cloud data server adjacent to a MANET is not logically connected to MANET, it cannot become a super-peer until it is. The architecture of the network and its overlay are shown in Fig. 1.

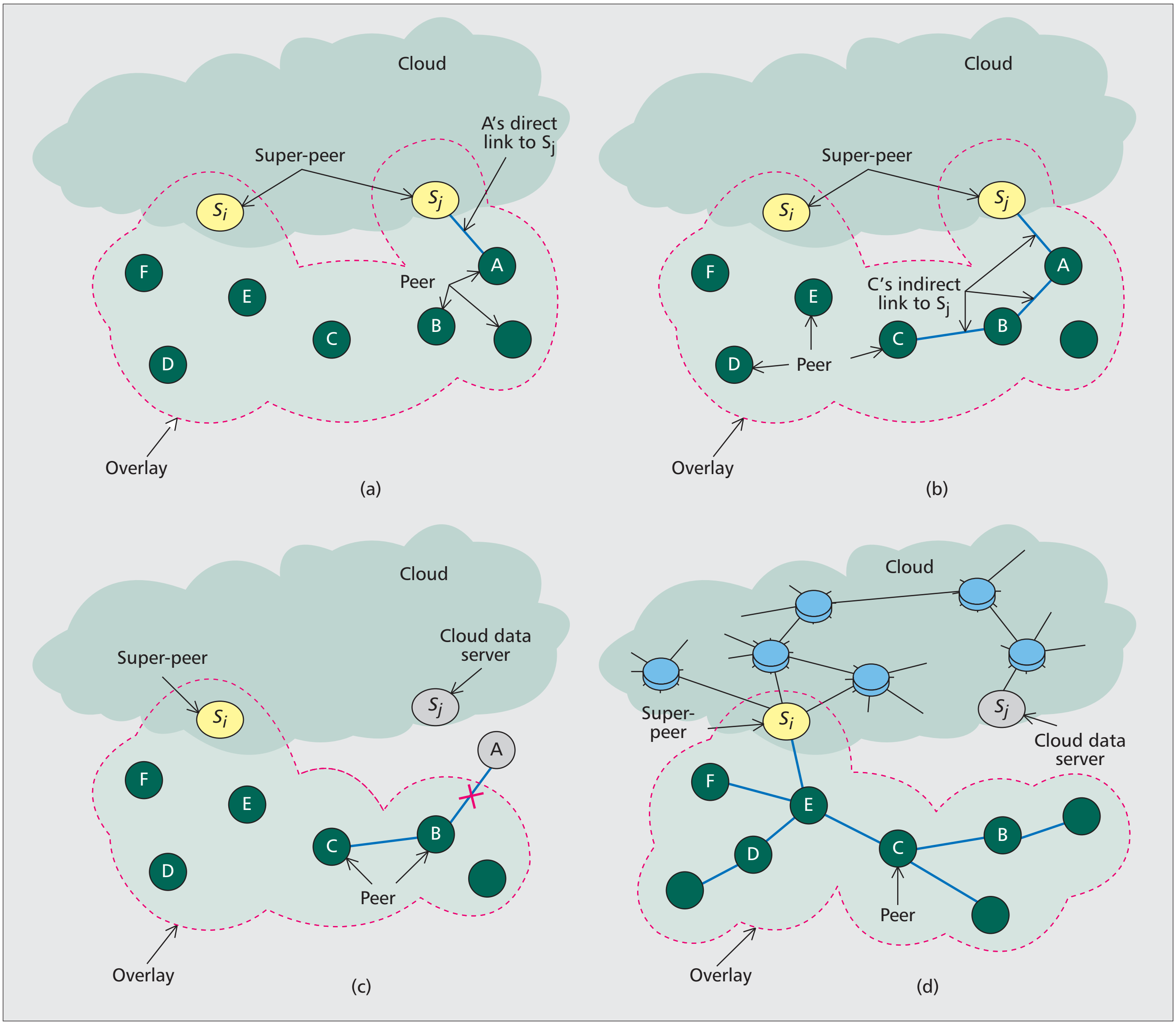

Figure 2. A link loss case in cloud-assisted mobile ad-hoc networks.

In the overlay, each peer maintains a tentative database containing information to be retrieved by routing and searching. Peers communicate with other peers using messages transmitted over the ad-hoc wireless network. There might be a *direct link* or an *indirect link* between any two peers. If two peers are able to send and receive messages directly, then there is a direct link between them. Otherwise, intermediary peers are required to forward transmitted messages, and hence there is an indirect link between these two peers. In this case the intermediary peers act as routers. For example, in Fig. 1 there are only two peers with each having one direct link to a super-peer. All other peers have indirect links to the super-peer. A peer sends queries for data services and receives the result from other peers using direct or indirect links. The overlay is self-organizing and allows peers to join or leave the overlay at any time.

The Problem

A traditional MANET can extend the functionality of services by exploiting the infrastructure and services of the cloud-assisted mobile ad-hoc network adopted in this article. Many potential P2P data services [14] can benefit from the network as well. The advantage of using the network has been demonstrated with *Contrail* developed for Windows Azure [15]. We are now motivated to examine the network uses.

Figure 2 depicts time-varying scenarios corresponding to typical use cases. Fig. 2a is the scenario in which peer A accesses and receives new data items from super-peer S_j via its direct link to S_j. At a later time, A will inform C and B of the received data, and then peer C can access S_j via its indirect link, which in turn relies on A's direct link to S_j (Fig. 2b). It then utilizes searching transactions provided by S_j to find the needed data. Normally, S_j caches data items retrieved by peers to make its searching transactions efficient. If data needed by C is available in S_j's cache, then S_j can efficiently retrieve the data and send it to C. Otherwise, S_j searches the queried data from other cloud data servers.

In the scenario depicted in Fig. 2c, let's suppose that C again tries to connect to S_j using the indirect link established in the past and currently saved in C's tentative database. However, this request fails, because A has already left the overlay. Consequently, C has to reconstruct the link to S_j using the overlay's routing transactions. However, the absence of A has caused the absence of S_j. As a result, the searching transactions provided by S_j are not available in the overlay anymore. Assume that C has an indirect link to another super-peer named S_i (Fig. 2d). Then, as queried from C, S_i initiates a searching process for the requested data in the cloud. Suppose now that C's requested data is available in S_j's cache. However, because S_i did not know about S_j's data items in advance, it triggers the searching transaction through all of its possible interfaces in the cloud. In this case, the cost (i.e. the energy consumed by routing and searching transactions) is obviously expensive, even though C can receive the queried data.

The above problem occurs due to link loss in the overlay. Actually, the mobility of the mobile devices mainly causes link loss. When a peer leaves the overlay, all links relying on the peer as a router will be lost. In addition, sometimes mobile devices turn on power-saving mode, which can be equivalent to link loss. Due to frequent link losses [4], energy consumption will be significant in a cloud-assisted MANET.

Motivation and Ideas

We propose a mechanism to solve the above problem causing excessive energy consumption. The idea is simple, but is practically feasible and powerful. We let super-peers recognize each other, or more precisely each other's data, when they join the overlay. Then they can cooperate to effectively allow routing and searching transactions when a link loss happens in the overlay. The link loss is caused from the natural characteristics of mobile devices.

We recall that search transactions are intrinsic functions of most P2P overlays, and the overlay routing protocol is often designed to make searching efficient. However, the performance of routing in the overlay is strongly affected by the relationship between the topology of the overlay and the underlying physical network [13]. When adding cloud data services to a MANET, a problem that arises is that routing in the cloud normally does not support changes in the network topology. Moreover, due to the computing capabilities and power limitations of mobile devices, routing protocol in a MANET must be kept simple. Therefore, designing an effective routing protocol for cloud-assisted mobile ad-hoc networks is a very challenging issue.

A possible design strategy is to keep the overlay routing protocols unchanged and to complement them with upper-layer functions via a P2P overlay [13]. Currently, there are many effective energy-aware routing protocols that have been established [3]. There are also many effective searching algorithms ([13]). To benefit from these existing protocols and algorithms, our proposed mechanism must have the ability to work with them. However, cooperation with the existing protocols and algorithms poses many challenges, such as ensuring quality of service and incorporating the trade-offs between performance and energy efficiency [1, 4]. In our case, we should deal with the trade-offs between performance and energy efficiency.

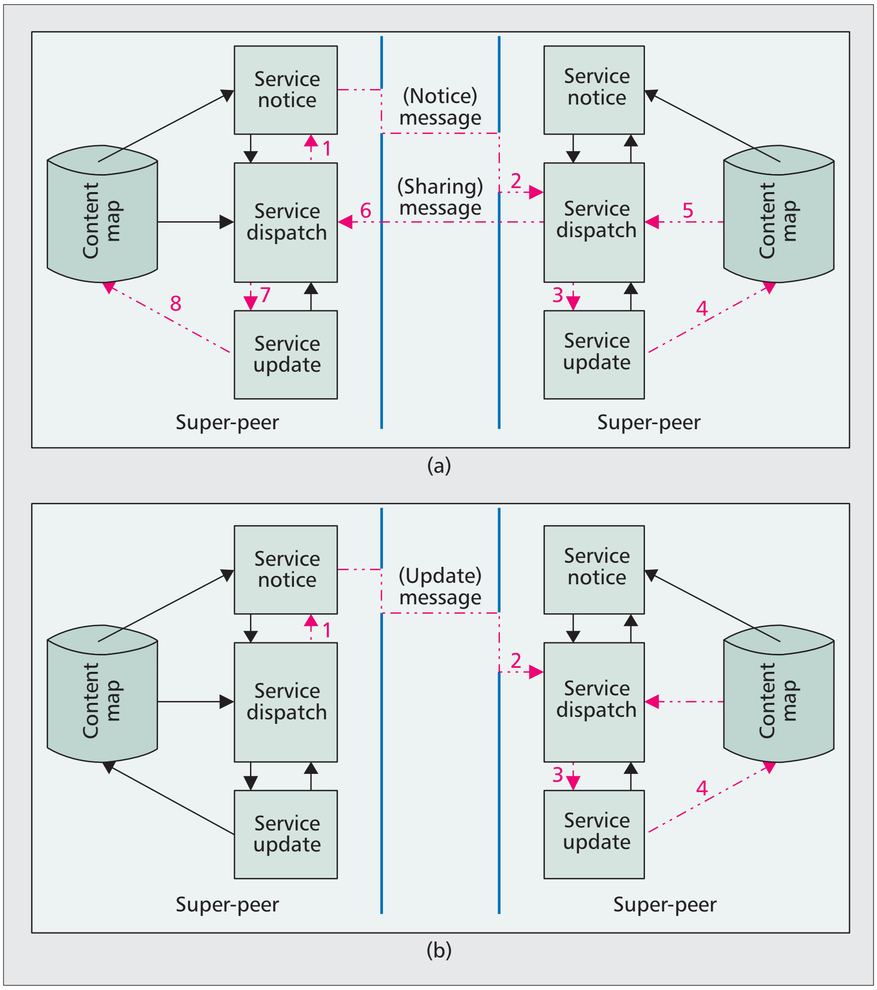

Figure 3. The proposed mechanism message flows.

Proposed Mechanism

In order to overcome the excessive energy consumption problem mentioned in the previous section, we propose a novel mechanism running among super-peers. The proposed mechanism operates at the application layer of the network protocol stack and consists of the following three procedures:

- The *Service Notice* (SN) procedure is executed by a new super-peer to inform other super-peers via the overlay about its new role change from a normal cloud data server to a super-peer.
- The *Service Update* (SU) procedure is invoked by a new super-peer to update its new database content map. The map contains the contents of other cloud data servers and other super-peers as well as its own database content. The SU procedure can be used when a new super-peer is made, or the database content of super-peers are simply updated without a new super-peer presence.
- The *Service Dispatch* (SD) procedure is used by a super-peer to directly request data services from other cloud data servers via the cloud, or to share its database with other super-peers.

Suppose each super-peer maintains a content map, which is a compact table of database content provided by all super-peers that have joined in the overlay. Then, any super-peer in the overlay can request data services from previous super-peers, now normal cloud data servers, via its SD procedure, which have left the overlay due to link loss. Initially, the content map of a super-peer is set to be empty when it joins the overlay for the first time. Then, the joining super-peer may

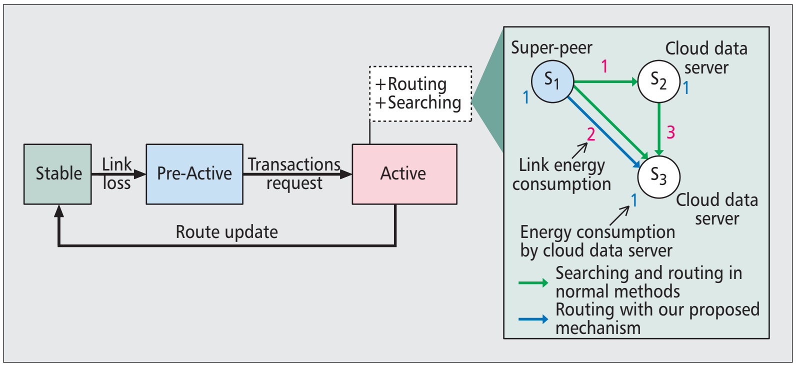

Figure 4. Energy consumption comparison by the different searching and routing of existing normal method and proposed mechanism, respectively.

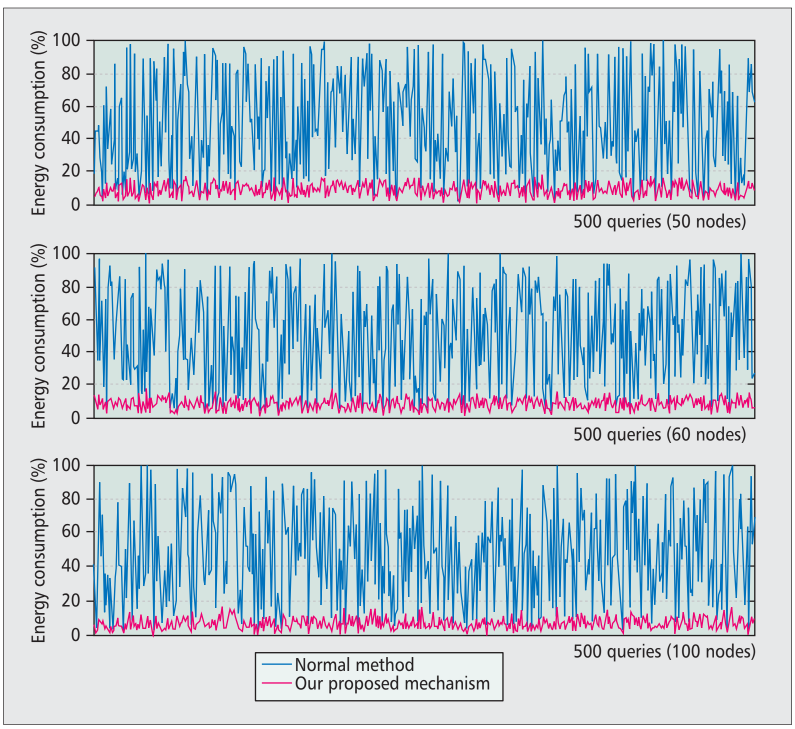

Figure 5. Energy consumption comparison by link routing only.

send a message by its SN procedure to inform other super-peers about its presence. If there is a super-peer with a non-empty database (content map) available in the overlay, it will share the content map using its SD procedure with the newly-joined super-peer. The super-peer in turn uses the SU procedure to update its own content map. The flowchart for these transactions is given in Fig. 3a. If the content map of a super-peer is changed, the updated information will spread among other super-peers (using the SN and SU procedures) that are currently available in the overlay (Fig. 3b).

As mentioned above, the SD procedure can be used to share content map information among super-peers. The specific case where a super-peer rejoins the overlay can be treated similarly to the process described in Fig. 3a. However, the rejoining super-peer needs to send its content map information together with a notice message.

Now we are interested in how much energy can be saved by the proposed mechanism when link loss occurs. We will justify the advantage of the proposed mechanism by simulation in the next section.

Experimental Results and Open Issues

Simulation and Results

In this section we create a simulation to validate the effectiveness of our proposed mechanism. As discussed in the problem definition section, data services provided by the cloud-assisted mobile ad-hoc network can be lost due to link loss. This is a worst case scenario in the overlay.

We set up a simulation to evaluate the overall energy consumed by network routing and searching transactions. We assume that the overlay supports the queries which can be extended to be submitted to the cloud. Then, in the experiments we simply compute energy consumption in the network with or without the use of our proposed mechanism.

At first, we need a model to describe our proposed mechanism algorithms. The model focuses on energy consumed by routing and searching transactions as shown in Fig. 4. We distinguish three states of the network as *stable*, *pre-active*, and *active*. The *stable* state means that all peers remain in the normal network condition. Thus, routing can be made without link loss. On the other hand, the network moves to the *pre-active* state in the case of link loss. In this state, it changes to the *active* state if there exists a peer that has sent a query to ask for a data service provided from a super-peer that has already left the overlay. In the *active* state, the routing information table is updated based on the link loss case information and then the network is back to its *stable* state. Whenever the network reaches the *active* state, it triggers two transactions: routing and searching. We then calculate the energy consumed by the two transactions under our proposed mechanism.

Now, considering the network in two situations (i.e. the network with and without the use of our mechanism), we can observe the differences in energy consumption during the two different situations within the cloud.

For this simulation we use the Dijkstra algorithm for routing. We consider a network with one super-peer and two cloud data servers, namely S_1, S_2, and S_3, respectively. The value assigned to a link between two nodes represents the energy consumed by that link by routing. The value assigned to each node represents the energy consumed by searching. We assume that searching energy consumption is uniquely 1 in all of the nodes. We assume that S_1 is currently in the overlay, that is, the super-peer and S_3 have already left the overlay, that is, now the cloud data server. Then, if we do not apply our proposed mechanism, the route will be S_1, S_2, and S_3. And the energy consumed for routing and searching by the query from S_1 to S_3 will be 3 ($1 + 2 = 3$) and 3 ($1 + 1 + 1 = 3$), respectively. But if we apply our proposed mechanism, then the query will be sent directly to S_3 from the super-peer S_1, and the results become 2 and 1, accordingly. The reason why we get a value of 1 for search energy consumption is that we do not need to search in super-peer S_1, because the super-peer already knows where the queried data is by our proposed mechanism. It is obvious that our proposed mechanism saves much more energy than the normal method.

We have implemented three different network models with

50, 60, and 100 cloud data servers, respectively, and the same number of 100 mobile devices for the three models. Suppose that whenever the network is in the active state, the queries sent by a peer (mobile device) always reach one activating super-peer. In the overlay, the link loss and queries are generated randomly. We also assign random integer values to represent the energy consumed by cloud data servers and links. Note that we use integers instead of specific energy measurement units such as Joule and Watt because they allow us to better illustrate our idea at an abstract level. We assume that searching energy consumption is uniquely 1 in all of the cloud data servers. We assume further that routing energy consumption of a link is an integer between 1 and 5. Then we can calculate the total energy consumed for routing and searching by a query as demonstrated above. Figures 5a, 5b, and 5c show the energy consumption comparison of routing transactions when there are 500 queries in the network with 50, 60, and 100 cloud data servers, respectively.

In the experimental result, 100 percent of energy consumption means the worst case of searching and routing all of the nodes in the entire network. Figure 5 shows that the existing normal method results in approximately 48 percent energy consumption on average over the three network models, while our proposed mechanism demonstrates a significant energy savings of 7.5 percent on average over the three different network models. We can notice that our proposed mechanism provides six times more energy savings.

We ran the simulation for both routing and searching transactions with three different network models. Our proposed mechanism results in significantly better energy savings also, as shown in Fig. 6. The proposed method gives nine times more energy savings on average than normal operational methods in the three models, that is, 49 percent energy consumption in normal methods and 5.5 percent in our proposed mechanism. Because the proposed mechanism reduces the number of searching transactions, it can save quite a lot of energy compared to the case when the mechanism is only used for routing transactions.

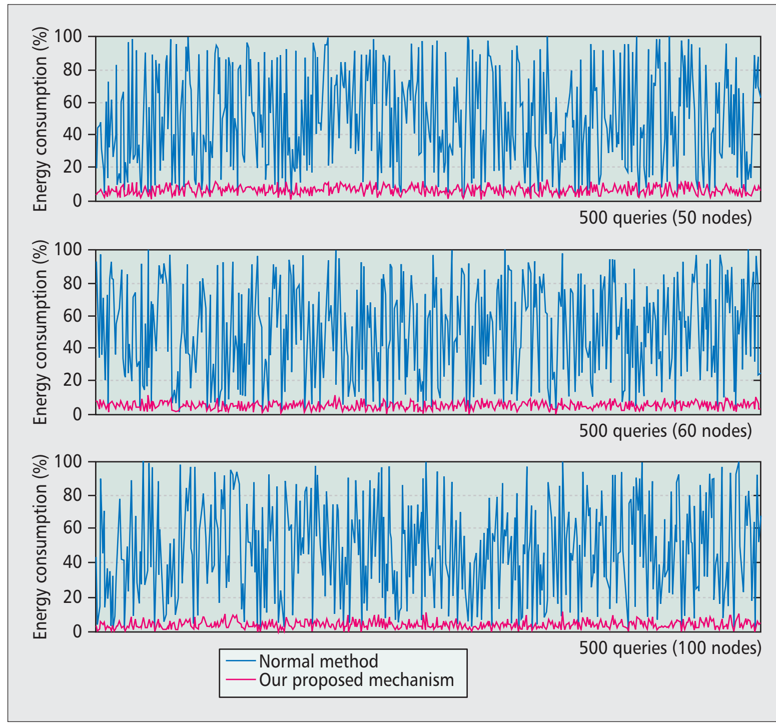

Figure 6. Energy consumption comparison of both searching and link routing.

Open Issues and Future Work

The numerical results have proven the effectiveness of our proposed mechanism, but we have not checked overall network performance when we apply the proposed mechanism. We don't consider the energy consumption of message exchanges as well as the content maps in the proposed mechanism, because they are a negligibly small portion, but we suggest reflecting them in future experimental work. How the proposed mechanism influences overall network performance would be good future work. As a security and energy efficiency issue, cloud servers and super-peers can possibly be forced to provide a fake content map, which would cause them to consume excessive energy. Thus, the issue would be another interesting future research topic.

Conclusion

We focus on green data centers providing data services for cloud-assisted mobile ad-hoc networks in 5G. We propose a practical and powerful energy saving mechanism to decrease searching and routing transactions in cloud data servers, because the transactions consume much more energy when links are lost. The experimental work has proven that our proposed mechanism significantly reduces energy compared to the normal operational method in data centers. Due to its simplicity, our proposed mechanism is particularly expected to attract many practical uses in the future.

Acknowledgment

This research was supported by a Korea University grant. Minho Jo is the corresponding author for this paper.

References

[1] A. Berl *et al.*, "Energy-Efficient Cloud Computing," *The Computer Journal*, vol. 53, no. 7, 2010, pp. 1045–51.
[2] F. Boccardi *et al.*, "Five Disruptive Technology Directions for 5G," *IEEE Commun. Mag.*, vol. 52, no. 2, 2014, pp. 74–80.
[3] T. Silva and A. Arsenio, "A Survey on Energy Efficiency for the Future Internet," *Int'l J. Computer and Communication Engineering*, vol. 2, no. 5, 2013, pp. 589–95.
[4] F. Liu *et al.*, "Gearing Resource-Poor Mobile Devices with Powerful Clouds: Architectures, Challenges, and Applications," *IEEE Wireless Commun.*, vol. 20, no. 3, 2013, pp. 14–22.
[5] L. Gkatzikis and I. Koutsopoulos, "Migrate or Not? Exploring Dynamic Task Migration in Mobile Cloud Computing Systems," *IEEE Wireless Commun.*, vol. 20, no. 3, 2013, pp. 24–32.
[6] L. Zhou and H. Wang, "Toward Blind Scheduling in Mobile Media Cloud: Fairness, Simplicity, and Asymptotic Optimality," *IEEE Trans. Multimedia*, vol. 15, no. 4, 2013, pp. 735–46.
[7] M. Chen *et al.*, "Enabling Technologies for Future Data Center Networking: A Primer," *IEEE Network*, vol. 27, no. 4, 2013, pp. 8–15.
[8] L. Zhou *et al.*, "Exploring Blind Online Scheduling for Mobile Cloud Multimedia Services," *IEEE Wireless Communications*, vol. 20, no. 3, 2013, pp. 54–61.
[9] Y. Wen, G. Zhang, and X. Zhu, "Lightweight Packet Scheduling Algorithms for Content Uploading from Mobile Devices to Media Cloud," *Proc. GLOBECOM'2011 Workshops*, Houston, Texas, USA, Dec. 2011, pp. 45–50.
[10] D. Camps-Mur *et al.*, "Device-to-Device Communications with WiFi Direct: Overview and Experimentation," *IEEE Wireless Commun.*, vol. 20, no. 3, 2013, pp. 96–104.
[11] X. Wang *et al.*, "Cloud-Assisted Adaptive Video Streaming and Social-aware Video Prefetching for Mobile Users," *IEEE Wireless Commun.*, vol. 20, no. 3, 2013, pp. 72–79.
[12] Y.-W. Lin, J.-M. Shen, and H.-J. Weng, "Cloud-Assisted Gateway Discov-

ery for Vehicular Ad hoc Networks," *Proc. NISS'2011*, Macau, Oct. 2011, pp. 237–40.
[13] J. F. Buford, H. Yu, and E. K. Lua, *P2P Networking and Applications*, Morgan Kaufmann Publishers, 2009.
[14] E. Talipov, Y. Chon, and H. Cha, "Content Sharing over Smartphone-Based Delay-Tolerant Networks," *IEEE Trans. Mobile Computing*, vol. 12, no. 3, 2013, pp. 581–95.
[15] P. Stuedi *et al.*, "The Cloud is the Router: Enabling Bandwidth-Efficient and Privacy-Aware Mobile Applications with Contrail," *Microsoft Technical Report*, 2010.

Biographies

MINHO JO is a professor in the Department of Computer and Information Science, Korea University, Sejong Metropolitan City, South Korea. He received his BA in the Dept. of Industrial Engineering, Chosun Univ., S. Korea in 1984, and his Ph.D. in the Dept. of Industrial and Systems Engineering, Lehigh University, USA in 1994, respectively. He is the founder and editor-in-chief of the *KSII Transactions on Internet and Information Systems*. He is an editor of *IEEE Wireless Communications*. He is now vice president of the Institute of Electronics and Information Engineers (IEIE), and vice president of the Korea Information Processing Society (KIPS). Areas of his current interests include cognitive radio, IoT, HetNets in 5G, green (energy-efficient) wireless communications, mobile cloud computing, 5G wireless communications, networks, optimization and probability in networks, network security, and massive MIMO.

NGUYEN DINH HAN is a senior researcher at Hung Yen University of Technology and Education, Viet Nam. He was a research professor in the Department of Computer and Information Science of Korea University from September 2013 to September 2014. He is now an editor of the *KSII Transactions on Internet and Information Systems*. His current research areas are the theory of code and applications, computer and network security, wireless communications, cognitive radio, and cloud computing.

YONGWHA CHUNG is a professor in the Department of Computer and Information Science, Korea University, Sejong Metropolitan City, South Korea. He received his Ph.D. in the Department of Electrical Engineering, University of Southern California, USA in 1997. He is an editor of the *KSII Transactions on Internet and Information Systems* and an editor for the *Journal of Next Generation Information Technology*. He is now vice president of the Korea Institute of Information Security and Cryptology. Areas of his current interests include high-performance computing, mobile computing, performance optimization, and network/information security.