Research Article

Virtualized ANR to Manage Resources for Optimization of Neighbour Cell Lists in 5G Mobile Wireless Networks

Yoonsu Shin and Songkuk Kim

School of Integrated Technology, Yonsei Institute of Convergence Technology, Yonsei University, Seoul, Republic of Korea

Correspondence should be addressed to Yoonsu Shin; sinowl@yonsei.ac.kr

In future, more devices such as wearable devices will be connected to the networks. This will increase simultaneous handovers. The coverage of a cell will be small because a superhigh frequency used in 5G wireless networks does not propagate very far. This trend will increase the number of neighbour cell lists and it will accelerate the change of neighbour cell lists since the coverage of cells can be altered by the environment. Meanwhile, the ANR technology will be essential in 5G networks. Since the network environment in the future is not similar to the present, the strategy of ANR should also be different from the present. First, since practical neighbour cell lists in each cell are changed frequently and individually, it is necessary to optimize them frequently and individually. Second, since the neighbour cell lists in each cell are not changed similarly, it is necessary to operate ANR flexibly. To respond to these issues, we propose to use network function virtualization (NFV) for ANR. To evaluate the proposed strategies, we measured additional resource consumption and the latency of handover if neighbour cell lists are not optimized when UEs perform handover simultaneously. These experiments are conducted using Amarisoft LTE-100 Platform.

1. Introduction

In the IoT era, the mass use of devices such as wearable devices and wireless sensors in mobility (e.g., vehicle, personal mobility, and watch) will be connected to the networks for convenience. This will cause an increase of simultaneous handover. Thus, the handover performance will become more important for ceaseless connection and QoS. In particular, if there will be an urgent situation to receive data from networks, the handover performance will be critical. Moreover, in the 5G era, the coverage area of cells will become smaller because a superhigh frequency will be used for high throughput. The superhigh frequency cannot propagate very far. Hence, more cells are necessary to cover the same area (i.e., massive small cells). These features will increase the number of neighbour cell lists, and it will accelerate the change of neighbour cell lists because the coverage of cells can be altered by the environment, which includes reflection, diffraction, and shadowing effects. In addition, these small cells will also increase the simultaneous handover because the boundary among cells will increase. Moreover,

nowadays, moving cells and small cells are usually used for data traffic rushes and radio shadow areas [1], and macrocells and massive small cells will coexist in 5G networks. In this complex network design, the configuration of neighbour cells becomes more difficult even though it is very important for handover between two neighbour cells.

Meanwhile, it is necessary to inquire assumption that an increase of neighbour cell lists will accelerate the change of neighbour cell lists. First, practical neighbour cell lists, according to the strength of the signal from other cells, are not fixed. This means that a strong signal from far-off cells can be received in the serving cell, and it propagates sufficiently far because radio signal strength is affected by environmental factors such as temperature and humidity [2–5]. This effect is also due to shadowing, fast fading, reflection, and so on [6, 7]. For example, authors in [6] emulated the propagation effects by increasing the standard deviation of the lognormal shadowing in the macrocells. In this simulation, the small cell had just one single neighbour macrocell as a neighbour cell list at the region with the no shadowing (i.e., a small cell was in a macrocell), but four neighbour cells were detected in a

small cell with the shadowing of 12 dB standard deviation. If the standard deviation would increase, the number of neighbour cell lists would increase. On the contrary, if the standard deviation would decrease, the number of neighbour cell lists would again decrease. In this simulation, the number of neighbour cell lists in a macrocell was varied from six to about 20, and the number of neighbour cell lists in a small cell was varied from one to about 10 according to shadowing standard deviation. Therefore, if there are many small cells in a macrocell, it is expected that the number of neighbour cell lists in a cell increases more and the neighbour cell lists in a cell change more frequently.

Due to massive small cells in 5G networks, automatic neighbour relation (ANR) technology, which automatically detects and configures neighbour cells, will be essential and become more important in 5G networks because manual configuration and optimization of neighbour cell lists in each individual small cell will become more costly and difficult. ANR detects a new cell's information including physical cell ID (PCI), E-UTRAN cell global identifier (ECGI), and the IP address from operations, administration, and maintenance (OAM) in order to execute handover to a new cell when a serving cell does not know the new cell and the serving cell receives the strong signal of the new cell. This handover method is called UE-Triggered ANR with OAM Support [8]. In other handover methods, ANR receives only the IP address from OAM and the other information from mobile phones [9]. Basically, ANR makes it possible for enhanced NodeB (eNodeB) to detect the neighbour cells on the basis of UE measurements [10]. The UE eventually sends Measurement Report messages, including PCI, to eNodeB when the UE gets a stranger signal than that from serving eNodeB, or periodically transfers this message [11].

Finally, since the network environment in the future (i.e., an increase of simultaneous handovers and a frequent change of neighbour cell lists) will not be the same as the present, the strategy of ANR should also be changed as we needed a new strategy of self-organizing networks in the past due to enterprise femtocells [12], and the strategies to be considered are discussed as below. First, it is necessary to optimize the neighbour cell lists frequently and individually because practical neighbour cell lists for each cell are changed frequently and individually due to the change of environment and the redundant neighbour cell lists could be burdensome to perform handover quickly. Although traditional ANR function has neighbour removal function [13], it does not consider this future condition (i.e., frequent change of practical neighbour cell lists). Recently, ANR algorithm for this complex network environment is researched [14], but it is only for overreached scenario. To apply the change of a cell's coverage according to a natural phenomenon, it is necessary to check the signal frequently and individual for optimizing neighbour cell lists. Second, it needs to operate ANR flexibly because the neighbour cell lists in some cells are rarely modified while those in other cells are frequently modified. For example, when a great number of people gather for a festival, game, or national holiday, it is necessary to operate new small cells or moving cells for a while. Additionally, if a tall building is built or if it is rainy or sunny, the cell coverage is altered, and this can also change the neighbour cell lists. Since this trend continues and even accelerates, these strategies will become more essential in the future.

For these strategies, we propose using network function virtualization (NFV) [15] for operating ANR. That is, ANR-virtual network function (ANR-VNF), in which network entry is virtualized and has ANR functionality, can be deployed and extended frequently and flexibly. For instance, in the case where the deviation of a floating population is large, such as that at a stadium, it is more efficient and inexpensive to operate moving cells than fixed cells for data offload. Also, in the case where the fluctuation of shadowing standard deviation is large due to weather, more resources are needed for optimizing neighbour cell lists quickly. In any case, in order to configure and optimize neighbour cell lists more quickly, it is necessary to extend the capacity of ANR functionality.

The remainder of this paper is organized as follows. Section 2 provides background on self-organizing networks (SON), ANR, NFV, and so on. Section 3 adduces the experiments for necessity of these strategies and expected hazard. Section 4 explains the proposed methods such as ANR-VNF in detail. Finally, Section 5 gives the conclusions.

2. Background and Related Work

2.1. Self-Organizing Networks (SON). Operating radio networks is a challenging task, especially in cellular mobile communication systems due to their latent complexity. This complexity arises from the number of network elements and interconnections between their configurations. In a heterogeneous network, it is difficult to handle the variety of technologies and their precise operational paradigms. Today, planning and optimization tools are typically semiautomated and management tasks need to be tightly supervised by human operators. This manual effort by the human operator is time-consuming, expensive, and error-prone and requires a high degree of expertise. SON can be used to reduce operating costs by reducing tasks at hand and to protect proceeds by minimising human error. The subsection below details SON taxonomies.

2.1.1. Self-Configuration. Configuration of base stations (eNBs), relay stations (RS), and femtocells is required during deployment, extension, and upgrade of network terminals. Configurations may also be needed when there is a change in the system, such as the failure of a node or a drop in network performance. In future systems, the conventional process of manual configuration needs to be replaced with self-configuration. It is predictable that nodes in future cellular networks should be able to self-configure all of their initial parameters including IP addresses, neighbour list, and radio access parameters.

2.1.2. Self-Optimization. After the initial self-configuration phase, it is significant to continuously optimize system parameters to ensure efficient performance of the system if all its optimization objectives are to be maintained. Optimization in legacy systems can be done through periodic drive

tests or analysis from log reports generated from network operating centers. Self-optimization includes load balancing, interference control, coverage extension, and capacity optimization.

2.1.3. Self-Healing. Wireless cellular systems are prone to faults and failures because of component malfunctions or natural disasters. In traditional systems, failures are mainly detected by the centralized O&M (Operation and Maintenance) software. Events are recorded and necessary alarms are set off. When alarms cannot be cleared remotely, radio network engineers are usually mobilized and sent to cell sites. This process could take days or even weeks before the system returns to normal operation. In future self-organized cellular systems, this process needs to be improved by consolidating the self-healing functionality. Self-healing is a process that relates the remote detection, diagnosis, and triggering of compensation or recovery actions to blunt the effect of faults in the network's equipment.

2.2. Automatic Neighbour Relation (ANR) [7–9]. The coverage of cells is limited because the cell cannot emit radio frequency with unlimited power, so there are many cells for covering a wide area. If one mobile phone moves from the coverage area of one cell to that of another, it would connect to the new cell and disconnect from the old cell. This procedure is called handover. In LTE radio access network, the cell consists only of eNodeB which communicates with each other directly via the X2 interface. Over this X2 interface, neighbouring eNodeBs communicate with each other to prepare and execute handovers. In order to provide seamless mobility in LTE, it is important to set up the X2 interface without omission because there will be no handover between neighbouring eNodeBs unless the X2 interface is set up and functioning.

The X2 interface is set up by using the neighbour cell lists in the Neighbour Relation Tables (NRT) of each eNodeB, so there is neither X2 interface nor handover between neighbour cells if one eNodeB omits the neighbour cell lists in its own NRT, which is caused by a moving cell or a newly added small cell. In this case, Automatic neighbour Relation (ANR) functionality can detect the new neighbour cell and add its list to the NRT automatically.

2.3. Network Function Virtualization [15]. Service provision within the telecommunications industry has traditionally been based on network operators providing physical proprietary devices and equipment for each function. These dedicated requirements for high quality, stability, and stringent protocol adherence have led to long product cycles, very low service agility, and heavy dependence on specialized hardware.

However, the requirements by users for more diverse and new (short-lived) services with high data rates continue to increase. Therefore, Telecommunication Service Providers (TSPs) must correspondingly and continuously purchase, store, and operate new physical equipment. All these factors lead to high CAPEX and OPEX for TSPs. Moreover, the resulting increase in capital and operational costs cannot result in higher subscription fees, so TSPs have been forced to find ways of building more dynamic and service-aware networks with the objective of reducing product cycles, operating, and capital expenses and improving service agility.

NFV [16, 17] has been proposed as a way to address these challenges by forcing virtualization technology to offer a new way to design, deploy, and manage networking services. The main idea of NFV is the decoupling of physical network equipment from the functions that run on them. The goal of NFV is to transform the way that network operators design networks by evolving virtualization technology to reinforce much of the network equipment onto standard servers, which could be located in data centers, distributed network nodes, and at the end user premises. It involves the implementation of network functions in software—VNFs—that can run on one or more industry standard physical servers and that can be moved to various locations in the networks as required without the need for installation of new equipment.

With this, NFV allows TSPs to get more flexibility to further open up their network capabilities and services to users and other services and the ability to deploy or support new network services faster and cheaper so as to realize better service agility. To achieve these benefits, NFV paves the way for a number of differences in the way network service provisioning is realized in comparison to current practice.

2.4. Virtualized Cellular Infrastructure. A basic architecture of LTE networks without NFV shows that the UE is connected to the Evolved Packet Core (EPC) over the LTE access network (E-UTRAN), in which the eNodeB is the base station for LTE radio. The EPC is made up of the Serving Gateway (S-GW), the Packet Data Network (PDN), the Gateway (P-GW), the Mobility Management Entity (MME), and the Policy and Charging Rules Function (PCRF). All these functions are based on dedicated equipment.

In virtualized cellular infrastructure shown in Figure 1, however, the network entries of EPC get virtualized into a data center. Also, a logic part of eNodeB gets virtualized into data center and only the RF part of eNodeB remains. This division of eNodeB is called C-RAN (cloud radio access network). C-RAN features centralized processing, collaborative radio, real-time cloud computing, and power efficient infrastructure [18]. C-RAN is composed of the BBU (baseband unit), OTN (optical transmission network), and the RRU (remote radio unit). The BBUs implement the base station functionality whereas the RRUs perform radio functions. Also, the BBUs and RRUs are applicable to typical RAN, from macrocell to femtocell. Thus, these BBUs can be centralized and this makes networks more efficient and optimized in terms of cost, resources, and energy through the orchestration management system [18, 19].

2.5. Amarisoft LTE-100 Platform. The Amarisoft LTE-100 platform is a software-based LTE station running on a PC. Like a virtualized cellular infrastructure, the Amarisoft LTE-100 platform provides LTE Enhanced Packet Core (EPC) and base station (eNB) on each PC. The EPC includes Mobility Management Entity (MME) with built-in Packet Gateway (P-GW), Serving Gateway (S-GW), and Home Subscriber
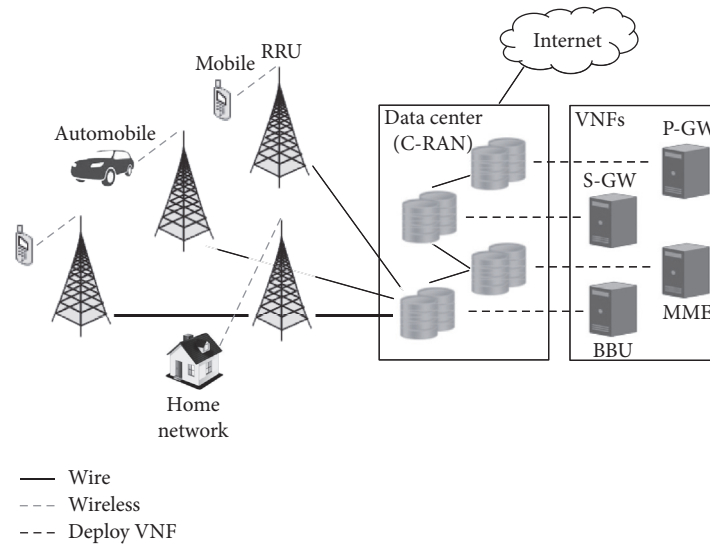
FIGURE 1: The architecture of virtualized cellular infrastructure.

Servers (HSS). The radio interface in the software-based LTE solution was handled by Ettus Research USRP N210. The antenna configuration used in the experiments was a Single-Input Single-Output (SISO) scheme.

The PHY layer complies with LTE release 13 and supports closed-loop power control, and the protocol layer also complies with LTE release 13 and implements the MAC, RLC, PDCP, and RRC layers. Also, it supports intra-eNodeB, S1, or X2 handovers. For network interface, it supports standard S1AP and GTP-U interfaces to the Core Network and the X2AP interface between eNodeBs.

Like Figure 2, the EPC is connected to the Internet, and the eNodeBs are connected to the EPC, and each eNodeB has an RF unit (USRP N210). It is necessary for each eNodeB to be registered to the EPC and to be connected with the X2 interface between eNodeBs. Then, each eNodeB recognizes other eNodeBs. However, for handover, each eNodeB must have the information of other eNodeBs in the Neighbour Relation Table (NRT). Naturally, it is desirable to have the information of the actual neighbour eNodeBs, but the handover is performed correctly although the NRT has the information of redundant eNodeBs that are not the actual neighbour eNodeBs.

3. The Experiments for the Strategy of ANR

Since small cells and macrocells are mixed in the transition period to 5G networks and the cell coverage is affected by the surrounding environment, neighbour cell lists will not be fixed in practice and will not be small in number; that is, the number of neighbour cell lists in the NRT of each eNodeB can be increased over the number of physically adjacent cells or can be decreased due to the environmental effects such as shadowing and fading. Furthermore, since the simultaneous handovers will increase due to IoT and massive small cells, it is expected that the handover performance will be more important and has hazard issues. Thus, it is
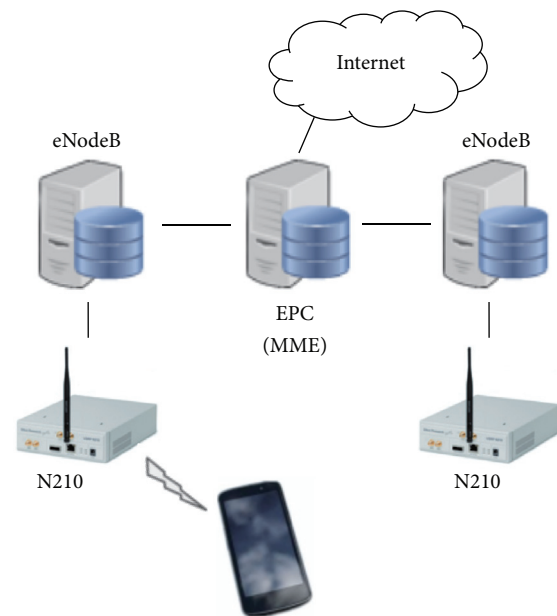

FIGURE 2: Amari LTE 100 platform configuration.

necessary to optimize the NRT (i.e., neighbour cell lists) because redundant neighbour cell lists could be burdensome to perform handover. In order to estimate the performance issues of ANR in 5G, we performed several experiments.

3.1. Experiment Design. In these experiments, the Amarisoft LTE-100 Platform [21], which has one Core Network (EPC) and two eNodeBs, is used, and each component (EPC, eNodeB) is installed in a common computer which has an Intel(R) Core(TM) i7-4790 @ 3.60 GHz CPU and 8 GB memory computer and has Ubuntu 14.04 operating system. This means that network elements like VNF are able to be carried out on a common computer. This platform was
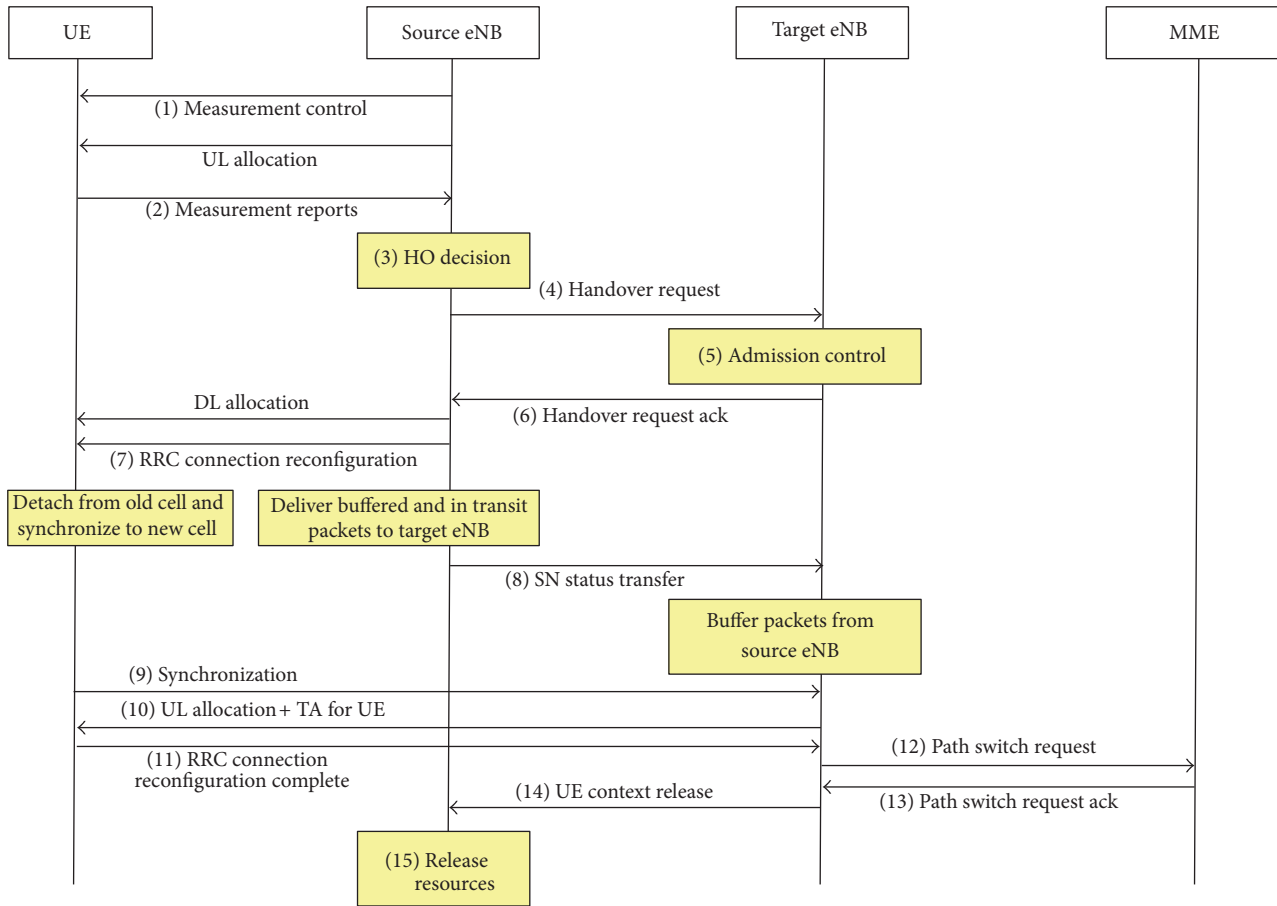
FIGURE 3: Handover procedure.

used in recent researches [22, 23]. Figure 2 shows the overall network configuration for experiments. There are two eNodeBs and one EPC, and it supports handover. Since there are just two eNodeBs, the optimized NRT of each eNodeB has just one neighbour cell, and the nonoptimized NRT has more neighbour cells. Also, two eNodeBs have the same handover threshold and different frequency, so the handover between two eNodeBs is interfrequency mobility handover. Several devices including a Galaxy S7, iPhone 6s, iPhone 6 plus, and iPhone SE are used as UEs.

For executing handover, the UEs are at a similar distance from two cells and perform handover simultaneously by decreasing the power of the serving cells with command at the same time. In order words, UEs do not move from serving cell to target cell; they are just fixed in the same position with light of sight (LOS). Other power adaptation is not considered. This forces the UEs to have the same handover condition (e.g., the power of the serving cell is less than that of neighbour cell). We increase the number of UEs that perform handover simultaneously (e.g., 2, 3, and 4 UEs), and all these handover experiments are executed over 10 times. During handover, to measure CPU usage, the Linux "top" command is used to measure the CPU usage of the eNodeB program during handover. This "top" command measures CPU usage per second which

makes it possible to evaluate the fluctuation of CPU usage for the running the eNodeB program.

For evaluating handover latency, the log information is used. For that, the handover procedure is necessary. Authors in [24] explained the intra-MME/S-GW handover procedure which means that only eNodeB (not MME and S-GW) is changed when handover is executed, and Figure 3 shows this intra-MME/S-GW handover procedure concretely. The handover time (latency) is the period from the Measurement Reports message (2) to the UE Context Release message (14). The handover is triggered by the UE that sends a Measurement Report message (2) to the serving cell, and at the handover completion, the target cell sends the UE Context Release message (14) to the serving cell to inform success of handover via X2AP [25]. Thus, the handover time can be measured by estimating the period from Measurement Report message (2) to UE Context Release message (14). To measure exact handover latency using log information, three computers are synchronized using NTP (network time protocol).

With the handover procedure log, we measure each UE's handover latency, which means the period of each UE's handover when all UEs complete handover simultaneously. We also measure total handover latency, which means the period from the start time of the first UE's handover to the

TABLE 1: During handover, the average fluctuation of target eNodeB's CPU usage (%) with respect to the number of neighbour cell lists.

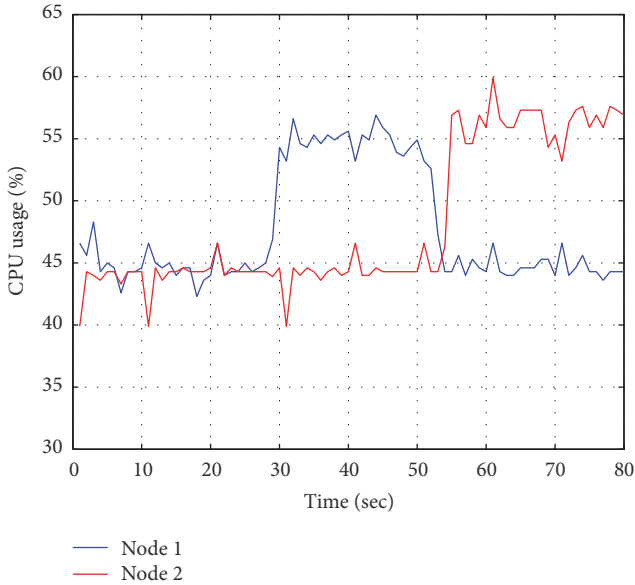| Number of neighbour cell lists	Average fluctuation of CPU usage [%]	SEM [%]
1	0.6125	0.280
11	1.1	0.558

FIGURE 4: Transition of CPU usage [%] during handover from node 1 to node 2.

TABLE 2: Average handover latency with respect to the number of neighbour cell lists when 1 UE performs handover.

Number of neighbour cell lists	Average latency [msec]	SEM [msec]
1	72.8	2.4
11	73.3	1.9
21	77.2	2.1

TABLE 3: Each UE's average handover latency with respect to the number of neighbour cell lists when 2 UEs, 3 UEs, or 4 UEs perform handover simultaneously.

	Number of neighbour cell lists	Average latency of handover [msec]	SEM [msec]
2 UEs	11	66.1	1.3
	21	83	3.5
3 UEs	11	68.9	1.2
	21	74.3	1.6
4 UEs	11	79.8	1.5
	21	79.3	1.4

end time of the last UE's handover when all UEs complete handover simultaneously.

3.2. Experimental Results. In Table 1, in the case of optimized NRT (i.e., 1 neighbour cell list), the average fluctuation of CPU usage is about 0.6% during handover, and this change is within the average fluctuation at idle. In the case of nonoptimized NRT (i.e., 11 neighbour cell lists), however, the average fluctuation of CPU usage is about 1.1%. It seems that about average 0.5% more of CPU usage is used due to the lookup of neighbour cell lists. In addition, the calculated means from two cases are significantly different at $\alpha = 0.05$ by using ANOVA (the analysis of variance) [26], and we indicate the standard error of the mean (SEM) in all Tables [26]. Even though only the number of neighbour cell lists is increased, the fluctuation of CPU usage is also increased during handover. Thus, it is necessary to optimize neighbour cell lists to stop resource waste.

Meanwhile, if a UE receives packets during handover, the CPU usage for download is transferred from serving cell to target cell. Figure 4 shows this movement of CPU usage from serving cell to target cell when one UE performs handover during download. It seems that resource usage for download is considerable, so proper handover (not ping-pong handover) is important for resource management as well as QoS because correct handover can distribute total resources efficiently.

Table 2 shows average handover latency according to the number of neighbour cell lists when one UE performs handover. There are two eNodeBs so that the number of optimal neighbour cell lists is just one. Thus, in Table 2, 21 neighbour cell lists mean that there are an additional 20 neighbour cell lists which do not exist in practice for various reasons. Average handover latency increases along with the number of neighbour cell lists. In particular, the average handover latency rises extremely high when the number of neighbour cell lists is 21. Unfortunately, however, three latencies are statistically equal using ANOVA at $\alpha = 0.05$.

Table 3 indicates the average of each UE's handover latency when 2, 3, and 4 UEs perform handover simultaneously. Although there are slight differences, the average latency of each UE's handover increases according to the increase of UE and neighbour cell lists. In particular, for the case of 2 and 3 UEs, the latency increases statically at $\alpha = 0.05$ when the number of neighbour cells is 21. These latencies, however, do not increase considerably, just 5–10 ms. In addition, it seems that the average of each handover latency does not significantly relate to the number of UEs which simultaneously perform handover. Thus, it is necessary to focus on average total handover latency in each case (i.e., 2, 3, and 4 UEs). To aid the reader's comprehension, Figure 5 describes each UE's handover latency and total handover latency when three UEs perform handover simultaneously.

Table 4 indicates the average of total UEs' handover latency when 2, 3, or 4 UEs perform handover simultaneously. According to the increase of UE and neighbour cell lists, the average of total handover latency is increased. In the case of 21 neighbour cell lists, the average total handover latency of 21 neighbour cell lists is increased statically at $\alpha = 0.05$ compared to the case of 11 neighbour cell lists. Table 5 shows each delay until the 2nd, 3rd, or 4th UE starts handover when 4UEs perform handover simultaneously and the number of
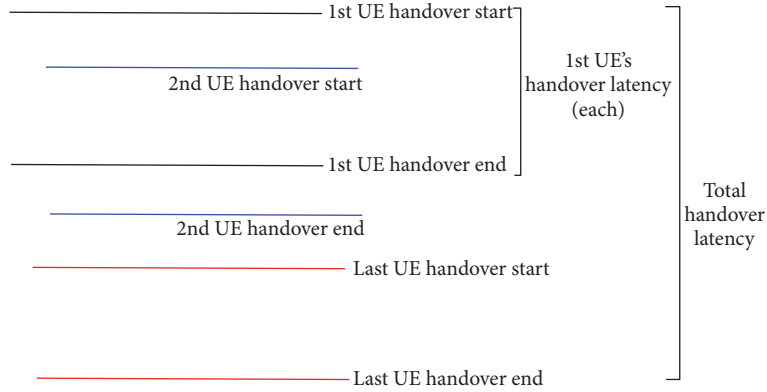
FIGURE 5: Each UE's handover latency and total handover latency when UEs perform handover simultaneously.

TABLE 4: Average total handover latency with respect to the number of neighbour cell lists when 2, 3, or 4 UEs perform handover simultaneously.

	Number of neighbour cell lists	Average total latency [msec]	SEM [msec]
2 UEs	11	324.3	79.9
	21	357.2	152.2
3 UEs	11	385.4	49
	21	529.4	32.7
4 UEs	11	513.2	25.9
	21	621.6	22.6

TABLE 5: Average delay until other UEs start handover since the start of the 1st handover.

	Number of neighbour cell lists	Average delay [msec]	SEM [msec]
2nd UE	11	28.2	6.3
	21	31.7	6.4
3rd UE	11	137.3	10.1
	21	134.2	25.9
4th UE	11	433.6	26.7
	21	506.6	23.5

neighbour cell lists is 11 or 21. When the number of neighbour cell lists is 21, the 4th UE starts handover more later (about 17%) than when the number of neighbour cell lists is 11. This delay leads to the increase of average total handover latency. This shows that as the number of neighbour cell lists and UEs that simultaneously perform handover increase, the average total handover latency increases in a step-like fashion.

As shown in Table 5, the start time (delay) of handover for each UE is different even though all UEs try to perform handover simultaneously. Moreover, this start time is increased according to the increase of UE and neighbour cell lists because the system resources are fixed and more resources will be required. It seems that if there are a lot of neighbour cell lists, the handover decision (3) time in Figure 3 increases because the lookup time of neighbour cell lists also increases, and if the number of UEs which perform handover simultaneously increases, this handover decision (3) time in Figure 3 also increases because more resources are needed.

Therefore, it seems that the average of total handover latency will increase considerably when the number of the neighbour cell lists will be large and these will be not optimized in 5G networks and the number of UEs that perform simultaneous handover will increase dramatically due to IoT and massive small cells. As a result, some UEs will have a delayed handover. This will result in the degradation of QoS.

4. Proposed ANR Model with NFV

As we mentioned, it will be essential to optimize neighbour cell lists in 5G networks. Although recent ANR technology includes the algorithms of optimization of neighbour cell lists and these algorithms are researched continuously, these algorithms are limited (e.g., overreached scenario) and are not sufficient to consider the future network conditions (i.e., an increase of simultaneous handovers and a frequent change of neighbour cell lists due to a natural phenomenon). In this paper, although there is not the algorithm to solve these issues, it is certain that more resources are needed to consider all of them.

In addition, it is also necessary to operate ANR technology flexibly. In some cells, since neighbour cell lists are frequently modified, it is necessary to extend the capacity of the ANR function to quickly optimize neighbour cell lists. Also, it is necessary to diminish the capacity of the ANR function in order to save resources when neighbour cell lists are rarely changed. For this flexible operation of ANR technology, we propose to use NFV for operating ANR. In this case, ANR is an important function for self-optimization as well as self-configuration. Like distributed SON [27], ANR-VNF, which means that network entry is virtualized and has ANR functionality, is deployed in each eNodeB and can be extended and also be diminished. The NFV's properties make it possible. NFV makes it possible to get wanted resources assigned in a common computer and to modify this assigned resource by redeploying VNF.
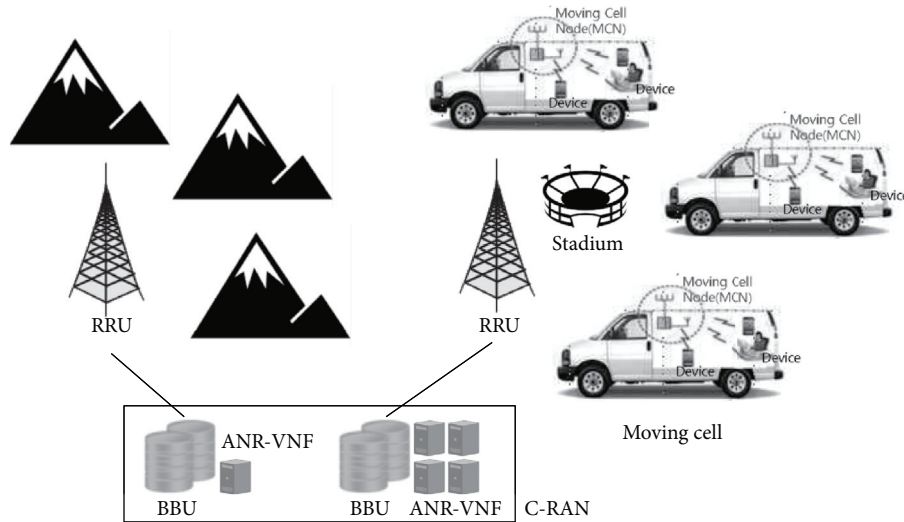
Figure 6: Conceptual diagram of proposed ANR model [20].

Figure 6 depicts a conceptual diagram of the proposed ANR-VNF model. In future, if network entities will be replaced with NFV, eNodeB will run on common computers of data centers like the BBU in this figure. The right side of Figure 6 depicts that ANR-VNF is extended in the BBU because moving cells are located near a stadium for data offload. On the contrary, the left side of Figure 6 shows that, in suburbs and rural areas, ANR-VNF is deployed with minimum resources in the BBU because neighbour cell lists in this eNodeB are rarely changed.

For ANR-VNF, it seems burdensome to deploy typical VMs (virtual machine). Thus, it is necessary to use a Linux container (e.g., Docker) to use ANR-VNF portably [28]. Unlike hypervisor virtualization, containers run in user space on top of an operating system's kernel. Thus, a container allows multiple isolated user space instances to be run on a single host [29]. In other words, if a virtualized eNodeB is deployed on a common computer, ANR-VNF can be run on this host. These containers' functionality makes the proposed strategy possible.

5. Conclusion

In 5G networks, the network environment will change in several cases. Small cells and moving cells will increase due to the frequency property and data offload. These changes will cause neighbour cell lists to be modified more frequently as we mentioned. In addition, since the coverage of cells will become small and many devices will be connected to networks due to IoT, there is no doubt that the number of simultaneous handovers will rise.

These changes will degrade handover performance if neighbour cell lists are not optimized frequently and individually, and this optimization will need more resources. The hazard from these changes is proved through several experiments. Therefore, we propose a new strategy of ANR (i.e., ANR-VNF) by using NFV to overcome this hazard. This strategy can make ANR able to respond to the change of network environments flexibly and efficiently for resource management and handover performance. In future work, we will focus on the algorithm to solve these issues and compare it with any other algorithm. In addition, it is necessary to implement and operate ANR-VNF in each eNodeB.

Competing Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.
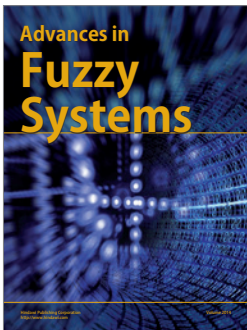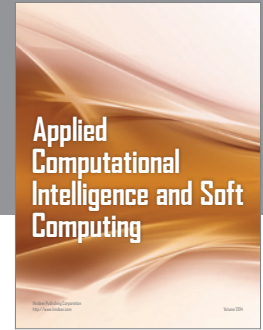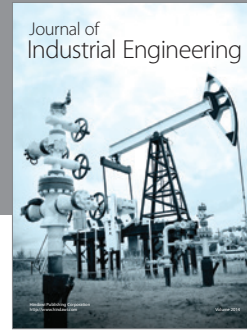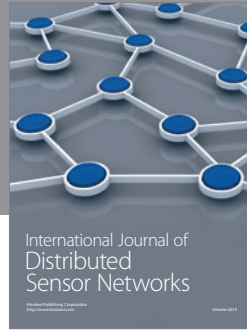
Acknowledgments

References

[1] J. G. Andrews, H. Claussen, M. Dohler, S. Rangan, and M. C. Reed, "Femtocells: past, present, and future," *IEEE Journal on Selected Areas in Communications*, vol. 30, no. 3, pp. 497–508, 2012.

[2] C. A. Boano, N. Tsiftes, T. Voigt, J. Brown, and U. Roedig, "The impact of temperature on outdoor industrial sensornet applications," *IEEE Transactions on Industrial Informatics*, vol. 6, no. 3, pp. 451–459, 2010.

[3] J. Lee and K. Chung, "An efficient transmission power control scheme for temperature variation in wireless sensor networks," *Sensors*, vol. 11, no. 3, pp. 3078–3093, 2011.

[4] J. Luomala and I. Hakala, "Effects of temperature and humidity on radio signal strength in outdoor wireless sensor networks," in *Proceedings of the Federated Conference on Computer Science and Information Systems (FedCSIS '15)*, pp. 1247–1255, IEEE, 2015.

[5] C. Ortega-Corral, L. E. Palafox, J. A. García-Macías, J. Sánchez-García, L. Aguilar, and J. I. Nieto-Hipólito, "Parameter optimization of a temperature and relative humidity based transmission power control scheme for wireless sensor networks," *International Journal of Distributed Sensor Networks*, vol. 2015, article no. 20, Article ID 921319, 2015.

[6] A. J. Fehske, I. Viering, J. Voigt, C. Sartori, S. Redana, and G. P. Fettweis, "Small-cell self-organizing wireless networks," *Proceedings of the IEEE*, vol. 102, no. 3, pp. 334–350, 2014.

[7] C. M. Mueller, H. Bakker, and L. Ewe, "Evaluation of the automatic neighbor relation function in a dense urban scenario," in *Proceedings of the IEEE 73rd Vehicular Technology Conference*, pp. 1–5, Yokohama, Japan, May 2011.

[8] S. Hamalainen, H. Sanneck, and C. Sartori, *LTE Self-Organizing Networks (SON)*, John Wiley & Sons, New York, NY, USA, 2012.

[9] J. T. Penttinen, *The Telecommunications Handbook: Engineering Guidelines for Fixed, Mobile and Satellite Systems*, John Wiley & Sons, Ltd, Chichester, UK, 2015.

[10] Y. Watanabe, Y. Matsunaga, K. Kobayashi, H. Sugahara, and K. Hamabe, "Dynamic neighbor cell list management for handover optimization in LTE," in *Proceedings of the IEEE 73rd Vehicular Technology Conference (VTC '11-Spring)*, Budapest, Hungary, May 2011.

[11] P.-C. Lin, "Minimization of drive tests using measurement reports from user equipment," in *Proceedings of the IEEE 3rd Global Conference on Consumer Electronics (GCCE '14)*, pp. 84–85, IEEE, Tokyo, Japan, October 2014.

[12] L. S. Mohjazi, M. A. Al-Qutayri, H. R. Barada, K. F. Poon, and R. M. Shubair, "Self-optimization of pilot power in enterprise femtocells using multi objective heuristic," *Journal of Computer Networks and Communications*, vol. 2012, Article ID 303465, 14 pages, 2012.

[13] TS ETSI. 136 300 v8. 12.0, Evolved Universal Terrestrial Radio Access (E-UTRA) and Evolved Universal Terrestrial Radio Access Network (E-UTRAN); Overall description; Stage 2 (3GPP TS 36.300 version 8.12.0 Release 8).

[14] D. Ortega-Sicilia, F. Cabrera Almeida, A. Sedeño Noda, and A. Ayala-Alfonso, "Design and evaluation of ANR algorithm for LTE real scenario with high interference," *Electronics Letters*, vol. 51, no. 24, pp. 2057–2058, 2015.

[15] R. Mijumbi, J. Serrat, J.-L. Gorricho, N. Bouten, F. De Turck, and R. Boutaba, "Network function virtualization: state-of-the-art and research challenges," *IEEE Communications Surveys and Tutorials*, vol. 18, no. 1, pp. 236–262, 2015.

[16] B. Han, V. Gopalakrishnan, L. Ji, and S. Lee, "Network function virtualization: challenges and opportunities for innovations," *IEEE Communications Magazine*, vol. 53, no. 2, pp. 90–97, 2015.

[17] R. Guerzoni, "Network functions virtualisation: an introduction, benefits, enablers, challenges and call for action," in *Proceedings of the SDN and OpenFlow World Congress*, 2012.

[18] J. Wu, Z. Zhang, Y. Hong, and Y. Wen, "Cloud radio access network (C-RAN): a primer," *IEEE Network*, vol. 29, no. 1, pp. 35–41, 2015.

[19] H. Hawilo, A. Shami, M. Mirahmadi, and R. Asal, "NFV: state of the art, challenges, and implementation in next generation mobile networks (vEPC)," *IEEE Network*, vol. 28, no. 6, pp. 18–26, 2014.

[20] Y. Hwang and J. Shin, "A user centric moving cell management mechanism in LTE-Advanced system," in *Proceedings of the 5th International Conference on Information and Communication Technology Convergence (ICTC '14)*, pp. 243–245, Busan, Korea, October 2014.

[21] "Software LTE base station on PC Amari LTE 100," http://www.amarisoft.com/.

[22] R. Trestian, Q.-T. Vien, P. Shah, and G. Mapp, "Exploring energy consumption issues for multimedia streaming in LTE HetNet small cells," in *Proceedings of the IEEE 40th Conference on Local Computer Networks (LCN '15)*, pp. 498–501, IEEE, Clearwater Beach, Fla, USA, October 2015.

[23] X. Xiong, T. Wu, H. Long, and K. Zheng, "Implementation and performance evaluation of LECIM for 5G M2M applications with SDR," in *Proceedings of the IEEE Globecom Workshops (GC Wkshps '14)*, pp. 612–617, December 2014.

[24] J. Han and B. Wu, "Handover in the 3GPP long term evolution (LTE) systems," in *Proceedings of the Global Mobile Congress (GMC '10)*, IEEE, Shanghai, China, October 2010.

[25] A. Hans, A. Sharma, K. Kumar, and N. Singh, "An overview of handoff procedure in LTE technology," in *Proceedings of the International Conference on Medical Imaging, m-Health and Emerging Communication Systems (MedCom '14)*, pp. 391–394, Greater Noida, India, November 2014.

[26] R. J. Freund, D. Mohr, and W. J. Wilson, *Statistical Methods*, Elsevier, Amsterdam, The Netherlands, 3rd edition, 2010.

[27] S. Feng and E. Seidel, *Self-Organizing Networks (son) in 3gpp Long Term Evolution*, Nomor Research GmbH, Munich, Germany, 2008.

[28] C. Boettiger, "An introduction to docker for reproducible research," *ACM SIGOPS Operating Systems Review*, vol. 49, no. 1, pp. 71–79, 2015.

[29] J. Turnbull, *The Docker Book: Containerization is the new virtualization*, James Turnbull, 2014.