SOCIETY FOR SCIENCE AND EDUCATION
UNITED KINGDOM

TMLAI TRANSACTIONS ON MACHINE LEARNING AND ARTIFICIAL INTELLIGENCE

Volume 6 No. 5
ISSN 2054-7390

Enhanced Handover Clustering and Forecasting Models Based on Machine Learning and Big Data

Luong-Vy Le[1], Bao-Shuh Paul Lin[2,3], Li-Ping Tung[3], Do Sinh[2]

[1]*College of Electrical and Computer Engineering, National Chiao Tung University, Hsinchu, Taiwan*
[2]*Department of Computer Science, National Chiao Tung University, Hsinchu, Taiwan*
[3]*Microelectronics & Information Research Center, National Chiao Tung University, Hsinchu, Taiwan*
leluongvy.eed03g@nctu.edu.tw, bplin@mail.nctu.edu.tw, lptung@nctu.edu.tw, dosinhuda.cs04g@nctu.edu.tw

ABSTRACT

In mobile networks, handover (HO) is one of the most important and complex KPIs (Key Performance Indicators), which directly affect to Quality of Service (QoS), Quality of Experience (QoE), and mobility performance. Moreover, its configuration parameters such as handover thresholds and handover neighbor lists are the key factors for implementing network optimization such as load balancing and energy saving. In a study before, the authors proposed clustering and forecasting models using ML algorithms and Time Series models to cluster, forecast, and manage the HO behavior of a huge number of cells. In this study, on the other hand, the authors firstly investigated more network KPIs to analyze their relationship with HO KPIs, and then, proposed new clustering, forecasting, and abnormal detection models that are expected to make them much more comprehensive. Finally, the performances of the proposed models were evaluated by applying them to a real dataset collected from the HO KPIs and other KPIs of more than 6000 cells of a real network during the years, 2016 and 2017. The results showed that the study was successful in identifying the relationship among network KPIs and significantly improving the performance of the HO clustering, forecasting, abnormal detection models. Moreover, the study also introduced the integration of emerging technologies such as machine learning (ML), big data, software-defined network (SDN), and network functions virtualization (NFV) to establish a practical and powerful computing platform for future self-organizing networks (SON).

Keywords: key performance indicators (KPIs); handover, Machine Learning; clustering, forecasting; SDN/NFV, SON; 5G; big data.

1 Introduction

In 5G networks, dense heterogeneous architectures of macrocells and small cells, are expected as a promising solution to overcome the exponential growth of broadband traffic (1000-fold capacity improvement) produced by massive IoT devices (over 25 billion devices are connected by 2020) and new vertical business services (e.g., healthcare, augmented reality, content delivery network (CDN), automotive system, entertainment). Although, the deployment of a huge number of small cells of ultra-dense networks (UDN) brings such benefit for network operators in improving indoor coverages, service quality, system capacity, energy saving, spectral efficiency, and the cost of expenditure, it carries many challenges for network management that may cause of the increase in the operational cost. Therefore,

DOI: 10.14738/tmlai.65.5411
Publication Date: 30th October, 2018
URL: http://dx.doi.org/10.14738/tmlai.32.5411

the SON in 5G must be significantly improved from the current 4G SON and pushed its functions to a next level that be able to provide full intelligence, faster computation, automatic management and optimization to fulfill network QoS and QoE [1][2].

In mobile networks, KPIs are used to judge the network performance, evaluate network operation quality, and statistical network traffic. Generally, they are collected on the hourly period and daily period by Operation and Maintenance Center (OMC) or by counters located at eNodeBs[3][4]. Their values are extremely important for network optimization to ensure that the system is operating normally at the peak of performance through evaluating the success or failure rate of different indicators reflecting the QoS regarding user perspective. For example, Radio Resource Control (RRC) success rate represents for the connection setup success rate, and Handover outgoing success rate (HOSR) represents the proportion of total number handover attempts that result in successfully completed handovers.

Recently, ML, big data, cloud computing, and SDN/NFV have been introduced as emerging technologies for the SON to deal with such requirements by reinforcing the SON talent in processing a massive data of HetNets and UDNs [5][6][7][8][9][10]. Moreover, the current development of many robust platforms like Apache Spark, Kafka, Zookeeper, IBM InfoSphere, and SDN controller (e.g., ONOS, OpenDaylight) carries a great opportunity to empower the SON with intelligence to make it much more comprehensive. This done to fully shift from being reactive to being a proactive SON with minimum human intervention by integrating various capacities: self-configuration, self-optimization, and self-healing. These functions facilitate the manual workload of the network and make the network management more economically. Recently, many researchers focused on developing ML, big data, and SDN/NFV models analyzing network KPIs to empower SON. For example, research [11] introduced a framework to empower the SON (called BSON) based on network KPIs and ML algorithms to solve challenges in the 5G SON; research [3] proposed and analyzed adaptive SON management using KPI measurements; study [4] analyzed the impact of SON function on the KPI behavior in realistic large-scale network scenarios; research [12] proposed and implemented a HO procedure based on a 5G SDN-based network architecture; moreover, research [13][14][15] investigated and developed many SDN-based architectures and algorithms for managing HO in HetNets.

In the previous study [1], we showed that HO management plays a crucial role in improving network quality, management efficiency, and mobility performance in mobile networks[1]. For example, an HO clustering model identifies cells that their HO behaviors are similar. As a result, it can support the self-configuration to provide plug-and-play functions, such as when a new component is deployed in the RAN (radio access network), it automatically configures HO threshold, radio parameters, and neighbor cell list [16][17]. On the other hand, an HO forecasting model that precisely predicts the future HO demand of cells can assist the self-optimization [18][19] to optimize and adjust network parameters (NPs) and handover parameter in response to real-time circumstance to guarantee that the system is working at its peak performance. Meanwhile, an abnormal HO detection model can support the self-healing to monitor the network and trigger rapid fault recovery by diagnosing the failure and then taking appropriate compensation mechanisms such as change the NPs to keep the network operating smoothly.

In this study, the authors propose several applications based on different ML algorithms for managing HO. Firstly, the relationship of HO and other KPIs is analyzed to identify KPIs that primarily affect to HO behaviors. Secondly, an HO clustering model is proposed to extract cells having the same HO patterns by exploring the complicated HO behavior of regularities in a dataset, and then, the HO patterns of the clusters and cells will be extracted. After that, we propose an HO forecasting model to forecast the future HO demand in cells or clusters based on several important KPIs using various ML algorithms. Besides, based on the results of clustering and forecasting models, we propose an effective abnormal HO detection model. Finally, the performance of these models will be evaluated and compared with those in the previous study.

The remainder of the paper is organized as follows: Section 2 introduces the experimental framework and computing platform based on SDN/NFV, big data, and ML; Section 3 reviews the network KPIs, HO characteristics, and ML applications for HO management; Section 4 analyzes the relationships of HO and other KPIs; Section 5 proposes and applies a new HO clustering model; Section 6 proposes and implements new HO forecasting models; Section 7 introduces an abnormal HO detection model based on clustering and forecasting results; and finally, Section 8 concludes the present study.

2 The Integration of Big Data, ML, and SDN/NFV to the 5G experimental platform

SDN/NFV, big data, and ML are considered as key ingredients and solutions for 5G to provide such significant capacities such as softwarization, virtualization, automation, high broadband bandwidth, etc. Specifically, in studies [1][12][13][14][15], they were applied to support new 5G architecture for developing various HO algorithms to perform low-latency handover.

2.1 Applying Big Data, ML, and SDN/NFV to 5G platform at NCTU

Recently, we have applied those technologies to establish a service-oriented architecture for the 4G/LTE&5G testbed located at Broadband Mobile Lab (BML), National Chiao Tung University (NCTU) [2][10][20][21][22][23]. There are essential revolutions in the 5G's main components as described in Fig.1: The C-RANs (Centralized/Cloud-Radio Access Network), Open5GSON, Open5GCore, and applications. The first revolution is in the C-RAN, which exploits small-cells, massive MIMO, and optical fibers to support low latency, high capacity, high-speed connections, cost-effective, and greener communication for a wide-area wireless connectivity of various types access technologies like LTE-E-UTRAN, UMTS-UTRAN, and Wireless Sensor Network (WSN). For example, in this architecture, a BBU (Base-Band Units) can connect to multiple RRUs (Remote Radio Heads) through optical fibers. The second revolution is in the Open5GCore, which is specifically illustrated as in research [23], in which Evolved Packet Core (EPC) is virtualized using hypervisor and container (Docker container) and its components such as P-GW, S-GW, Authorization and Accounting (AAA), home subscriber server (HSS), and mobility management entity

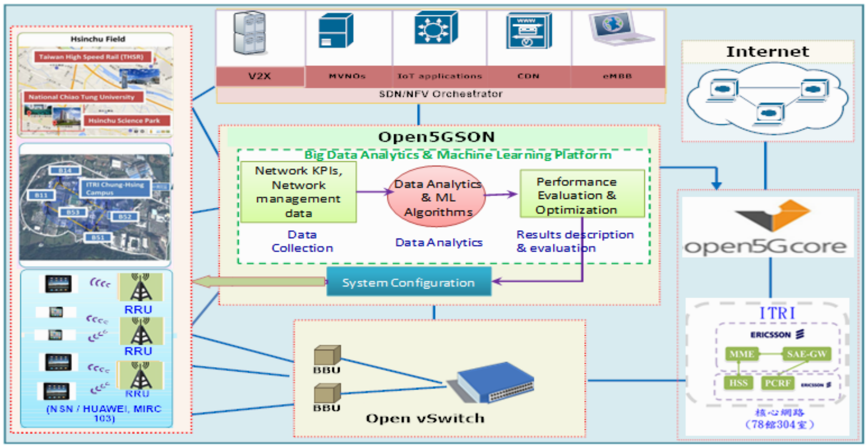

Figure 1. Experimental platform for Big Data analytics & ML at NCTU

(MME) are run on commodity hardware under the management of SDN/NFV orchestrations. As a result, in each component, the control plane is decoupled from the data plane, for example, the P-GW will contain the P-GW-C and P-GW-U. The third significant revolution is in the application layer in which SDN/NFV are deployed to enable a better support for vertical and diversified 5G services under a single network infrastructure [8]. Each application is a virtual entity deployed in a Docker container controlled by SDN/NFV orchestrations. Therefore, the deployment of 5G-based services become more flexible and efficient, such as end-to-end network slicing, network management as a service, IoT services, and massive machine type communications. The final crucial revolution is in the computing platform (e.g., Open5GMEC or Open5GSON), which works as a new component interacting with both the 5G RAN and Core to provide robust and real-time computing capabilities. It can be considered as a promising solution to simplify the core and eliminate a massive amount of data routing and processing at the core. As a result, core components can be simplified. The following section presents the computing platform in detail.

2.2 Applying Big Data and Machine Learning to the Computing Framework

The proposed computing platform described in Fig.2 is driven by 5G service requirements.

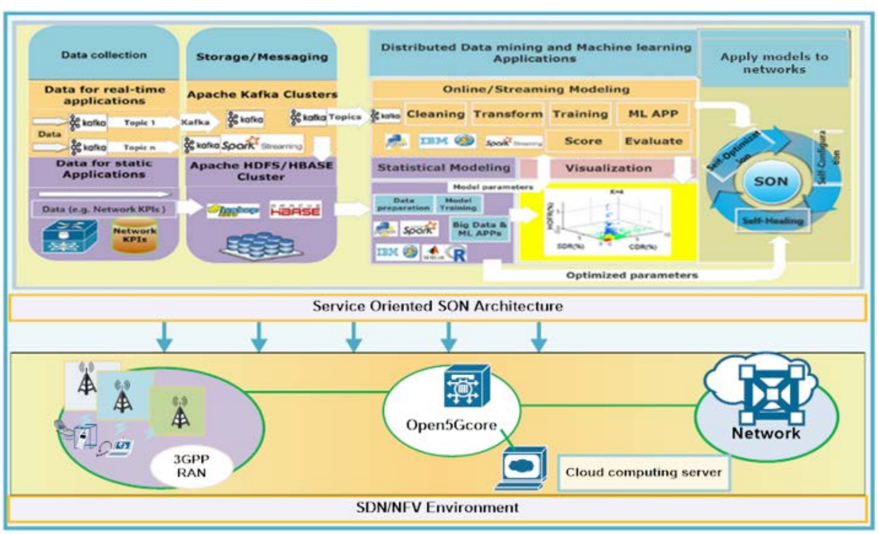

Figure 2. Applying Big Data & ML model to computing framework

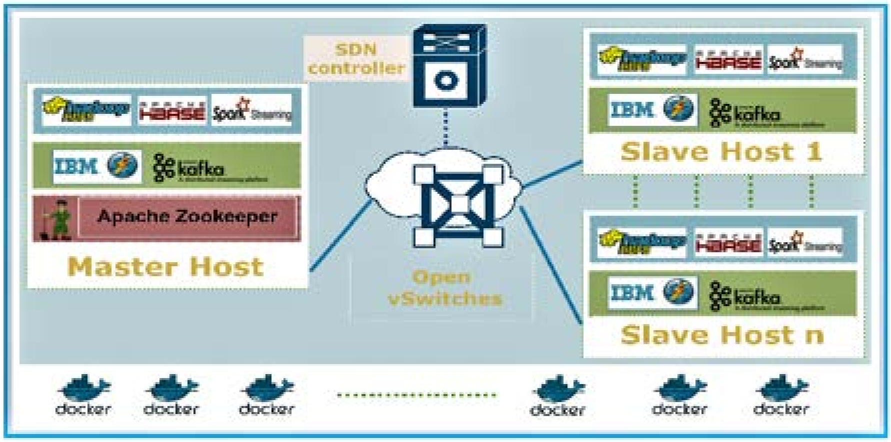

Figure 3. Distributed computing platform

It aims to provide a comprehensive SON with powerful capacities for different types of 5G applications and functions by interacting with the RAN and CORE to modify their configurations. Since it is necessary to collect and process a huge amount of data in a timely and autonomous manner to build SON applications, the computing platform integrates and utilizes state-of-the-art software, computing platforms, and programming languages, for different purposes. For example, Apache Kafka, a robust distributed streaming platform with high fault-tolerant, is used for building real-time streaming applications and data pipelines using publish-subscribe-based messaging; R can support data processing such as filtering, clearing, extraction, aggregation, visualization, and building ML and data mining models; finally, other platforms like HBase, Spark, InfoSphere, ZooKeeper, Matlab are used for data collection, storage, transformation, processing, etc.

The practical deployment of those components is a distributed computing platform as described in Fig.3, in which a host runs as the master and many hosts work as workers or executors. They connect to each other and work in the SDN/NFV environment under the control of an SDN controller. SDN/NFV program and control the network easily because the control plane is centralized with a global view. Moreover, as can be seen, in a physical machine, we can deploy multiple software components concurrently and independently, and each software component (e.g., Kafka) can work separately in its cluster under the monitor of ZooKeeper and SDN/NFV orchestrations. This makes the computing framework more powerful with full of intelligence and automation providing crucial capabilities, such as flexibility, reliability, scalability, and high computing performance. Moreover, since the components in each host run on independent Docker containers running on Linux systems of a sharing the OS' kernel, the deployment of the components becomes instant and efficient without downtime, while consuming small resources. On the other hand, each entity also provides a different approach for fault tolerance such as Apache Kafka can use partition and replication methods to improve its reliability, another example is in the case of a ZooKeeper cluster, the backup master will substitute the role of the primary master when a failure occurs in the primary master.

In the platform, a client can submit its jobs to the master or any worker, and once a computing application is submitted, the master of each relating component (e.g., Spark) distributes the job to its workers, and based on the computing load and the available computing resources, the master automatically adjusts and optimizes number of executors. In summary, this distributed computing for Open5GSON enables parallel data processing and building big data and machine learning models for on-demand services.

3 Handover KPIs and ML-Based Applications

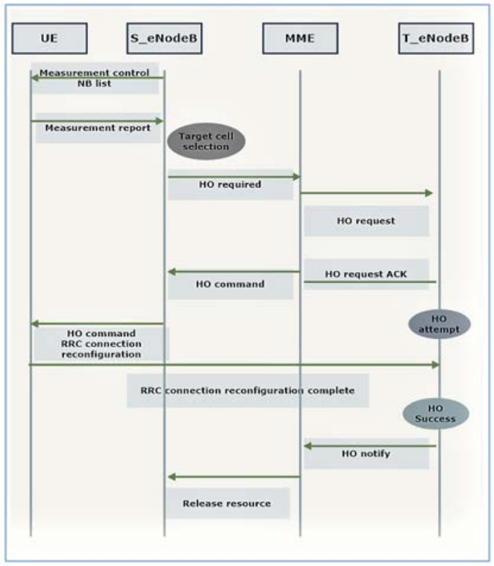

Figure 4. Inter-eNodeBs handover process

In cellular networks, handover KPIs are used to evaluate the mobility performance, which is significant to customer experience or QoE. The HO KPIs are defined for different handover types such as intra-frequency, inter-frequency, intra-eNodeB, inter-eNodeB, and inter-RAT (Radio Access Technology). This section illustrates the HO procedure, how to calculate HO KPIs, describes HO characteristics, and introduces several ML applications to manage HO KPIs.

3.1 Handover Process and KPIs

Fig.4 describes inter-eNodeBs HO process, this is the most popular HO type in cellular networks. Firstly, the UE is requested to send its measurement reports relating to some parameters such as the received signal level known as RSRP (Reference Signal Received Power) and RSRQ (Reference Signal Received Quality) of the serving cell and neighboring cells for the serving cell. If the received power and quality from the serving cell is less than a defined threshold (or much less than other cells a threshold value), the serving eNodeB selects the best target cell from the candidate list and triggers HO procedure by sending HO required message to MME, and the MME sends HO request to the target eNodeB [18] [16]. The latter will determine if there are available resources to provide for the new user. In the favorable case, the response messages are sent to the MME and the serving eNodeB, which then send the HO command to

Table 1. Examples of network KPIs

Date	Time	3G Cell name	No-SHO Attempt	No_SHO_Suc	SHOSR(%)	No_HHO Attempt	No_HHO Suc	HHOSR(%)	CS Traffic (Erl)	PS Traffic (MB)	RAB Success	RAB Attempt	2G Cell name	No SDCCH Attempt	No TCH Ass Attempt	No Incoming HO Attempt	No Outgoing HO Attempt	No SDCCH Attemp	Traffic Volume on SDCCH	TCH Traffic
04.07.2017	0:00	3HSC001	472	472	100	58	58	100	1.3333	287.4	58	58	QN001	602	17	190	280	1506	601	11.67
04.07.2017	1:00	3HSC001	269	269	100	15	15	100	1.575	653.24	139	139	QN001	347	13	110	140	1078	346	11.53
04.07.2017	2:00	3HSC001	131	131	100	12	12	100	1.4167	235.75	149	149	QN001	280	1	20	20	1118	279	1.94
04.07.2017	3:00	3HSC001	115	115	100	16	16	100	0.5583	127.71	109	109	QN001	278	6	10	60	1295	277	2.22
04.07.2017	4:00	3HSC001	245	245	100	40	40	100	1.1917	1014.4	139	139	QN001	308	5	80	70	1565	307	4.17
04.07.2017	5:00	3HSC001	238	238	100	14	14	100	1.1583	496.86	141	141	QN001	484	22	230	320	1203	483	35.28
04.07.2017	6:00	3HSC001	386	386	100	45	44	97.78	0.0167	224.66	75	75	QN001	1231	184	2080	2800	1311	1230	204.03
04.07.2017	7:00	3HSC001	853	853	100	96	96	100	1.0583	38.311	69	69	QN001	2682	265	3310	4030	1467	2681	278.89
04.07.2017	8:00	3HSC001	1376	1376	100	309	308	99.68	0.275	17.809	13	13	QN001	3534	436	5360	6430	1459	3533	430.56
04.07.2017	9:00	3HSC001	1637	1637	100	345	345	100	1.8667	101.02	131	131	QN001	3226	303	3520	4250	1479	3225	282.92
04.07.2017	10:00	3HSC001	2106	2106	100	543	543	100	1.4083	37.67	71	71	QN001	2334	321	4240	5310	1861	2333	407.92
04.07.2017	11:00	3HSC001	2106	2106	100	547	545	99.63	0.3917	26.91	14	14	QN001	1966	258	3530	4290	1434	1965	312.78
04.07.2017	12:00	3HSC001	1910	1908	99.9	380	380	100	1.075	190.73	135	135	QN001	1649	170	3260	3760	1557	1648	329.58

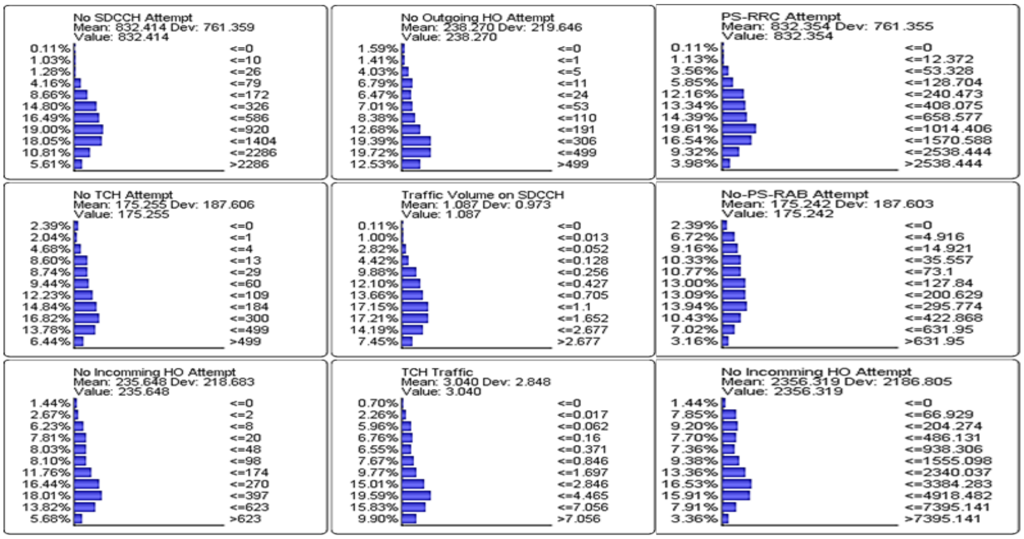

Figure 5. Marginal probability distributions of HO and several KPIs

confirm the target cell and RRC reconfiguration information for the UE. Finally, the connection is switched to the target cell, which then notifies to the MME to release the old resources at the serving eNodeB. During the HO procedure, there are several important HO KPIs are collected at both eNodeBs: the source eNodeB counters collect outgoing HO KPIs, and the target eNodeB counters collect incoming HO KPIs. For example, in Fig.4, the incoming HO attempts is calculated at the point when the target eNodeB sends the Handover Request Acknowledge message to the MME; and the number of successful incoming HO is measured at the point when the target eNodeB receives the RRC Connection Reconfiguration Complete message sent from the UE. Table 1 is an example of several typical KPIs and their hourly values of a 3G cell and a GSM cell, from early morning to noon duration. They represent for number of incoming HO attempts, number of successful incoming HO, incoming HO success rate for soft HO and hard HO; voice traffic (CS traffic), data traffic (PS traffic); number of RAB (Radio Access Bearer) attempts and success, number of TCH (traffic channel) attempts, etc. Moreover, to summarize the distribution values of several important KPIs, their hourly values were clustered in 10 groups by applying K-means algorithm, and the marginal probability distributions are shown in Fig.5, which provides a deeper understanding of the characteristics of these KPIs. This monitor histogram also illustrates a comparison of KPIs statistical values, such as the probability and value of each cluster, the mean values, and the deviation values.

3.2 Applying Machine Learning to HO Management

Recently, Machine learning widely exhibited as the breakthrough solution to reinforce 5G SON and MEC talent for solving various problems automatically [1][2][17]. Fig.6 introduces possible ML-based applications for managing HO in mobile networks, the following describes several HO applications.

Clustering HO models use unsupervised learning to group a set of cells that their HO behaviors are similar. The popular and efficient ML algorithms for clustering models are K-means, relevance determination Automatic (ARD), Mixtures of Gaussians, etc.

HO forecasting aims to accurately forecast the future HO demand of a cell, an eNodeB, even for an area. It has crucial roles in improving QoE and QoS, HO management, load balancing, and congestion avoidance by setting relevant HO parameters. The HO forecasting models usually use real-time HO data collected from various counters, and the prevalent method for HO forecasting is Time Series models, which are based primarily on the historical patterns of HO to forecast the future one, and the suitable ML algorithms are dynamic models like neural networks (NN), Linear Dynamical Systems, and Gaussian Process (GP).

HO diagnosis and decision making models analyze the current condition of a UE or network parameters such as UE tracking to predict HO trend of the UE at an early stage, and then, the SON can provide timely controlling actions to ensure the accuracy of HO decision. The most suitable ML algorithms for decision making are dynamic ML algorithms like HHM, Kalman filter, dynamic Bayesian networks, reinforcement learning, and deep learning.

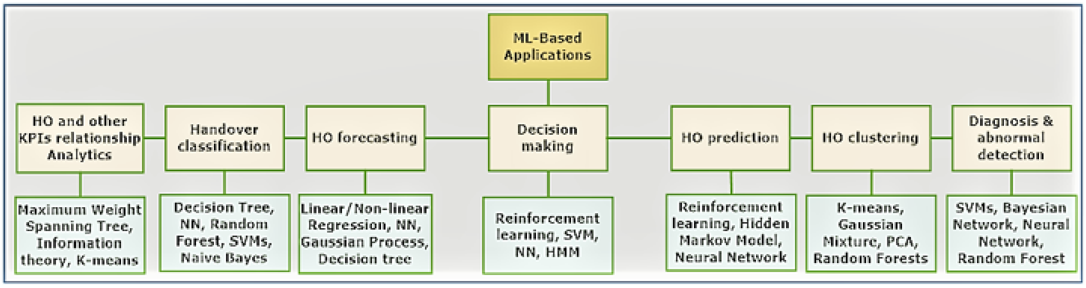

Figure 6. ML-Based applications for HO management

4 Analytics of the relationships of HO and other KPIs using ML

In 5G networks, each eNodeB is expected to have more than 2000 parameters that produce a massive amount of data or KPIs data [4]. Moreover, since network KPIs significantly affect each other, understanding the fundamental relationships between them has an important role in managing and analyzing the network behaviors for optimizing purposes like fault detection. This study proposes Maximum weight spanning tree (MWST), an unsupervised learning algorithm, to tackle this challenge. It analyzes and builds a structural network describing the relationships between KPIs, and then, provides a deeper awareness of KPIs characteristics. Moreover, based on the structure, we can extract the crucial KPIs that relate and impact to the HO KPIs. In general, the model analyzes a set of variables from the pure collected data without having any specific acknowledgment about the relationship between them. The goal is to find the tree that maximizes the data likelihood, in other words, find the tree with the greatest total weight. A weight is a mutual information between 2 nodes.

This study analyzes the hourly values of 9 typical network KPIs from the collected data to find out their relationship, especially, we also analyze the future number of HO attempts as a normal KPI in this model. This is important for HO forecasting applications in the next section. In addition, the Time is also considered as a KPI because it impacts directly to other KPIs such as the traffic of the cell.

MWST is a popular algorithm for structural learning in the Bayesian network. It is a constrained algorithm, which has one parent per node. As the result, the learning time is much faster than those of other algorithms. Fig.7 and Fig.8 show the structural networks representing relationships between the variables after analyzing the dataset for GSM cells and 3G&4G cells, respectively. It is noticeable that two structural networks describe exactly the hypothetical relationships of cellular network KPIs, and they also provide such useful and interesting information. For example, in the case of GSM KPIs, we can learn the procedure of making a call, it starts by requiring an SDCCH (standalone dedicated control channel) for the signaling and the call setup purpose, and then a TCH is assigned to UE (number of TCH attempts) for carrying user information (speech or data), which consumes the traffic of the serving cell.

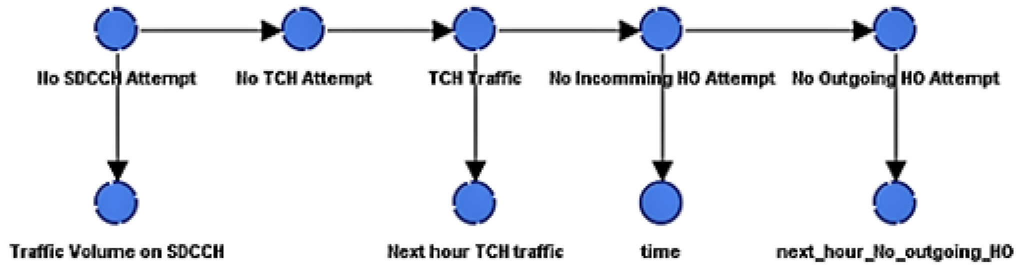

Figure 7. The relationship structure of GSM KPIs

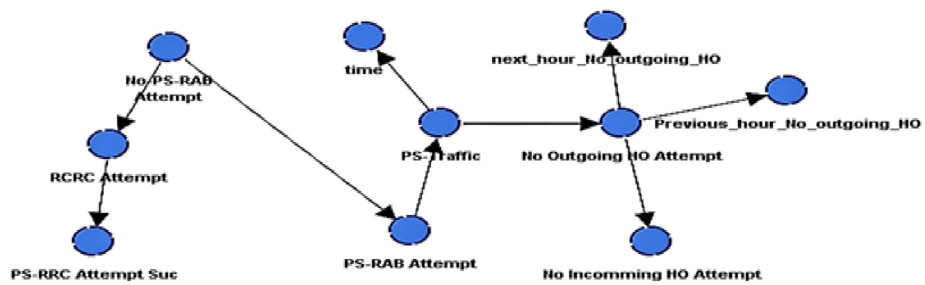

Figure 8. The relationship structure of 3G&4G KPIs

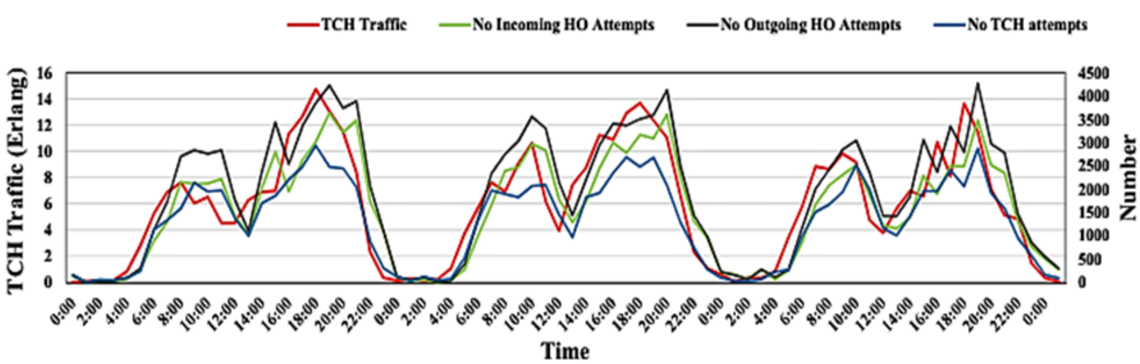

Figure 9. Hourly profiles of several KPIs

Meanwhile, the handovers usually happen during the call. To show the relationships clearer, Fig.9 shows the similar patterns of four KPIs (TCH traffic, number of incoming and out outgoing HO attempts, and number TCH attempts) during 3 days. On the other hand, the structural network also shows that the number of SDCCH attempts, the number of TCH attempts, and the number of incoming handover impact to the SDCCH traffic, the TCH traffic, and the number of outgoing HO, respectively. Finally, it is important to identify several factors that impact to the future number of HO attempts such as the TCH traffic, the time, and the number of incoming HO. The detailed analysis of the relationships will be discussed in the HO forecasting section. In brief, this application is a practical solution for future networks to analyze relationships of a huge number of KPIs and NPs.

5 Improving the Performance of Handover Clustering Model

HO patterns of cells in cellular networks are complicated, depending on time scales (e.g., hour, day), cell configuration, and the geographic location of the cell. Since understanding the HO behavior of cells is important for network management, in the research [1], the authors proposed a clustering model to put cells that have the most similar HO patterns into a cluster. Moreover, the model performance was investigated by comparing the HO behaviors among cells in a cluster and cells between clusters. Besides, we also extracted basic HO characteristics and the distributions of HOs under a day and a week of different clusters. However, the previous clustering model was built based on the HO features only without considering other KPIs, as the result, there were some cells in the same cluster that their HO patterns were similar but the probability of HO that occurs during a call of each cell might be different. To deal with this challenge, this study enhances the previous model by considering other features.

5.1 Features selection

The first main factor that impacts to the HO pattern of a cell is the probability that UEs move into or out of the cell when they are making calls at a specific time of a day. For example, cells that cover highways or high speech rails usually have a higher HO probability than those of cells that cover offices or buildings. Moreover, the HO probability of a cell is represented by the relationship between its traffic and number of HO attempts. As the result, if the traffic of these cells is similar, the number HO attempts of the cells in the first group (cover high-speed rails) must be significantly greater. On the other hand, the number of HO attempts also depend on the time such as during the day, during weekdays and weekends, monthly, even yearly [1]. For example, cells that cover school campuses usually have a higher number of HO in the daytime duration, while cells that cover night market areas usually have a higher HO trend in the night time duration. Hence, to achieve higher performance of clustering models, input features must accurately represent the two main factors affecting the HO behaviors of the cells.

Different from the previous study, the clustering model was built based on the HO features only, this study considers features extracted from different KPIs: the values of traffic KPIs and HO KPIs are used to capture

Table 2. Input features selection for the clustering model

	Time (Hour)	0:00	1:00	2:00	3:00	4:00	5:00	6:00	7:00	8:00	9:00	10:00	11:00
Cell 3QI0015	Avarage HO values	391.13	267.80	194.93	169.53	190.60	269.80	647.80	1248.53	1837.20	2212.93	2074.20	2153.20
	Percentage HO (%)	1.15	0.79	0.57	0.50	0.56	0.79	1.90	3.66	5.39	6.49	6.09	6.32
	Avarage values	179.57	102.70	3.70	12.02	11.78	51.97	41.31	65.73	111.49	124.45	105.03	155.06
	Percentage (%)	5.63	3.22	0.12	0.38	0.37	1.63	1.30	2.06	3.50	3.90	3.29	4.86
	Time	12:00	13:00	14:00	15:00	16:00	17:00	18:00	19:00	20:00	21:00	22:00	23:00
	Avarage values	1918.00	1703.40	1808.53	1882.33	2006.40	2131.20	2201.73	2287.13	2546.60	2053.73	1281.20	595.47
	Percentage (%)	5.63	5.00	5.31	5.52	5.89	6.25	6.46	6.71	7.47	6.03	3.76	1.75
	Avarage Traffic values	180.98	198.90	182.67	103.12	115.89	123.88	119.26	191.86	293.68	254.52	220.99	237.85
	Percentage Traffic (%)	5.68	6.24	5.73	3.23	3.63	3.89	3.74	6.02	9.21	7.98	6.93	7.46
	Weekday	Mon	Tue	Wed	Thu	Fri	Sat	Sun	Avarage HO values	Average hourly values	Avarage daily values	Avarage weekly values	
	Avarage HO values	33073.50	31112.00	32829.67	36496.00	33527.50	34630.00	37467.00					
	Percentage HO (%)	13.87	13.04	13.76	15.30	14.06	14.52	15.71		1419.725	34073.4	238514	
	Avarage Traffic values	1675.80	5541.71	2177.91	3516.97	4095.82	2420.91	2725.33	Avarage Traffic values	1419.72	34073.40	238513.80	
	Percentage Traffic (%)	7.51	9.76	15.76	24.83	18.35	10.85	12.21					

the HO probability of a cell, the percentage values of traffic and HO in hourly, daily and weekly distribution capture the changes of HO behaviors in the time. Table II shows an example of the input features for the clustering model (cell QI0015), the sample consists of 130 features. This section compares the performance of two clustering models. The first model is the clustering model of the previous study, which uses 65 features relating to the number of HO attempts. The second model or the new model uses all 130 features as shown in Table 2.

5.2 Experimental Set Up Performance Analytics

Experimental set up: This experiment uses K-means to cluster 2000 cells into 40 clusters, the number of clusters is high enough to guarantees that the number of clusters is manageable and the HO behaviors of cells in the same group are most similar without many clusters that cover a small number of cells. Since K-mean is a distance-based algorithm, all input attributes are normalized into a comparable range by using z-score normalization.

Performance analytics: Each clustering model divided 2000 cells into 40 groups, whose sizes were summarized as in Table 3. After identifying and assigning cells into their respective cluster, the number HO attempts of each cluster were determined by the average values of all cells belonging to it. Fig.10 and 11 show the hourly HO patterns of all 40 clusters extracted from each model, respectively, for 4 days. As can be seen, the HO pattern of each cluster was identified and distinguishable from others, the difference

Table 3. Handover clustering result summarization

Cluster	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
Clustering result (first model)	27	51	19	130	13	33	52	78	29	83	138	7	66	17	6	66	40	17	79	28
Clustering result (second model)	36	57	84	56	53	56	12	20	103	74	72	97	55	37	50	74	46	58	61	62

Cluster	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39
Clustering result (first model)	5	3	13	4	72	41	34	19	31	63	89	82	109	78	87	90	16	40	19	126
Clustering result (second model)	43	29	93	43	45	40	48	43	44	60	26	26	43	23	75	33	31	19	10	63

can be average values, patterned shapes, cluster sizes, etc. However, it is noticeable that with the same input dataset, the cluster patterns of the second model separate more smoothly and steadily from others than those of the first model. For example, in Fig.10, there are some clusters that their patterns change dramatically and unpredictably during different days.

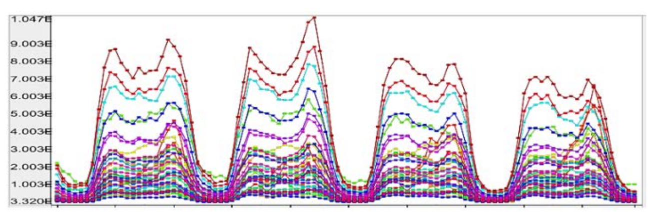

Figure 10. HO profiles of all Clusters extracted from the first model

That means the second model produced the clusters that their behaviors were more straightforward to identify, analyze, and manage.

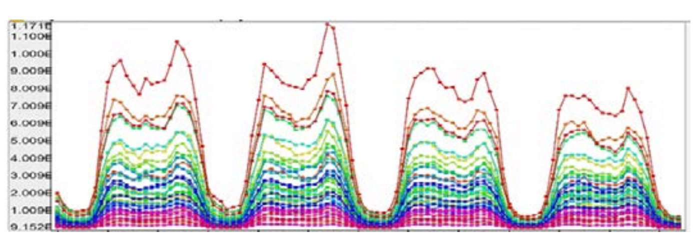

Figure 11. HO profiles of all Clusters extracted from the second model

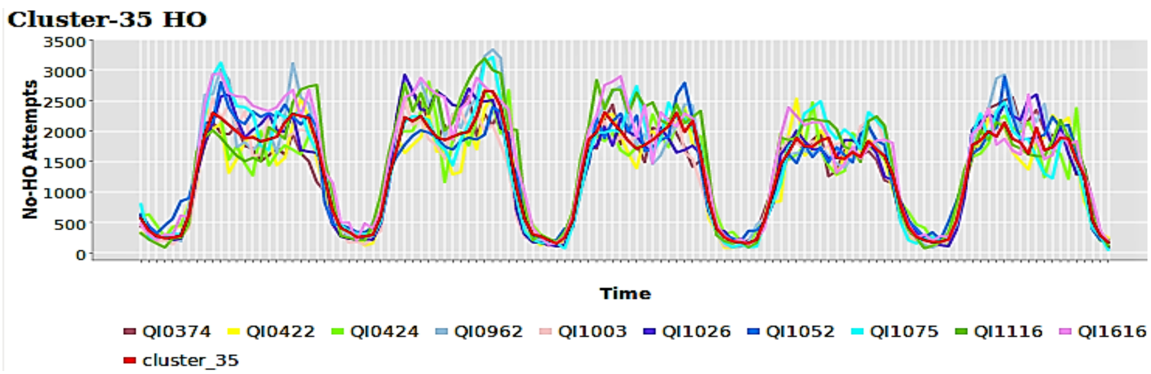

Figure 12. HO profiles of 10 cells in Cluster-35 of the first model

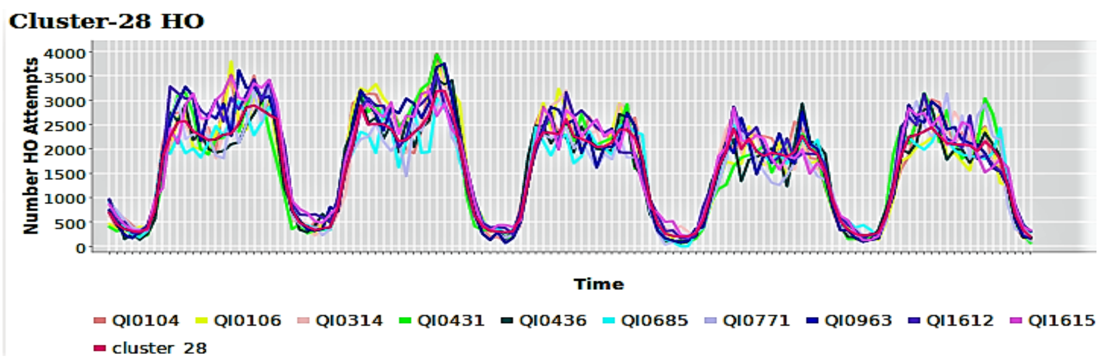

Figure 13. HO profiles of 10 cells in Cluster-28 of the second model

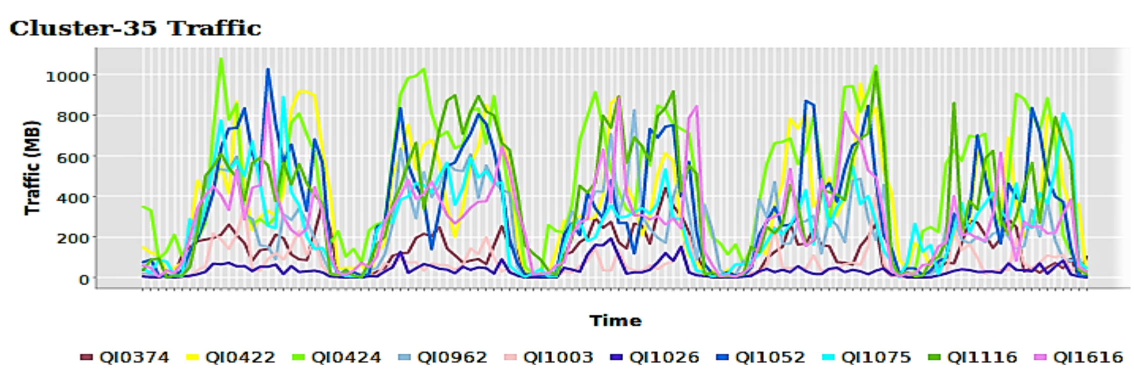

Figure 14. Traffic profiles of 10 cells in Cluster-35 of the first model

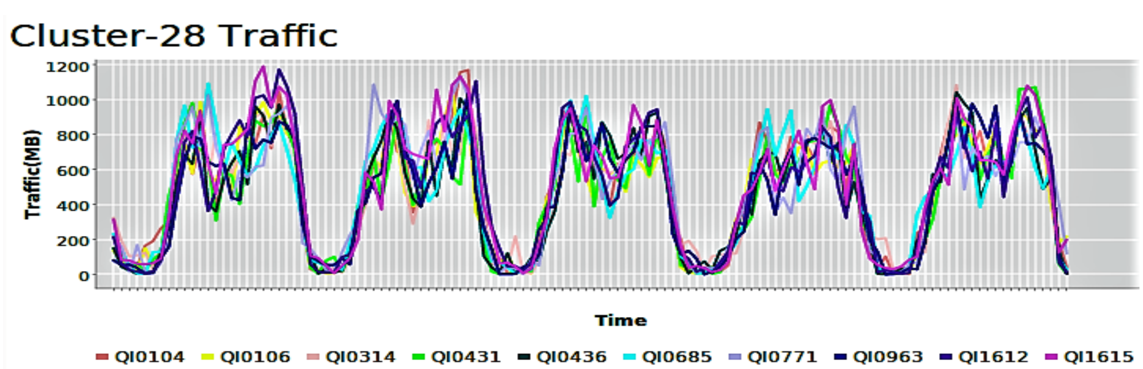

Figure 15. Traffic profiles of 10 cells in Cluster-28 of the second model

Next, to investigate the HO patterns of a cluster and its cells, each cluster of each model, Cluster-35 of the first model and Cluster-28 of the second model, were selected. Their HO patterns are quite similar and their sizes are large enough (90 and 44, respectively) so that they are able to represent for the common pattern of clusters in each model to make the analysis more generally. Fig.12 and 13 show the HO patterns of these two clusters and their cells for 5 days. Here, for the sake of clarity, only 10 cells of each cluster were randomly selected and displayed. It is visible that all of the HO patterns in the same cluster are quite similar, and they only fluctuate around the cluster patterns. As the result, the HO pattern of a cluster can be used to represent the HO pattern of its cells.

Similarly, the traffic patterns of the cells in Cluster-35 and Cluster-18 are shown in Fig.14 and Fig.15, respectively. It is obvious that traffic behaviors of the cells in Cluster-35 vary significantly and possess different characteristics in time-domain, while those of the cells in Cluster-28 are quite similar during all time. That means the first model clustered cells that have the same HO patterns into a cluster without considering their traffic behaviors. In contrast, the second model clusters cells that the same both HO behaviors and traffic behaviors into a cluster.

To make the comparison clearer, Figs. 16 (a), (b), (c), and (d) show the correlation matrices of the HO patterns, as well as the traffic patterns of the cells in the two clusters, respectively, represented by colors. As can be seen, the cells in each cluster have high correlations of the HO patterns with each other. In other

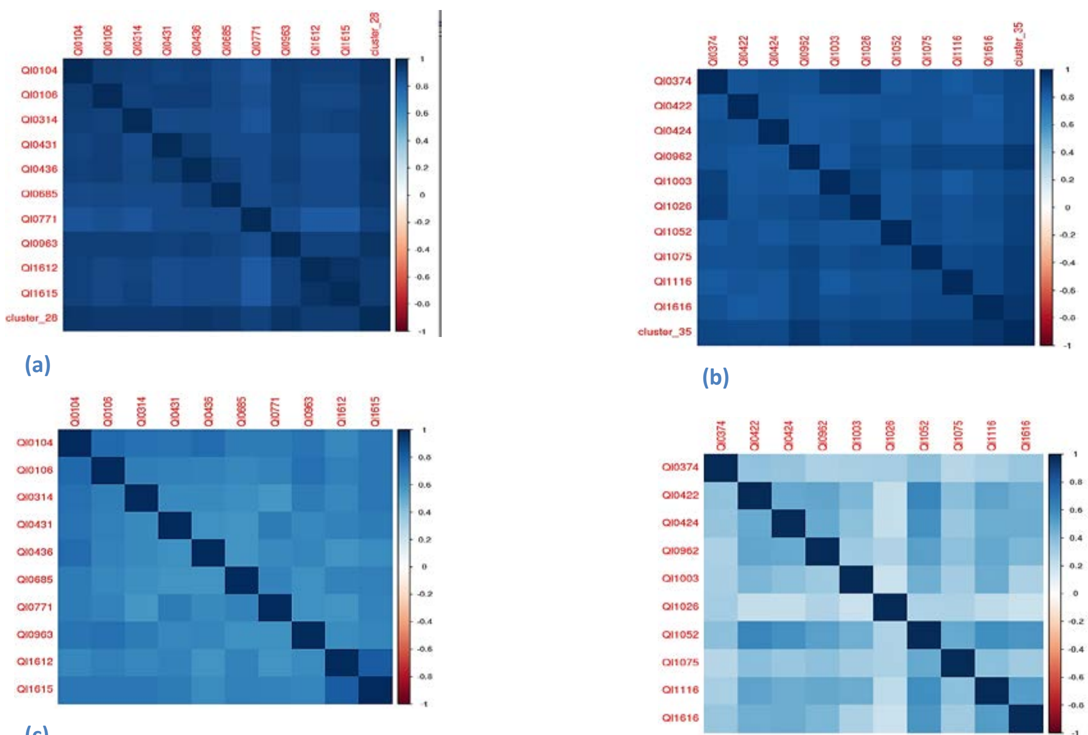

Figure 16. Correlation matrix of HO and Traffic of cells
(a) HO of 10 cells in Cluster-35; (b) HO of 10 cell in Cluster-28
(c) Traffic of 10 cells in cluster-35; (d) Traffic of 10 cells in cluster-28

words, their HO patterns are similar. However, compared with the cells in Cluster-35, those of Cluster-28 have much higher correlations of traffic patterns with each other.

In summary, the second model with the input features significantly improves the HO clustering model that can group cells into accurate clusters associated with their HO, traffic behaviors, and HO probability. This is crucial for networks operators to manage HO behaviors of a huge number of cells easier and more efficiently. For example, all cells in the same cluster covering areas with a high HO probability (e.g., along high-speech rails) should be assigned with the same suitable parameters (e.g., HO threshold) to ensure the continuity of services. Moreover, in the previous study, based on the clustering result, we extracted and classified clusters with different HO patterns in daily and weekly periods, such as weekday-higher pattern, the weekend-higher pattern, and the equally distributed pattern, which can be utilized for load balancing and energy saving applications [1].

6 Improving Performance of Handover Forecasting Models

Handover forecasting plays a crucial role in improving network quality and influences directly the mobility performance of networks. In the previous study, the authors proposed and implemented different forecasting models to forecast hourly HO demand in cells using various powerful ML algorithms. Especially, the model, which utilized HO clustering results to forecast HO patterns of any cell in a cluster by just using one forecasting model, significantly improved the performances of HO forecasting model. It was considered as a practical solution that can be applied in the industry due to the high accuracy and computing efficiency. Generally, Time-series approach, which bases primarily on the historic patterns of a parameter to predict its future values, is the most popular model for HO forecasting and prediction, while the utilization of other network parameters is limited.

In this study, the authors propose a practical model for HO forecasting based on network KPIs and ML algorithms. its process comprises two main steps: Step 1 analyzes the relationship between the HO attempts in the next hour and other KPIs in the past to determine the KPIs that have a significant impact to the future HO; Step 2 builds forecasting model based on the selected KPIs from the first step using ML algorithms. Furthermore, the performance of the proposed model will be evaluated by applying various popular ML algorithms.

6.1 Analysis of the relationship of future HO demand and network KPIs

The fundamental background for analyzing the relationship of the future HO demand and other KPIs is mainly based on information theory concepts, such as mutual information (MI) and relative mutual information (RMI). Moreover, the Bayesian network (BN) from BayesiaLab is used to analyze the probabilistic relationships among parameters and present their relationships as a directed acyclic graph (DAG) model. In this model, the next-hour HO attempt is the target node, and other KPIs such as current-hour HO attempt, current-hour TCH traffic, etc. are observation nodes. Here, Naïve Bayes, one of the most popular supervised learning algorithm of BN, is used to build the DAG.

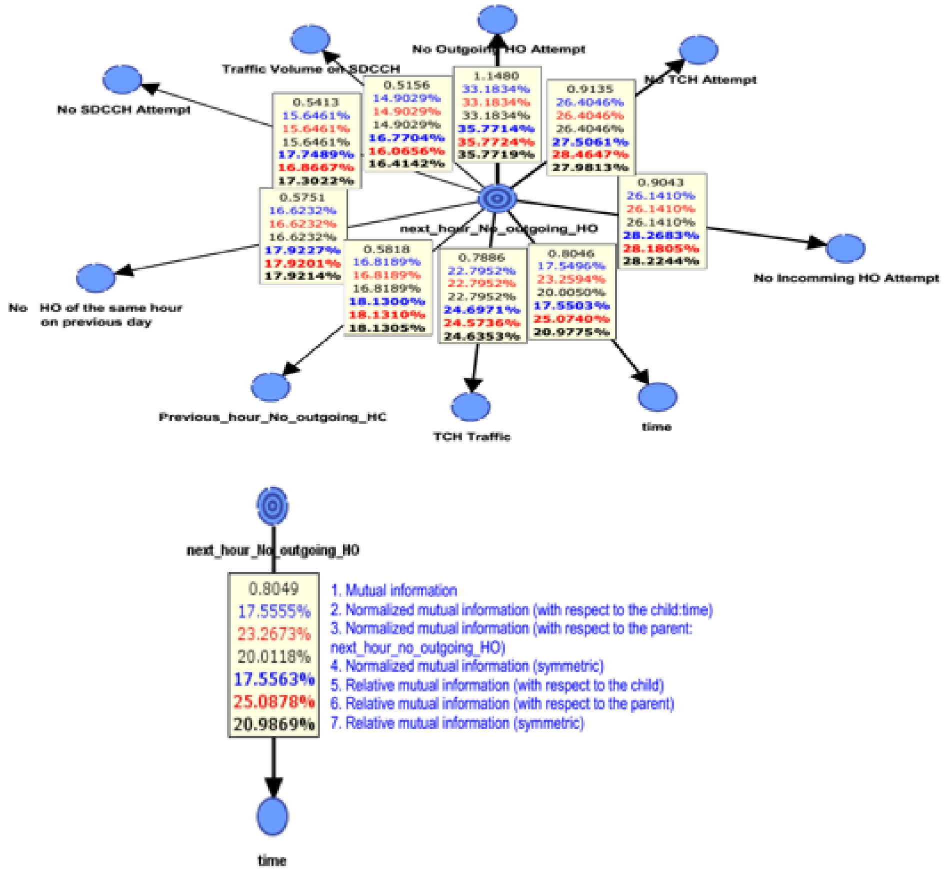

Figure 17. Graphical representation of the relationships between the next-hour HO attempts and the KPIs

Fig.17 presents the analytics results, in which the relationship between an observation node and the target node is represented by 7 information concepts defined in the right side of Fig.17. For example, the top number in the boxes is the Mutual Information, which is a measure of mutual dependence between two variables; it represents the reduction in entropy or the information it would gain, on average, about a variable if the knowledge of the other is given. Therefore, variables that have higher MI values are more significant in forecasting the target variable due to their close relationship. Similarly, the RMI is an important concept, which determines how much information that it would gain in percentage of a variable through observing another. The result shows that the number HO attempt of the current hour is the most important KPI for the HO forecasting model since it provides the highest information gain with 35.77% about the future HO. Other significant KPIs that have a primary impact on the future HO are the number of TCH attempts, the number of incoming HO, the Time, and the TCH traffic. It is noticeable that their MI and RMI values are higher than those of other historic HO KPIs, which are usually used as the main features of time series forecasting models, such as the number HO of the previous hour, the number HO of the same hour on the previous day.

6.2 HO forecasting model based on important KPIs

This section evaluates the performances of the proposed model called KPI model, which is based on relevant KPIs that have high MI and RMI values with future HO. Here, five KPIs are selected for the model: the number of HO outgoing attempts, the number of TCH traffic attempts, the number of incoming HO attempts, the Time, and the TCH traffic of the current hour. The hidden meanings behind these KPIs are that they present for the observation of the past HO pattern, the number of calls, and the mobility of Users. the performance of the KPI model is compared to the performance of the time series model, which introduced in the previous study [1]. The time series model is based mainly on the historic traces of HO to forecast future patterns. That means it uses x(t-1), x(t-2), …, x(t-N) to predict x(t), where x(t) is the future number of HO, x(t-1) is the current number of HO, and x(t-2),... , are the past number of HO demand. Here, 7 steps behind the past (N=7 (hour) will be used for hourly HO forecasting.

6.3 Machine learning algorithms

In this subsection, several popular forecasting algorithms are used to evaluate the performance of the proposed HO forecasting model. The first type of forecasting models is based on Auto-Regressive (AR) method, which is widely used for practical regression and time series model, both linear regression (LR) and polynomial regression (PR) algorithms are deployed. The second type is Neural Network (NN), which is a state-of-the-art non-linear algorithm providing a remarkable ability to derive meaning from complicated data. It is usually used to handle real data characteristics such as non-linearity to extract and detect patterns, forecast and predict trends that are too complex to be solved by other algorithms. The last ML algorithms investigated for the HO forecasting is Gaussian process (GP) algorithm, which is a kernel-based model that can provide a practical solution with a significant improvement in performance for both classification and regression applications.

6.4 Experimental setup and performance evaluation

Experimental setup:

Practically, building a forecasting model for a whole network can cause its performance degradation because HO behaviors of cells are diversity. However, building a model for each cell, it might be an overwhelming problem and inefficient. Fortunately, the clustering results in the previous section proved that the HO behaviors of all cells in the same cluster were similar to one another and similar to the cluster pattern. Therefore, building a forecasting model for all cells in a cluster can guarantee both accuracy and efficiency. For example, in our case, we only need to build 40 forecasting models for 40 clusters. Hence, the training and testing dataset for each experiment were selected as below: a training dataset of 200000 data samples and a testing dataset of 50000 data samples were selected randomly from the data of different cells in the same cluster, and samples of the testing dataset must be different from the training dataset.

Performance evaluation: The performance of each model and ML algorithm was evaluated through 4 performance metrics: the mean absolute error (E_{ma}), the root mean square error (E_{rms}), the coefficient of determination (R^2), and the correlation coefficient (Correl).

$$E_{ma} = \frac{1}{n} * \sum_{i=1}^{n} |p_i - r_i|$$
$$E_{rms} = sqrt(\frac{1}{n} * \sum_{i=1}^{n} (p_i - r_i)^2)$$
$$R^2 = 1 - \sum_{i=1}^{n} (p_i - r_i)^2 \,/\, \sum_{i=1}^{n} (r_i - \frac{1}{n} * \sum_{i=1}^{n} r_i)^2$$
$$Correl = R$$

Where r_i is the real value, p_i is the forecasted value, and n is the total number of test samples.

6.4.1 Experimental result

We examined the performance of HO forecasting models through HO patterns and the performance metrics. Firstly, Fig.18 shows the forecasted result for a cluster, cluster-25 (of the first clustering model), of the two models and the four ML algorithms for continuous 8 days. For the sake of clarity, in this figure, the forecasted HO traces of each ML algorithm and model appears only for two continuous days. It is obvious that the HO forecasted traces and the actual trace are quite similar and close to each other during all time. That means, all the ML algorithms and 2 models can accurately capture the future HO attempts of the cluster. Similarly, Fig.19 describes the experimental results when applying the same forecasting models for cell QI006 in Cluster-25. Furthermore, the statistical performances metrics of the models and ML algorithms were summarized in Table 4 to provide a detailed comparison of their performances. The first evaluation is the performance of ML algorithms when applying them to both models: LR and PR algorithms gave quite similar performances, PR performance was a little bit higher than LR performance even if the maximum polynomial degree of the PR algorithm was set to a high number; NN and GP gave better performance, and GP gave the best performance. The second evaluation is the performance of the Time Series model and the KPI model. As can be seen, when the same ML algorithm was applied, the KPI models significantly improved the forecasting performance such as higher R^2 , $Correl$, and smaller E_{ma}, E_{rms}.

In summary, the KPI model is a practical solution for handling complex patterns with different characteristics of HO behaviors. It can precisely capture the future HO demand of a cluster and any cell in this cluster with just using one model. This is crucial in industrial networks like HetNets and UDNs, which covers a huge number of cells.

Table 4. Hourly HO forecasting performances

Performance evaluation	Real values			
Model	R^2	Correl	E_{ma}	E_{rms}
LR + Time Series	0.92	0.959	137.00	218.00
LR + KPI	0.98	0.990	58.55	106.46
PR + Time Series	0.93	0.964	135.18	215.14
PR + KPI	0.98	0.990	57.40	102.42
NN + Time Series	0.953	0.976	84.90	148.06
NN + KPI	0.993	0.996	31.62	56.19
GP + Time Series	0.973	0.986	53.63	114.97
GP +KPI	0.995	0.997	25.25	50.94

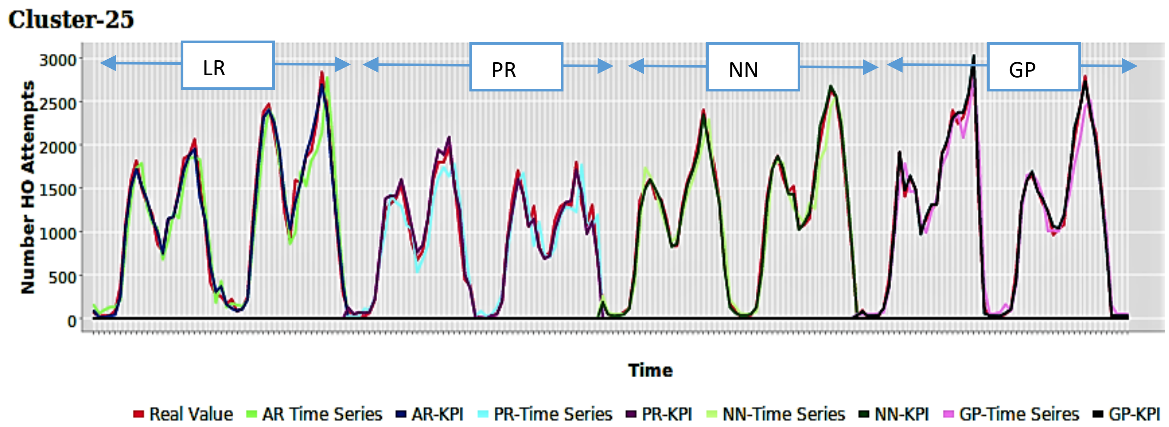

Figure 18. Forecasted hourly HO attempts results for Cluster-25

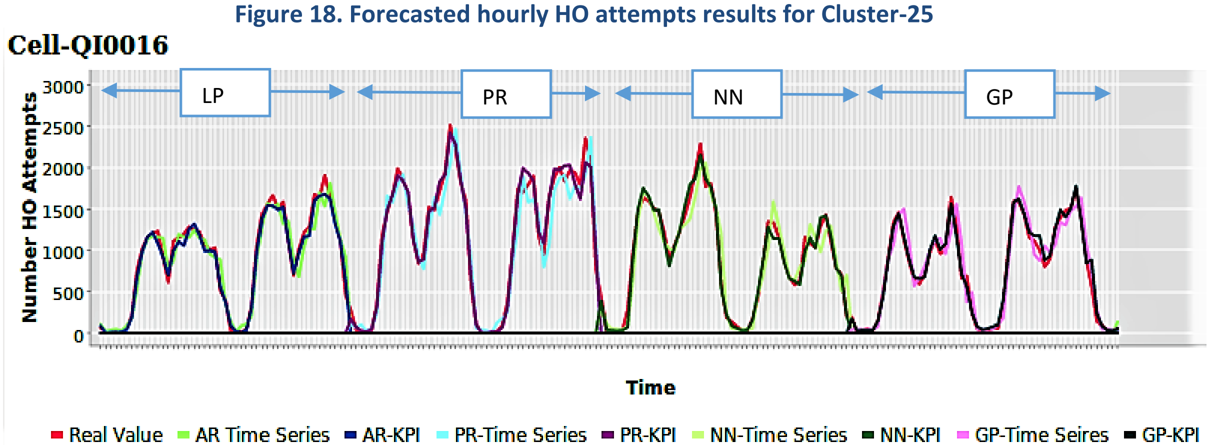

Figure 19. Forecasted hourly HO attempts results for Cell-QI0016 in Cluster-25

7 Abnormal HO detection based on HO Clustering and Forecasting Results

One of the most essential applications based on KPIs analysis, HO clustering, and forecasting is abnormal detection. Abnormal problems are unavoidable challenges of mobile networks while operating, such as equipment failures, frequency interference, and missing configuration of neighbor cells. Since most of the issues result in abnormal KPIs and HO behaviors, in the conventional network, network operators manually analyze a set of factors from a huge amount of collected KPIs dataset or through the driving test to detect abnormal in operating cells. This may take a long time and need a lot of efforts. Hence, it is in need of an urgent approach that is able to automatically and effectively detect abnormal problems.

As discussed before, the second clustering model puts cells with similar probability of HO attempts (the same traffic and HO patterns) together. However, the relationship of HO and traffic patterns is dramatically changed by abnormal issues that often produce abnormal KPIs. For example, Fig.20 shows the driving test result of an abnormal cell, cell QNI006. As can be seen, the test mobile equipment (UE) was consecutively handovered or ping-pong HO among cell QNI006 with its neighbor cells, as the result, there were many HO attempts, even dropped ongoing calls due to low C/I (the carrier to interference ratio) and HO failures. The root cause of the problem was the inter-cell interference of cell QNI006 with its new active neighbor. Thus, their Received Signal Levels (RxLev) were overlapped with each other, and

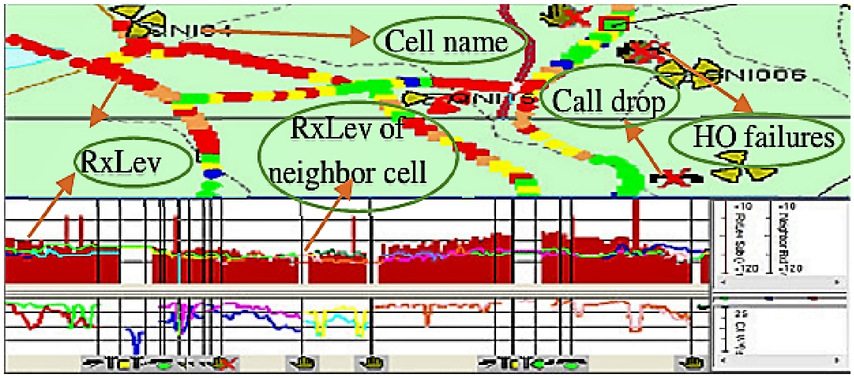

Figure 20. An example of abnormal Handover in a cell

the C/I was much lower than the regular value (C/I >12 dB). Consequently, this cell produced a high number of HO attempts, low traffic, high Call Drop Rate (CDR), and high HO-probability.

In the previous study, we proposed a useful approach called two alarm levels to detect abnormal

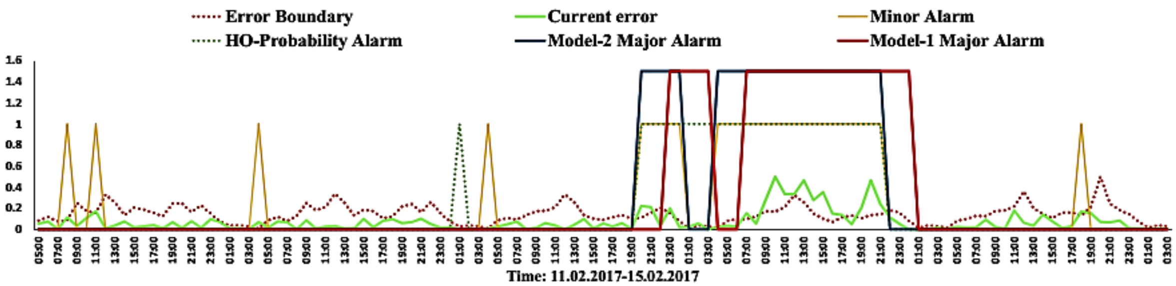

Figure 21. Abnormal detection time series of cell QNI006

situations based on HO forecasting. In this study, we propose a new practical abnormal detection based on HO forecasting and HO relationship analysis. Both models were built based on experimental results after extracting and analyzing the abnormal and failure situations through their KPIs of more than 6000 cells for 2 years. Each model involves three steps, and their first two steps are similar.

The first step: Define a parameter called *error boundary* for each hour based on calculating average errors and standard deviation errors between the forecasted HO and the real one using its moving window of 7 past values at the same hour (or during 7 days):

$$error\ boundary = average\ (error) + 3 * standard\ deviation(error)$$

(the factor 3 here is a practical factor, its value is usually in the range [2, 10]). (In real situations, we also use *error boundary* to define a parameter called HO boundary for HO congestion avoidance applications).

The second step (trigger minor alarm): calculate and compare the current error with the *error boundary*.

$IF\ current\ error\ > Boundary$, then trigger minor alarm (or $minor\ alarm = 1$).

The minor alarm notifies that there was an irregular HO behavior during the last hour. It could due to the movement of group users or an abnormal increase in traffic consumption. Since the minor alarm does not provide much meaning, it is necessary to have a further process to make the abnormal detection more reliable.

The third step of the first model (trigger model-1 Major Alarm): Both HO forecasting models are based on the past HO so that they will update any minor event causing increasing in predicted error. However, if the errors occur continuously, that means the HO behavior has been changed or there was a serious problem in the cell. To detect the major problem, the average 5 past values of minor alarm is calculated. If it exceeds 0.5 then trigger the major alarm (*or Model-1* $Major\ Alarm\ = 1.5$)

The third step of the second model (trigger model-2 Major Alarm): Similar to the *error boundary*, here we define a parameter called *HO-probability boundary* for each hour based on its moving window of 7 past values

HO-probability $= number\ of\ HO\ aptempts/Traffic$

HO-probability boundary $= average\ (value) + 3 * standard\ deviation(value)$

Next, we need to calculate and compare the current HO-probability with its boundary

$IF\ current$ HO-probability $>$HO-probability boundary, then trigger the HO-probability alarm (or HO-probability $alarm = 1$).

This value notifies that the regular relationship between HO and traffic was changed during the last hour. Finally, if both the HO-probability alarm and the minor alarm happen concurrently, then trigger the mode-2 major alarm.

If HO-probability alarm $*\ minor\ alarm = 1,$ then mode-2 $Major\ Alarm = 1.5.$

Fig. 21 illustrates the time series patterns of error boundary, current error, minor alarm, HO-Probability Alarm, Model-1 Major Alarm, and Model-2 Major Alarm of cell QNI006 since 11.02.2017 to 15.02.2017. In this figure, the values of the current error and error boundary were normalized in the range [0, 0.5] using the min-max approach. As can be seen, there was a critical issue in the cell at around 19 pm on 13.02.2017. The second model immediately triggered its major alarm while the first model required an extra delay. The problem is caused by inter-cell interference of cell QNI006 with its new active neighbor as described in Fig.20.

In summary, the second model based on the HO forecasting and HO probability is a practical solution for abnormal detection that can be applied UDNs due to the fact that it is effective and simple to implement. Moreover, it is possible to improve the model performance by using short-term forecasting models (e.g., minutes, seconds).

8 Conclusion

A comprehensive architecture for handover analytics, accurate HO clustering and forecasting, and effective HO management is the most critical challenge in UDNs and HetNets to improve mobility performances and make networks more coordinated and efficient. Firstly, establishing the relationship between HO KPIs and other KPIs is important to identify the key factors that affect HO patterns in cells, such as time series, the number of TCH attempts, and the traffic pattern. Secondly, the process proposed in this study can capture the dynamic behavior of HO and thus proposes a comprehensive approach for HO clustering, which is important for self-organization to manage HO patterns of a huge number of cells easily and efficiently. Finally, the new HO forecasting model provides an accurate HO forecasting, which is crucial for the SON to develop robust optimization applications, such as handover control, offloading traffic from the Macro-cells, congestion control, and abnormal detection. In this scenario, the SON become more robust, active and efficient in terms of managing, optimizing, and improving the network performance.

ACKNOWLEDGMENT

This paper is particularly supported by "the Center for Open Intelligent Connectivity from The Featured Areas Research Center Program within the framework of the Higher Education Sprout Project" by the Ministry of Education (MOE) in Taiwan, and the Ministry of Science and Technology of Taiwan under Grants: MOST 107-2221-E-009-056.

REFERENCES

[1] Le Luong Vy; Li-Ping Tung; Bao-Shuh Paul Lin, "Big data and machine learning driven handover management and forecasting," in *IEEE Standards for Communications and Networking (CSCN), Helsinki, 2017*, 2017, pp. 214–219.

[2] Luong-Vy Le; Bao-Shuh Lin; Do Sinh, "Applying Big Data, Machine Learning, and SDN/NFV for 5G Early-Stage Traffic Classification and Network QoS Control," *Trans. Networks Commun.*, vol. 6, no. 2, pp. 36–50, 2018.

[3] S. Lohmüller, L. C. Schmelz, and S. Hahn, "Adaptive SON management using KPI measurements," *Proc. NOMS 2016 - 2016 IEEE/IFIP Netw. Oper. Manag. Symp.*, no. Noms, pp. 625–631, 2016.

[4] S. Hahn, M. Schweins, and T. Kurner, "Impact of SON function combinations on the KPI behaviour in realistic mobile network scenarios," in *2018 IEEE Wireless Communications and Networking Conference Workshops, WCNCW 2018*, 2018, pp. 1–6.

[5] L. Le, D. Sinh, B. P. Lin, and L. Tung, "Applying Big Data, Machine Learning, and SDN/NFV to 5G Traffic Clustering, Forecasting, and Management," *IEEE NetSoft*, vol. 870, no. NetSoft, pp. 168–176, 2018.

[6] O. G. Aliu, A. Imran, M. A. Imran, and B. Evans, "A Survey of Self Organisation in Wireless Cellular Communication Networks," *IEEE Commun. Surv. Tutorials*, vol. 15, no. 1, pp. 336–361, 2013.

[7] P. V Klaine, M. A. Imran, O. Onireti, and R. D. Souza, "A Survey of Machine Learning Techniques Applied to Self-Organizing Cellular Networks," *IEEE Commun. Surv. Tutorials*, vol. 19, no. 4, pp. 2392–2431, 2017.

[8] D. Sinh, L. Le, B. P. Lin, and L. Tung, "SDN / NFV - A new approach of deploying network infrastructure for IoT," *27th Wirel. Opt. Commun. Conf. (WOCC), Hualien, Taiwan, 2018*, pp. 1–5, 2018.

[9] D. Sinh, L. Le, L. Tung, and B. P. Lin, "The Challenges of Applying SDN / NFV for 5G & IoT," *14th IEEE - VTS Asia Pacific Wirel. Commun. Symp. (APWCS), Incheon, Korea, Sep 2017*.

[10] B. P. Lin, F. J. Lin, and L. Tung, "The Roles of 5G Mobile Broadband in the Development of IoT, Big Data, Cloud and SDN," *Commun. Netw.*, vol. 8, no. 1, pp. 9–21, 2016.

[11] A. Imran and A. Zoha, "Challenges in 5G: How to empower SON with big data for enabling 5G," *IEEE Netw.*, vol. 28, no. 6, pp. 27–33, 2014.

[12] J. Prados-Garzon, O. Adamuz-Hinojosa, P. Ameigeiras, J. J. Ramos-Munoz, P. Andres-Maldonado, and J. M. Lopez-Soler, "Handover implementation in a 5G SDN-based mobile network architecture," *2016 IEEE 27th Annu. Int. Symp. Pers. Indoor, Mob. Radio Commun.*, pp. 1–6, 2016.

[13] T. Nguyen-Duc and E. Kamioka, "An extended SDN controller for handover in heterogeneous wireless network," *2015 21st Asia-Pacific Conf. Commun. APCC 2015*, pp. 332–337, 2016.

[14] E. B. Hamza and S. Kimura, "A Scalable SDN-EPC Architecture Based on OpenFlow-Enabled Switches to Support Inter-domain Handover," *2016 10th Int. Conf. Innov. Mob. Internet Serv. Ubiquitous Comput.*, pp. 272–277, 2016.

[15] C. Chen, Y. T. Lin, L. H. Yen, M. C. Chan, and C. C. Tseng, "Mobility management for low-latency handover in SDN-based enterprise networks," *IEEE Wirel. Commun. Netw. Conf. WCNC*, vol. 2016–Septe, no. Wcnc, 2016.

[16] I. Elgendi, K. S. Munasinghe, and A. Jamalipour, "Mobility management in three-Tier SDN architecture for DenseNets," *IEEE Wirel. Commun. Netw. Conf. WCNC*, vol. 30 October, no. Wcnc, pp. 214–219, 2017.

[17] P. V. Klaine, M. A. Imran, O. Onireti, and R. D. Souza, "A Survey of Machine Learning Techniques Applied to Self Organizing Cellular Networks," *IEEE Commun. Surv. Tutorials*, vol. 19, no. 4, pp. 2392–2431, 2017.

[18] M. Boujelben, S. Ben Rejeb, and S. Tabbane, "SON Handover Algorithm for Green LTE-A/5G HetNets," *Wirel. Pers. Commun.*, vol. 95, no. 4, pp. 4561–4577, 2017.

[19] J. Rizkallah and N. Akkari, "SDN-based vertical handover decision scheme for 5G networks," in *2018 IEEE Middle East and North Africa Communications Conference, MENACOMM 2018*, 2018, pp. 1–6.

[20] L. Le, D. Sinh, L. Tung, and B. P. Lin, "A practical model for traffic forecasting based on big data, machine-learning, and network KPIs," in *2018 15th IEEE Annual Consumer Communications & Networking Conference (CCNC)*, 2018, pp. 1–4.

[21] B. P. Lin, L. Tung, F. Tseng, I. Hsieh, Y. Wang, and S. Chou, "Performance Estimation of MAR for Outdoor Navigation Applications based on 5G Mobile Broadband by using Smart Mobile Devices," in *Conference: IEEE VTS APWCS 2015, Singapore*.

[22] B. S. Lin *et al.*, "The design of big data analytics for testing & measurement and traffic flow on an experimental 4G/LTE network," *2015 24th Wirel. Opt. Commun. Conf. WOCC 2015*, pp. 40–44, 2015.

[23] H. C. Chang *et al.*, "Empirical Experience and Experimental Evaluation of Open5GCore over Hypervisor and Container," *Wirel. Commun. Mob. Comput.*, vol. 2018, no. i, 2018.