IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received February 5, 2021, accepted February 19, 2021, date of publication March 12, 2021, date of current version March 22, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3065820

Extracting Cell Patterns From High-Dimensional Radio Network Performance Datasets Using Self-Organizing Maps and *K*-Means Clustering

SHAOXUAN WANG AND RAMON FERRÚS, (Member, IEEE)
Signal Theory and Communications Department, Universitat Politècnica de Catalunya, 08034 Barcelona, Spain

Corresponding author: Shaoxuan Wang (shaoxuan.wang@upc.edu)

This work was supported in part by the Spanish Research Council and FEDER funds through SONAR 5G under Grant TEC2017-82651-R, in part by the European Commission's Horizon 2020 research and innovation program through the 5G-CLARITY Project under Grant 871428, and in part by the program of China Scholarships Council under Grant 201808390034.

ABSTRACT Mobile Radio Networks produces many of Operations, Administration, and Maintenance (OAM) data used by operators for network operational assurance. These data include multiple and diverse performance measurements and indicators that characterize the behavior of the radio cells. Being able to properly cluster the apparently dissimilar behaviors exhibited by a large number of individual cells into a reduced set of prototype patterns constitutes a valuable tool to support multiple processes such as cell configuration optimization or fault performance root cause analysis. While powerful clustering methods such as Self Organized Maps (SOM) exist, there is practically no literature showing the applicability of these methods of OAM datasets with a high number of attributes (>20) collected from live network deployments. Moreover, the applicability of the clustering methods does not come free of open questions since, for instance, when using SOM there is no explicitly obtained information about clusters after the SOM training in the underlying data, so the *k*-means technique for grouping SOM units has to be applied afterward. In this context, this paper describes a methodology to cluster radio cells based on a combination of SOM and K-means methods. The methodology is applied to extract cell patterns of the characterization of the long-term behavior (15 days' observation period) and short-term behavior (hourly observation periods) of mobile cells. OAM datasets collected from a live 4G/LTE network deployed in a major European city are used in the analysis.

INDEX TERMS Self-organizing maps, *k*-means, mobile access network, OAM, cluster, long-term behavior, short-term behavior.

I. INTRODUCTION

Tightly integrated with Artificial Intelligence (AI) technologies, the exploitation of data analytics is anticipated to being a game-changer for network operators at all levels, ranging from top business, service, and network management levels (e.g. customers care management, service fault management, network performance management) down to the level of driving the operation of specific functions embedded within the network nodes (e.g. traffic routing and Quality of Service [QoS] parameter selection based on network data analytics), for instance, how to implement data transactions among mobile users in customers care management field is a hot research issue [1]. In particular, the Radio Access Network (RAN) is a data-rich environment where data is continuously gathered in the form of radio measurements or other system indicators (e.g., performance indicators, alarm conditions). However, data cached Mobile Network Operators (MNOs) associated with spatial-time in RAN is too complex and huge to explore and analyze. Therefore, the rise of data mining and analysis enables MNOs to effectively monitor cell performance and manage network resource allocation is emerging recently [2]. Among many data analysis methods, clustering as a simple and effective analysis method has attracted the attention of many researchers. It is one of the standard methods of dealing with a large dataset in many areas as well, such as industry, agriculture, and economic system [3], etc.

The associate editor coordinating the review of this manuscript and approving it for publication was Tariq Umer.

Clustering is the method used to group data onto sets having similar characteristics. It could be applied to observe similar patterns of the data. Well-formed clusters are those, who are properly segregated and represent an order. Labeled data is easier to cluster as a penalty and a reward system can be put into place to facilitate the efficient clustering of the data [4]. However, it is difficult to cluster the unlabeled data, since, there is no specific standard against which the clustering can be tested and the data is large enough to be properly clustered by human intervention. The requirement for clustering methods was researched a long time ago and some different clustering algorithms (i.e., k-means and SOM) have been developed. However, lately, there has been an increase in the use of these two concepts to properly identify clusters. At the same time, these two algorithms have shown lots of advantages and disadvantages in the clustering part in data analysis researching as well.

In this paper, we use the SOM-K algorithm to analyze RAN datasets, which consist of temporary reference values for numeric attributes with Matlab. First, we implement the SOM algorithm, and the input data is clustered into the SOM for training. Since SOM simulation is complex and time-consuming, thus nearly accurate clustering results are possible and required in the initial assembly, the number of iterations can be greatly reduced (e.g., can be set to 350 times), and there is no need to wait for the algorithm to converge completely. Second, after training is completed, the network makes each node of the output layer become a neuron, which is sensitive to a certain pattern class through the method of self-organization, and the corresponding internal weight vector of each node becomes the central vector of each input pattern class. This center vector can be used as a primary center vector in the k method algorithm for performing accurate secondary aggregation.

The rest of the paper is organized as follows. Section II provides some background on SOM and K-means clustering techniques along with an overview of the related work and the novelties of this work. Section III presents the OAM dataset used in the analysis. The clustering methodology based on a SOM-K model is described in Section IV and Section V provides the simulation results and discussion. Finally, Section VI draws the conclusions.

II. BACKGROUND

A. SOM AND K-MEANS FOR CLUSTERING

SOM is an unsupervised learning neural network model that can be used for clustering, high-dimensional reduction, and visualization [5], [6]. It is a readily explainable, simple, and highly visual automatic data-analysis method [7], [8], which is widely applied to clustering problems and data exploration in pattern recognition, data compression, and mining [9], [10].

SOM clustering is different from other artificial neural networks because they apply competitive learning instead of error-correction learning, and in a sense, they use a neighborhood function to preserve the topological properties of the input space [11], [12]. The biggest advantage of the SOM algorithm is that it can map data from high-dimensional space to low-dimensional space. Besides, the SOM algorithm can automatically classify data based on the similarity between the datasets and reduce noise [13]. However, the most serious defect of the SOM algorithm is that it cannot provide accurate clustering information after clustering and the SOM clustering has high complexity and slow learning speed [14]. Compared to the SOM, k-means clustering is a method that divides the datasets into different categories through the iterative process based on the similarity in the sample, which makes the sample difference in internal clusters smaller, while the sample difference in different clusters is greater [15], and it is a dynamic clustering algorithm suitable for the small and medium-sized data clustering. However, there are inherent short-comings 1) The k-means algorithm requires k to be given in advance, but the value of k is usually difficult to determine; 2) The k-means is sensitive to noise and outliers and has a poor performance in high-dimensional data clustering [16]. Balancing the pros and cons of these two algorithms, we propose a two-stage clustering algorithm that combines the SOM and k-means and applied it to cell clustering in RAN. Also, the comparison of SOM-K clustering and k-means clustering results are shown in the appendix.

B. RELATED WORK

SOM and k-means have been used in the telecommunication field for monitoring the communication system as well. For example, the authors of [17] proposed a SOM-ward's clustering method, and [18] proposed a SOM + K-medoids clustering method with low-dimensional label datasets for LTE network anomaly detection as well. These two anomaly detection systems are verified effectiveness by analyzing their performance and comparing them with the reference mechanism. Unlike authors of [17] using SOM algorithms, the authors in [19] presented a data-driven method that uses feature selection and k-means to group LTE cells into clusters with common uplink behavior and the result shows that the uplink speed of 4G networks has increased by 7%. In particular, authors in [20] using k-means clustering to detect any abnormal behavior with label datasets from the LTE core network, and improve network performance by adding other resources. At the same time, a clustering algorithm based on k-means is proposed to group users to define spatial beams to evaluate the capability of the proposed automation solution at the UMTS cellular network [21], although the method successfully grouped UEs with labeled datasets, the class deviation was higher and the calculation time was longer, which took more than half an hour (i.e. the calculation time of SOM-K is about ten minutes [22]). This is because it will cost much time to choose a suitable k if input data is from an unknown probability distribution [23]. Simultaneously, the work of [7] is very similar to our work, but they extracted the behavioral pattern of the 3G network based on low-dimensional datasets (i.e., they only use 2-dimensional

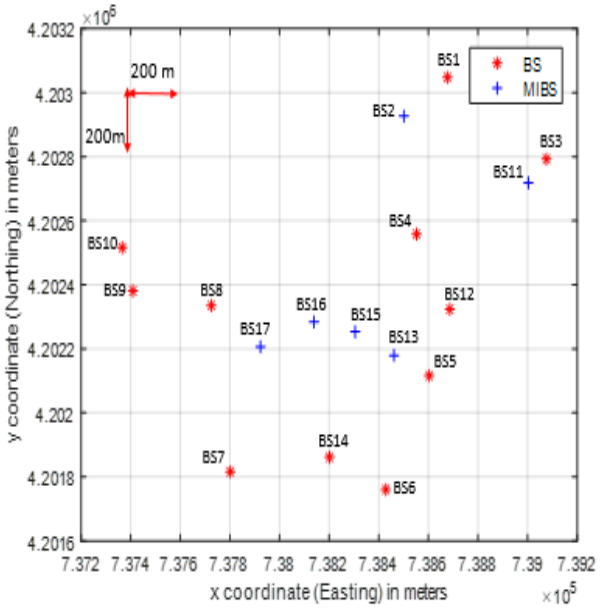


FIGURE 1. Base stations distribution of LTE.

datasets). While compared with previous work, the novelty and contributions of this paper are summarized as follows:

1. The SOM-K algorithm is elaborated and described in detail. At the same time, the SOM-K algorithm for a variety of scenarios is validated by comparing with the k-means algorithm, including different time-domain environments (i.e., long- and short-term), which is crucial to describing the applicability of the algorithm.

2. Compared with [7] and [17] using simple and low dimensional feature datasets, we use more complex and higher dimensional datasets at RAN (i.e., 29-dimensional datasets) to analyze the performance of LTE cells.

III. RADIO PERFORMANCE DATASET

The analyzed data set consists of OAM performance measurements extracted from an UMTS/LTE network deployed in a major European city. The measurements were collected for 15 days, from 12th September to 26th September in 2017, and include data of 63 LTE cells operated in 11 macro-cell base stations (BS) and 6 micro-cell BSs covering an area of about 3.2 km2. The geographical distribution of the BSs is illustrated in Fig. 1 (The coordinates are dislocated from the real ones, but relative distances are kept). Red points represent the macro-cell BSs and blue crosses represent the micro-cell BSs. All the LTE cells are operated in the macro-cell BSs except one LTE cell provided at micro-cell BS2. The most common configuration of the macro-cell BSs consists of 3 sectors with 2 LTE carriers of 20 MHz plus 1 LTE carrier of 10 MHz (i.e., 9 LTE cells in total). For example, this configuration is used in macro-cells BS1, BS6, BS7, and BS8. Other supported configurations include 6 LTE cells distributed among 3 sectors (used in BS3 and BS10), 4 LTE cells in 2 or 3 sectors (used in BS9 and BS12), 3 LTE cells in 3 sectors (used in BS4 and BS5), and 1 LTE cell in a single sector (used in micro-cell BS2). In all cases, channel bandwidths of 20 and 10 MHz are used.

TABLE 1. OAM performance measurement collected per cell.

Category	Features		Values (Max/Mean/Min)
UEs	Average #*UEs* in UL		48.2/1.3/0
	Average #*UEs* in DL		100.6/1.5/0
	Max #*UEs* in UL		79.6 /4.98/0
	Max #*UEs* in DL		213/5.9/0
	Total Max #*UEs* in eNB		1388/712/3
	Total Average #*UEs* in eNB		1186/606/0
	Carrier Aggregation capable UEs (%)		45/7.45/ 0
Data volume	Data Traffic (MB)– Hourly data traffic in UL		2307/ 19/0
	Data Traffic (MB)– Hourly data traffic in DL		4431/208.3/0
Throughput	Max Cell Throughput (Mbps) in UL		48.2/2.87/0
	Max Cell Throughput (Mbps) in DL		232.8/ 31.2 /0
	Mean Cell Throughput (Mbps) in UL		26/ 0.7/ 0
	Mean Cell Throughput (Mbps) in DL		87/12.7/ 0
Physical Resource Block (PRB) Utilization	Average PRB Usage per Time Transmission Interval (TTI) (%) in UL		95.8/4.6/ 0
	Average PRB Usage per TTI (%) in DL		97.9/7.6/ 0
Handover (HO) Failure Indicators	Intra eNB HO Failure Rate (%)		66.7 /0.36 / 0
	Inter eNB HO over X2 Failure Rate (%)		25/ 0.17/ 0
	Inter eNB HO over S1 Failure Rate (%)		2.8/0.01/ 0
Radio Resource Control / Data Radio Bearer (RRC(DRB) Failure Indicators	RRC Drop Ratio (%)		15.9/ 0.6/0
	DRB Setup Failure Rate (%)		44.4/ 0.72/ 0
	RRC Setup Failure Rate (%)		8.97/0.41/ 0
Channel Quality	Average CQI		15/9.83/ 0
	Average Physical Uplink Share Channel (PUSCH) SINR		34/ 13 /-9
	Average Physical Uplink Control Channel (PUCCH) SINR		28/ 7 /-10
	MCS Distribution (%)	Low (MCS0-9)	15.5/7.84/0
		Medium (MCS10-19)	32.1/16.7/0
		High (MCS20-28)	52.4/31.6/0
Circuit Switched fallback (CSFB) attempts	CSFB attempts in idle mode		1421/ 24.7/ 0
	CSFB attempts in connected mode		898/10.7/ 0
Latency	Intra eNB Latency in DL (ms)		526/0.87/ 0
	Intra eNB Latency in UL (ms)		1381/22.9/ 0

A total of 29 types of performance measurements are collected per cell. These are illustrated in Table 1, which is grouped into 9 categories. The measurements are sampled for periods of 15 minutes, except for the average/max UEs per eNB, CQI (Channel Quality Indicator), and SINR PUCCH (Physical Uplink Share Channel)/PUSCH (Physical Uplink Control Channel) average measurements, which are available for periods of one hour. This results in a total of 1404 samples per cell (14 days and 96 samples/day, and 60 samples/day in the last day) and a total of 85977 samples of the aggregate of cells (i.e., we have 63 cells and each cell have 1404 samples except for cell U, V, W of BS 10 are only kept one-week data, which is only 579 samples per cell), with each sample containing 29 different features. Global statistics (max, mean, and min values) of the collected performance measurements are provided with Table 1 to illustrate their range of variation.

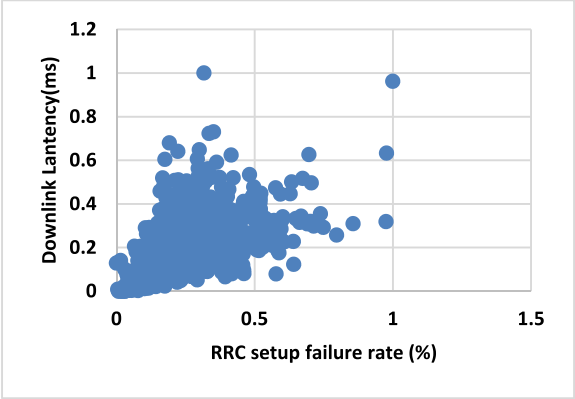


FIGURE 2. Scatter plot of RRC setup failure rate and downlink latency.

Fig. 2 illustrates the scatter plot of the RRC setup failure rate and downlink latency within 1404 samples of cell #7. RRC setup failure rate is the ratio of the number of failed RRC connection establishment and the total number of RRC connection establishment. At the same time, latency is the time takes to get a packet of a specific point. The latency time is generally the sum of response delays and transmission delay and it is also measured in ms. The reason why we choose these two features' scatters plots is that the correlation between them is very strong, and their correlation coefficient value is 0.613. This means they are highly correlated. It could better characterize the correlation between these two features and the specific description of the correlation coefficient will be discussed in the next paragraph. At the same time, Fig. 3 shows the time evolution of downlink maximum throughput in the whole 15 days at three different LTE cells, which are all located in BS6. And the average downlink throughput is different from these three cells. For instance, cell#3 has the highest average downlink throughput, while cell#1 has the lowest, and the average values of these two cells are 23.3 and 5.23 Mbps, respectively. Cell#2 has a medium performance among these three cells and the value of average throughput is 16.1 Mbps. Simultaneously, we can find that the throughput is fluctuating over time of each cell, that is because the usage of data traffic is different from time to time between different users.

Moreover, a correlation matrix heat map of the 29 features in different cells is shown in Fig. 4. This was obtained by using Pearson's correlation coefficient [24] for each feature pair, which studies the 1404 samples of the whole 15 days per LTE cell. As the heat map differs in the number of cells, so we calculate the average values of the correlation coefficients achieved with all cells to represent the heat map of the entire cells. Green and red represent positive and negative correlations between different features, respectively. For instance, the upper left corner of Fig. 4 is roughly green, which means that the features of this area are positively correlated. On the contrary, the features of the red region in the middle of Fig. 4 show the meaning of negative relationships.

IV. CLUSTERING METHODOLOGY USING SOM-K

A. SOM TOPOLOGY AND PARAMETERS

The topology of the SOM neural network is illustrated in Fig. 5. It consists of two layers, namely, the input layer and the competition layer. Both layers are made up of several neurons, also called nodes or units. The working principle of SOM is projecting a collection or sequence of data input items from the input layer's neurons into the competition layer's neurons [25]. The data input items are typically represented as a set of I vectors $Xi = [x_{i1}, x_{i2}, \ldots, x_{in}]^T$ where $i = 1 \ldots, I$, and n is the dimension of the input space given by the number of features of the individual input data items. On the other hand, the competition layer of the SOM model is a rectangular or hexagonal grid of P neurons, each associated with a weight vector $W_j = [w_{j1}, w_{j2}, \ldots, w_{jn}]^T$, where $j = 1 \ldots, P$. The proper value of P is one of the parameters to be determined when building the SOM model. The neurons in the competition layer are connected to the adjacent ones with a neighborhood relation. Each neuron, except the ones on the border of the map, has four or six direct neighbors, depending on choosing a rectangular or hexagonal grid structure, respectively. For the determination of the weight vectors, a training procedure is used, which typically is implemented using the batch computation algorithm presented in [10]. It is based on a competition approach by which input vectors are compared with weight vectors to select the winning neurons, referred to as the Best Matching Units (BMU), based on the least distance criterion such as the Euclidean distance. During this process, when an input sample is fed to the SOM model, its Euclidean distance to all weight vectors is computed and the weights of the BMU and the other neurons close to it in the SOM grid are adjusted to the input vector [22]. The magnitude of the change decreases with many iterations and with the grid-distance from the BMU [12]. The SOM is an unsupervised algorithm that follows an iterative process until the network converges. In particular, in a given t-th iteration, the basic process of SOM can be described as follows:

1. Randomly select a new sample of the input dataset [7]: $X(t) = X_i$ with i have randomly chosen between 1 and I.
2. Find the BMU of $X(t)$: search for a neuron $q \in 1, \ldots, P$ in the competition layer. A BMU is calculated for each input sample of the training data by finding the neuron which has the smallest distance between the sample. The BMU and all of its neighboring neurons, assigned through the topology and neighborhood radius, are shifted towards the input sample. Both the size of the neighborhood and the strength of the shift will decrease over-time to help with convergence.

$$min_j \left\{ \left\| W_j - X(t) \right\| \right\} = \left\| W_q - X(t) \right\| \quad (1)$$

where $\|.\|$ is the distance measurement and we use the Euclidean distance in this work.

3. Update the weights of the BMU and their neighbors to reduce the distance between the input sample $X(t)$

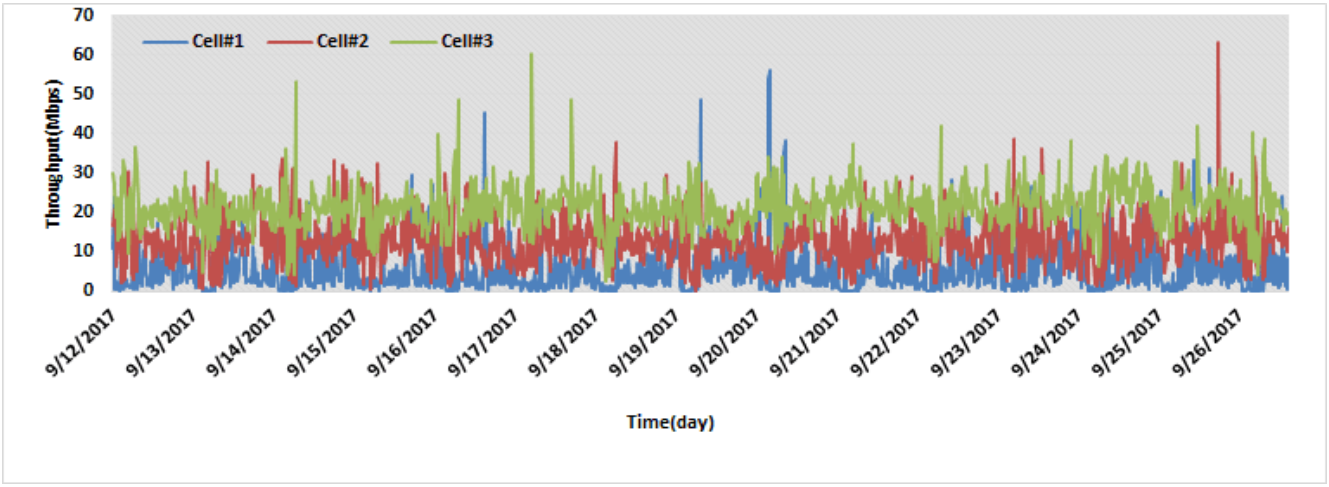


FIGURE 3. The time evolution of downlink throughput in the whole period among different cells.

FIGURE 4. Heatmap of correlation coefficient of different features.

according to the following rule [10]:

$$W_j(\mathrm{t}+1) = W_j(\mathrm{t}) + \rho(\mathrm{t})^* h_{ci}(\mathrm{t})[X(\mathrm{t}) - W_j(\mathrm{t})] \quad (2)$$

where $W_j(\mathrm{t}+1)$ is the updated weight vector, $\rho(\mathrm{t})$ is the learning rate at t-th iteration, which is usually a monotonically decreasing function of the number of iterations and its range is from 0 to 1, and $h_{ci}(\mathrm{t})$ is the neighborhood function and usually denote as a Gaussian function, which is centered around the BMU.

$$h_{ci}(t) = \exp(-\frac{\|r_i - r_q\|^2}{2\sigma^2}) \quad (3)$$

where r_q depicts the coordinates of the winner unit, and r_i denotes the coordinates of an arbitrary unit on the competition

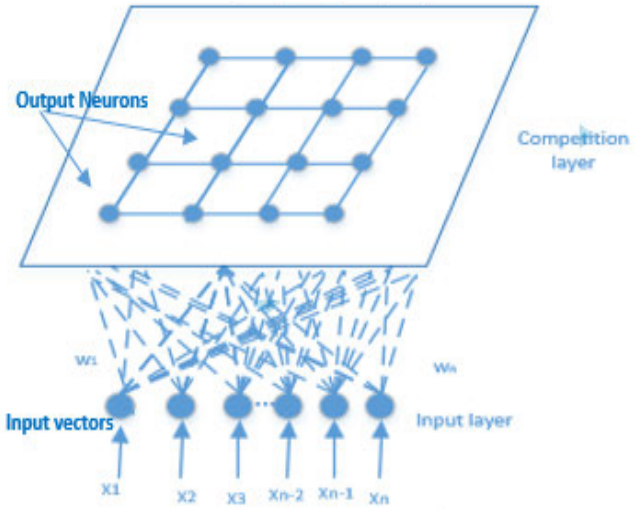


FIGURE 5. Network topology of SOM.

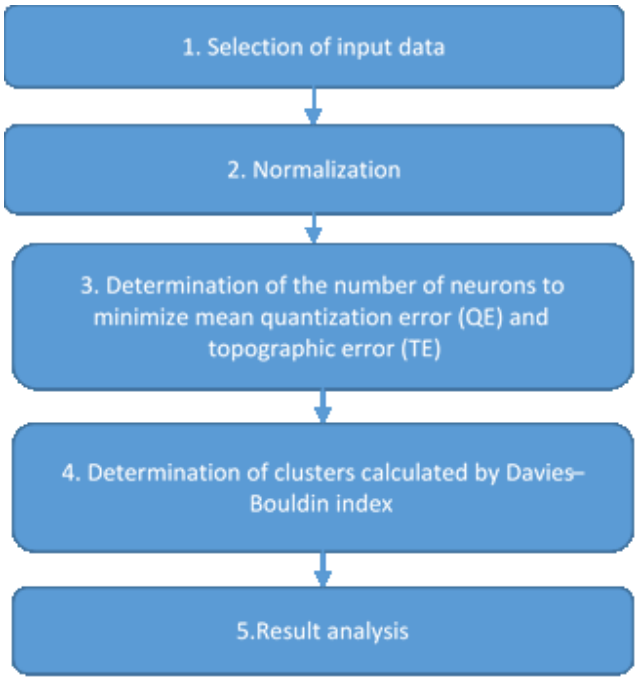


FIGURE 6. SOM-based clustering methodology.

layer lattice of the map and is the width of the neighborhood. It is necessary that $h_{ci}(t)$ close to 0 when t closes to ∞ for the algorithm to converge. During learning, the learning rate and the width of the neighborhood function are decreased, typically to a linear fashion (to know more detailed neighborhood functions see Ref. [8]).

4. The three above steps are repeated until the learning rate is no longer changing (i.e., network convergence).

B. CLUSTERING METHODOLOGY

The methodology used for cell clustering by using the SOM model is illustrated as a flow chart in Fig. 6. In our case, the dimension of the input space is n = 29 and the maximum number of data items are I = 85977, as previously discussed in section III.

Before training the SOM model, input data have to be normalized. Otherwise, directly using raw data as input data, might cause huge deviations because of the range of numerical values taken but each of the features differs greatly. Therefore, standardization is applied using the following formula [26]:

$$O_{ij} = \frac{x_{ij} - \min(x_j)}{\max(x_j) - \min(x_j)} \tag{4}$$

where x_{ij} is the raw value of feature j in data item i, $\min(x_j)$ and $\max(x_j)$ are the minimum and maximum data values of feature j in whole data items respectively. O_{ij} is the normalized value and the result of normalization is in the range of [0,1].

The next step after the normalization of the input data is the determination of the most appropriate number of neurons P to be used in the competition layer of the SOM model. This is solved in our case by relying on the computation of the mean quantization error (QE) and topographic error (TE), which are the main measurements to assess the quality of the SOM model [27]. In particular, QE is computed as the average distance between input data vectors and their BMUs [28] and TE represents the percentage of data vectors for which the BMU and the second winning neuron are not adjacent [29]. In general terms, the smaller the QE and TE, the better the operation result and performance of the SOM model [8], [30]. However, when deciding the number of neurons, the correlation effects between QE and TE have to be accounted [31], [32]. For topographic error, the smaller the map size, the lower error. However, when the map size is bigger, the topographic error is the highest. The reason is that SOM can simulate the topology structure, and the topology structure will change based on the map size. When the map size is small, the structure of the topology is simple and the error is small. When the map size is large, the structure becomes complex, and the error rate increases more than the increased rate of the map size. As the map size becomes larger, the rate of topographic error less than the rate of map size. The topographic error decreases naturally. Typically, QE is considered as the dominant parameter for choosing the map size [27] when they exhibit similar values of TE. Formally, QE and TE are computed as follows:

$$\mathrm{QE} = \frac{1}{R}\sum_{t=1}^{R} \left\| X\left(t\right) - W_{\mathrm{q}}(t) \right\| \tag{5}$$

$$\mathrm{TE} = \frac{1}{R}\sum_{t=1}^{R} d\left(X\left(t\right)\right) \tag{6}$$

where $X\ (t)$ is the input data item at the t-th iteration; $W_{\mathrm{q}}\ (t)$ is the BMU's weight vector of the sample $X\ (t)$; $d\ (X\ (t)) = 1$, if the first BMU and the second BMU of $X\ (t)$ are not adjacent and $d(X(t)) = 0$ otherwise, and R is the number of iterations until network convergence. Based on the above, in this step, we train the SOM model using the Matlab SOM toolbox

2.0 with a different number of neurons ranging from 4 (i.e., 2×2 grid) up to 100 neurons (i.e., 10×10 grid). For each configuration, we compute the value of QE and TE by using the *som_quality* (sMap) package and among all the trained models, the one with the lowest QE and TE will be selected. Let us denote P_{opt} to the size of the SOM model that delivers the best performance in terms of TE/QE.

In the next step, the k-means algorithm is used to cluster the P_{opt} weight vectors obtained by SOM into a smaller number of vectors $Q \leq P_{opt}$ so that overall clustering accuracy is improved [33]. The obtained Q vectors are called centroids or cluster centers. In particular, the k-means calculation is performed by selecting the optimal clustering number Q that minimizes the Davies & Bouldin index (DBI), which represents the ratio of the sum of centroid intra-cluster distances and inter-cluster distances [34]. The combined SOM-k clustering algorithm, which is named quadratic clustering, can maintain the self-organizing characteristics of the SOM network and the high efficiency of the k-means algorithm, and the small selection range of the initial clustering center value of the k-means algorithm. The DBI for Q clusters is defined by the following expression:

$$v_{DB}(Q) = \frac{1}{Q}\sum_{m=1}^{Q} \max_{l \neq m} \left\{ \frac{S_c(Q_m) + S_c(Q_l)}{d_{ce}(Q_m, Q_l)} \right\} \quad (7)$$

- Q_m and Q_l represent the m-th and l-th clusters within the set of Q clusters.
- $Sc(Q_m) = \frac{\sum_i \|X_i - C_{Q_m}\|}{N_{Q_m}}$ is the computation of the centroid intra-cluster distance. $S_c(Q_m)$ measures the average of all pair-wise distances from samples of the cluster to the cluster centroid [35]. X_i is an n-dimensional feature vector assigned to the cluster Q_m. C_{Q_m} is the centroid of the cluster Q_m [36].
- $d_{ce}(Q_m, Q_l)$ is the so-called inter-cluster distance between two generic clusters Q_m and Q_l. At the same time, it is computed as the distance between their centroids.

The DBI is computed for several values of Q and the one that minimizes $v_{DB}(Q)$ is taken as the optimal value [37], denoted in the following as Q^* [38], [39].

Finally, the characters of different clusters are analyzed by comparing the distribution of different features of each cluster.

V. USE CASE

A. LONG-TERM BEHAVIOR CELL PATTERNS

We try to obtain the cell patterns based on the long-term behavior (long-term behavior is the performance of cell patterns throughout the whole 15 days) of the cells as observed during the entire measurement collection period. According to average values of the entire measurement period are first computed for all the cells, the input data onto 63-row vectors (one per cell) with 29-column vectors of each cell (one per feature) was selected. We represent input data as a 63×29 matrix. In the simulation work, we have tried many group iterations as well (e.g., from 50 to 1000) and found when iteration times are 350, the accuracy of simulation will be stable and reached 0.92).

TABLE 2. Value of QE and TE.

Map size	QE	TE
2*2	1.66	0
3*3	0.79	0
4*4	0.95	0
5*5	0.76	0
6*6	0.296	0
7*7	0.46	0
8*8	0.43	0
9*9	0.4	0
10*10	0.41	0.5

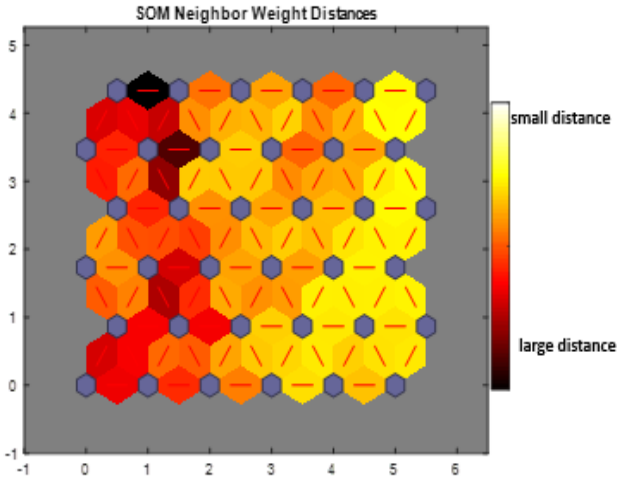


FIGURE 7. SOM neighbor weight distance.

To determine the map size, the value of QE and TE are computed for configurations ranging from 2×2 neurons up to 10×10 neurons. Obtained values are given in Table 2, as we can see, QE and TE get the minimum when the configuration reaches 6×6 (QE $= 0.296$ and TE $= 0$).

The weight vectors associated with each neuron move to become the center of a cluster of input vectors. Besides, neurons that are adjacent to each other in the topology should also move close to each other in the input space, therefore, it is possible to visualize a high-dimensional input space in the two dimensions of the network topology. One visualization tool of the SOM is the weight distance matrix (also called the U-matrix) is shown in Fig. 7, the blue hexagons represent the neurons. The red lines connect neighboring neurons. The colors in the regions containing the red lines indicate the distances between neurons. The U-matrix is a matrix whose elements represent the mean distance of each map unit from its neighboring units, so relative maximums of this distance matrix indicate cluster borders, whereas low values indicate the clusters themselves [25].

The SOM sample hits are shown in Fig. 8: each neuron shows the number of input vectors for its classification. The relative number of vectors for each neuron are shown by the

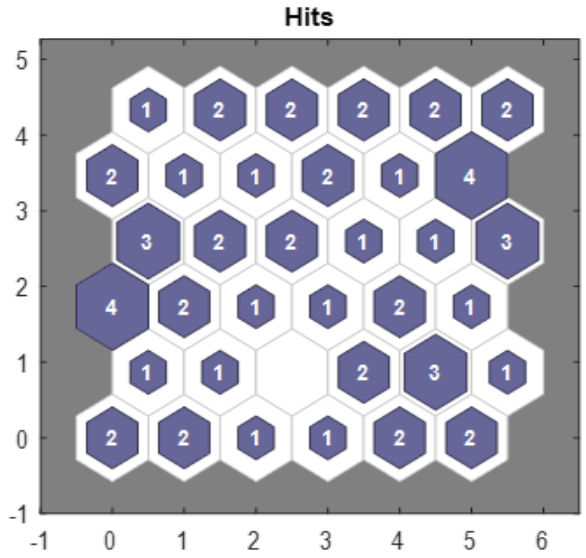


FIGURE 8. SOM hits of each neuron.

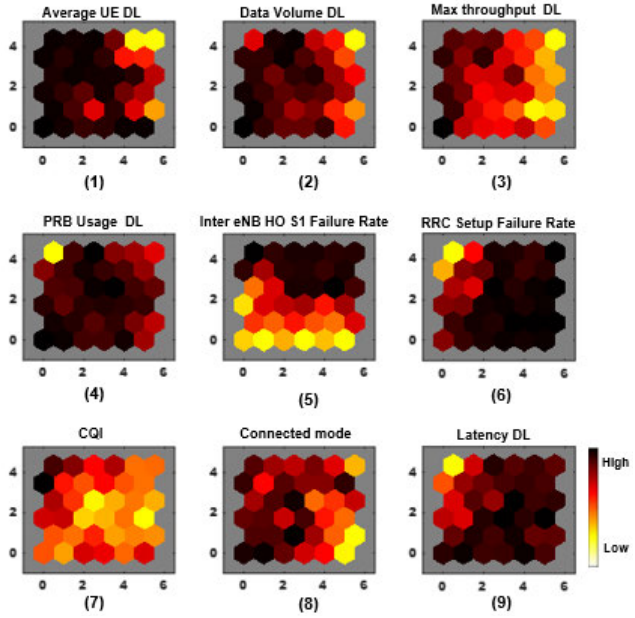


FIGURE 9. Weight planes of different features.

size of the colored patches and the default topology of the SOM is hexagonal. This figure shows the neuron locations in the topology and indicates how many of the training data are associated with each of the neurons. The topology is a 6-by-6 grid, so there are 36 neurons. The initial number cluster of SOM is 35 and the maximum number of hits associated with any neuron are 4.

To divide input vectors into different clusters, we are using SOM weight planes to visualize the SOM topology; The weight planes in 9 different features selected from 9 different categories in Table 1 are shown in Fig. 9, and they are also representing the correlations between different features.

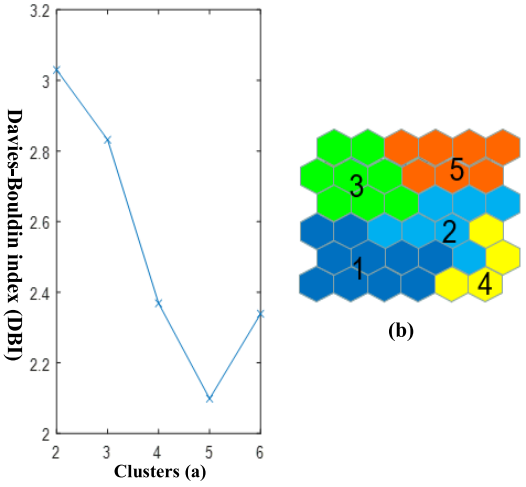


FIGURE 10. The minimum value of DBI and responding clusters.

At the same time, they are visualizations of the weights that connect each input to each of the neurons (the darker colors higher weights, the lighter colors the lower weights). If the connection patterns (e.g., color changing and distribution are similar) of the two inputs were very similar, you can assume that the inputs are highly correlated. For instance, Fig. 9(1) and (2), (3) and (4) almost have similar color changes and distributions, therefore, we can determine the Average UE DL (downlink) and Data volume DL, max throughput DL and PRB Usage DL have a positive correlation. Fig. 9(6) and (9) almost have a similar color distribution, which means that the latency DL and RRC setup failure rate have a strong correlation. However, Fig. 9(5) has a different color changing and distribution with the other 8 features, therefore, we determine Inter eNB HO S1 Failure Rate has a weak correlation between the other 8 features. On the other hand, Fig. 9(6) and (7), (7) and (9) have an opposite color distribution in each neuron, which means that these two groups' features have negative correlation characteristics, respectively.

For the specific clustering, we can find from Fig. 10 (a), which shows the minimum value of DBI are 2.1 and its corresponding cluster volume is 5. As shown in Fig. 10(b), we use different colors to represent different clusters, and the characteristics of specific cells in each cluster are shown in Fig. 11 as well.

Fig. 11 illustrates the distribution of the feature of different clusters of long-term behavior cell patterns and Table 3 shows that we successfully clustered 63 LTE cells, and the specific characteristics of each cluster are also described in this Table. For instance, from Fig. 11, we could find that cluster#5 has the best performance, while cluster#2 has the worst performance among the whole 5 clusters. That is because #5 has the most UEs, the best CQI, and connected mode, but #2's performance is the worst in the whole 5 clusters at these three features. Unlike these two clusters, cluster#1, #3, and #4's performance is medium. For instance, cluster#1 and cluster#3 have similar characteristics of data volume, max

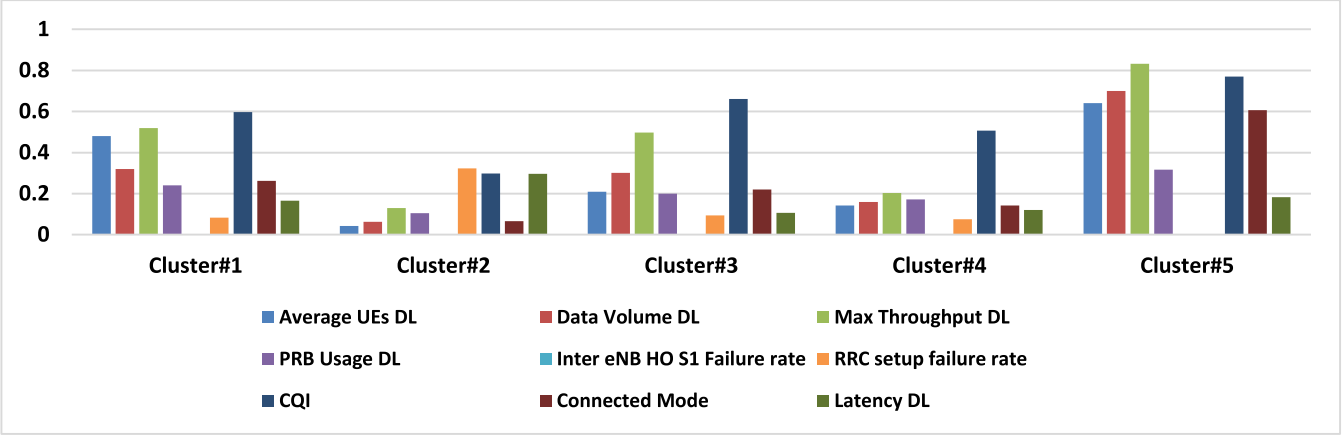


FIGURE 11. Features distribution of different clusters under long-term behavior cell patterns.

TABLE 3. Number of cells in different clusters by using SOM-K.

Clusters	Number of cells	Defining characteristics
1	16(60,62,56,21,38 ,35,37,57,18,5,6,2 5,44,43,48,49)	Medium: UEs, Data volume, Throughput, RRC setup failure rate, PRB usage, CQI, Connected mode and Latency Low: Inter eNB HO S1 failure rate
2	12(61,54,63,29,47 ,51, 33,14,41,42,17,1)	High: RRC setup failure rate and Latency. Low: UEs, Data volume, Throughput, PRB usage, CQI Connected mode, and Inter eNB HO S1 failure rate
3	14(32,34,46,9, 15,7,8,19,58,59,1 0,50,4,11)	Medium: UEs, CQI, Data volume, Throughput, RRC setup failure rate, Connected mode, and PRB usage. Low: Latency and Inter eNB HO S1 failure rate
4	6(20,55,3,2,23, 24)	Medium: Connected mode and CQI Low: UEs, Data volume, Throughput, Inter eNB HO S1/RRC setup failure rate, PRB usage, Latency.
5	15(39,16,30,31, 36,45,12,27,26, 22,28,13,52,53, 40)	High: UEs, Data volume, Throughput, PRB usage, CQI, and Connected mode Medium: Latency Low: Inter eNB HO S1/RRC setup failure rate.

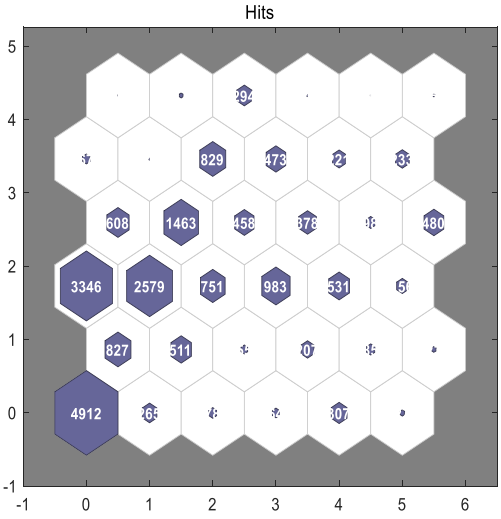


FIGURE 12. SOM hits of each neuron.

throughput, and PRB usage DL are very close between these two clusters. However, since the UEs in #3 are less than #1, the latency of cluster #1 is higher than #3. At the same time, although the number of UEs in #4 is less than #3, the reason why RRC set up failure rate and latency of #4 is close to #3 is the CQI of #4 is lower than #3.

B. SHORT-TERM (DAILY) BEHAVIOR CELL PATTERNS

We tried to obtain cell patterns based on the short-term behavior (short-term behavior is hourly cell performance among different cells in the first stage and analysis of the percentage of the time distribution of the cells in the second stage) of the entire cell observed during the entire measurement collection period. To differentiate long-term behavior, which captures the behavior of the cell over a 24h period (1 day). In the short-term behavior analysis, we are expected to do a similar clustering analysis but now taking as input samples cell performance indicators that capture the behavior of the cell over 1h period and the whole 15 days are divided into 350 hours (due to last day we only have 14 hours' data) per cell. Cell U, V, W of BS 10 are only kept one-week data as well. Therefore, we obtained 21341 groups of data in hours by preprocessing the data of 63 cells. According to average values of the entire measurement period are first computed for all the cells, the input data onto 21341-row vectors of 29-dimensions (one per feature) was selected, which input data is represented as a 21341×29 matrix.

For the specific clustering, to determine the map size, the value of QE and TE are computed for configurations ranging from 2×2 neurons up to 10×10 neurons. QE and TE reach the minimum when the topology is 6×6 (QE $=$ 0.026 and TE $=$ 0.047).

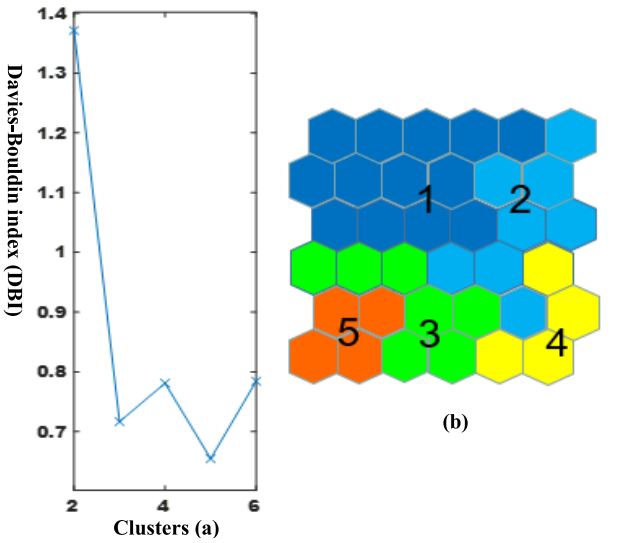


FIGURE 13. The minimum value of DBI and responding cluster.

The SOM sample hits are shown in Fig. 12. The initial number cluster of SOM is 36 and the maximum number of hits associated with any neuron are 4912, which means there are 4912 samples of this SOM unit. Unlike long-term is clustering 63 cells based on 15 days and each cluster is a group of cells. Short-term behaviors are clusters of states in different cells per hour, not clusters in cells. At the same time, the data is more complicated in the short-term, and the features are 29 as well. Therefore, compared with the long-term, the distribution of hits of different clusters of the short-term is more uneven.

We can find from Fig. 13(a), which shows the minimum value of DBI are 0.65 and its corresponding cluster volume is 5. We use different colors to represent different clusters are shown in Fig. 13(b). Therefore, the number cluster of short-term behavior is 5 and the specific clusters of short-term behavior are shown in Fig. 14.

From Fig. 14 and Table 4, we can find the behavior of hours of cluster #4 has the best performance among these 5 clusters. We can see that all features' performance in #4 is far above the other clusters, such as #4 has the best UEs, CQI, and connected mode (i.e., UEs proceed with a connected mode when they completed the RRC connection establishment) among these 5 clusters. Simultaneously, the sample time of cluster #4 is almost concentrated on the rush hour of 9 am and 3 pm, which is mainly working time and the data usage is a peak period. Since the values of all features are the lowest, the performance of cluster#1 is the worst. Because the period of it is almost from 2 am to 6 am, there is a little person need to use mobile phones to communicate or surfing the internet. Focus on clusters #2 and #3, cluster#2 has a lower RRC setup failure rate and higher connected mode. The other features of these two clusters are similar. That is because, in the case of a similar number of UEs, the value of CQI of #3 is smaller than#2. This means that compared with # 2, the channel quality of #3 is worse. And the sample time of cluster#3 is almost from 6 am to 8 am, and 10 am to 2 pm, which the medium period of data usage. But the sample time of cluster#2 almost comes from 4 pm to 6 pm, which just happens before getting off work and this is a period of sub-peak data usage. Finally, cluster#5 has a sub-worse performance among these 5 clusters and it is only better than #1, for instance, the RRC setup failure rate of cluster #5 is higher than #3 and the connected mode is lower. The reason for this phenomenon is that under the premise that the number of UEs is similar to #3, the CQI and connected mode during this period are weaker than #3. The sample time of this cluster is almost from 7 pm to 1 am, which is the period of getting off work.

TABLE 4. Number of cells in different clusters by using SOM-K.

Clusters	Number of different hours in whole cells	Defining characteristics
1	4645	Low: UEs, Data volume, throughput, Inter eNB HO S1/RRC setup failure rate, PRB usage, CQI, Connected mode, and Latency.
2	2841	Medium: Connected mode, UEs, Data volume, Throughput, PRB usage, CQI, and Latency Low: Inter eNB HO S1 /RRC setup failure rate.
3	6676	High: RRC setup failure rate Medium: Data volume, Throughput, Connected mode, CQI, PRB usage, and Latency Low: UEs, RRC, Inter eNB HO S1 failure rate,
4	655	High: UEs, Data volume, Throughput, PRB usage, CQI, Connected mode, and Latency. Medium: RRC setup failure rate Low: Inter eNB HO S1 failure rate
5	6524	High: RRC setup failure rate Medium: Latency Low: UEs, Data volume, Throughput, CQI, PRB usage, Inter eNB HO S1 failure rate, and Connected mode

TABLE 5. Distribution of cells in different load clusters at short-term behavior.

Load degree / Cells	Cluster1(%) Low load	Cluster2(%) Sub-high load	Cluster3(%) Medium load	Cluster4(%) High load	Cluster5(%) Sub-low load
I:5,6,12,13,16,21,22,25,26,27,28,29,31,32,35,36,37,38,39,40,43,44,45,48,49,52,53,57,60,62	10.2 - 14.5	14.3 - 15.8	30.8 - 39.1	3.7-5.1	20-26.8
II:2,3,4,7,8,9,10,11,15,18,19,20,23,24,32,34,46,50,58,59	16-19.1	11.7 - 14.5	30.5 - 32.5	1.1-3.5	31.6 - 34.7
III:61,54,63,29,47,51,33,14,41,42,17,1	21.4 - 27.7	5.8-8.1	18.5 - 26.1	0-0.55	30.7 - 37.1

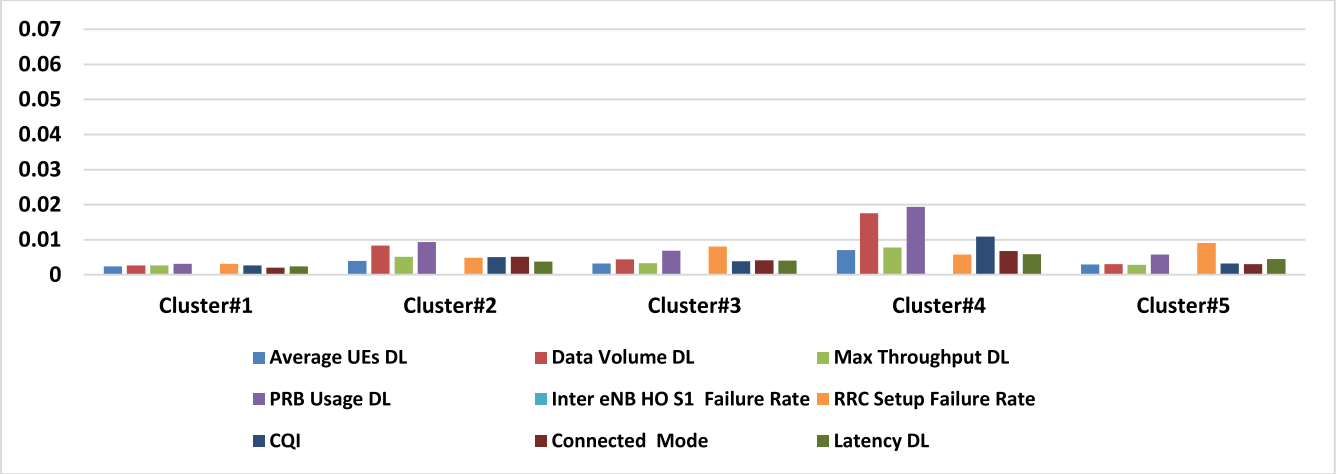


FIGURE 14. Features distribution of different clusters under short-term behavior of hours.

TABLE 6. Distance comparison between *k*-means and SOM-K.

	Intra-cluster centroid distance (*k*-means)	Intra-cluster centroid distance (SOM-K)	Inter-cluster centroid distance (*k*-means)	Inter-cluster centroid distance (SOM-K)
Cluster#1	1.05	0.405	1.01	1.08
Cluster#2	0.63	0.577	0.59	0.88
Cluster#3	0.68	0.655	0.53	0.96
Cluster#4	1	0.505	0.63	0.80
Cluster#5	0.52	0.470	0.71	0.85

TABLE 7. Number of cells in different clusters by using *k*-means.

Clusters	Number of cells	Defining characteristics
1	14	High: UEs, Data volume, Throughput, PRB usage, CQI, and Connected mode Medium: Latency Low: Inter eNB HO/RRC setup failure rate.
2	20	High: RRC setup failure rate and Latency. Medium: PRB usage and CQI Low: UEs, Data volume, Throughput, Inter eNB HO S1 failure rate, and Connected mode
3	12	Medium: UEs, Data volume, Throughput, RRC setup failure rate, PRB usage, CQI, Connected mode, and Latency. Low: Inter eNB HO S1 failure rate
4	9	High: UEs, Data volume, Throughput, CQI Medium: PRB usage and Connected mode Low: Inter eNB HO S1/RRC setup failure rate and Latency
5	8	High: RRC setup failure rate and Latency. Medium: PRB usage and CQI Low: UEs, Data volume, Throughput, Inter eNB HO S1 HO rate, and Connected mode

Table 5 shows the cell distribution of different load degrees in the short-term, the top row is the load and the left column are three types of cells from the whole 63 LTE cells.

TABLE 8. Distance comparison between *k*-means and SOM-K.

	Intra-cluster centroid distance (*k*-means)	Intra-cluster centroid distance (SOM-K)	Inter-cluster centroid distance (*k*-means)	Inter-cluster centroid distance (SOM-K)
Cluster#1	0.002	0.043	0.48	1.37
Cluster#2	0.433	0.038	0.02	0.72
Cluster#3	0.009	0.038	0.80	0.78
Cluster#4	0.049	0.051	0.57	0.65
Cluster#5	1.126	0.080	0.44	0.78

TABLE 9. Number of cells in different clusters by using *k*-means.

Clusters	Number of different hours in whole cells	Defining characteristics
1	5719	Low: UEs, Data volume, Throughput, PRB usage, Inter eNB HO S1/RRC setup failure rate. CQI, Connected mode, and Latency
2	1238	Medium: UEs, Data volume, Throughput, Connected mode, PRB usage, CQI, RRC setup failure rate, and Latency. Low: Inter eNB HO S1 failure rate
3	4224	High: RRC setup failure rate Low: UEs and Data volume, Throughput, Inter eNB HO S1 failure rate, PRB usage, CQI, Connected mode, and Latency.
4	7475	High: RRC setup failure rate, and Latency Low: PRB usage Connected mode UEs, Data volume, throughput, CQI, and Inter eNB HO S1 failure rate
5	2685	High: RRC setup failure rate, latency, PRB usage, CQI UEs, Data volume, Throughput, and Connected mode Low: Inter eNB HO S1 failure rate

Firstly, we can move on the column of high and low load degrees, for instance, let us focus on the low load degree in cells III, the time distribution range is 21.4%-27.7%, which is the highest range of these three types of cells. And the

lowest percentage range is 10.2%-14.5% in cells I. On the other hand, focus on the high load degree column, the highest time percentage range is 3.7%-5.1% in cells I, and the lowest percentage range is 0-0.55% in cells III. This demonstrates that cell I is the best cells' group and cell III is the worst cells' group. No matter in high or low load, cell II's performance is moderate.

Secondly, analyze from sub-high/low load, let us focus on the sub-high load part, we can find that the cell with the highest percentage range is cells I, and their time distributions partially overlap with cells II as well, for instance, the time percentage range of cells I and II are 14.3%-15.8% and 11.7%-14.5%, respectively. The same situation occurs to cells II and III in the sub-low load cluster as well. Meanwhile, the highest time percentage range of the sub-low load is 30.7%-37.1% in cells III and the lowest proportion range is 20%-26.8% in cells I. Lastly, the medium load is the area with the most overlapping distribution and the respective ranges are not much different except for the cells III. For instance, the range 30.8%-39.1% in cells I and 30.5%-32.5% in cells II have much overlap time distribution of this period. The three group cells with overlapped time percentage range of different load degrees can be explained that the user data consumption of different cells in this period is the same.

We can also find that the cells I, II, and III in the short-term have a certain relationship with the five clusters cells in the long-term. For example, cells in III in the short-term are the worst performance cells and it is also the worst cells in long-term (e.g., cluster#2). II and I cells are the cells at sub-optimal and optimal cells in the long-term as well. Therefore, we can conclude that the performance of the cells at different load degrees in the short-term is highly correlated with the previous long-term cell clustering result, and it is a specific performance of the long-term behavior cell patterns in the short-term behavior cell patterns.

VI. CONCLUSION

This article is based on the data mining method and applied it to the research of LTE cell behavior, which is analyzing the characteristics of each cluster and tapping potential high-quality cells according to different key performance indicators (KPIs). We propose a combination method called SOM-K and apply it to high-dimensional data set analysis of LTE cells. SOM-K has provided the results of combining traditional methods and expertise verified according to long- and short-term behaviors of cell patterns. The SOM-K results can be routinely understood, thereby increasing confidence in the new analysis and its applicability in the field of LTE networks. In the course of this work, it is worth noting that compared with the clustering results of high-dimensional data using SOM-K, the clustering results of high-dimensional data using the k-means algorithm alone are not sufficient to provide effective cell pattern analysis.

The analysis based on the SOM-K used in the LTE network has not yet been popularized. Our work demonstrated that SOM-K could be used in LTE cell clustering performance analysis and obtain a better cluster result. At the same time, the advantage of the analysis method based on SOM-K is that the two-step measurement method is used in the analysis to improve the clustering results. Besides, the clustering effect of this method of high-dimensional data is significantly better than the traditional k-means algorithm. All in all, the use of data analysis and mining in LTE network optimization also means that the SOM-K method can now be used to adjust network performance, and the algorithm successfully uses big data to cluster cells with the same characteristic pattern of almost real-time.

In the future, the operation of 5G networks will be mainly driven by services and the trend of the 5G network will further increase the use of data. It is worth noticing that SOM-K detection based on the analysis of abnormal network behavior has been successfully used in GSM data for a long time. Therefore, exploring the use of this algorithm for data usage prediction and network performance management (i.e., intelligent network energy saving and consumption reduction based on network mobility pattern management) in the 5G domain will be a new researching direction, which could provide business values in reducing the operating expenditure and improve the QoS of the cellular network operators.

APPENDIX

A. COMPARISON OF SOM-K AND K-MEANS IN LONG-TERM BEHAVIOR

Since the SOM algorithm cannot achieve odd clustering, such as the 5 clusters in this article, therefore, we just compared the k-means and SOM-K algorithms and set their cluster number to 5 clusters of comparison, and the following conclusions are drawn.

Table 6 shows the intra- and inter-cluster centroids distance between k-means and SOM-K. Intra-centroid cluster distance is also called within-cluster distance, which is representing the distance from samples of the cluster to the cluster centroid. On the contrary, the inter-cluster distance is the distance between two different clusters' centroids. These are two very important unsupervised learning clustering indicators. We can find that the intra-centroids distance between k-means is higher than SOM-K. The inter-centroids distance of k-means is lower than SOM-K, Therefore, according to the shorter distance of intra-clusters and the larger inter-cluster distance the better cluster performance [40]. The k-means cluster performance in high dimension data set is worse than SOM-K.

Comparing Table 3 with 7 we can find that, cluster#5 in SOM-K has the same performance when compared to cluster#1and #4 in k-means, which are all the best performance among the whole 5 clusters. The same situation occurs to clusters #1 and #2 in SOM-K and clusters #3 and #5 in k-means as well, respectively. Cluster #2 in SOM-K and cluster#5 in single k-means have similar performance, which is all the worst clusters. And cluster#1in SOM-K and cluster#3 in k-means are all medium performance clusters. The only difference between these two algorithms is

cluster#4 in SOM-K and cluster#2 in k-means, for instance, cluster#4 in SOM-K has a low RRC setup failure rate and latency, but cluster#2 in k-means has a high RRC setup failure rate and latency. And the number of cells in cluster #2 is larger than cluster#4. All in all, it can also illustrate that k-means cluster performance is not well in high dimension datasets, and applying the dimension reduction method (i.e., SOM) to reduce the dimension of the data first and then use k-means will have a better clustering effect.

B. COMPARISON OF SOM-K AND K-MEANS IN SHORT-TERM BEHAVIOR

Table 8 shows the intra- and inter- centroids cluster distance between k-means and SOM-K. We can find that the value of the intra-centroids distance from k-means is more than SOM-K in clusters #2 and #5. But the distance from cluster#1 and #3 is a bit smaller than SOM-K. That is, compared with SOM-K, the distance distribution of the intra-cluster distance from k-means is more unbalanced and the gap is larger. At the same time, the inter-centroids distance from k-means is lower than SOM-K, therefore, according to the shorter distance of intra-clusters and the larger inter-cluster distance the better cluster performance [40]. The k-means cluster performance in high dimension data set is worse than SOM-K.

Comparing Table 4 with 9 we can find that, cluster#5 in SOM-K has the same performance when compared to cluster#3 and #4 in k-means, which are all having a higher RRC setup failure rate and the performance is worse. That means the channel qualities of these clusters are terrible. The same situation occurs to cluster# #1 in SOM-K and cluster #1 in k-means as well, respectively. Cluster #1 in SOM-K and cluster#1 in k-means have a similar performance, which is the worst performance cluster and all the features' value is very low, which means there is a little person to use a mobile phone. Cluster #2 in SOM-K and cluster#2 in k-means are all medium performance clusters as well. The only difference between these two algorithms is cluster#4 in SOM-K and cluster#5 in k-means, for instance, cluster#4 in SOM-K has a medium RRC setup failure rate and high latency, but the RRC setup failure rate and latency at cluster#5 in k-means are all high and the other features' value is similar. However, these two clusters are all the best clusters of their clusters. All in all, we can find that the cluster results' characteristics by using k-means have many repetitive features' performances in a different cluster (i.e., cluster#3 and #4), but the clustering results of SOM-K are hierarchical. Therefore, it could illustrate that k-means cluster performance is not well in high dimension and large amount datasets, but SOM-K has the exact opposite performance when compared to k-means in high dimension dataset clustering.

REFERENCES

[1] J. Du, C. Jiang, Z. Han, H. Zhang, S. Mumtaz, and Y. Ren, "Contract mechanism and performance analysis for data transaction in mobile social networks," *IEEE Trans. Netw. Sci. Eng.*, vol. 6, no. 2, pp. 103–115, Apr. 2019.

[2] P. Chiu, J. Reunanen, R. Luostari, and H. Holma, "Big data analytics for 4.9G and 5G mobile network optimization," in *Proc. IEEE 85th Veh. Technol. Conf.*, Sydney, NSW, Australia, Jun. 2017, pp. 1–4.

[3] G. Andrienko, N. Andrienko, P. Bak, S. Bremm, D. Keim, T. von Landesberger, C. Pölitz, and T. Schreck, "A framework for using self-organising maps to analyse spatio-temporal patterns, exemplified by analysis of mobile phone usage," *J. Location Based Services*, vol. 4, nos. 3–4, pp. 200–221, Sep. 2010.

[4] D. K. Roy and H. M. Pandey, "A new clustering method using an augmentation to the self-organizing maps," in *Proc. 8th Int. Conf. Cloud Comput., Data Sci. Eng.*, Noida, India, Jan. 2018, pp-739-743.

[5] T. Kohonen, "Self-organizing maps," *Proc. IEEE*, vol. 78, no. 9, pp. 1461–1480, Sep. 1990.

[6] J. Vesanto and E. Alhoniemi, "Clustering of the self-organizing map," *IEEE Trans. Neural Netw.*, vol. 11, no. 3, pp. 586–600, May 2000.

[7] K. Raivio, O. Simula, J. Laiho, and P. Lehtimä, "Analysis of mobile radio access network using the self-organizing map," in *Proc. 8th Int. Conf. Integr. Netw. Manage.*, Colorado Springs, CO, USA, Mar. 2003, pp. 439–451.

[8] T. Kohonen, *Self-Organizing Maps*, vol. 3, 3rd ed. Berlin, Germany: Springer-Verlag, 2001, pp. 177–189.

[9] M. N. M. Sap and E. Mohebi, "Hybrid self-organizing map for overlapping clusters," *Int. J. Signal Process., Image Process. Pattern Recognit.*, vol. 1, no. 1, pp. 11–20, Dec. 2008.

[10] T. Kohonen, "Essentials of the self-organizing map," *Neural Netw.*, vol. 37, pp. 52–65, Jan. 2013.

[11] J. Suykens. *K.U. Leuven, Esat-Stadius Kasteelpark Arenberg 10*. Accessed: Oct. 14, 2014. [Online]. Available: http://www.esat.kuleuven.be/stadius

[12] T. Kohonen and T. Honkela, "Kohonen network," *Scholarpedia*, vol. 2, no. 1, p. 1568, 2007.

[13] Y. Zheng, "Research on model of distribution intrusion detection system based on SOM and K-means," Tianjin Univ. Technol., Tianjin, China, (in Chinese), Tech. Rep., 2008, vol. 29, no. 3, pp. 37–39.

[14] L. Binmei, "Study on improved ad application of self-organizing map neural network," (in Chinese), *Comput. Eng. Appl.*, vol. 45, no. 31, pp. 134–137, Jul. 2009.

[15] Z. He, J. Chen, and M. Gao, "Feature time series clustering for lithium battery based on SOM neural network," in *Proc. 13th IEEE Conf. Ind. Electron. Appl. (ICIEA)*, May 2018, pp. 358–363.

[16] J. N. Xu, L. W. Zhang, L. Xu, and L. Su, "Research on K-means clustering algorithm based on improved genetic algorithm," (in Chinese), *Microcomput. Appl.*, vol. 31, no. 4, pp. 11–15, Aug. 2011.

[17] A. Gomez-Andrades, P. Munoz, I. Serrano, and R. Barco, "Automatic root cause analysis for LTE networks based on unsupervised techniques," *IEEE Trans. Veh. Technol.*, vol. 65, no. 4, pp. 2369–2386, Apr. 2016.

[18] X. Liu, G. Chuai, W. D. Gao, K. S. Zhang, and X. Y. Chen, "KQIs-driven QoE anomaly detection and root cause analysis in cellular networks," in *Proc. IEEE Global Commun. Workshops*, Waikoloa, HI, USA, Dec. 2019, pp. 1–6.

[19] A. Gaber, M. M. Zaki, and A. M. Mohamed, "4G uplink power control tuning approach using unsupervised machine learning," in *Proc. Int. Conf. Inf. Netw.*, Barcelona, Spain, Jan. 2020, pp. 437–442.

[20] S. Swarnalaxmi, I. Elakkiya, M. Thilagavathi, A. Thomas, and G. Raja, "User activity analysis driven anomaly detection in cellular network," in *Proc. 10th Int. Conf. Adv. Comput.*, Chennai, India, Dec. 2018, pp. 159–163.

[21] P. Savazzi and L. Favalli, "Dynamic cell sectorization using clustering algorithms," in *Proc. 65th IEEE Conf. Veh. Technol.*, Dublin, Ireland, Apr. 2007, pp. 604–608.

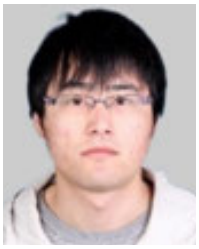
[22] J. Laiho, K. Raivio, P. Lehtimaki, K. Hatonen, and O. Simula, "Advanced analysis methods for 3G cellular networks," *IEEE Trans. Wireless Commun.*, vol. 4, no. 3, pp. 930–942, May 2005.

[23] M. S. Mahmud, M. M. Rahman, and M. N. Akhtar, "Improvement of K-means clustering algorithm with better initial centroids based on weighted average," in *Proc. 7th Int. Conf. Electr. Comput. Eng.*, Dhaka, Bangladesh, Dec. 2012, pp. 647–650.

[24] K. A. S. Immink and J. H. Weber, "Minimum pearson distance detection for multilevel channels with gain and/or offset mismatch," *IEEE Trans. Inf. Theory*, vol. 60, no. 10, pp. 5966–5974, Oct. 2014.

[25] A. Ultsch and H. P. Siemon, "Kohonen's self-organizing feature maps for exploratory data analysis," in *Proc. Int. Conf. Neur. Netw.* Dordrecht, The Netherlands: Kluwer, Jul. 1990, pp. 305–308.

[26] J. L. Chen, R. M. Peng, and S. Z. Li, "Self-organizing feature map neural network and K-means algorithm as a data excavation tool for obtaining geological information from regional geochemical exploration data," *Geophys. Geochem. Exp.*, vol. 41, no. 5, pp. 919–927, Oct. 2017.
[27] R. R. Cai, H. W. Zhang, H. L. Bu, and Y. Zhang, "Research on variation of runoff and sediment load based on the combination patterns in the middle and lower yellow river," (in Chinese), *Shuili Xuebao*, vol. 5, no. 6, pp. 732–742, Jun. 2019.
[28] A. T. Le. *Improving Feature Map Quality of SOM Based on Adjusting the Neighborhood Function*. Accessed: Oct. 14, 2019. [Online]. Available: https://www.intechopen.com/online-first/improving-feature-map-quality-of-som-based-on-adjusting-the-neighborhood-function
[29] K. Kiviluoto, "Topology preservation in self-organizing maps," in *Proc. IEEE Int. Neural Netw.*, Washington, DC, USA, Jun. 1996, pp. 294–299.
[30] E. Germen, "Improving the resultant quality of Kohonens self-organizing map using stiffness factor," in *Proc. 1st Int. Conf. Natural Comput.*, in Lecture Notes in Computer Science, Changsha, China, Aug. 2005, pp. 353–357.
[31] G. Polzlbauer, A. Rauber, and M. Dittenbach, "A vector field visualization technique for self-organizing maps," in *Proc. Pacific-Asia Conf. Knowl. Discovery Data Mining*, Jun. 2004, pp. 399–409.
[32] G. Polzlbauer, "Survey and comparison of quality measures for self-organizing maps," in *Proc. 5th Workshop Data Anal.*, Sliezskydom, Slovakia, Jun. 2004, pp. 67–82.
[33] F. Zhang, J. Wang, and X. P. Wang, "Recognition of spatial framework for water quality and its relation with land use/cover types from a new perspective: A case study of Jinghe Oasis in Xinjiang, China," *Natural Hazards Earth Syst. Sci.*, vol. 358, pp. 1–18, Oct. 2017.
[34] S. Petrovic, G. Alvarez, A. Orfila, and J. Carbó, "Labelling clusters in an intrusion detection system using a combination of clustering evaluation techniques," in *Proc. 39th Int. Conf. Syst. Sci.*, Kauai, HI, USA, Jan. 2006, p. 129.
[35] M. Beccali, M. Cellura, V. Lo Brano, and A. Marvuglia, "Forecasting daily urban electric load profiles using artificial neural networks," *Energy Convers. Manage.*, vol. 45, nos. 18–19, pp. 2879–2900, Nov. 2004.
[36] J. Mwasiagi, H. Xiubao, W. Xinhou, and C. Qing-Dong, "The use of K-means and Kohonen self-organizing maps to classify cotton bales," in *Proc. Conf. BWCC*, New Orleans, LA, USA, Jan. 2007, pp. 379–383.
[37] K. Lu, Q. Wang, J. Xue, and W. Pan, "3D model retrieval and classification by semi-supervised learning with content-based similarity," *Inf. Sci.*, vol. 281, pp. 703–713, Oct. 2014.
[38] D. L. Davies and D. W. Bouldin, "A cluster separation measure," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. PAMI-1, no. 2, pp. 224–227, Apr. 1979.
[39] N. Bolshakova and F. Azuaje, "Cluster validation techniques for genome expression data," *Signal Process.*, vol. 83, no. 4, pp. 825–833, Apr. 2003.
[40] Y. Liu, Z. Li, H. Xiong, X. Gao, and J. Wu, "Understanding of internal clustering validation measures," in *Proc. IEEE Int. Conf. Data Mining*, Sydney, NSW, Australia, Dec. 2010, pp. 911–916.



SHAOXUAN WANG was born in Xi'an, China, in 1990. He received the B.S. degree in automatic from Xi'an Jiaotong University City College, Xi'an, in 2012, and the master's degree in communication electronics from the Tallinn University of Technology, Tallinn, Estonia, in 2016. He is currently pursuing the Ph.D. degree with the Universitat Politècnica de Catalunya, Barcelona, Spain. His research interests include 5G network slicing, neural networks, and machine learning.

RAMON FERRÚS (Member, IEEE) received the B.S. and M.S. degrees in telecommunications engineering and the Ph.D. degree from the Universitat Politècnica de Catalunya (UPC), Barcelona, Spain, in 1996 and 2000, respectively. He is currently a tenured Associate Professor with the Department of Signal Theory and Communications, UPC. His research interests include system design, functional architectures, protocols, resource optimization, and network and service management in wireless communications. He has participated more than ten research projects within the 6th, 7th, and H2020 Framework Programmes of the European Commission, taking the responsibility of WP leader in H2020 VITAL and FP7 ISITEP projects. He has also participated in numerous national research projects and technology transfer projects for public and private companies. He has participated in 3GPP and ETSI standardization activities. He is the coauthor of one book on mobile and one book on mobile broadband public safety communications. He has coauthored over 120 papers mostly in IEEE journals and conferences.