Enhanced Resource Allocation Scheme for the LoRaWAN Harmonization

*Note: Sub-titles are not captured in Xplore and should not be used

Zhifu Zhang
Department of Electrical Engineering
City University of Hong Kong
Hong Kong, China
zhifzhang3-c@my.cityu.edu.hk

Yang Wei
Department of Electrical Engineering
City University of Hong Kong
Hong Kong, China
ywei22-c@my.cityu.edu.hk

Hao Wang
Department of Electrical Engineering
City University of Hong Kong
Hong Kong, China
hwang272-c@my.cityu.edu.hk

Kim Fung Tsang
Department of Electrical Engineering
City University of Hong Kong
Hong Kong, China
ee330015@cityu.edu.hk

IECON 2022 – 48th Annual Conference of the IEEE Industrial Electronics Society | 978-1-6654-8025-3/22/$31.00 ©2022 IEEE | DOI: 10.1109/IECON49645.2022.9968466

***Abstract*—LoRa Wide Area Network (LoRaWAN) is one the of the most popular Internet of Things (IoT) technologies for long-range and low-cost communication. At present, LoRaWAN has been applied in a variety of applications, including localization, smart metering, etc. However, the increasing number of LoRaWAN devices would degrade their quality of service (QoS). There are two main reasons. The first reason is the competition of bandwidth and channel resources between large number of connected end devices. Another one is the redundant channel resources allocation configurations of most end devices to achieve better transmission reliability. To address these challenges, this work proposes an enhanced resource allocation scheme based on both k-means and k-prototype classification algorithms to mitigate the affection of ALOHA scheme and the multi-gateway interference problem. In this proposed scheme, intense network resources under dense end device scenario and the redundant claim of network resources configuration in end devices are considered. An outlier improved spreading factor distribution method is also proposed to reduce the negative effect of the problems. By evaluating and comparing packet loss rate and the relative distribution of spreading factors, an average of 22% increment in transmission performance of LoRaWAN networks is achieved.**

Keywords*—*LoRaWAN, Spreading Factors, K-means, K-Prototypes, Harmonization, Internet of Things

I. Introduction (*Heading 1*)

Currently, the Internet of Things (IoT) has been the one of the most attractive technologies in both research and industry area, given that it generates the possibility that the machine type devices can communicate with other kind of machine type devices (i.e., machine-to-machine (M2M) communication [1]) which in turn stimulates many emerging applications (e.g., Smart Metering, etc.). In general, an IoT network interconnects variety of sensor nodes collecting the sensing data (e.g., temperature data, humidity data, etc.) and transmits the collected sensing data to server for further processing. To embrace the development of IoT, many communication technologies are proposed.

Conventionally, the ZigBee, Bluetooth, Wi-Fi [2] are the leading technology in IoT regarding the efficient data rate. However, given the scale of IoT applications increases drastically and convergent to be deployed in large-scale. The aforementioned technologies become cost-inefficient to support large-scale applications (e.g., remote monitoring [3], etc.). To deal with this challenge, researchers propose the Low Power Wide Area (LPWA) technologies recently. There are three leading LPWA technologies, namely Long Range (LoRa), Narrowband-IoT (NB-IoT), and Sigfox [4], respectively. All these LPWA technologies can provide kilo-meter-level coverage and uA-level power consumption in principle. For LoRa, to generate the large-scale network, it usually applies the LoRa wide area network (LoRaWAN) protocol proposed by LoRa-alliance [4] to interconnect hundreds of LoRa end devices. NB-IoT and Sigfox are mainly managed by the operators, and they lack the degree of freedom on deploying private applications, whereas LoRaWAN supports the deployment of applications in private. Thus, in this paper, we mainly focus on the LoRaWAN.

However, in actual deployment of LoRaWAN, a commonly existing issue of low quality of service (QoS) is inevitable with increasing LoRa end device connected. The issue occurs due to two reasons. Firstly, LoRaWAN is mainly deployed in unlicensed band with limited frequency resources (5MHz in general [18]), which in turn provides limited communication channels. Furthermore, LoRaWAN used pure Aloha multi-access scheme, in which connected end devices transmit in randomly select wireless channels. Application of pure Aloha scheme in high-density scenarios, such as kilo-number end devices, increases possibility of collisions while decreasing the scalability and reliability of the network, further causing to more serious packet loss. Secondly, some LoRa devices may be configured with excessive channel resource occupation, resulting in insufficient channel resources to other LoRa devices which also causing to serious packet loss.

Thus, to overcome the aforementioned two challenges, an outlier improved spreading factor distribution (OISFD) method is proposed to mitigate the negative affection of the mentioned harmonization in network resources issues. OISFD method is based on k-means algorithm and its variant k-prototype algorithm. OISFD method makes improvements targeting on disadvantages of both algorithms that aim at different scenarios. In this work, the issue of intensity in the wireless transmission resources allocation is categorized as Single Gateway (Single GW) scenario in the remaining of the work. And the issue of redundant settings of each end device is categorized and referred to as Multi-Gateway (Multi GW) scenario for reference.

The performance of QoS for LoRaWAN is based on the packet loss rate (PLR) which is defined as the ratio of the failed uplink (i.e., transmission from end device to gateway) message to total uplink messages in a LoRaWAN. Evaluation of method performance is based on the PLR comparisons between scenarios where 2 different versions of the method were applied for experiments. One of the versions is OISFD-k-means-based (KMB) method, the other is OISFD-k-prototype-based (KPB) method.

In this paper, 2 contributions were proposed and listed:

- OISFD method is proposed for low QoS issues. Version OISFD-KMB is proposed for single gateway scenario. Version OISFD-KPB is proposed for multi-gateway scenario when mitigating the analyzed issues.
- Via applying OISFD method to both scenarios, an average of 8.854% of decrease in PLR of the LoRaWAN is observed.

The remaining of this paper is organized into 6 sections. Section II presents the related works. Section III explains functioning mechanics of methodologies applied in the work. Section IV introduces the experiment and the testbed setup. Section V gives experiment results and analysis with discussion of the collected data from the experiment setup. Section VI, conclusion, gives conclusive summarize. Section VII shows all references.

II. Related Works

Recent research works on LoRaWAN focused on the energy consumption and single gateway SF allocation optimization issues. [5] analyzed the LoRaWAN network through a game theoretical approach with optimization target of reaching the Bayesian Nash equilibria. [6] used a frequency-channel-based approach for the allocation of SF for each end device aiming at enhancing the overall network performance. The proposed method dynamically adjusts the SF based on the instantaneous transmission channel realization. [7] features dynamic SF allocations as well, [7] improves the original adaptive data rate (ADR) algorithm. The method uses ordered weighted averaging (OWA) as a decision-making algorithm capable of considering configurations of the transmission channel parameters. The author in [8] focused on the co-SF and inter-SF interference caused high PLR issue and proposed an Interference-Aware Spreading Factor Assignment (I-ASF) method for better allocation of SF for network harmonization. Interference in single gateway scenario is the primary concern of this work. [9] considers both the SF interference and the power constrains issue by proposing a mixed-integer non-linear optimization method. A low-complexity many-to-one matching algorithm between SFs and end-devices is constructed for the allocation of SF. In [10], the author used distance-based K-Means algorithm for the allocation of the SF in single gateway scenario. No interference issues and inter-SF issue are considered. To further enhance the LoRaWAN network scalability and the applicability for multi-gateway scenario and interference resistance, this work further proposed a method to harmonization of LoRaWAN.

III. Methodology

A. LoRaWAN Spreading Factor

Spreading factor (SF) (Ranging from 7-12 [18]), a parameter defined by LoRaWAN standard published by LoRa Alliance, is a parameter defined to differentiate different transmission speeds in LoRa radio transmission for each of the end devices connected to LoRaWAN based on the radio transmission parameters (includes receive signal strength indicator (RSSI), signal-to-noise ratio (SNR), PLR, Distance between end devices (ED) and the gateways (GW)…) of LoRa. SF is the categorical output of the proposed OISFD method. By assigning different SF for each individual ED, LoRaWAN distributes different levels of network resources to each ED correspondingly. SF has 6 levels categorizing transmission data rate ranging from 7 to 12. Of which the minimum level 7 gives the highest data rate and the least network coverage. The SF level 12 gives the lowest data rate and the largest network coverage.

Table I lists the detailed parameters and coding specifications of each level of SF.

TABLE I. Spreading Factor Level and Data Rate [11]

Data Rate Code	Spreading Factor	Channel Width	Coding Rate	Data Rate
0	12	125kHz	4/6	250bps
1	11	125kHz	4/6	440bps
2	10	125kHz	4/5	980bps
3	9	125kHz	4/5	1760bps
4	8	125kHz	4/5	3125bps
5	7	125kHz	4/5	5470bps

Different SF allocates different levels of resources to each end device. Constrained by the gateway's radio limitations (duty cycle, antenna sensitivity, etc.), the portion of end device obtaining a certain level of SF is limited. Hence to distribute the network resources fairer and to achieve harmonization of the wireless network resources in LoRaWAN, the distance between the end devices and the gateway, RSSI, SNR, the original SF need to be comprehensively considered. This work uses the distribution of spreading factor as the main network resources distribution indicator and the output of the proposed method.

B. Introduction of Base Algorithm: K-means Algorithm

K-means algorithm is a Euclidian-distance based, unsupervised, local optimized machine learning (ML) clustering algorithm [12]. K-Means++ [13] algorithm is an improved K-Means algorithm that optimized the process of choosing the initial K cluster centroids. K-Means++ does not need centroids data point as the input. It random chooses K centroids from the input data set first and categorized the data points into the corresponding cluster based on the Euclidean distance between each of the data point and the chosen centroids. Then the algorithm recalibrates centroids by calculating means of clustered data points and use it as centroids of next categorization. The above steps are repeated until the centroids converges. At this point, the result is stable and taken as output. K-Means algorithm has been one of the easiest, fastest unsupervised ML clustering algorithms with relative low time complexity while obtaining effectively good results in clustering applications. The advantages of K-Means algorithm include:

1) K-Means algorithm obtains good elasticity when dealing with large data set and easy to understand.

2) K-Means algorithm performs best when the input data is convex clustering alike.

3) K-Means algorithm obtains time complexity of $O(n^2)$[8], where n is the size of the input data set. The time complexity of K-Means algorithm is simpler compared with most other common clustering algorithms.

C. Disadvantages of K-means algorithm

Being an unsupervised ML method, K-Means needs input K as the result clusters' number. The clustering result of K-Means algorithm is heavily affected by extreme data points, also called outliers. To sum up, the disadvantages of K-Means algorithm include:

1) K-Means algorithm requires manually predicted and assigned K value, the number of clusters, to reach the best clustering performance.

2) K-Means clustering result is sensitive to initial centroids. Different initialization of K centroids gives drastically different clustering results.

3) K-Means clustering result is sensitive to abnormal and extreme values that are off-cluster.

The k-means algorithm obtains simplicity and relative time-efficient features, while showing drawbacks of lack of automation and is outlier sensitive.

D. Improvements to K-means Algorithm

1) Gap Statistic Analysis

Gap statistical analysis (GSA) method is proposed to automatically predicting the number of clusters of an input data set [14]. The predicted data set is then sent to the K-Means algorithm as input, users do not need to input K to the method anymore.

In gap statistic method, a loss function defined by K-Means algorithm is used as the main indicator of choices. The loss function is formulated as

$$D_k = \sum_{i=1}^{N} \left\| x^i - \mu_{c^i} \right\|^2, k \in K \tag{1}$$

where the D_k is defined as the loss function when the K value is assigned by k, K is the range for k values. x^i is defined as each data point in the data set. c^i is defined as the clustered centroid of x^i. μ_{c^i} is defined as the data values of centroids c^i. D_k effectively takes the sum of square of the vector distance between each data point and whose corresponding cluster centroid as the total loss function of K-Means algorithm. The larger the loss function output is, the worst the clustering result is for K-Means algorithm.

$$Gap(k) = \mathrm{E}(\log D_k) - \log D_k \tag{2}$$

Gap(k) is the gap value defined by GSA. It is calculated by the comparison between the expectation $\mathrm{E}(\log D_k)$ and the actual value calculated from the loss function as well. The expectation, $\mathrm{E}(\log D_k)$, of the logarithmic of D_k, which is the loss function defined by K-Means algorithm, is calculated by Monte Carlo Method simulated results. The range taken by Monte Carlo Method is max and min value in input data set. The k value with the largest Gap(k) is then chosen as the input of K to the K-Means algorithm. Because the k value with the largest gap is the k value that gives largest decline of total distance between all the data set and the corresponding centroids. Hence the method chooses the K value with the largest gap value as the predicted K value.

GSA can predict the of number of clusters in K-Means algorithm. However, the Monte Carlo method takes relatively large amount of time to calculate. Hence in this work, the input to the GSA is the means of all devices' input data values instead of all the input data values. Hence the time needed for GSA can be drastically reduced while maintaining a good accuracy.

2) Inter Quartile Range (IQR) Method

Sensitivity to abnormal values is one of the disadvantages of K-Means. The Inter Quartile Range (IQR) method is used to eliminate the outlier points in the data set for better prediction accuracy. A box plot is demonstrated for explaining the mechanism of IQR method.

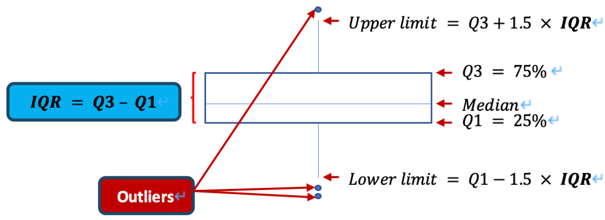

Fig. 1. Boxplot of IQR definition

Firstly, the features of input data set are extracted. The Q1 value is defined as the 25% data point value regarding the data set in a whole with ascending order. The Q3 value is defined as the 75% data point respectively. The median of the data set is denoted as the median. Then, IQR value is calculated by

$$IQR = Q3 - Q1 \tag{3}$$

This range is then used in the calculation of upper and lower bounds of outlier elimination. Upper limit and lower limit used to classify the outliers are calculated by

$$Upper\ limit = Q3 + 1.5 \times IQR \tag{4}$$

$$Lower\ limit = Q1 - 1.5 \times IQR \tag{5}$$

Based on these calculated boundaries, data points outside of these boundaries are eliminated in the following clustering process.

E. OISFD-K-Means-Based (KMB) Method

The outlier improved spreading factor distribution – K-Means-Based (KMB) method is introduced in this section. It has below few improved features:

- OISFD-KMB method requires only the data set to be trained and predicted and requires no more input parameter.
- OISFD-KMB predict the K value based on input data set automatically compared with K-Means algorithm.
- OISFD-KMB method is more immune to outliers in the data set, the sensitivity of OISFD to the extreme data point are reduced due to the improvement made regarding input of the K-Means algorithm. The outliers are eliminated by application of IQR method.

The OISFD-KMB method explained by order of workflow.

Input: of OISFD-KMB method is the data set containing N end devices' 20 transmission records each.

Output: of OISFD-KMB are clustering centroids and clustered labels for each end device.

- Step 1, the means of each end device's transmission information are calculated.
- Step 2, $\mathrm{E}(\log D_k)$ is calculated by simulation, the simulation upper bound is the max value taken from the input data set in each column. Same, the simulation lower bound is the min value taken from the input data

set in each column. The simulation data set size remains identical with the input data set containing N end devices' 20 transmission records each.

- Step 3, D_k is calculated using mean values calculated in step 1 with K value ranges from 1 to 6. In this process, the K values excluded by elbow method implemented in K-Means++ algorithm are excluded from K range as well.
- Step 4, $Gap(k)$ are calculated for K value range from 1 to 6. The K value with the largest gap is chosen.
- Step 5, input data set is filtered by IQR method.
- Step 6, the means of filtered data are calculated as the input to the K-Means algorithm.
- Step 7, filtered data set from step 6, and K value chosen from step 4 are input to the K-Means algorithm, outputting clustered centroids and end devices data rate labels (SF).

F. *Introduction to K-prototype Algorithm*

K-Prototype algorithm is a variant of K-Means algorithm. It handles both numerical data type and categorical data type. For numerical data type, K-Prototype algorithm uses Euclidean Distance to calculate the similarity of the data points. For categorical data type, K-Prototype uses Hamming distance [16]. The total distance is given by both distance calculation with a weight assigned to balance the affection of numerical data type and categorical data type. The D(x), distance between data points is given by

$$D(x, C_k) = \sum_{i=1}^{N} (x_i - c_i)^2 + \mu \sum_{j=1}^{M} \delta(x_j, c_j) \qquad (6)$$

while

$$\delta(x, c) = \begin{cases} 1, if\ x\ != c \\ 0, if\ x == c \end{cases} \qquad (7)$$

Where C_k is the centroid. c_i is the numerical data value of C_k where i ranges from 1 to N, i.e., the first N numerical data columns. c_j is the categorical data value of C_k where j ranges from 1 to M, i.e., the last M categorical data columns. μ is the weight for balance Hamming distance and Euclidean distance. K-Prototype algorithm differs K-Means algorithm in the similarity calculation, also, K-Prototype algorithm requires input of weighting coefficient to balance Hamming distance and Euclidean distance. However, the weight parameter can also be automatically assigned by K-Prototype algorithm itself.

1) *Reverse of Hamming Distance Results*

The OISFD-KPB method is proposed for multi-gateway scenario where redundant claiming of network resources between different gateways is the major issue responsible for high PLR and low QoS. However, LoRaWAN gateways typically communicates with end devices using 8 frequency channels [18]. When the end devices are using different frequency channels, same SF settings between them will have interference whose affection can be omitted. Based on this fact, the Hamming distance of results of 'frequency' column in the OISFD-KPB method is modified into below expression:

$$\delta(x, c) = \begin{cases} 0, if\ x\ != c \\ 1, if\ x == c \end{cases} \qquad (8)$$

when the frequencies of two end devices become identical, the results should tend to categorize these two end devices into two different spreading factor clusters. Vice versa. This relationship results in the counter-Hamming-distance calculation. When the frequencies of two end devices are same, the dissimilarity of the two end devices are more that that of two end devices are different in frequencies. Hence, in the OISFD-KPB method, the Hamming distance for K-Prototype application was modified to (8). sK-Prototype algorithm shares the same advantages and disadvantages with K-Means algorithm. Except for the changes in the similarity calculation between centroids and data points. Thus, the advantages of K-Prototype algorithm are:

- K-Prototype algorithm obtains good elasticity when dealing with large data set.
- K-Prototype algorithm obtains time complexity of $O(n^2)$[17], where n is the size of the input data set. The time complexity of which is simpler compared with most other common clustering algorithms.
- K-Prototype algorithm handles both numerical data type and categorical data type.

The advantages and disadvantages of K-Prototype are similar to K-Means. Hence, by applying GSA and IQR, the disadvantages of K-Prototype algorithm can be improved.

G. *OISFD-K-Prototype-Based (KPB) Method*

The outlier improved spreading factor distribution – K-Prototype-Based (KPB) method is introduced. By applying the improvement targeting disadvantages of K-Prototype algorithm, the OISFD-KPB is proposed. Similar to OISFD-KMB method, OISFD-KPB method has below few features:

- OISFD-KPB method requires 2 inputs. The first one is the data set to be trained and predicted. The second one is the categorical columns in the data set selecting columns of data needed as input.
- OISFD-KPB predict the K value based on input data set automatically using gap statistic method.
- OISFD-KPB method is more immune to outliers in the data set, the sensitivity of OISFD to the extreme data point are reduced due to the improvement made regarding input of the K-Prototype algorithm. The outliers are eliminated by application of IQR method.

Here the method is explained by the order of workflow:

Inputs: OISFD-KMB method are the data set containing N end devices' 20 transmission records each and the categorical columns indicator.

Output: OISFD-KMB are clustering centroids and clustered labels for each end device.

- Step 1, calculate the means of each end device's transmission data. The categorical data columns are excluded from calculation of means.
- Step 2, $E(\log D_k)$ with K range from 1 to 6 are calculated using Monte Carlo simulation, the simulation upper bound is the max value of input data set in each column. The simulation lower bound is the min value of input data set in each column. The simulation data set size remains identical with the input

data set containing N end devices' 20 transmission records.

- Step 3, D_k is calculated using mean values from step 1 with K value ranges from 1 to 6. In this process, the K will also be filtered by elbow method.
- Step 4, $Gap(k)$ are calculated for K value range from 1 to 6. The K value with the largest gap is chosen.
- Step 5, input data set is filtered by IQR method.
- Step 6, the means of filtered data are calculated as the input to the K-Prototype algorithm, the categorical data columns (input) are excluded.
- Step 7, filtered data set from step 6, and K value chosen from step 4 are input to the K-Prototype algorithm, outputting clustered centroids and end devices data rate labels (SF).

IV. Experiment

For verifying the actual performance of the proposed OISFD method, a testbed consists of 66 LoRa end devices and 4 LoRaWAN gateways and Chirpstack network server were setup. The models of end devices include NETVOX R718N17, NETVOX R718G, Elsys EMS5k and Elsys ERS-CO2 sensors. All with default firmware configurations. The sensors functionality is air quality sensing related. The 4 deployed gateways are MultiTech MTCDT-L4E1 with firmware upgraded to mLinux 5.3.31 instead of A.E.P. The Chirpstack network server is version 3.16.3, Chirpstack application server is version 3.17.7, Chirpstack gateway bridge is version 3.14.2.

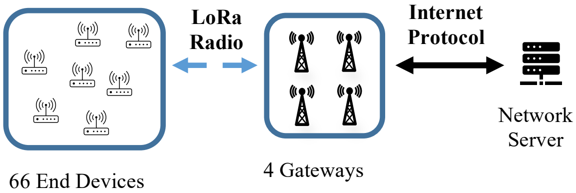

Fig. 2 Deployment of Testbed

In the testbed, the method was deployed in the network server. The algorithm replaces the default ADR algorithm. The data was collected from the persistent database in the network server. The data include 12 columns for each record, which are receiving time, device EUI, device name, the data rate level, frame count, the frame port, the frequency channel used, the gateway number count, the gateway id, primary indicator, RSSI and SNR. These data columns are collected using data export for analyzing.

V. Results and Analysis

A. Single Gateway Scenario

For single GW scenario, the result is mainly analyzed by the distribution of SF within 1 gateway's cover range, and the overall packet loss rate (PLR) of 1 gateway's connected end devices. The PLR comparisons between K-Means algorithm and OISFD-KMB, on 4 single gateway LoRaWAN are analyzed in details.

1) Spreading Factor Distribution Comparisons.

In this sub-section, the SF distributions of three scenarios are listed and compared with each other. The "pre-processing" section gives the SF distribution using the default ADR algorithm, the "K-Means" section presents the SF distribution of proposed K-Means-based method in [10] for resources allocation. The "OISFD-KMB" and "OISFD-KPB" columns shows the SF distribution in the respective application scenarios. The first gateway with id = '00800000A0008074', represented by GW 074.

TABLE II. GW 074 SF Comparisons

SF	Pre-processing (ADR)	K-Means	OISFD-KMB
SF = 12	21	22	24
SF = 10	14	11	9
SF = 7	1	2	3

Gateway with id = '647FDAFFFE00C79B', represented by GW 79B.

TABLE III. GW 79B SF Comparisons

SF	Pre-processing (ADR)	K-Means	OISFD-KMB
SF = 12	37	32	36
SF = 10	28	32	16
SF = 7	1	2	14

Gateway with id = '647FDAFFFE00C778', represented by GW 778.

TABLE IV. GW 778 SF Comparisons

SF	Pre-processing (ADR)	K-Means	OISFD-KMB
SF = 12	37	44	52
SF = 10	21	14	6

Gateway with id = '647FDAFFFE006B32', represented by GW B32.

TABLE V. GW B32 SF Comparisons

SF	Pre-processing (ADR)	K-Means	OISFD-KMB
SF = 12	37	32	27
SF = 10	23	19	17
SF = 7	1	10	17

From the results of each individual gateway, the distribution of spreading factors under K-means optimized and OISFD optimized are improved in different level. The distribution of SF is observably more utilized in the OISFD optimized scenario than k-means optimized and pre-processing scenarios.

2) Single gateway's PLR Comparison and Analysis

TABLE VI. PLR Comparison for Single-Gateway Scenario

GW	Pre-processing (ADR)	K-Means	OISFD-KMB
GW 074	0.21629	0.19260	0.17025
GW 79B	0.21547	0.18023	0.13502
GW 778	0.62613	0.59682	0.51003
GW B32	0.59840	0.54991	0.55185
Mean	0.41407	0.37989	0.34178

From the collected and summarized PLR of 4 gateways in different situations, the performance improvement was analyzed below. Average PLR decrement Between K-Means and OISFD-KMB:

$$0.37989 - 0.34178 = 3.811\% \quad (9)$$

Average PLR decrement of OISFD-KMB compared with ADR method:

$$0.41407 - 0.34178 = 7.229\% \quad (10)$$

From above summarized results regarding PLR, the calculation of PLR decrement of OISFD-KMB was 7.229%, the PLR decrement between K-Means and OISFD-KMB was 3.811%. These results proved the effectiveness of OISFD-KMB method in the application of single gateway scenario on the issue of network resource distribution of LoRaWAN network. The actual PLR was reduced from 41% to 34% generally for 4 gateways individually. The networks' overall QoS was improved, and the negative affection of intensity of network resources was further reduced.

B. Multi-Gateway Scenario SF Distribution Comparisons.

TABLE VII. SF COMPARISON FOR MULTI-GATEWAY SCENARIO

SF	Pre-processing (ADR)	OISFD-KPB
SF = 12	75	66
SF = 10	56	67
SF = 7	4	3

From the summarized numerical results of spreading factor, or data rate level distribution for the deployed 4 gateways in single LoRaWAN networks. One is able to observe that the distribution of spreading factor of all devices within the cover range of LoRaWAN are more evenly distributed with application of OISFD-KPB method for network resource allocation.

TABLE VIII. PLR COMPARISON FOR MULTI-GATEWAY SCENARIO

	Pre-processing (ADR)	OISFD-KPB
Mean PLR of 4 Gateways	0.414079236	0. 309280501

PLR decrement of OISFD-KPB method on multi-GW scenario.

$$0.414079236 - 0.309280501 = 10.4794226\% \quad (11)$$

From above PLR table for multi GW scenario applying OISFD-KPB method, PLR decreased by 10.479%. The actual PLR was reduced from 41% to 30%. By applying this method, the redundant configuration settings caused negative affection to large scale LoRaWAN network was minimized. These results proved the effectiveness of OISFD-KPB method in the application of multi GW scenario on the issue of network resource distribution of LoRaWAN network.

VI. CONCLUSION

In this work, a harmonization method for improving QoS and better allocation of network resources of LoRaWAN is proposed based on k-means and k-prototype algorithms. The method targets the issues brought by the intensity of network resources under the condition of kilo-number end devices connected to single gateway in LoRaWAN and the issue of redundant claiming of the network resources configuration in the end devices in the network. Experiment was conducted for verification and evaluation based on the PLR of the network per different testing scenarios. The method generally decreased overall PLR of LoRaWAN by 8.854%, thus serving as an improvement targeting LoRaWAN harmonization.

VII. REFERENCES

[1] O. A. Amodu and M. Othman, "Machine-to-machine communication: An overview of opportunities," *Computer Networks,* vol. 145, pp. 255-276, 2018.

[2] S. J. Danbatta and A. Varol, "Comparison of Zigbee, Z-Wave, Wi-Fi, and bluetooth wireless technologies used in home automation," in *2019 7th International Symposium on Digital Forensics and Security (ISDFS)*, 2019: IEEE, pp. 1-5.

[3] M. Sharma, M. K. Singla, P. Nijhawan, S. Ganguli, and S. S. Rajest, "An Application of IoT to Develop Concept of Smart Remote Monitoring System," in *Business Intelligence for Enterprise Internet of Things*: Springer, 2020, pp. 233-239.

[4] K. Mekki, E. Bajic, F. Chaxel, and F. Meyer, "A comparative study of LPWAN technologies for large-scale IoT deployment," *ICT express,* vol. 5, no. 1, pp. 1-7, 2019.

[5] A. Tolio, D. Boem, T. Marchioro, and L. Badia, "Spreading factor allocation in LoRa networks through a game theoretic approach," in *ICC 2020-2020 IEEE International Conference on Communications (ICC)*, 2020: IEEE, pp. 1-6.

[6] R. Hamdi, M. Qaraqe, and S. Althunibat, "Dynamic spreading factor assignment in LoRa wireless networks," in *ICC 2020-2020 IEEE international conference on communications (ICC)*, 2020: IEEE, pp. 1-5.

[7] J. Babaki, M. Rasti, and R. Aslani, "Dynamic spreading factor and power allocation of LoRa networks for dense IoT deployments," in *2020 IEEE 31St annual international symposium on personal, indoor and mobile radio communications*, 2020: IEEE, pp. 1-6.

[8] A. Farhad, D.-H. Kim, P. Sthapit, and J.-Y. Pyun, "Interference-aware spreading factor assignment scheme for the massive LoRaWAN network," in *2019 International Conference on Electronics, Information, and Communication (ICEIC)*, 2019: IEEE, pp. 1-2.

[9] L. Amichi, M. Kaneko, E. H. Fukuda, N. El Rachkidy, and A. Guitton, "Joint allocation strategies of power and spreading factors with imperfect orthogonality in LoRa networks," *IEEE Transactions on Communications,* vol. 68, no. 6, pp. 3750-3765, 2020.

[10] M. Asad Ullah, J. Iqbal, A. Hoeller, R. D. Souza, and H. Alves, "K-means spreading factor allocation for large-scale LoRa networks," *Sensors,* vol. 19, no. 21, p. 4723, 2019.

[11] Y. Lalle, L. C. Fourati, M. Fourati, and J. P. Barraca, "A comparative study of LoRaWAN, SigFox, and NB-IoT for smart water grid," in *2019 Global Information Infrastructure and Networking Symposium (GIIS)*, 2019: IEEE, pp. 1-6.

[12] J. A. Hartigan and M. A. Wong, "Algorithm AS 136: A k-means clustering algorithm," *Journal of the royal statistical society. series c (applied statistics),* vol. 28, no. 1, pp. 100-108, 1979.

[13] D. Arthur and S. Vassilvitskii, "k-means++: The advantages of careful seeding," Stanford, 2006.

[14] R. Tibshirani, G. Walther, and T. Hastie, "Estimating the number of clusters in a data set via the gap statistic," *Journal of the Royal Statistical Society: Series B (Statistical Methodology),* vol. 63, no. 2, pp. 411-423, 2001.

[15] A. Chaturvedi, P. E. Green, and J. D. Caroll, "K-modes clustering," *Journal of classification,* vol. 18, no. 1, pp. 35-55, 2001.

[16] A. Bookstein, V. A. Kulyukin, and T. Raita, "Generalized hamming distance," *Information Retrieval,* vol. 5, no. 4, pp. 353-375, 2002.

[17] M. K. Pakhira, "A linear time-complexity k-means algorithm using cluster shifting," in *2014 international conference on computational intelligence and communication networks*, 2014: IEEE, pp. 1047-1051.

[18] "LoRaWAN® Specification," LoRa Alliance®, 14-Oct-2021. [Online]. Available: https://lora-alliance.org/about-lorawan/. [Accessed: 25-May-2022].