Efficient k-means based clustering scheme for mobile networks cell sites management

Jocelyn Edinio Zacko Gbadoubissa, Ado Adamou Abba Ari, Abdelhak Mourad Gueroui

Journal of King Saud University – Computer and Information Sciences 32 (2020) 1063–1070

Contents lists available at ScienceDirect

Journal of King Saud University – Computer and Information Sciences

journal homepage: www.sciencedirect.com

Efficient k-means based clustering scheme for mobile networks cell sites management

Jocelyn Edinio Zacko Gbadoubissa [a,c], Ado Adamou Abba Ari [b,c,*], Abdelhak Mourad Gueroui [b]

[a] African Institute for Mathematical Sciences (AIMS-Cameroon), P.O. Box 608, Limbé, Cameroon
[b] LI-PaRAD Lab, Université Paris Saclay, University of Versailles Saint-Quentin-en-Yvelines, 45 Avenue des États-Unis, 78035 Versailles cedex, France
[c] LaRI Lab, University of Maroua, P.O. Box 814, Maroua, Cameroon

ARTICLE INFO

Article history:
Received 24 July 2018
Revised 8 October 2018
Accepted 30 October 2018
Available online 1 November 2018

Keywords:
Clustering
K-means
Geometry of a circle
Mobile networks
OpenCellID

ABSTRACT

Telecommunication network infrastructures in Africa and the Middle East regions, are deployed and operated in challenging environments that are highly scattered particularly in rural areas. Moreover, considerable number of cell sites are located in areas difficult to access. Furthermore, low income in rural areas does not allow a fast return on investment since the cost of deployment and operation of a cell site is considerable. These issues lead to a difficult human resource management, particularly, in the assignment of technicians to cell site for maintenance purpose. In this paper, an optimized scheme for costs of maintenance operations on cell sites is proposed. We used the k-means clustering algorithm for allocating field technician to a pool of cell sites. Moreover, to alleviate the k-means sensitivity to initialization, we proposed an initialization method that is based on the geometry of a sphere. We conducted series of experiments with sample of thousands of cell towers from OpenCellID and the results demonstrate the effectiveness of the proposal.

1. Introduction

1.1. Background

Information, and Communication Technologies play a key role in Africa development, as well as in developing countries. In the last decade, the telecommunication market in Africa and the Middle East has enjoyed penetration growth and profitability far above the averages. These regions represented 8% of the global telecommunications market in 2015 and contributed nearly 20% of the economic profit pool (Boniecki et al., 2016). According to a study carried out by International Finance Corporation in 2013, Nigeria and Ghana have reached a mobile subscriber base of 107 and 25 million respectively. This base is driven by strong growth in mobile penetration and mobile coverage of 80% of the population. These countries, both combined, have a total of 35000 sites, of which 50% are located in areas without commercial grid power, and mostly in rural remote location often difficult to access. This prevents the operators from carrying out maintenance operations on the network effectively. Hence, this situation affects the cost of maintenance operations, the network availability, the quality of service and the quality of experience.

In developing countries, Mobile Networks Operators (MNO) such as Vodaphone, Vodacom, Orange, Mobile Telephone Network (MTN) and Airtel prefer to divest from cell towers management in order to focus on service delivery. Towers companies such as Ericsson, Helios Towers Africa, Eaton towers, IHS Towers and American Tower Corporation are contracted to manage telecommunications towers. Unfortunately, MNO and Tower companies in Africa are faced with many challenges such as equipment monitoring and maintenance of existing passive infrastructure, operational leakages (diesel pilferage), security, surveillance and environmental control. Nevertheless, Intelligent Site Asset Management solutions have been developed, and their goal is to maximize service delivery potential and benefits, while minimizing related risks and costs (Dietrich, 2016).

* Corresponding author at: LI-PaRAD Lab, Université Paris Saclay, University of Versailles Saint-Quentin-en-Yvelines, 45 Avenue des États-Unis, 78035 Versailles cedex, France.
E-mail addresses: adoadamou.abbaari@gmail.com (A.A.A. Ari), mourad.gueroui@uvsq.fr (A.M. Gueroui).

Peer review under responsibility of King Saud University.

Production and hosting by Elsevier

https://doi.org/10.1016/j.jksuci.2018.10.015
1319-1578/

Despite these solutions, Tower companies use field technicians to ensure some maintenance tasks which cannot be performed by Site Asset Management tools. These tasks can be refueling, fixing physical damage on a cell site and others activities related with maintaining and operating a cell site. Hence, some Tower companies adopt the strategy of assigning a field technician to a pool of cell sites; and a technician is positioned to a predetermined location such that he can move easily to each site. In case that a cell site fails to function due to physical damage for example, the responsible field technician is required to the critical site to fix the problem.

However, in Africa and Middle East, several cell sites are located in regions difficult to access particularly in rural remote location so that moving a technician to these sites is a very costly operation in time and money. Therefore, in this paper we investigate on optimal predetermined locations for technicians and an optimal allocation of cell sites such that the cost due to a field technician and maintenance operation is minimized. This problem can be considered as that of grouping objects based on their similarities.

In other words, this issue can be assimilated to a clustering problem. Given a set of n objects in d-dimensional space and an integer $k \geqslant 2$, a k-clustering problem is defined as finding a partition of the n objects in k groups called clusters in such a way that an objective function is optimized (Ari et al., 2016; Rahman et al., 2015; Wu, 2012; Moussaoui et al., 2006; Tan et al., 2005). Finding the ideal partition which optimizes the objective function is difficult and makes the k-clustering problem *NP*-hard (Ari et al., 2018; Titouna et al., 2018; Kel'manov and Pyatkin, 2015; Ageev et al., 2014). Nevertheless, some researchers have proposed methods which seek approximate partitions that is also called local optima (Lloyd, 1982).

The k-means algorithm is one of famous k-clustering algorithms which seek approximate solutions (Steinley, 2006). The k-means uses the Euclidean distance as a similarity measure to gather objects into the same cluster. It is widely studied and used. Despite its popularity, the k-means suffers from some problems such as its sensitivity to initial cluster centers. Considering the cell sites as n objects in a 2-dimensional space and the number of technicians as the integer k, our work is to adapt the k-means algorithm with respect to the problem addressed in this paper.

1.2. Authors' contributions

One of the purposes of this paper is to address resource allocation problem, in telecommunications domain. In our context, we consider technicians as resources. In this paper, we address the problem of clustering mobile network cell sites in rural areas. We formulated the k-clustering as a graph theory problem. We adapt the k-means algorithm to partition a set of data points into a number of clusters while minimizing the Sum of Squared Error (SSE) or Inter cluster. We propose an initialization method called geometric initialization based on the geometry of a sphere to position the k initial cluster centers. We conducted intensive simulation and the results show that our method outperforms the random initialization. Briefly, our main contributions can be summarized as follows:

- Graph theory based formulation of the k-clustering problem.
- Adaptation of k-means algorithm to partition clusters.
- Proposition of a geometric initialization scheme based on the geometry of a circle.
- Simulation of the proposed schemes to demonstrate its effectiveness compared to existing schemes.

1.3. Organization of the paper

The rest of the paper is organized as follows: Section 2 we study the used k-means algorithm; In Section 3 the k-clustering problem and the fitness function derivation are given; Section 4 presents our proposed geometric initialization scheme; and then this is followed by the performance evaluation in Section 5; Finally the conclusion and direction of future work are given in Section 6.

2. K-means algorithm

The *k-means* algorithm belongs to the class of exclusive clustering. It is one of the famous clustering methods which have been studied in the last decades due to its simplicity (Celebi et al., 2013, 2012). It is a method of vector quantization that was originally used in signal processing.

The k-means algorithm given in Algorithm1 starts by selecting k cluster centers randomly. Then it assigns each data point to its closest center with respect to the Euclidean distance measure. Next, it recomputes the positions of all cluster centers. It repeats the two last steps until no point can move from one cluster to another.

Algorithm 1: Pseudo-code of k-means

input : S set of n data points $x \in \mathbb{R}^d$ and an integer $k \geq 2$
output: k clusters $C_1, C_2, \cdots, C_k$

Select randomly k clusters centers $c_1, c_2, \cdots, c_k$
repeat
  foreach *data point* x *in* S **do**
  forall *cluster centers* c_j, $1 \leq j \leq k$ **do**
    if $\|x - c_j\| < \|x - c_l\|$, $j \neq l$, $1 \leq j, l \leq k$ **then**
      Assign x to C_j
  ;
  Take the center of mass of every cluster as cluster center: $c_j = \frac{1}{n_j} \sum_{x \in C_j} x$, $n_j = |C_j|$
until *The objective function is minimized*;

Where,

- c_j denotes a cluster center, with $c_j \in \mathbb{R}^d$, $1 \leqslant j \leqslant k$ and $k < n$;
- with $C_j \subset S, \cap_{j=1}^{k} C_j = \varnothing$, $\bigcup_{j=1}^{k} C_j = S$, $1 \leqslant j \leqslant k$;

To each cluster is associated a cost $Intra(C_j)$ in Eq. (1).

$$Intra(C_j) = \sum_{x \in C_j} \|x - c_j\|^2. \tag{1}$$

The goal of the k-means algorithm is to partition the set S of n data points into k clusters such that the objective function $F(S, k)$ given in Eq. (2) is minimized.

$$F(S, k) = \sum_{j=1}^{k} \sum_{x \in C_j} \|x - c_j\|^2 \tag{2}$$

From the proof of Theorem 1, we can find a real $\alpha > 0$ such that the problem represented by the minimization of F is reduced to P_2.

2.1. Shortcomings of k-means

Despite its popularity, the k-means algorithm suffers from some problems. The major one is its sensitivity to the initialization. Indeed, the random selection of the k initial cluster centers affects

the quality of clusters obtained. To overcome this problem, in (Zhang et al., 2015) a set of sensor nodes candidates to initial centers are created based on a threshold parameter. Then initial clusters centers are selected in such a way that they are furthest away from each other. The authors in (Ray and De, 2016) proposed a midpoint algorithm for initial clusters centers selection. For every point, it finds the distance to the origin, then partitions the points (sorted by distances) into k groups. The k initial centers are the middle point of their groups.

In order to improve the results of k-means, some researchers used the seeding technique. Particularly in the k-means++, where randomized seeding technique is used as initialization method for the standard k-means (Arthur and Vassilvitskii, 2007). To select the k initial cluster centers, it first chooses a center in a random uniform way. Then, the $k-1$ remaining centers are selected with the probability p (Arthur and Vassilvitskii, 2007).

Reddy et al. (2012) proposed a technique to initialize k-means which uses the Voronoi diagram in order to obtain global optimum results. Some authors proposed deterministic methods, like KKZ and PCA-part, for initializing k-means Su and Dy (2007). The idea behind KKZ is to pay attention to the data points that are most far apart from each other, since those data points are more likely to belong to different clusters (Su and Dy, 2007). The PCA-part starts with one cluster (the cluster with the greatest sum o squared Euclidean distance), and cuts it in half. Then it selects the next cluster to partition, and repeat the process until k are obtained (Su and Dy, 2007).

2.2. Convergence of the k-means clustering algorithm

The k-means clustering algorithm converges. To show that k-means converges, it is sufficient to show that the value of the objective function at time $t+1$ is less or equal than its value at time t. To explain why, consider that if it was true, no partition could be visited twice since the number of possible partitions is finite and is given by the Stirling formula given in Eq. (3).

$$\frac{1}{k!}\sum_{j=0}^{k}(-1)^{k-j}\binom{k}{j}j^n \approx \frac{k^n}{k!}. \tag{3}$$

Let us denote by:

- $F_t(S,k)$ the objective function at time t,
- C_k^t the k^{th} cluster at time t,
- c_k^t the center of the k^{th} cluster at time t,
- $F_{t+1|t}(S,k)$ the objective function at time $t+1$ with clusters C_k^{t+1} and centers c_k^t. That is, some data points have migrated, but the positions of the cluster centers are not yet updated.

We want to show that $F_{t+1}(S,k) < F_t(S,k)$. We first show that $F_{t+1|t}(S,k) < F_t(S,k)$ (see Eq. (A.4)). Since a point moves from a cluster to a new one if and only if its Euclidean distance to the new cluster center is minimal we have the relation given in Eq. (A.4).

Next, we show that $F_{t+1}(S,k) < F_{t+1|t}(S,k)$. From Eq. (10) we get:

$$F_{t+1}(S,k) < F_t(S,k) \tag{4}$$

3. K-clustering problem formulation and fitness function derivation

3.1. Overview of clustering problem

Clustering is the task of grouping objects in such a way that objects in the same group are more similar to each other than to those in other groups. It is applied in Marketing to find groups of customers with similar behavior; in Biology, to classify plants and animals; in Insurance to identify groups of motor insurance policy holders with a high average claim cost and to identify fraud. Moreover, clustering algorithms should satisfy the following requirements (Gupta, 2014; Kononenko and Kukar, 2007): scalability, i.e., they can be applied to large amounts of data; ability to deal with different types of features; ability to discover clusters with arbitrary shapes; minimal requirements for domain knowledge to determine input parameters; ability to deal with noise and outliers; insensitivity to order of input records; high dimensionality, which means if we increase the size of the dimension, the algorithms still can perform; interoperability and usability of the clustering results, i.e., outputs of the algorithms.

In addition, cluster analysis or clustering suffers from numerous problems such as: not addressing all the requirements adequately by the clustering algorithms; the large time complexity of these techniques due to the large number of data points and large dimensions; the effectiveness of these algorithms depending on the definition of the metric; and the result of the clustering algorithm can being interpreted in different ways (Zhang et al., 2017; Rashid, 2011).

We define clustering as the grouping of similar objects, and there are two main approaches to measure the similarity: distance measures and similarity measures. Here we focus on the distance measures to estimate the similarity between data points. By default in k-means, the similarity measure between data points is based on the Euclidean distance or the squared Euclidean distance (*see* Eq. (5)).

$$\|x-y\| = \left(\sum_{i=1}^{d}(x_i-y_i)^2\right)^{\frac{1}{2}} \quad \text{Where } x,y \in \mathbb{R}^d. \tag{5}$$

3.2. Graph theory based k-clustering problem formulation

The problem can be formulated in terms of graph theory where the vertices are defined by the points of the Euclidean space, and the lengths of the edges are given by the distances between points. The vertices can also be considered as data points to cluster and the weights of edges denote the proximity of the points (Ageev et al., 2014). For numerical data, this proximity can be defined by the Euclidean distance. If the numerical data contain a measurement error or perturbation, then the minimum of squared Euclidean distances is usually applied. Thus, the proximity of elements is expressed in terms of the squared Euclidean distance between points (Ageev et al., 2014). The Quadratic Euclidean Max-Cut Problem (QMCP) corresponds to this case. To show that the k-clustering problem is *NP*-hard, we need to show that it is reducible to the Quadratic Euclidean Max-Cut problem (QMCP) which is proven to be *NP*-hard. In (Ageev et al., 2014), it is shown that the QMCP is *NP*-hard by polynomial-time reduction to the Minimum Bisection Problem (MBP). Without loss of generality, we assume that $k=2$, i.e., a 2-clustering problem. To prove the NP-Hardness of the 2-clustering problem, let us generalize the case $k>2$. (Kel'manov and Pyatkin, 2015) studied the *NP*-hardness of 2-clustering techniques with the following objective functions:

- F_1: the sum over all clusters of the sums of squared Euclidean distances between the elements of the cluster.
- F_2: the sum of the sums of squared distances between the elements of the cluster to its centroid (or cluster center).

3.3. Fitness function derivation

Let us give the mathematical formulations of the mentioned objective functions F_1 and F_2 by deriving the fitness function.

3.3.1. Formulation of objective function F_1

Given a set S of n points x in $\mathbb{R}^d$, find a partition of S into two non-empty subsets C_1 and C_2 such that the objective function $F_1(S,2)$ (*see* Eq. (6)) is minimized.

$$F_1(S,2) = \sum_{x\in C_1}\sum_{y\in C_1}\|x-y\|^2 + \sum_{x\in C_2}\sum_{y\in C_2}\|x-y\|^2 \tag{6}$$

3.3.2. Formulation of objective function F_2

Given a set S of n points x in $\mathbb{R}^d$, find a partition of S into two non-empty subsets C_1 and C_2 such that the objective function $F_2(S,2)$ (*see* Eq. (7)) is minimized.

$$F_2(S,2) = n_1 \times \sum_{x\in C_1}\|x-c_k\|^2 + n_2 \times \sum_{y\in C_2}\|y-c_k\|^2 \tag{7}$$

where $n_k = |C_k|$ is the number of elements of C_k, and c_k is the center of mass of C_k.

3.4. NP-hardness of the objective functions

Let us call problems P_1 and P_2 the problems represented by the minimization of F_1 and F_2 respectively. *(N.B: There exists also problems represented to the maximization of F_1 and F_2)*

Definition 1. $\langle x,y\rangle$ is the inner product of x and y.

Theorem 1. *The problems P_1 and P_2 are NP-hard.*

To prove that, let us consider the objective function F_1. Then, we have:

$$\begin{aligned} F_1(S,2) &= \sum_{x\in C_1}\sum_{y\in C_1}\|x-y\|^2 + \sum_{x\in C_2}\sum_{y\in C_2}\|x-y\|^2 \\ &= \underbrace{\sum_{x\in S}\sum_{y\in S}\|x-y\|^2}_{Const} - \underbrace{\sum_{x\in C_1}\sum_{y\in C_2}\|x-y\|^2}_{F_{qmcp}(S,2)} \end{aligned} \tag{8}$$

where *Const* is a constant as the overall sum of squared distance between points of S, and F_{qmcp} is the objective function of the Quadratic Max-Cut Problem.

Therefore the problem P_1 is reduced to the QMCP. To prove the *NP*-hardness of the problem P_2, we use the relation given in Eq. (A.2) for a fixed $z \in S_k$,

$$\sum_{x\in C_k}\|x-z\|^2 = \sum_{x\in C_k}\langle (x-c_k)+(c_k-z),(x-c_k)+(c_k-z)\rangle \tag{9}$$

By applying the linearity property of the inner product we get:

$$\begin{aligned} \sum_{x\in C_k}\|x-z\|^2 &= \sum_{x\in C_k}\|x-c_k\|^2 + \sum_{x\in C_k}\|c_k-z\|^2 \\ &\quad + 2\sum_{x\in C_k}(xc_k - xz - c_kc_k + c_kz) \\ &\Rightarrow \sum_{x_i\in S_k}\|x_i-x\|^2 = \sum_{x_i\in S_k}\|x_i-c_k\|^2 + n_k\|c_k-x\|^2 \end{aligned} \tag{10}$$

where $\sum_{x\in C_k} x = n_k c_k$

We prove the Eq. (10) for every $z \in C_k$ (see Eq. (A.3)), and we get

$$\begin{aligned} \sum_{x,y\in C_k}\|x-y\|^2 &= n_k\left(\sum_{x\in C_k}\|x-c_k\|^2 + \sum_{y\in C_k}\|c_k-y\|^2\right) \\ &\Rightarrow \sum_{x,y\in C_k}\|x-y\|^2 = 2n_k\sum_{x\in C_k}\|x-c_k\|^2 \end{aligned} \tag{11}$$

Now we are ready to prove the *NP*-hardness of the problem P_2.

By applying formula given in Eq. (11) to the objective function F_2, we have $F_2(S,2) = \frac{1}{2}F_1(S,2)$. Therefore we deduce the *NP*-hardness of the problem P_2 from the one of problem P_1.

4. Geometric initialization scheme

We propose a novel method for initializing k-means algorithm. Our method is based on ideas from Voronoi diagram (Reddy et al., 2012), *k*-means++ (Arthur and Vassilvitskii, 2007), the Lloyd's algorithm applied to vector quantization (Lloyd, 1982), and the geometry of a sphere in d dimensional Euclidean space. The purpose of this algorithm is to select the k initial cluster centers, i.e., technicians, in such a way they are as far as possible from each other.

4.1. Geometric initialization in 2-dimensions

Given a set S of n data points (or objects) and k initial cluster centers, firstly we select the first cluster center as the center of mass of all points, and call it c_0. Then, we evaluate the Euclidean distance between c_0 and the furthest point. This distance will be considered as the radius of a circle with center c_0. To select the $(k-1)$ remaining clusters centers, we shall need some geometric considerations. We divide the circle into $(k-1)$ sectors with same size; and place the centers on the lines which split the circle into sectors. Each cluster center, i.e., field technician, is placed at equal distances from the center of mass and the distance between every two consecutive centers is equal. Figs. 1 and 2 give details on the functioning of this model. This initialization technique is summarized in Algorithm 2.

Algorithm 2: Pseudo-code of the initialization scheme

Input: S set of n data points $x = (x_1, x_2)$ and k = number of clusters centers.

Step 1: Select the first cluster center c_0 such that $c_0 = \frac{1}{n}\sum_{x\in S} x$.

Step 2: Find $r := \max_{x\in S}\|x-c_0\|$ such that $(y_1-z_1)^2 + (y_2-z_2)^2 = r^2$, where $c_0 = (z_1,z_2)$ and (y_1,y_2) is an arbitrary point of $\mathbb{R}^2$.

Step 3: Select the $(k-1)$ centers, i.e., all the points of coordinates

$$c_j = (\delta\cos\theta_j,\ \delta\sin\theta_j), \text{ with}$$
$$\delta < \tfrac{r}{2},\quad \theta = \tfrac{2\pi}{k-1},$$
$$(j-1)\times\theta \le \theta_j < j\times\theta,\ \ j = 1,\cdots,k-1.$$

4.2. Geometric initialization in 3-dimensions

Recall S and k, with $S \subset \mathbb{R}^3$. Then, we proceed exactly as in the case of 2-dimensions. Since we are in 3-dimensions, we get a 2-sphere of center c_0 with the Euclidean distance between c_0 and the furthest point of S as radius. To select the $k-1$ remaining centers, we split the 2-sphere into $(k-1)$ spherical wedges (see Fig. 3) of same size; and place each center on (or in) its corresponding spherical wedge.

Algorithm 3: Pseudo-code of the initialization scheme for 3-dimensions

Input: S set of n data points $x \in \mathbb{R}^3$ and k = number of clusters centers.

Step 1: $c_0 = \frac{1}{n}\sum_{x\in S} x$

Step 2: $r := \max_{x\in S}\|x-c_0\|$

Step 3: Select the $(k-1)$ centers, i.e., all the points of coordinates

$$c_j = (\delta\cos(\phi),\ \delta\sin(\phi)\cos(\theta_j),\ \delta\sin(\phi)\sin(\theta_j)),$$

where

$$\delta < \tfrac{r}{2},\quad \theta = \tfrac{2\pi}{k-1},\quad \phi \in [0,\pi],$$
$$(j-1)\times\theta \le \theta_j < j\times\theta,\ \ (j = 1,\cdots,k-1)$$

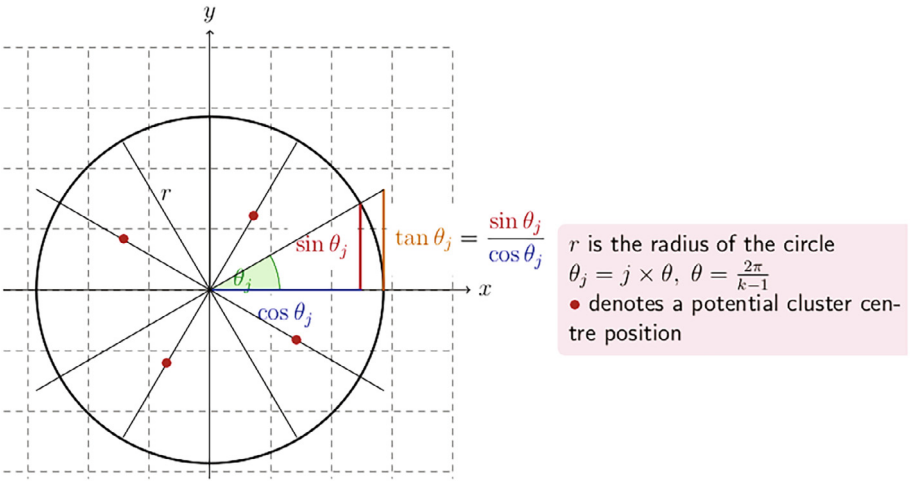

Fig. 1. Geometrical description of our initialization method.

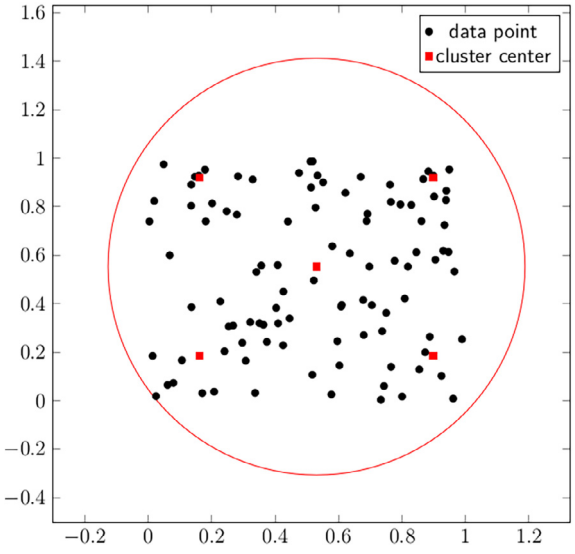

Fig. 2. Illustration of the geometric Initialization.

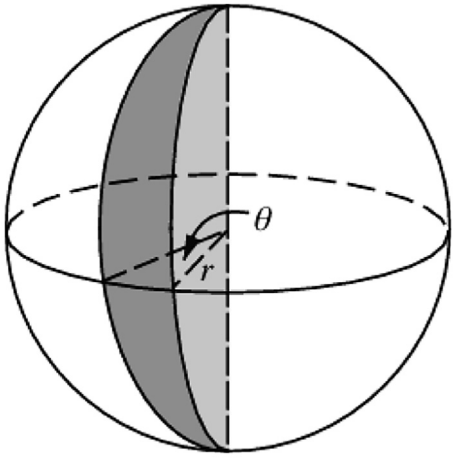

Fig. 3. Spherical wedge of angle θ and radius r.

4.3. Geometric initialization in d-dimensions ($d > 3$)

When $d > 3$ the coordinates of the $k-1$ remaining centers are given below:

$$c_j = \begin{cases} \delta\cos(\phi_1) \\ \delta\sin(\phi_1)\cos(\phi_2) \\ \delta\sin(\phi_1)\sin(\phi_2)\cos(\phi_3) \\ \vdots \\ \delta\sin(\phi_1)\cdots\sin(\phi_{n-2})\cos(\phi_{n-1}) \\ \delta\sin(\phi_1)\cdots\sin(\phi_{n-2})\sin(\phi_{n-1}) \end{cases}$$

where

$\phi_1, \cdots, \phi_{n-2} \in [0, \pi],\quad \phi_{n-1} = \theta_j,$
$(j-1) \times \theta \leqslant \theta_j < j \times \theta,\quad (j = 1, \cdots, k-1)$

5. Performance evaluation

5.1. Evaluation metrics

Results of a clustering algorithm can be interpreted in several ways. However, there exists formal methods to analyze the results of clustering. *Cluster validity* is a process of evaluating clustering results. In general, we distinguish the internal measures from the external measures of cluster validity (Zhao, 2012). Many internal measures of cluster validity for clustering are based on the notions of cohesion, separation (Tan et al., 2005), or the inter cluster. In this work, we measure the performance of our algorithm based on the following criteria:

- **Inter cluster**: Also called Sum of Squared Error (SSE); it is the objective function that we try to minimize in this paper.
- **Speed of convergence**: Here we consider the running time of the algorithm.

5.2. Data set and algorithms implementation

We use the sample of data provided by OpenCellID to compare our proposed scheme. OpenCellID (by Unwired Labs) is a collaborative community project that collects GPS positions of cell towers and their corresponding location area identity (Wikipedia contributors, 2018). In OpenCellID database, a cell tower is defined

Table 1
Attributes of cell tower required for the performance evaluation.

Attribute	Description
Lon	Longitude
Lat	Latitude
Range	Approximate area within which the cell could be

by the following attributes: Radio, mcc, mnc, lac/tac/nid, cid, longitude, latitude, range, samples, changeable, created, updated, averagesignal (OpenCellID contributors, 2018). However, we will perform our analysis based on three attributes (see Table 1).

To evaluate the performance of our initialization scheme, we compare it to classical k-means and k-means++. We implemented these algorithms in Python. However, code sources of k-means and k-means++ were taken from (The Data Science Lab, 2014); then, adapted according to the data sets.

5.3. Results

5.3.1. Inter cluster (or SSE)

Table 2 and Fig. 4 show the inter cluster obtained when we applied k-means, k-means++ and geokmeans on 2000 cell towers from OpenCellID. Fig. 4 shows the average SSE obtained when we applied these algorithms. We ran these algorithms 20 times. Since the goal is to minimized the inter cluster, hence the smallest values are desired. As we can observe on Fig. 4, the values of inter cluster obtained with geokmeans are the smallest. However, as we can observe in the figure, when the number of data points is 10 and 80, the SSE of geokmeans is not the smallest. This fact is explained by the fact that the performance of our initialization scheme, i.e., geokmeans, is strongly based on an optimal choice of δ. However, by means of mathematical drawings and simulations, we realized that for geokmeans to tend to optimal solutions, the value of should be lower than the half of the radius $(\delta < \frac{r}{2})$. During the simulations phase, for each instance of k, we ran geokmeans several times with different values of $\delta < 2$; then we plot the statistical mean of the SSE as shown in Fig. 4. Due to the fact that we are still looking for the optimal value of δ, it may happen that the statistical mean of the SSE values is not the smallest. Therefore, we can conclude that our initialization method outperfoms random initialization and the seeding technique of k-means++ in term of inter cluster.

5.3.2. Speed of convergence

In this section, we compare the various algorithms based on their running time. From Table 3 we can observe that all the algorithms are competitive. Nevertheless, we claim that k-means

Table 2
Comparison of SSE, with n = 2000 cell towers.

k	k-means			k-means++			geokmeans		
	max	ave	min	max	ave	min	max	ave	min
10	3,16	2,45	1,41	3,10	2,19	1,03	3,17	2,34	1,68
20	6,38	3,87	1,51	5,55	3,99	2,05	5,00	3,19	1,18
30	8,75	5,22	1,68	9,01	5,73	1,58	6,85	4,23	1,28
40	9,57	6,40	1,98	9,66	6,26	1,51	9,87	5,19	2,31
50	12,46	7,41	3,38	13,93	8,15	3,64	12,2	5,33	1,54
60	17,19	10,38	2,49	16,73	13,08	10,28	13,73	8,43	2,45
70	17,52	9,46	2,15	19,38	11,99	3,77	14,52	7,31	3,25
80	18,98	10,08	5,05	19,89	14,14	8,49	20,34	12,39	6,67
90	18,68	11,46	3,95	17,35	10,19	4,40	22,15	9,42	3,24
100	24,20	11,80	4,62	24,67	12,97	2,69	18,20	7,49	2,76

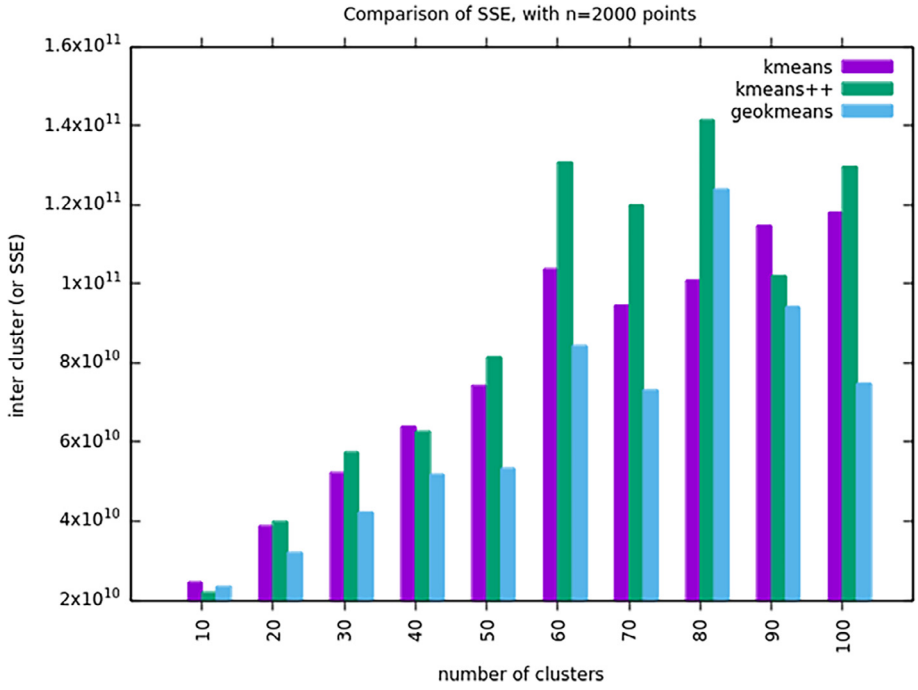

Fig. 4. Comparison of the average of final SSE.

Table 3
Comparison of running time, with n = 2000 cell towers.

k	k-means			k-means++			geokmeans		
	max	ave	min	max	ave	min	max	ave	min
10	10,59	7,00	3,20	9,91	8,03	3,43	10,78	6,68	3,52
20	21,45	16,93	13,93	20,81	15,51	12,37	23,24	18,91	12,23
30	44,17	24,83	9,51	52,98	27,54	16,97	44,02	23,46	11,99
40	59,89	43,15	17,36	65,85	39,08	20,98	61,56	44,01	15,16
50	87,10	45,00	21,28	71,76	49,16	12,04	82,59	47,06	27,00
60	79,64	42,52	19,16	61,01	33,40	12,58	110,00	43,21	13,22
70	85,85	41,11	19,47	92,86	38,49	23,36	89,75	41,02	13,41
80	34,09	21,79	9,10	36,44	25,27	15,06	82,59	36,74	18,52
90	67,09	29,49	12,75	44,07	25,53	12,27	58,19	26,64	12,39
100	43,65	25,20	13,45	57,43	27,79	12,40	39,46	22,85	11,40

with our initialization scheme is faster than k-means++. Unlike k-means++, the selection of the i_{th} center does not depend on the $(i-1)_{th}$ center. After the selection of the first cluster center c_0, the $(k-1)$ remaining clusters centers are selected simultaneously. However, during the implementation of geometric initialization, these remaining clusters centers are selected sequentially.

6. Conclusion

Enhancing the cell sites maintenance operations remains an active issue faced by telecommunication operators. In rural areas, the dispersion of the towers leads to a non evident assignment of field technicians to a group of cell site for maintenance interventions. An optimal clustering of field technicians and cell sites has a strong influence on mobile network availability as well as human resource management. In this paper, we investigated the problem of clustering mobile network cell sites in rural areas. To achieve that we implemented the k-means clustering algorithm with random initialization to geometric initialization, with a view to optimizing costs of maintenance operations on telecommunications cell sites. The k-clustering is formulated as a graph theory problem. For optimization purpose, we adopted the k-means algorithm for partitioning a set of data points into a number of clusters. A geometric initialization based on the geometry of a sphere has been proposed. Intensive simulations were made to compare algorithms with a sample of thousands of cell towers provided by OpenCellID, into 3-dimensional Euclidean space, in terms of inter cluster and speed of convergence. The results showed that k means with geometric initialization outperforms the classical k-means and k-means++. As prospect, we intend to redefine the similarity measure of k-means in such a way that it can be adapted to this optimization problem.

Acknowledgement

We like to thank the editor and the anonymous reviewers for their valuable remarks that helped us in better improving the content and presentation of the paper. We also like to thank AIMS-Cameroon for the support of some of its assistant teacher, especially Mrs. Nathalie WANDJI for her significant support that helped us to improve the work achieved in this paper.

Appendix A

A.1. Quadratic Euclidean Max-Cut Problem

Given a set $S = \{x_1, \cdots, x_n\}$ of points of $\mathbb{R}^N$, the QMCP is to find a partition of S into two subsets S_1 and S_2 such that $F_{qmcp}(S,2)$ (*see* Eq. (A.1)) is maximized (Ageev et al., 2014).

$$F_{qmcp}(S,2) = \sum_{x_i \in S_1} \sum_{x_j \in S_2} d(x_i, x_j)^2 \tag{A.1}$$

where $d(x_i,x_j) = \|x_i - x_j\|_2 \Rightarrow d(x_i,x_j)^2 = \|x_i - x_j\|_2^2$

A.2. Minimum Bisection Problem (MBP)

Given a graph $G = (V,E)$ with $|V| = n$ vertices. The minimum bisection problem seeks to partition the set of vertices V into two disjoint subsets V_1 and V_2 of the same size such that the cost is minimized (Cygan et al., 2014). The cost of a bisection is the number of edges (e_i, e_j), $i \neq j$ such that $e_i \in V_1, e_j \in V_2$. As a graph-partitioning problem, the MBP is an *NP*-hard problem (Cygan et al., 2014). Its time complexity is $O\left(2^{O(k^3)} n^3 \log^3 n\right)$ and it can be solved in polynomial time for $k = O\left(\sqrt[3]{\log n}\right)$ (Cygan et al., 2014).

A.3. Linearity property of inner product

$$\begin{aligned}
&\langle (x_i - c_k) + (c_k - x), (x_i - c_k) + (c_k - x)\rangle \\
&= \langle x_i - c_k, x_i - c_k\rangle + \langle x_i - c_k, c_k - x\rangle + \langle c_k - x, x_i - c_k\rangle \\
&\quad + \langle c_k - x, c_k - x\rangle \\
&= \|x_i - c_k\|^2 + \|c_k - x\|^2 + 2\langle c_k - x, x_i - c_k\rangle \\
&= \|x_i - c_k\|^2 + \|c_k - x\|^2 + 2(c_k - x) \\
&\quad \cdot (x_i - c_k)\langle (x_i - c_k) + (c_k - x), (x_i - c_k) + (c_k - x)\rangle \\
&= \|x_i - c_k\|^2 + \|c_k - x\|^2 + 2(x_i c_k - x_i x - c_k c_k + c_k x)
\end{aligned} \tag{A.2}$$

$$\begin{aligned}
&\sum_{x \in C_k} \|x - y\|^2 = \sum_{x \in C_k} \|x - c_k\|^2 + n_k \|c_k - y\|^2 \\
&\sum_{x \in C_k} \|x - w\|^2 = \sum_{x \in C_k} \|x - c_k\|^2 + n_k \|c_k - w\|^2 \\
&\qquad \cdots \\
&\sum_{x \in C_k} \|x - t\|^2 = \sum_{x \in C_k} \|x - c_k\|^2 + n_k \|c_k - t\|^2
\end{aligned} \tag{A.3}$$

$$\begin{aligned}
F_{t+1|t}(S,k) &= \sum_{j=0}^{k} \sum_{x \in C_j^{t+1}} \|x - c_j^t\|^2 \\
&\leqslant \sum_{j=0}^{k} \sum_{x \in C_j^t} \|x -, c_j^t\|^2 = F_t(S,k)
\end{aligned} \tag{A.4}$$

$$\sum_{x\in C_j^{t+1}} \|x-c_k^t)\|2 > \sum_{x\in C_j^{t+1}} \|x-c_k^{t+1}\|^2, \Rightarrow \sum_{j=0}^{k}\sum_{x\in C_j^{t+1}} \|x-c_k^t\|^2$$
$$> \sum_{j=0}^{k}\sum_{x\in C_j^{t+1}} \|x-c_k^{t+1}\|^2, \text{ i.e } F_{t+1|t}(S,k)$$
$$> F_{t+1}(S,k). \tag{A.5}$$

References

Ageev, A., Kel'manov, A., Pyatkin, A., 2014. NP-hardness of the euclidean max-cut problem. Doklady Math. Springer, 343–345.

Ari, A.A.A., Labraoui, N., Yenke, B.O., Gueroui, A., 2018. Clustering algorithm for wireless sensor networks: the honeybee swarms nest-sites selection process based approach. Int. J. Sens. Netw. 27 (1), 1–13.

Ari, A.A.A., Yenke, B.O., Labraoui, N., Damakoa, I., Gueroui, A., 2016. A power efficient cluster-based routing algorithm for wireless sensor networks: honeybees swarm intelligence based approach. J. Netw. Comput. Appl. 69, 77–97.

Arthur, D., Vassilvitskii, S., 2007. The advantages of careful seeding. In: Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms. Society for Industrial and Applied Mathematics, pp. 1027–1035.

Boniecki, D., Marcati, C., Abou-Zahr, W., Alatovic, T., El Hamamsy, O., 2016. Middle east and africa: Telecommunications industry at cliff's edge. time for bold decisions. TMT Practice 1, 1–64.

Celebi, M.E., Kingravi, H.A., Vela, P.A., 2013. A comparative study of efficient initialization methods for the k-means clustering algorithm. Exp. Syst. Appl. 40 (1), 200–210.

Cygan, M., Lokshtanov, D., Pilipczuk, M., Pilipczuk, M., Saurabh, S., 2014. Minimum bisection is fixed parameter tractable. In: Proceedings of the forty-sixth annual ACM symposium on Theory of computing. ACM, pp. 323–332.

Dietrich, N., 2016. Tower companies and intelligent site asset management. IMQS www.imqs.co.za 1, 1–2.

Gupta, G., 2014. Introduction to data mining with case studies. PHI Learning Pvt, Ltd.

Kel'manov, A., Pyatkin, A., 2015. NP-hardness of some quadratic euclidean 2-clustering problems. Doklady Math. Springer, 634–637.

Kononenko, I., Kukar, M., 2007. Machine Learning and Data Mining: Introduction to Principles and Algorithms. Horwood Publishing.

Lloyd, S., 1982. Least squares quantization in pcm. IEEE Trans. Inf. Theory 28 (2), 129–137.

Moussaoui, O., Ksentini, A., Naimi, M., Gueroui, M., 2006. A novel clustering algorithm for efficient energy saving in wireless sensor networks, 2006 International Symposium on Computer Networks. IEEE, pp. 66–72.

OpenCellID contributors, 2018. OpenCellID contributors. http://wiki.opencellid.org/wiki/View the data, [Online; accessed 23-May-2018].

Rahman, M.A., Islam, M.Z., Bossomaier, T., 2015. Modex and seed-detective: two novel techniques for high quality clustering by using good initial seeds in k-means. J. King Saud Univ.-Comput. Inf. Sci. 27 (2), 113–128.

Rashid, T., 2011. Clustering.

Ray, A., De, D., 2016. Energy efficient clustering protocol based on k-means (eecpk-means)-midpoint algorithm for enhanced network lifetime in wireless sensor network. IET Wireless Sens. Syst. 6 (6), 181–191.

Reddy, D., Jana, P.K., et al., 2012. Initialization for k-means clustering using voronoi diagram. Procedia Technol. 4, 395–400.

Steinley, D., 2006. K-means clustering: a half-century synthesis. Br. J. Math. Stat. Psychol. 59 (1), 1–34.

Su, T., Dy, J.G., 2007. In search of deterministic methods for initializing k-means and gaussian mixture clustering. Intell. Data Anal. 11 (4), 319–338.

Tan, P.-N., Steinbach, M., Kumar, V., 2005. Cluster analysis: basic concepts and algorithms. In: Introduction to Data Mining. Pearson International Edition. Pearson Addison Wesley, Boston, pp. 487–568.

The Data Science Lab, 2014. Improved seeding for clustering with k-means++. https://datasciencelab.wordpress.com/2014/01/15/improved-seeding-for-clustering-with-k-means/, [Online; accessed 28-May-2018].

Titouna, C., Ari, A.A.A., Moumen, H., 2018. Fdra: Fault detection and recovery algorithm for wireless sensor networks. In: Younas, M., Awan, I., Ghinea, G., Catalan Cid, M. (Eds.), Mobile Web and Intelligent Information Systems. MobiWIS. Lecture Notes in Computer Science, Vol. 10995. Springer, Cham, pp. 72–85.

Wikipedia contributors, 2018. Opencellid — Wikipedia, the free encyclopedia. https://en.wikipedia.org/w/index.php?title=OpenCellID&oldid=840269231, [Online; accessed 23-May-2018].

Wu, J., 2012. Cluster analysis and k-means clustering: an introduction. In: Advances in K-means Clustering. Springer, pp. 1–16.

Zhang, G.-Y., Wang, C.-D., Huang, D., Zheng, W.-S., 2017. Multi-view collaborative locally adaptive clustering with minkowski metric. Expert Syst. Appl. 86, 307–320.

Zhang, Y., Sun, H., Yu, J., 2015. Clustered routing protocol based on improved k-means algorithm for underwater wireless sensor networks, 2015 IEEE International Conference on Cyber Technology in Automation, Control, and Intelligent Systems (CYBER). IEEE, pp. 1304–1309.

Zhao, Q., 2012. Cluster Validity in Clustering Methods. Publications of the University of Eastern Finland.