DEPARTMENT OF INFORMATION TECHNOLOGY, NITK SURATHKAL
IT 301 PARALLEL COMPUTING
PC LAB 7
Date: 30 September 2020
Submission Date : 5th September 2020

Note:
Implement the following using OpenMP.
Every component in each lab carries marks [10 Marks]
For final grading 10% is based on regular lab evaluation.

Consider random deployment of sensor nodes in field to sense the environment. The nodes are deployed randomly and the position of each sensor node is sent to centralised server. The server would like to cluster these nodes. Use K-means algorithm to cluster the nodes. Write an OpenMP program to cluster the sensor nodes and compare the result with sequential and parallel approach.
For implementation consider the following things.

1. Assume 1000 sensor nodes are deployed in 1000m x 1000m area. Generate the position of each node using random function. **[1 Mark]**
2. Justify in what way the parallel approach used by you is better for implementation (eg task, section, for etc) **[2 Marks]**
3. Implement the algorithm to make 2 clusters, 4 clusters and 8 clusters. Compare the result with sequential algorithm. **[3 Marks]**
4. Using some graphical tools, plot the clusters and positions of each node. **[1 Mark]**
5. Program Code and results **[3 Marks]**

Step 1:	Accept the number of clusters to group data into and the dataset to cluster as input values
Step 2:	Initialize the first K clusters - Take first k instances or - Take Random sampling of k elements
Step 3:	Calculate the arithmetic means of each cluster formed in the dataset.
Step 4:	K-means assigns each record in the dataset to only one of the initial clusters - Each record is assigned to the nearest cluster using a measure of distance (e.g Euclidean distance).
Step 5:	K-means re-assigns each record in the dataset to the most similar cluster and re-calculates the arithmetic mean of all the clusters in the dataset.

Fig 1: Generalised Pseudocode of Traditional k-means