# Hierarchical Clustering based Spatial Sampling of Particulate Matter Nodes in IoT Network

by

rajashekar.reddy , Sachin Chaudhari

Report No: IIIT/TR/2021/-1

Centre for Communications
International Institute of Information Technology
Hyderabad - 500 032, INDIA
August 2021

# Hierarchical Clustering based Spatial Sampling of Particulate Matter Nodes in IoT Network

C. Rajashekar Reddy, S. Chaudhari
International Institute of Information Technology-Hyderabad (IIIT-H), India
Emails: rajashekar.reddy@research.iiit.ac.in, sachin.chaudhari@iiit.ac.in

***Abstract*—For understanding an environmental variable in a given geographical space, finding the optimal number of nodes is a tedious task. For this purpose, a framework is proposed in this paper based on hierarchical agglomerative clustering along with geographical distance based cluster representation. The proposed framework helps remove the redundant nodes in a practical IoT network by choosing the optimal nodes based on the target reconstruction error in the spatially interpolated map. The approach is employed on the data collected by an IoT network of ten particulate matter (PM) nodes on the campus of IIIT Hyderabad, India. The performance of the proposed approach is also compared with that of the brute force approach, which provides the lower bound on the reconstruction error. The results show that the proposed approach performs very closely to the brute force approach in terms of the reconstruction error with much fewer computations.**

***Index Terms*—Clustering Analysis, Cluster Representative, Optimal Nodes, Particulate Matter, Spatial Sampling**

## I. Introduction

IoT based monitoring networks help us understand the spatio-temporal variation of the parameters of interest. The environmental parameters like particulate matter (PM) vary on a small spatio-temporal scale from the emission site, as shown in [1] and [2]. Deciding how fine this spatio-temporal scale is required for a tolerable error in the spatially interpolated map is a challenging task. On the one hand, if the monitoring sensors are sparsely placed, it would not give a good spatio-temporal understanding of the measured parameter. On the other hand, if the monitoring sensors are large in number, it will increase the volume of data transmitted and the processing capability required at the sink or cloud. In most sensor deployment cases, the number of nodes is determined by intuition or resource constraints or domain knowledge or any evidence-based understanding of the environment.

Previous works on determining the number of optimal nodes in a sensor network vary from using analytical approaches to simulation-based and geometric-based approaches [3]–[6]. Mathematically, environmental phenomena in a space can be modeled as a spatio-temporal random field. Now the problem becomes sampling the assumed random field. In [3] and [4], the random field is modeled as a Gaussian process, and the sampling strategies are discussed accordingly. Within a spatial setting, we use geometrical arrangements like Voronoi tessellation [5] for deciding the optimal location of nodes. In [6], the Monte-Carlo simulation-based approach is used to decide the sensor nodes. However, there are issues with these approaches. None of the approaches are data-centric and do not use an already existing sensor network. In [3] and [4], the main issue is how the approaches depend on the nature of the process. The issue with the approach in [5] is that the environmental parameters are not always in convenient geometric arrangements. The approach in [6] needs many computations and iterations. Also, there is a dearth of a complete end-to-end framework that includes clustering the nodes and choosing cluster representative (removing redundant nodes) followed by performance evaluation using spatial interpolation, which is the focus of this paper.

Specific contributions of this paper are:

- A framework is proposed based on hierarchical agglomerative clustering and geographical distance based cluster representation for selecting the optimal number of sensor nodes in an IoT network. The proposed approach trades-off the number of nodes required in an IoT network with the desired reconstruction error in the spatial interpolated map.
- The framework is employed on the IoT network of ten PM nodes in the IIIT-H campus for different cases of linkages and distance metrics. Spatial interpolation is used to improve the spatial resolution with the obtained number of sensors at each hierarchy.
- The proposed approach is compared with the brute force approach in terms of root mean square error (RMSE) after the reconstruction. The variability in the spatial data with the change in the number of nodes considered is presented to visualize the trade-off between the reconstruction accuracy and the nodes used.

Unlike the clustering approaches in [3]–[6], which are not data-centric, hierarchical agglomerative clustering is a data-driven approach with no initial assumptions. It does not dependent on geographical arrangements and allows us to decide the number of clusters required based on an acceptable error threshold. It works well when the availability of data points is less [7]. Also, note that none of the approaches [3]–[6] use an already existing sensor network as is done in this work.

The rest of the paper is organized as follows. Section II presents the details of IoT network deployment followed by basic data processing. Section III presents the proposed framework in detail. Section IV present the results while Section V concludes the paper.


Fig. 1. Deployment and Node Locations

## II. Methodology

### A. Sensor Network

Fig. 1 shows the IoT network for PM monitoring considered in this paper. Ten nodes are deployed in the IIIT-H campus, Hyderabad, India with area of 66 acres (0.267 $\mathrm{km}^2$). The sensor nodes are developed at IIIT-H using ESP8266 based NodeMCU microcontroller, SDS011 PM sensor, and DHT22 sensor for temperature and relative humidity as shown in [1]. SDS011 PM sensor gives the values of PM2.5 (particulate matter with aerodynamic diameter 2.5 micrometers and smaller) and PM10 (particulate matter with aerodynamic diameter 10 micrometers and smaller). The microcontroller samples the data at an interval of 15 seconds and sends it periodically via WiFi to *ThingSpeak* [8], which is a cloud-based IoT platform for storing and processing data using MATLAB.

### B. Data cleaning and Preprocessing

The following tasks are done to convert the raw data from the ten sensor nodes into a usable dataset:

- The fiirst task is to remove outliers or extreme values, which may otherwise further processing. The outlier removal is done using a clustering-based unsupervised technique, a density-based clustering algorithm (DBSCAN) [9].
- The second task is to average data so that to observe the significant trend in the dataset containing random fluctuation. Since a single dataset with the same timestamps is required, the data points are averaged into a single timestamp within every minute timeframe giving a dataset with 1 minute sampled data points.
- The third task is to remove unreliable sensor values. The sensor SDS011 used for measuring PM2.5 and PM10 is affected by high relative humidity values. So, the data points at which the relative humidity RH $\geq$ 75% are dropped.
- The fourth task is to keep data points only for those time instances, where data is available for all sensors simultaneously, which is necessary for the problem considered in this paper. The ten sensor nodes did not work simultaneously in several instances due to different failures related to network, power supply connection, sensors or micro-controller. After considering the instances where ten simultaneous values for PM2.5 and PM10 are available, the dataset is reduced to 9,737 data points denoted by $N$ and each node's data as $\mathbf{x}_i$ of length $1 \times N$ for PM2.5 and PM10 separately. The data matrix $\mathbf{X}$ of dimensions $10 \times N$ is formed by stacking $\mathbf{x}_i$ as rows separately for PM2.5 and PM10.

## III. Proposed Framework

Fig. 2 shows the framework proposed. The framework includes three sections, hierarchical agglomerative clustering, cluster representative selection, and spatial interpolation and performance evaluation. The initial step is to perform hierarchical agglomerative clustering on the data matrices of PM2.5 and PM10. Here, cophenetic correlation is used to select the best clustering solution. The dendrogram and cophenetic distances corresponding to the chosen solution and the priority order calculated using the geographical distance matrix are used in the next step for cluster representative selection. For performance evaluation, spatial interpolation is done using the selected nodes from the previous step, and reconstruction error is calculated.

### A. Hierarchical Agglomerative Clustering

Hierarchical clustering offers a flexible and non-parametric approach to cluster data [7]. It presupposes very little in the way of data characteristics or prior knowledge on the part of the analyst. The primary motivation for using hierarchical clustering is its ability to partition the data, which has its corresponding hierarchy. The agglomerative approach suits our problem of identifying similar nodes as it gives a better understanding of the hierarchy of grouping the nodes. It considers each point as a partition of equal hierarchy and then moves forward with joining the least dissimilar clusters. We use a distance metric to determine the least dissimilar two points or clusters and group them as a cluster at the following hierarchy using specific linkage criteria. The hierarchy of clusters is expressed in the form of a dendrogram [7] for graphical representation. The quality of the solution is measured using cophenetic correlation [10]. This measure can be used to compare alternative cluster solutions obtained using different algorithms. The data matrices of PM2.5 and PM10 are the inputs to this stage. The above steps are explained in detail below:

*1) Distance or Dissimilarity:* To group data, we need a measure for quantifying the similarity relative to each other. Many times a dissimilarity measure also called a distance measure, is defined, which is minimized. We calculate the dissimilarity matrix for the input data X as $\mathbf{D}_{i,j} = \mathrm{dist}(\mathbf{x}_i, \mathbf{x}_j)$ where $\mathbf{x}_i$ and $\mathbf{x}_j$ denote the the $i$th and $j$th rows of the matrix $\mathbf{X}$ and dist is the dissimilarity measures used. The dissimilarity measure used in this paper are

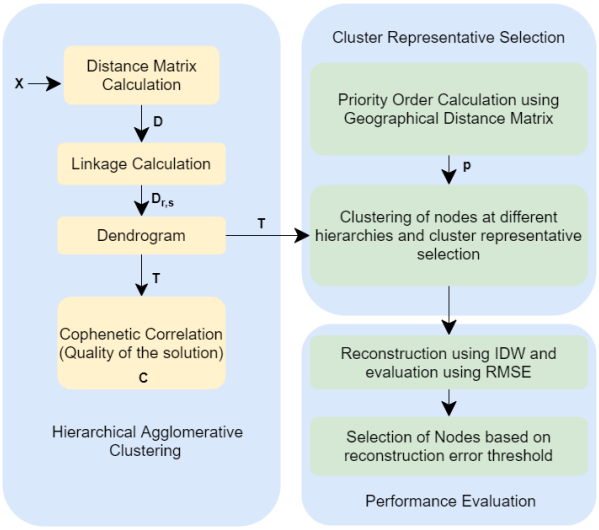



Fig. 2. Hierarchical Agglomerative Clustering based Framework

- Euclidean distance: The Euclidean distance measure is the most commonly used distance metric. For linkage criteria like Ward's method, centroid linkage, and median linkage, Euclidean distance is the only appropriate one. The Euclidean distance is given as

$$\text{dist}(x_i, x_j) = \sqrt{\sum_{m=1}^{N} (\mathbf{x}_{i,m} - \mathbf{x}_{j,m})^2},$$

where $\mathbf{x}_{i,m}$ and $\mathbf{x}_{j,m}$ denotes the $m$th data point in $\mathbf{x}_i$ and $\mathbf{x}_j$ respectively.

- Correlation coefficient based dissimilarity: Correlation coefficient is a statistical tool that quantifies the similarity between the variables compared in pairs. In this paper, Pearson's $r$ is used as a correlation coefficient and is given as

$$r(\mathbf{x}_i, \mathbf{x}_j) = \frac{\sum_{m=1}^{N}(\mathbf{x}_{i,m} - \bar{\mathbf{x}_i})(\mathbf{x}_{j,m} - \bar{\mathbf{x}_j})}{\sqrt{\sum_{m=1}^{N}(\mathbf{x}_{i,m} - \bar{\mathbf{x}_i})^2}\sqrt{\sum_{m=1}^{N}(\mathbf{x}_{j,m} - \bar{\mathbf{x}_j})^2}}.$$

In terms of strength, it varies from -1 to 1. To obtain the dissimilarity, we consider the subtraction of the correlation coefficient from 1.

$$\text{dist}(\mathbf{x}_i, \mathbf{x}_j) = 1 - r(\mathbf{x}_i, \mathbf{x}_j).$$

*2) Linkage:* The dissimilarity matrix $\mathbf{D}$ is the input to this stage. A linkage, in simple terms, is the distance between two clusters. The distance matrix is updated according to the linkage criteria used after grouping two clusters, and only a single row represents the two objects grouped. For example, if a cluster $\mathcal{R}$ is made by grouping $\mathcal{P}$ and $\mathcal{Q}$, the values associated with $\mathcal{P}$ and $\mathcal{Q}$ in the distance matrix $\mathbf{D}$ are dropped. Values calculated for the remaining clusters for $\mathcal{R}$ using the linkage criteria are added. For the following explanation, $\mathcal{R}$ and $\mathcal{S}$ are the clusters considered, $n_r$ and $n_s$ are the number of objects in $\mathcal{R}$ and $\mathcal{S}$, respectively and $\mathbf{x}_{\mathcal{R},i}$ and $\mathbf{x}_{\mathcal{S},i}$ are the $i$th objects in clusters $\mathcal{R}$ and $\mathcal{S}$, respectively.

- Single Linkage: Single linkage, which is similar to the nearest neighbor, uses the smallest distance between the objects in the clusters considered and is calculated as

$$\mathbf{D}_{r,s} = \min(\text{dist}(\mathbf{x}_{\mathcal{R},i}, \mathbf{x}_{\mathcal{S},j})), i \in (1...n_r), j \in (1...n_s).$$

- Complete Linkage: Complete linkage, which is also known as farthest neighbor, uses the largest distance between the objects in the clusters considered and is calculated as

$$\mathbf{D}_{r,s} = \max(\text{dist}(\mathbf{x}_{\mathcal{R},i}, \mathbf{x}_{\mathcal{S},j})), i \in (1...n_r), j \in (1...n_s).$$

- Average Linkage: Average linkage uses the average distance between all pairs of objects in the clusters considered and calculated as

$$\mathbf{D}_{r,s} = \frac{1}{n_r n_s}\sum_{i=1}^{n_r}\sum_{j=1}^{n_s}(\text{dist}(\mathbf{x}_{\mathcal{R},i}, \mathbf{x}_{\mathcal{S},j})).$$

- Ward's Method: This method uses the increase in the total within-cluster sum of squares resulting from joining two clusters. The within-cluster sum of squares is defined as the sum of the squares of the distances between all objects in the cluster and the cluster's centroid. The sum of squares metric is equivalent to the following metric.

$$\mathbf{D}_{r,s} = \sqrt{\frac{2n_r n_s}{n_r + n_s}}\|\mathbf{x}_{\mathcal{R}}{}' - \mathbf{x}_{\mathcal{S}}{}'\|_2,$$

where $\|.\|_2$ represents the Euclidean distance and $\mathbf{x}_{\mathcal{R}}{}'$ and $\mathbf{x}_{\mathcal{S}}{}'$ represent the centroids of $\mathbf{x}_{\mathcal{R}}$ and $\mathbf{x}_{\mathcal{S}}$

*3) Dendrogram:* A dendrogram [7] is the graphical representation of the hierarchical relationship between the objects to work out the best way to allocate them to the clusters. The dendrogram is the final output of the agglomerative clustering stage. The key to interpreting a dendrogram is to focus on the height at which any two objects are joined. The height of the dendrogram indicates the order in which the clusters are joined. This height is used to calculate the cophenetic distance which is used for calculating cophenetic correlation. The cophenetic distance between two objects is the height of the dendrogram where the two branches that include the two objects merge into a single branch. The `dendrogram` function from [11] has been used in this paper.

*4) Cophenetic Correlation:* The cophenetic correlation gives the quality of the clustering solution and is calculated using the dendrogram, which gives the cophenetic distances. The cophenetic correlation for a tree is the linear correlation coefficient between the cophenetic distances and the original dissimilarities used to construct the tree. It measures the tree's quality and how faithfully the dendrogram represents the dissimilarities between the observations. The cophenetic distance in a dendrogram is defined as the height of the link at which those two observations are first joined, which is the distance between the two sub-clusters merged by that link. The magnitude of cophenetic correlation $c$ is very close to 1 for a high-quality solution. Suppose the objects in the dendrogram, which is the simplified model in which the close data is

grouped, is given by $\mathbf{T}_i$ for the $i$th object. The cophenetic correlation is defined as

$$c = \frac{\sum_{i<j}(\mathbf{D}_{i,j} - \bar{\mathbf{D}})(\mathbf{T}_{i,j} - \bar{\mathbf{T}})}{\sqrt{\sum_{i<j}(\mathbf{D}_{i,j} - \bar{\mathbf{D}})^2}\sqrt{\sum_{i<j}(\mathbf{T}_{i,j} - \bar{\mathbf{T}})^2}},$$

where $\mathbf{D}_{i,j}$ is the distance between $i$ th and $j$ th objects obtained from the dissimilarity matrix $\mathbf{D}$ calculated earlier, $\mathbf{T}_{i,j}$ is the cophenetic distance between the objects $\mathbf{T}_i$ and $\mathbf{T}_j$ in the dendrogram and $\bar{\mathbf{D}}$ and $\bar{\mathbf{T}}$ are the averages of $\mathbf{D}$ and cophenetic distances between the objects. Cophenetic distance $\mathbf{T}_{i,j}$ is the height of the node at which the two points $\mathbf{T}_i$ and $\mathbf{T}_j$ are first joined together in the dendrogram.

### B. Cluster Representative Selection

The agglomerative clustering used starts with the base case of considering ten sensor nodes as a singleton cluster, which is the actual field deployment. Each cluster, which might be a singleton or group of nodes, is again grouped into another cluster after each stage. For selecting a representative sensor node for each cluster, we employ a priority order based on the geographical distance between nodes to be calculated while deploying the IoT network as shown in the Algorithm 1. The geographical distance matrix is represented by $\mathbf{G}$ and $\mathbf{G}_{i,j}$ which is the $(i, j)$ entry is the geographical distance between $i$th and $j$th node. $\mathbf{G}$ is used to calculate the priority order $\mathbf{p}$.

**Algorithm 1** Priority Order

```
procedure PRIORITYORDER(G)
    p={i,j}                          ▷ where G_{i,j} = max(G)
    num=2
    while num <= 10 do
        k=index with max value of Σ_{l∈p} G_l
        p.append(k)
        num++
    return p
```

### C. Spatial Interpolation

Spatial interpolation is used to reconstruct the data and calculate the reconstruction error in terms of RMSE with every decrease in the number of sensor nodes. In this paper, inverse distance weighing (IDW) is used, which is one of the simplest and popular deterministic spatial interpolation techniques with reasonable accuracy for estimating the geographically varying PM values [12]. IDW follows the principle that the nodes that are closer to the location of estimation will have more impact than the ones which are farther away [13]. IDW uses a linearly weighted combination of the measured values at the nodes to estimate the parameter at the location of interest. The weight corresponding to a node is a function of the inverse distance between the node's location and the location of the estimate. In this paper, weights are chosen to be inverse distance.

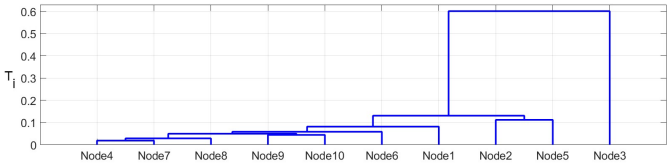
## IV. Results and Analysis

The framework mentioned above is applied to clean and pre-processed datasets. The results section is organized as follows, IV-A explains the dendrogram obtained for the data and quality of the solution using the cophenetic correlation values. IV-B deals with the geographical distance-based priority order and the clustering of nodes based on heights in the corresponding dendrogram IV-C present the results for spatial interpolation and reconstruction errors.

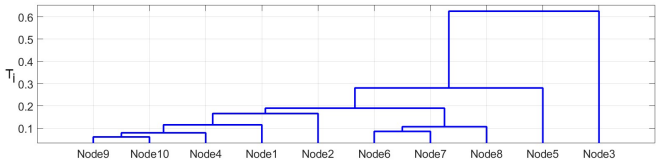
### A. Dendrogram and Cophenetic Correlation

The graphical representation of the hierarchy of the clustered nodes is shown in the form of a dendrogram. Figs. 3(a) and 3(b) present the hierarchy of clustering of nodes for PM2.5 and PM10 respectively. Each of the leaves represents a sensor node. The framework is applied to the data using all the other mentioned linkages, but the dendrograms are not presented due to space constraints. The dendrograms only for complete linkage are shown in Fig. 3. The leaves represent the singleton clusters of single nodes. The distance at which the two leaves combine helps us decide which nodes to group. For example, in Fig. 3(a) Node4 and Node7 are the closest as they have the smallest vertical distance at which they are combined. So, Node4 and Node7 will be grouped, and a cluster representative needs to be chosen for the cluster, which the priority order will define. In the next stage, Node8 is the closest with the group of Node4 and Node7. Furthermore, as explained earlier, a cluster representative for this group at this hierarchy needs to be chosen again.

The cophenetic correlation values in different cases of linkages is presented in Table. I shows that the cophenetic correlation value for the correlation-based distance is very close to 1, indicating the good quality of the obtained solution. The correlation value for Ward's method using the Euclidean distance metric is less comparatively for both PM2.5 and PM10. This implies that the quality of the solution using the correlation-based distances is better than the solution obtained using Euclidean distance.




(a) PM2.5




(b) PM10

Fig. 3. Dendrogram for PM2.5 and PM10 values.

### B. Cluster Representative Priority Order

The geographical distance matrix is represented by $\mathbf{G}$ and $\mathbf{G}_{i,j}$ which is the $(i, j)$ entry is the geographical distance between $i$th and $j$th node. The matrix $\mathbf{G}$ is given by

TABLE I
COPHENETIC CORRELATION VALUES FOR DIFFERENT LINKAGES

| **Linkage Criteria** | **PM2.5** | **PM10** |
|---|---|---|
| Ward's Method (Euclidean Distance) | 0.8290 | 0.8448 |
| Complete (Correlation based Distance) | 0.9885 | 0.9697 |
| Single (Correlation based Distance) | 0.9888 | 0.9753 |
| Average (Correlation based Distance) | 0.9888 | 0.9748 |

$$\mathbf{G} = \begin{bmatrix} 0 & 465 & 244 & 127 & 309 & 374 & 289 & 384 & 434 & 567 \\ 465 & 0 & 330 & 339 & 219 & 136 & 206 & 376 & 301 & 117 \\ 244 & 330 & 0 & 163 & 113 & 314 & 265 & 469 & 192 & 447 \\ 127 & 339 & 163 & 0 & 194 & 257 & 177 & 331 & 337 & 444 \\ 309 & 219 & 113 & 194 & 0 & 226 & 204 & 425 & 160 & 335 \\ 374 & 136 & 314 & 257 & 226 & 0 & 86 & 244 & 363 & 203 \\ 289 & 206 & 265 & 177 & 204 & 86 & 0 & 221 & 360 & 288 \\ 384 & 376 & 469 & 331 & 425 & 244 & 221 & 0 & 581 & 402 \\ 434 & 301 & 192 & 337 & 160 & 363 & 360 & 581 & 0 & 404 \\ 567 & 117 & 447 & 444 & 335 & 203 & 288 & 402 & 404 & 0 \end{bmatrix}.$$

The priority order is calculated using the geographical distance matrix $\mathbf{G}$ and using the Algorithm 1 as follows:

$$p \leftarrow \text{Node8} = \text{Node9} > \text{Node1} > \text{Node10} > \text{Node3} > \text{Node2} > \text{Node5} > \text{Node4} > \text{Node6} > \text{Node7}$$

Using the dendrogram in Fig. 3 and the priority order $p$ calculated above, the clusters at different level of hierarchy for PM2.5 and PM10 data using complete linkage and correlation based distance would be as

$$\mathbf{PM2.5}: (\text{All Nodes}) \rightarrow \text{Node7} \downarrow\rightarrow \text{Node4} \downarrow\rightarrow \text{Node10} \downarrow \rightarrow \text{Node6} \downarrow\rightarrow \text{Node1} \downarrow \text{Node5} \downarrow\rightarrow \text{Node2} \downarrow \rightarrow \text{Nodes(3,8,9)}$$

$$\mathbf{PM10}: (\text{All Nodes}) \rightarrow \text{Node10} \downarrow\rightarrow \text{Node4} \downarrow\rightarrow \text{Node7} \downarrow \rightarrow \text{Node6} \downarrow\rightarrow \text{Node1} \downarrow\rightarrow \text{Node2} \downarrow\rightarrow \text{Node5} \downarrow \rightarrow \text{Nodes(3,8,9)}$$

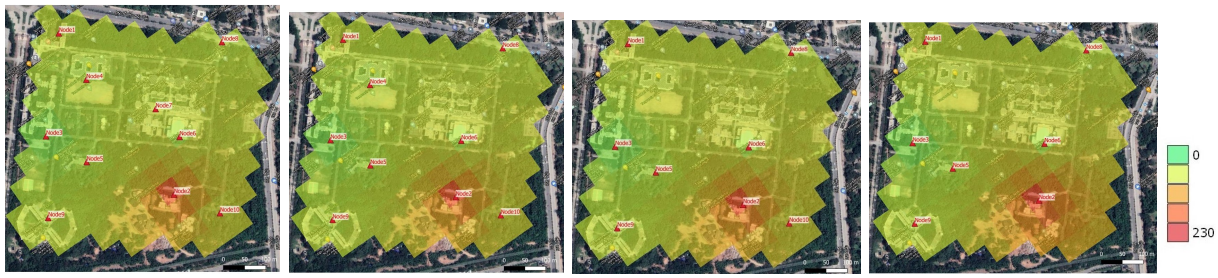
Here we start with the actual deployment of 10 Nodes and proceed with dropping a sensor node at each level of the hierarchy, which indicates the specific node is not representative of the cluster grouped at that hierarchy. For example, in Fig. 3(a), each node is considered as a singleton cluster at the initial stage. Using the least distance at which two leaves of the dendrogram is connected, Node4 and Node7 are grouped. However, priority order $p$ gives higher priority to Node4 when compared to Node7. So at this stage, Node7 is dropped, and Node4 is selected as the representative for the cluster Nodes(4,7). In the next stage, we group the leaves with the next least distance at which they are being connected. In this case, Node4, which is representative of the earlier cluster, and Node8 are considered. As $p$ gives higher priority to Node8 than Node4, Node4 is dropped, and Node8 is selected as the representative for the cluster Nodes(7,4,8). We continue the process until the required minimum number of nodes have reached, in this case, 3 Nodes for dendrograms of both PM2.5 and PM10.

### C. Spatial Interpolation and Reconstruction Error

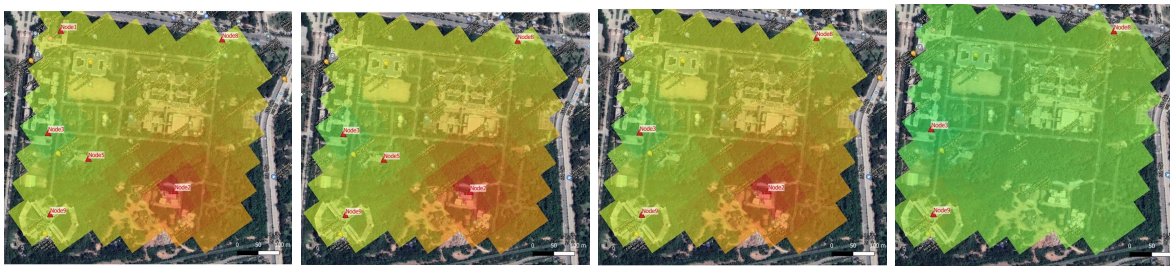
IDW based interpolation is used to reconstruct the dropped nodes' data at each hierarchy of clustering. The IDW is applied on a grid of the considered spatial domain. The spatial interpolation plots for different nodes and the spatial variation is shown in Fig. 4. The number of nodes used for interpolation is from 10 to 3, as shown in Fig. 4. The interpolation plots show that the estimated values in the spatial domain vary a lot with the number of nodes considered. The amount of difference is highly noticeable from 6 Nodes. 3 Nodes were not able to identify the hotspot at Node2. To quantify the variation of the estimated values from the actual values, the reconstruction error for each number of nodes considered in the IoT network is calculated as RMSE and shown in Fig. 5. Fig. 5(a) and Fig. 5(b) respectively show the change in the RMSE with the number of nodes considered for four different linkage criteria. The RMSE for different nodes in different linkages and distance metric cases is compared with the brute force approach. In the brute force approach, all the possible combinations are used for determining the optimal number of nodes at every stage. The main reason for comparing with the brute force approach is that it limits the best possible set of nodes. Any other algorithm will not be able to work better than this. We try to be as close as possible to this limit. As mentioned earlier in section IV-A, Ward's Method with Euclidean distance works the worst compared to the other linkage criteria that used correlation-based distance metric in terms of the cophenetic correlation. The same is shown in terms of the RMSE error also. The resultant error is significantly less and comparable to the inherent sensor error of SDS011 for PM2.5 and PM10. The error between the brute force approach and our approach is very low, and it follows it very closely. The main advantage of our framework is the fewer number of computations compared to the brute force, which considers all the possible combinations at each stage. Another observation is the increase in the error with the decrease in the number of nodes and, in some cases, not able to detect the hotspot location as shown in 4(i). This indicates the need for an optimally decided dense deployment of the sensor nodes. It can also be observed that complete linkage with correlation-based distance works better in most cases. For a particular RMSE threshold or required number of clusters, we can decide the optimal number of nodes and reuse the sensor nodes in different locations.

## V. CONCLUSION

In this paper, a hierarchical agglomerative clustering-based framework has been proposed for deciding the number of optimal nodes in an IoT network. The framework has been employed and tested on the IoT network of PM sensor nodes. The framework with different linkages and distance metrics has been compared with the brute force approach in terms of the RMSE and the number of nodes. The least error obtained for correlation-distance-based linkages and further complete linkage worked better in almost all the cases. The framework proposed has significantly fewer computations than the brute

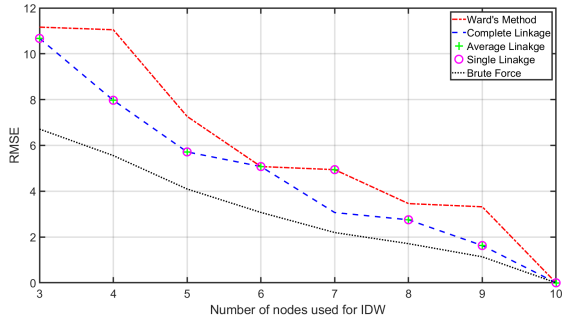



(a) 10 Nodes (b) 9 Nodes (c) 8 Nodes (d) 7 Nodes

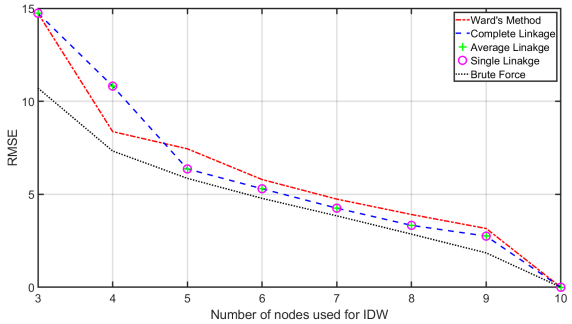


(f) 6 Nodes (g) 5 Nodes (h) 4 Nodes (i) 3 Nodes

Fig. 4. IDW based Spatial Interpolation of PM2.5 for different number of nodes




(a) PM2.5




(b) PM10

Fig. 5. Reconstruction Error v/s the Number of Nodes considered

force approach. The framework helps to decide the optimal number of nodes required in a spatial setting based on the required error threshold. Thus, using the error vs. the number of nodes, we can understand the trade-off between the spatial resolution and tolerable reconstruction error.

## ACKNOWLEDGEMENT

This research was supported partly by National Geospatial Programme (NGP), India, under grant no. 2073 (2020), PRIF Social Incubator Program (2019) and the Ministry of Electronics and Information Technology (MEITY) under grant no. 3070665 (2020), with no conflict of interests.

## REFERENCES

[1] C. Rajashekar Reddy et al., "Improving Spatio-Temporal Understanding of Particulate Matter using Low-Cost IoT Sensors," *IEEE PIMRC*, 2020.
[2] S. Johnston et al., "City Scale Particulate Matter Monitoring Using LoRaWAN Based Air Quality IoT Devices," *Sensors*, 01 2019.
[3] R. Cristescu and M. Vetterli, "On the optimal density for real-time data gathering of spatio-temporal processes in sensor networks," in *Fourth Int. Symp. Inf. Process. Sensor Netw.*, 2005, pp. 159–164.
[4] A. Krause et al., "Near-optimal sensor placements: maximizing information while minimizing communication cost," in *5th Int. Conf. Inf. Process. Sensor Netw.*, 2006, pp. 2–10.
[5] B. Lu et al., "Mobile sensor networks for modelling environmental pollutant distribution," *Int. J. Syst. Sci.*, vol. 42, no. 9, pp. 1491–1505, 2011.
[6] C. C. Castello et al., "Optimal sensor placement strategy for environmental monitoring using wireless sensor networks," in *42nd Southeastern Symposium on System Theory (SSST)*, 2010, pp. 275–279.
[7] Murtagh F and Contreras P, "Algorithms for hierarchical clustering: an overview, II," *WIREs Data Mining Knowl Discov*, 2017.
[8] *ThingSpeak*, accessed 14 Apr. 2021, https://thingspeak.com/.
[9] M. Çelik et al., "Anomaly detection in temperature data using DBSCAN algorithm," in *Int. Symp. Innovations Intell. Syst. Appl.*, 2011, pp. 91–95.
[10] Saracli.S et al., "Comparison of hierarchical cluster analysis methods by cophenetic correlation," *J. Inequal Appl*, 2013.
[11] "Matlab statistics and machine learning toolbox," R2020a, The MathWorks, Natick, MA, USA.
[12] S. A. Sajjadi et al., "Measurement and modeling of particulate matter concentrations: Applying spatial analysis and regression techniques to assess air quality," *MethodsX*, vol. 4, pp. 372–390, 2017.
[13] S. Chaudhari et al., "Spatial interpolation of cyclostationary test statistics in cognitive radio networks: Methods and field measurements," *IEEE Trans. Veh. Technol.*, vol. 67, no. 2, pp. 1113–1129, 2018.