Design of Smart Street Lighting System with Energy Saving Function and Monitoring the System by Smart Grid using Hadoop

A Thesis Submitted

In Partial Fulfillment of the Requirements

for the Degree of

Bachelor of Technology

in

Computer Science and Engineering

Submitted by

ANUPAM SEN
(Roll No: **16CS8081**)

Under the supervision of

DR. PARAG KUMAR GUHA THAKURTA
(Assistant Professor)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

National Institute of Technology Durgapur

West Bengal-713209

India

June 2020

NATIONAL INSTITUTE OF TECHNOLOGY DURGAPUR

CERTIFICATE

It is certified that the work contained in the thesis entitled "**Design of Smart Street Lighting System with Energy Saving Function and Monitoring the System by Smart Grid Using Hadoop"** has been carried out by **Anupam Sen (16CS8081)** under the guidance of **Dr. Parag Kumar Guha Thakurta,** the data reported herein is original and that this work has not been submitted elsewhere for any other Degree or Diploma.

Anupam Sen
(Roll No: **16/CS/8081)**
Place: **Department of Computer Science and Engineering, NIT Durgapur**
Date: **02/06/2020**

This is to certify that the above declaration is true.

Acknowledgement

We have no words to express our gratitude and thanks to **Dr. Parag Kumar Guha Thakurta**, Assistant Professor, Computer Science and Engineering Department, NIT Durgapur, for his precious guidance and effectual care which happens to be the psyche of this thesis report. We consider them as our great advisers and will continue to seek their guidance in future accomplishments.

The thesis report couldn't be furnished without experience and versatility so we would like to express our heartly thanks to all the respected professors of the Computer Science and Engineering Department for their valuable technical and moral suggestion and also constant encouragement, without which this thesis report would not come into existence.

We are grateful to express our gratitude and appreciation to our friends, all those who have provided many helpful suggestions and also encouraged us from time to time in completion of this project.

Date: **02/06/2020**

Anupam Sen
Roll No: **16/CS/8081**
Department of Computer Science and Engineering
National Institute of Technology Durgapur
West Bengal-713209, India

Table of Contents

Abstract

An implementation of Hadoop in Smart Lighting Systems is proposed in this paper. In this project, the street light system, in which lights are on when needed and light-off when not needed. Our smart street light system consists of a LED light, a brightness sensor, a motion sensor and a short-distance communication network. The lights turn on before pedestrians and vehicles come and turn off or reduce power when there is no one. Smart Grids (SGs) are developing as an encouraging technology to confront the energy efficiency issue, presently supported in traditional electrical grids, by disseminating important information in a real-time mode among the various SG units. The Hadoop framework has been advanced to effective growth of comprehensive data in MapReduce applications. Hadoop users define the application calculation logic in terms of mapping and reduction work, which are often described as MapReduce applications. The big data analytics association has authorized MapReduce as a programming model for transforming extensive data on distributed systems. In this paper, we presented a HadoopMapReduce framework where the audit log files (Big Data) are stored in a Hadoop environment using Map-Reduce technique. The Smart grid under surveillance generates Gigabytes of data (log files) which becomes an issue of storage limitation. This data is mapped and reduced into a few Kilobytes or Megabytes. Hence, this technique enables Big Data to store very efficiently. The MapReduce algorithm is executed and our experimental results show significant improvement based on our presented Hadoop-MapReduce framework.

Chapter 1: Introduction

Currently, in the whole world, enormous electric energy is consumed by the street lights, which are controlled by means of the embedded brightness sensors. A sensor light, which is controlled by the brightness sensor and the motion sensor, is sometimes used [1]. It only turns on for a while when the motion is detected in front of the light and it is dark. However, it usually is too late to turn the light on when a person or a car comes in front of it. The light should turn on before a person or a car comes.

The Microgrid is a small-scale grid that is constructed to provide power for local communities and an aggregation of multiple distributed generators (DGs) such as renewable energy sources, conventional generators, in association with energy storage units which work together as a power supply network. In other words, we can see Microgrids as electricity distribution systems that have loads and energy resources that are driven in controlled or coordinated ways.

A smart grid can be defined as an evolved grid system that manages electricity demand in a sustainable, trustworthy and economic manner. Smart Grid permits real time tracking and regulation of power systems.

The word “Big-Data” has been adopted to characterize data sets which are so huge that conventional means of storing data, authority, search, analytics, security, and other functions have become a threat [2]. Big Data is outlined by the huge capacity of digitized data that can come from many origins and in various data formats. The sheer capacity of data makes it tough to evaluate or to figure out. These complications have changed the programming paradigms in order to conveniently control big data both in context of performance & programmability. There is an essential and synergetic ‘similarity' between SGs and Cloud computing: on one hand, SGs are producing Big Data and are in essential demand of storage and processing power. On the other hand, Cloud computing has been modified with the fundamental goal of facilitating compute power (e.g., storage and processing) as a commodity. Cloud computing covers all the load of deploying and managing the required IT framework and yields profit to meet scalability. Hadoop (Apache Hadoop) acts the clustering stage to store and process data. This needs HDFS (Hadoop Distributed File System) for accumulating data and the MapReduce programming that deals with distributed data. The Big Data produced by SGs (primarily produced by the SG sensors and meters) comes in the Key-value pairs. The Hadoop HDFS maintains storage and corresponding problems (e.g., data replication, loss recovery), and the Hadoop-MapReduce works the important "jobs" on selected pieces of the big data. Cloud computing is an evolving technology which provides authenticity, flexibility, scalability, elasticity and is also cheaper to cloud users in terms of cost saving.

Dissertation Structure

The entire work is organised as follows:

Chapter 1 provides a brief introduction

Chapter 2 discusses the related works for completeness.

Chapter 3 discusses the mapreduce algorithm

Chapter 4 presents the system model which includes the network model, energy model. It also provides a discussion about the performance metrics for the proposed work.

Chapter 5 provides an overview of the method of solution proposed in this project work.

Chapter 6 discusses the experimental results, and provides an analysis of the results obtained.

Chapter 7 concludes and discusses the future scope of the proposed work.

Chapter 2: Related Work

As the most popular open source technology ecosystem of Big Data,Hadoop is now widely used. The excellent scalability of Hadoop has been fully verified in the industry. HBase as BigTable was implemented as open source in the Hadoop system.

There have been various outstanding attempts to move Hadoop into cloud [2], [3]. Amazon Elastic MapReduce (Amazon EMR), Cloud-Hadoop [3] and Joyent solution for Hadoop [2] are official cloud platforms.

There are basically three ways to process the time series data:

1.Use of Relational Database

2.Use of Real-time Database

3.Use of Hadoop(HBase)

We generally go for the third approach to process the time series data because the first two approaches have many disadvantages.

Reasons for not using the relational database approach are the following:

1.Data storage isolated and separated:

According to the entity relationship model,a time series data record is saved as a single line.If we use a relational database to store the time series data in accordance with the increasing demand for accessing large amounts of time series data,it will cause pressure and may not fill the entire data.

2.Data access speed is inversely proportional to the scale of data:

As the amount of data increases sharply,the access speed becomes slower and slower.Meanwhile in order to improve the query performance,large amounts of data are indexed into relational databases,which consumes large amounts of system resources.

3.Unable to meet the demand of real-time processing of time series data:

Time series data generated in real time must be loaded into the database and accessed via the database.For massive time series data,relational databases cannot handle high-speed loading and meet the demand.

Reasons for not using the real time database approach are the following:

1.Unable to support management of large scale measuring points:

For a national or provincial smart grid,they need to support the measuring point on the scale of hundred millions or even billions.

2.Insufficient scalability:

Current real time database supports only stand-alone deployment and cannot be clustered to support large scale data applications.

3.**Insufficient high availability and data security:**

Real time databases generally have no high availability features and the security of the data stored also depends on the hardware and database software itself completely and does not support data redundancy backup.

Chapter 3: MapReduce Algorithm

MapReduce is a framework in which we can write applications to process huge amounts of data, in parallel, on large clusters of commodity hardware in a reliable manner. MapReduce is a processing technique and a program model for distributed computing based on java. The major advantage of MapReduce is that it is easy to scale data processing over multiple computing nodes.

Fault Detection:

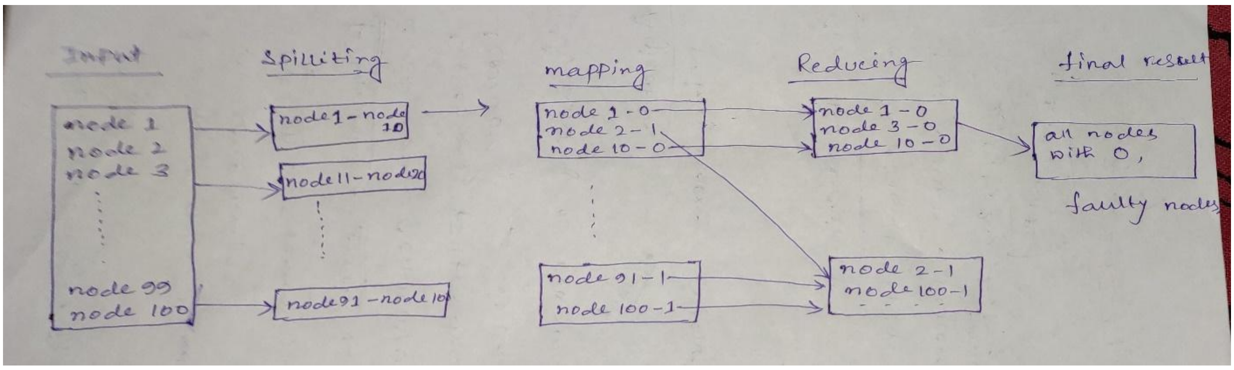
Input- The nodes on the smart grid to which all the devices are connected.

Output- Faulty Nodes.

Nodes in the input are considered as the object the class “NODE”;
The structure of class NODE is described below….

```
NODE{
        Euipment_name;
        Voltage Range;
        Power Required;
        Current Range;
        Status;
Check(NODE);  //This is operation
};
```

The function Check() returns the status of the device represented by the Node passed as argument;



Description:

Large amount of data is taken as input. In splitting, a huge amount of data is splitted into small pieces and processing will be performed in parallel. These small pieces will be assigned in multiple numbers of nodes.

Now the operation is to check whether the device connected with the input node is working or not. This is done by calling the function Check(NODE). If there is any fault it returns '0' otherwise return '1'. This is mapping.

In reducing all nodes whose value is '0' separated from the nodes whose value is '1'.

Then the algorithm returns the nodes having value 0, those are the faulty nodes. This is considered as the final result.

Chapter 4: System Model

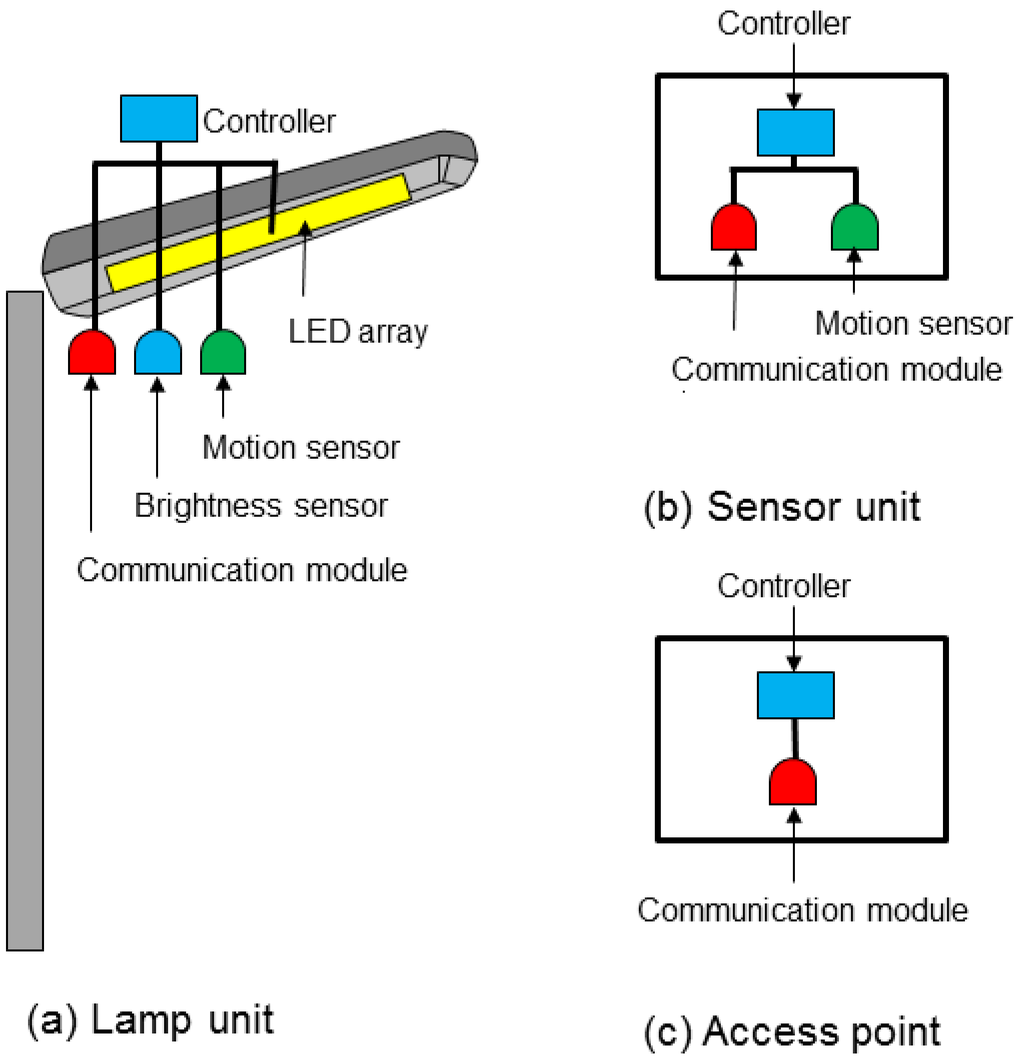


Figure 1.1: shows the components, with which our smart street light system is realized

(a) Lamp unit:

It consists of a power-adjustable LED array, the brightness sensor, the motion sensor, the communication device, such as a module, and the controller. It turns on for several minutes under the conditions that a motion is detected in the defined area by the sensors including its own sensor. Then, it sends the message to other units. It turns off or reduces power under the condition that any motion is not detected in the defined area.

(b) Sensor unit: It consists of the motion sensor, the communication device and the controller. It sends out the message to other units under the condition that motion is detected. This unit is placed at many locations, such as at electric poles, at house gates, at house fences and inside or outside of the door, to ensure that every street light turns on before pedestrians notice that. As for power supply, the solar battery can be a good option.

(c) Access point:

It consists of the communication device and the controller. It is used in the case that the distance between the lamp units and the sensor units are too large to communicate with each other.

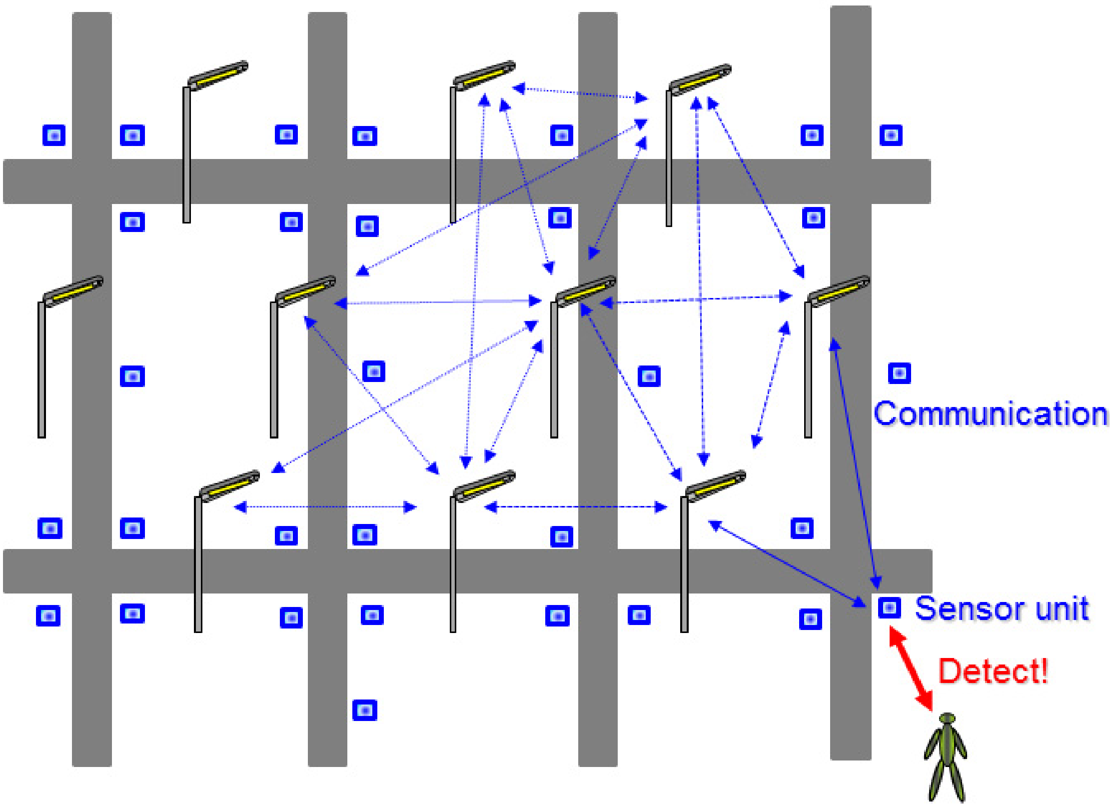


Figure 1.2 : The components for the smart street light

Figure 1.2 shows an example of our smart street light system. The street lights turn on before the pedestrians come and turn off or reduce power when there is no one by means of a distributed-installed sensor network.

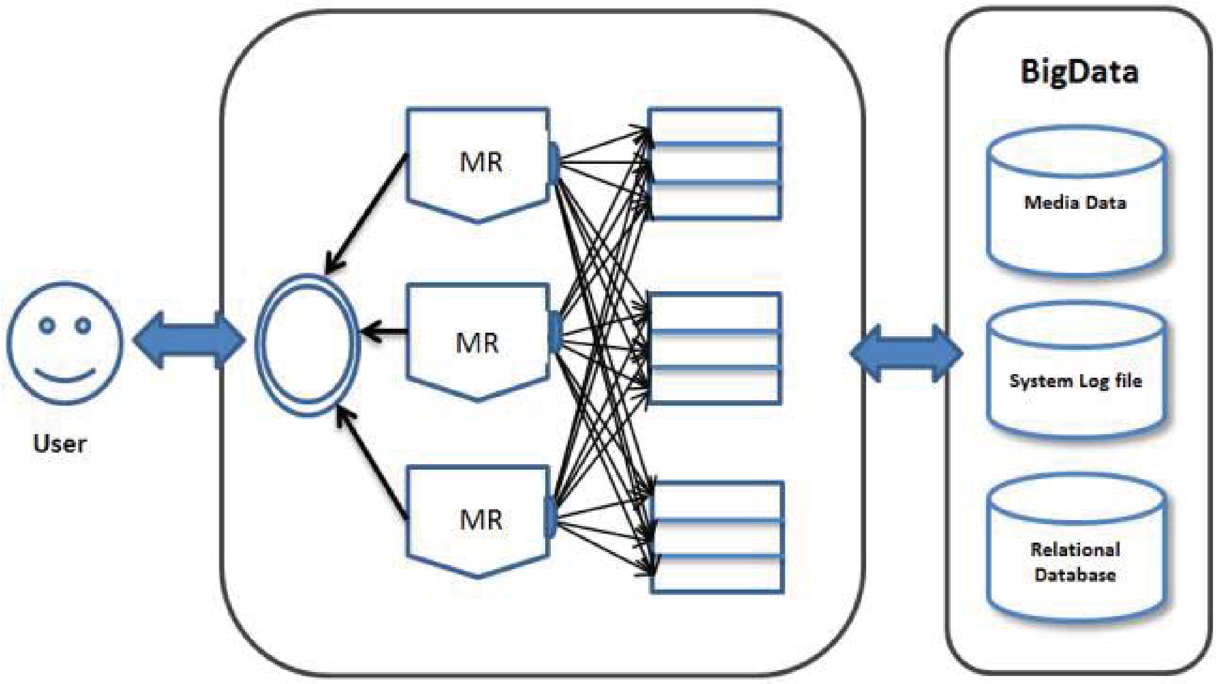


Figure 2: Hadoop Framework

An Apache open source framework named ‘Hadoop’, created in Java, uses simple programming models to refine huge datasets over clusters of computers distributively. Its framework does a task in an environment that facilitates storage and computations over clusters of computers. Hadoop is constructed to extend from single server to thousands of machines, each providing local calculation and storage.

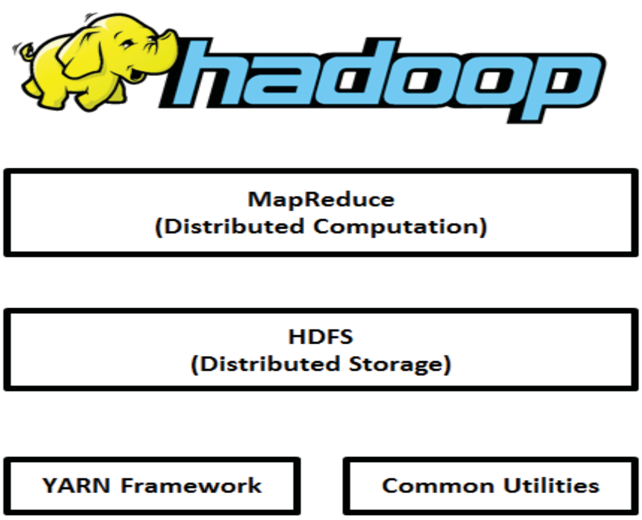


Figure 3: Four Moules in Hadoop

Now Apache Hadoop is a certified brand of the Apache Software Infrastructure. Hadoop manages implementation, uses and applications using the MapReduce algorithm, where the data is refined in parallel on various CPU nodes. Precisely, Hadoop framework is competent enough to establish applications adequate for running on clusters of computers and they could execute entire analytical analysis for a large load of data. (Refer Fig. 2) Hadoop framework consists of the following four modules:

❖ **Hadoop Common:**These are Java libraries and utilities required by other Hadoop modules.

❖ **Hadoop YARN:** This is a framework for job scheduling and cluster resource management.

❖ **Hadoop Distributed File System (HDFS):** A distributed file system that facilitates high-throughput authority to application data.

❖ **Hadoop MapReduce:** This system is used for parallel processing of huge data sets which is completely dependent on the YARN system.

How Does Hadoop Work?

It is quite expensive to build bigger servers with heavy configurations that handle large scale processing, but as an alternative, you can tie together many commodity computers with single-CPU, as a single functional distributed system and practically, the clustered machines can read the dataset in parallel and provide a much higher throughput. Moreover, it is cheaper than one high-end server. So this is the first motivational factor behind using Hadoop that it runs across clustered and low-cost machines.

Hadoop runs code across a cluster of computers. This process includes the following core tasks that Hadoop performs −

❖ Data is initially divided into directories and files. Files are divided into uniform sized blocks of 128M and 64M (preferably 128M).

❖ These files are then distributed across various cluster nodes for further processing.

❖ HDFS, being on top of the local file system, supervises the processing.

❖ Blocks are replicated for handling hardware failure.

- ❖ Checking that the code was executed successfully.
- ❖ Performing the sort that takes place between the map and reduce stages.
- ❖ Sending the sorted data to a certain computer.
- ❖ Writing the debugging logs for each job.

Advantages of Hadoop:

Hadoop framework allows the user to quickly write and test distributed systems. It is efficient, and it automatically distributes the data and work across the machines and in turn, utilizes the underlying parallelism of the CPU cores.

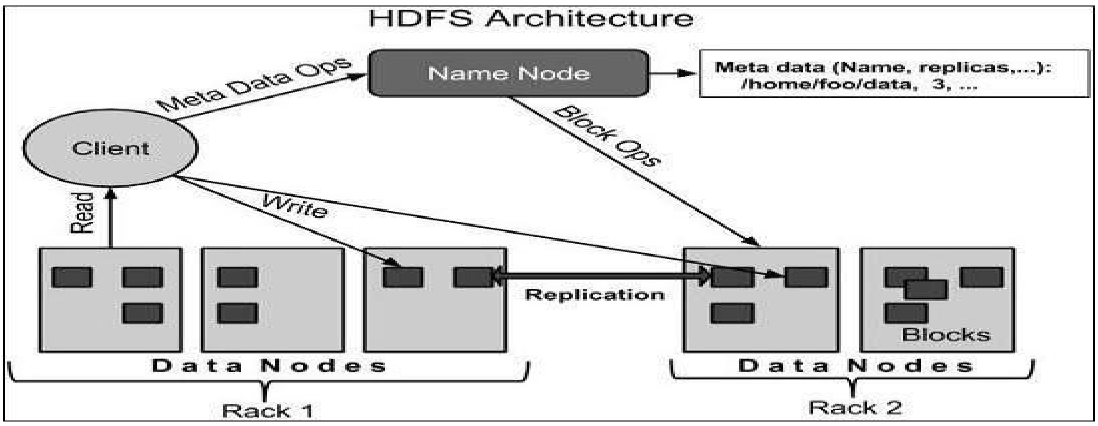
Hadoop does not rely on hardware to provide fault-tolerance and high availability (FTHA), rather Hadoop library itself has been designed to detect and handle failures at the application layer.

Servers can be added or removed from the cluster dynamically and Hadoop continues to operate without interruption.

Another big advantage of Hadoop is that apart from being open source, it is compatible on all the platforms since it is Java based.

HDFS Architecture:

Given below is the architecture of a Hadoop File System.



HDFS follows the master-slave architecture and it has the following elements.

1) Name node:

The namenode is the commodity hardware that contains the GNU/Linux operating system and the namenode software. It is a software that can be run on commodity hardware. The system having the namenode acts as the master server and it does the following tasks −

Manages the file system namespace.

Regulates client's access to files.

It also executes file system operations such as renaming, closing, and opening files and directories

2) Datanode:

The datanode is a commodity hardware having the GNU/Linux operating system and datanode software. For every node (Commodity hardware/System) in a cluster, there will be a datanode. These nodes manage the data storage of their system.

Datanodes perform read-write operations on the file systems, as per client request.

They also perform operations such as block creation, deletion, and replication according to the instructions of the namenode.

4.1 Network Model

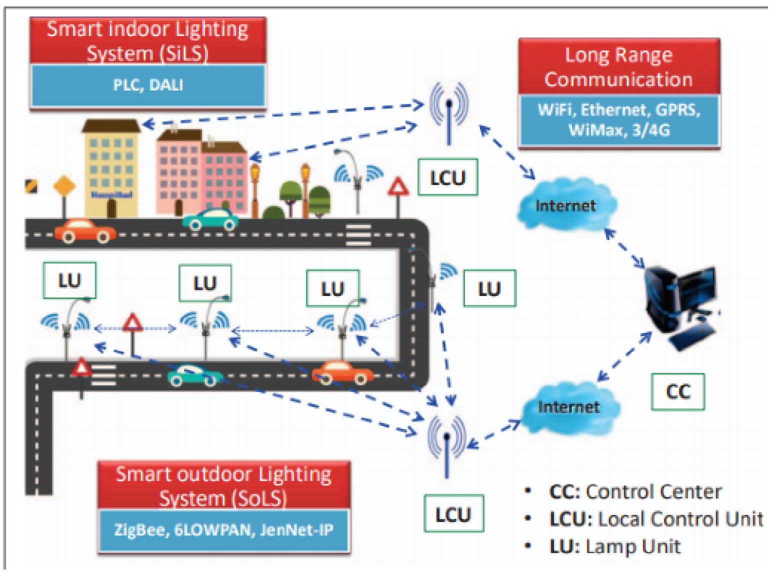


Figure 5: Overview of Smart Lighting System in Smart Grid Environment

In this section, we introduce basic components of smart lighting systems (SLSs) and give detailed categorization based on different features of SLSs. The term smart in a lighting system in a smart city environment refers to its being autonomous and efficient which is achieved by the features of the IoT technology. Normally, efficiency for one lamp can be increased significantly with the advances in lamp technology (e.g., LED). Despite improvements, there is still great potential in the current lighting systems to save energy. For instance, by introducing different sensors (e.g., light sensor, motion sensor, etc.) and efficient communication schemes, a lighting system can be autonomous, efficient, and interoperable with other components of the smart city. Such a sensor-equipped lighting system that can communicate with other lamp units and a central management system efficiently is referred to as Smart Lighting System (SLS). In addition to the reduction in power consumption, an SLS also provides a reduction in the maintenance cost since faults in the lamps can be easily detected and fixed. Conventional lighting approaches which are proposed to monitor and control the energy consumption of lighting systems have been focused on either manual solutions (e.g., using LED bulbs, efficient wiring, etc.) or arranging on/off duration of the lamps for certain periods of a day. While manual solutions suffer from higher implementation cost, timer-based solutions waste the power in less populated regions during uncrowded nights. However, to consider these drawbacks, an SLS uses multiple sensors (motion sensors, light sensors, fog sensors, etc.). These sensors are used to adjust the on/off time according to human presence and light intensity. Figure 1 gives a detailed view of an SLS in the context of a smart city. In an SLS, there are three basic components of the system architecture:

1. Lamp Unit (LU): In recent years, significant improvements have been achieved in lamp manufacturing technologies. The introduction of LEDs in lighting systems has reduced energy consumption remarkably. Moreover, the use of reflection and refraction features of light in manufacturing offers efficient intensity control of a lamp unit. In addition to these features, different control mechanisms (wired or wireless) provide seamless integration with LU and other components of an SLS. Typically, an LU of an SLS consists of a controller and several sensors connected to it, which gathers data and provides the ability to communicate with each other as well as with a control unit called Control Center (CC). The data can be gathered from motion sensors or light sensors. While motion sensors are used to detect the pedestrians and cars, ambient light sensors like LDR can be used to both check the operational condition (intensity, on/off) of the lamp and the intensity of the daylight. Some other types of sensors like the temperature sensor can also be deployed to detect the fault in the power lines. In summary, LUs of SLS should have following three features:

1. Lamps used in SLS must be energy efficient and should have easy maintenance steps.
2. LUs must be composed of necessary and sufficient sensors to provide automatic control to the overall system. Also, sensors must be deployed to provide the intelligent on-off scheme to reduce power consumption.
3. As an SLS must utilize both centralized and localized control schemes, an efficient and reliable control system is needed with LUs to handle various conditions.

2. Local Control Unit (LCU): Local control unit collects the data from an array of LUs through a short range communication protocol (e.g., IEEE 802.15.4 protocols such as 6LoWPAN or Bluetooth Low Energy etc.) transmit the data to the Control Center. In some distributed architectures, they can build a mesh network topology and have the ability to talk to each other as well.

3. Control Center (CC): The Control Center collects all types of data from LCUs and stores it on a server. With recent advancements in cloud computing it is also possible to store data in the cloud instead of a server. With this, a more cost-effective SLS model is possible as storing data from all the LCUs for large urban areas requires large storage units on the server. Then, different data analysis tools are then used to visualize and analyze such data. Based on the results of these analysis process; different decisions can be made and actions can be taken to control overall SLS by the administration.

Moreover, based on the ambience, SLSs can be used in either Indoor or Outdoor lighting. Smart indoor lighting systems (SiLS) are used in the public service buildings like hospitals, schools or big factories, and companies.

4.2 Smart Grid Energy Model

Figure 6: Hadoop Environment

Smart grid can be defined as electricity networks that can Intelligently integrate the actions of all users connected to it – generators, consumers and those that do both – in order to efficiently deliver sustainable, economic and secure electricity supplies.

The U.S. defined the smart grid of the future in a similar way that incorporates the digital technology to improve reliability, security and efficiency of the electric system through information exchange, distributed generation and storage resources for a fully automated power delivery network.

Advantages of Smart Grid:

Normally, the smart grid can be assessed with a Smart Grid Architecture Model (SGAM), which is a 3-dimensional framework that merges domains, zones and layers together. The conventional structure of power systems can be found in the domains as generation, transmission, distribution, DER (Distributed Energy Resources) and customer premises. The zones which present the layout of power system management are composed of market, enterprise, operation, station, field and process. On top of the first two dimensions, the layout of interoperability layers includes the component, communication, information, function and business layers. SGAM as an architectural overview can be used to find the limitations and commonalities of existing smart grid standards.

Data Analysis in Smart Grid:

We have discussed a Smart Grid system embedded with an information layer that allows for two-way communication between the central controllers and local actuators as well as logistic units to respond digitally to urgent situations of physical elements or quickly changing electric demand. Now, the information which is communicating between the smart grid and the above discussed model of street light can be categorized as Big Data.

Concept of Big Data:

The definition of big data is not very clear and uniform at present. But there is a consensus among different descriptions: this is an emerging technical problem brought by a dataset of large volume, various categories and complicated structures which needs novel framework and techniques to excavate useful information effectively. Therefore, the definition of big data depends on the ability of data mining algorithms and the corresponding hardware equipment to deal with large volume datasets. It is a relative concept instead of an absolute definition. Big data can be understood as the amount of data beyond technology's capability to store, manage and process efficiently as the data size increases along with the evolution of ICT technologies.

Chapter 5:Proposed Approach

Our approach is to merge the Smart Lighting Systems in a Smart Grid with Hadoop.

The targets of our development are as follows:

❏ Easy installation and extension: Each unit can be installed one by one to the network by setting the parameters. The system is autonomous-distributed controlled. No hosta computer is needed.

❏ Low cost: Only the parts of mass production are used.

❏ Easy update: The firmware of each unit can be updated easily. The control algorithms should be developed for the situations, such as a quiet residential area, a shopping street, a part, a main road and a mountain road.

❏ Self-diagnosis: The worst event is that the light does not turn on when the pedestrian comes. Each unit records the failures, in which the motion is detected in front of it without the advanced notification from the other units

There are following requirements to implement our system-

- ❏ A modified switching center
- ❏ Street light with multi sensor (Optical, Motion & Speed,)
  - i) A modified switching center, it controls the switching system of different area
  - ii) Street light with multi sensor
    - a) Optical sensor
    - b) Motion sensor
    - c) Speed sensor
    - d) Visibility sensor for determining visibility under different weather conditions.
    - e) Low power consuming and bright LED Light with fast.
  - iii) Low power consuming and bright LED Light with fast response time that is It can respond instantly on switch on/off request

iv) Brightness controller

v) Classification of area

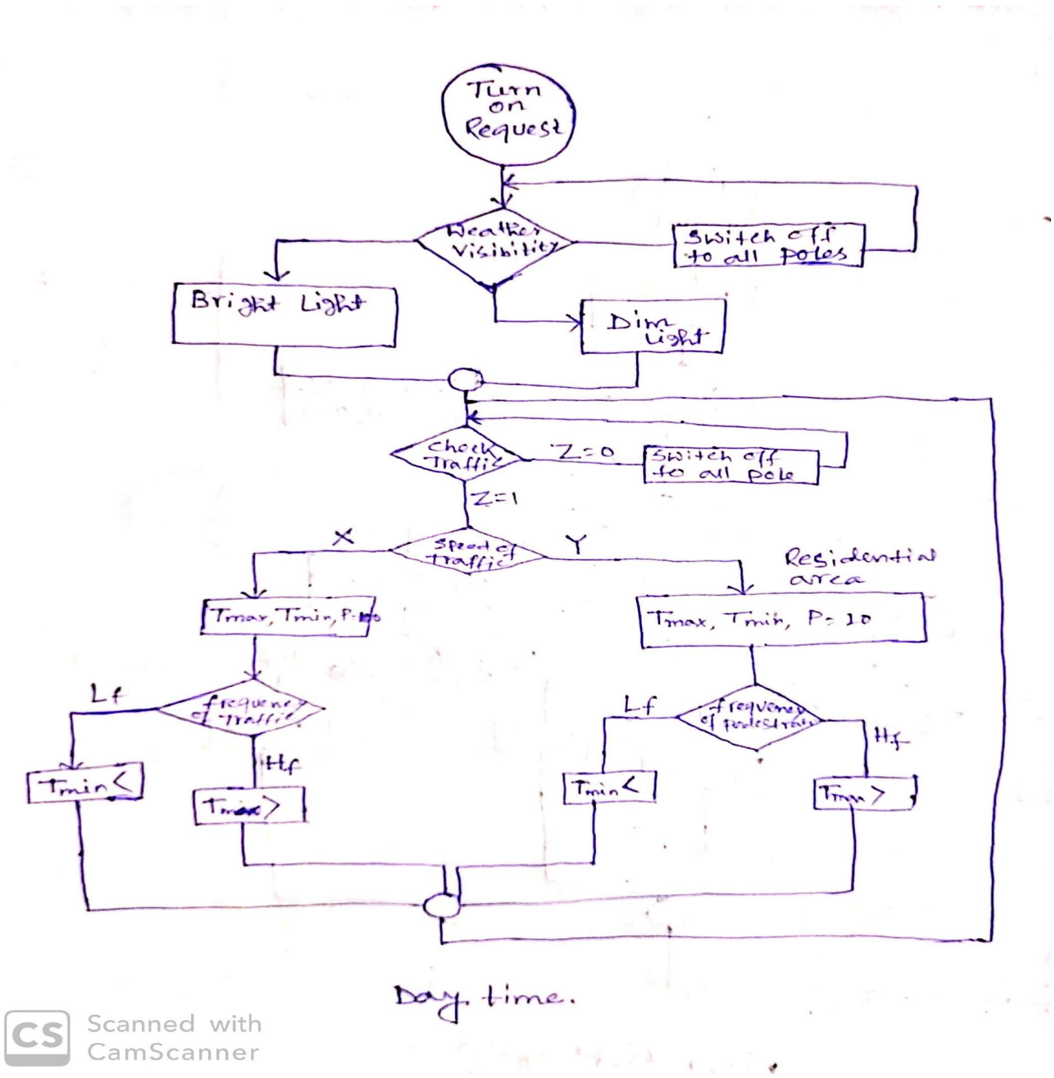
Highway area: Where vehicle motion is available. Low frequency area of pedestrians.

Some parameters are used for modeling algorithms and analyzing the system.

V: Voltage of each light.
Vd: Dim voltage.
Tmin: Minimum turn on time of LED.
Tmax: Maximum turn on time of LED.
S0: Switch off.
S1: Switch on.
P: no of poles.
Hf: High frequency of traffic.
Lf: Low frequency of Traffic.
X: Fast moving traffic.
Y: Slow moving traffic.
Z: Detection of traffic(0/1)
W: Total Power consumption.

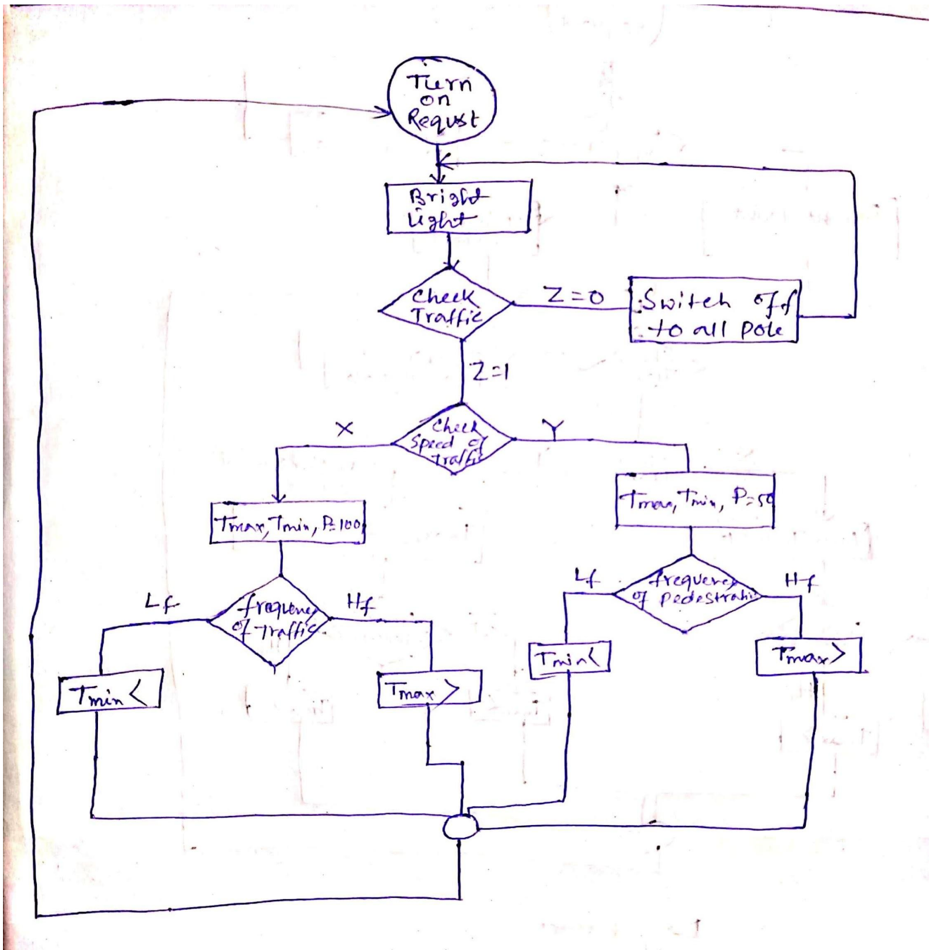
In Day time:

When we implement the algorithm, the sensor continuously checks the weather and visibility conditions. And the brightness of the LED will be adjusted according to the visibility levels. If the weather condition is good and there is no Fog, the LED remains off. In case of bad weather conditions when LED is on it Checks whether there is any traffic in the area. If there is no traffic, turn off the switch (S0) of all poles in the area. Now if there is motion available, it checks the speed of traffic. During the peak traffic hours the lights will remain turned on. The algorithm performs different functions for Highways and pedestrian areas. In Highway when vehicle motion is detected, it turns on the switch of more contiguous LED poles (we assumed P = 100) for a predefined time duration (Tmax). Now it checks the frequency of traffic and changes the turn on time accordingly. Go to step 3 and repeat the process. Now in case of pedestrian areas it skips steps 4, 5 and jumps to step 7. It turns on the switch off less number of contiguous LED poles(We assumed P = 50) for a predefined time duration. Here, less power is consumed. So the control continuously checks the availability of power storage and stores the extra power in the storage. Then control goes to step 3 again and repeat the process.



Day time flow chart

In Night time: In this case the algorithm remains the same. The only change is it won't check any weather conditions. The LED remains on with high brightness. The rest of the algorithm remains the same as day time.



night Light System

Power Consumption:

W1 = X*Hf(Power consumption in highway area in high frequency vehicle),
W2 = X*Lf(Power consumption in highway area in low frequency vehicle)
W3 = Y*Hf(Pedestrian area in high frequency motion).
W4 = Y*Lf(Pedestrian area in low frequency motion).

Therefore, Total Power Consumption, W = W1 + W2 + W3 + W4.

The Algorithm follows following steps:

i) Typically the MapReduce model is organized on delivering the computer to where the data is located.

ii) MapReduce program performs in three steps, viz. mapping stage, shuffling stage, and reducing stage.

1) Map stage: The Mapper's function is to operate the input data obtained as a file or directory, and is reserved in the Hadoop file system (HDFS). The input file is moved to the mapper function line by line. The Mapper operates the data and generates various small pieces of data.

2) Reduce stage: This stage is the fusion of the Shuffle stage and Reduce stage. The Reducer's function is to operate on the data that arrives from the Mapper section. After operating, it creates a new set of output, which will be accumulated in the HDFS.

iii) During a MapReduce job, Hadoop passes the Map and Reduce tasks to the correct servers in the cluster.

iv) The framework maintains all the contents of data passing such as issuing tasks, checking task completion, and replicating data over every cluster between the nodes.

v) Almost all of the evaluating happens on nodes with data on specific disks that decreases the network traffic.

vi) After finalizing the given tasks, the cluster gathers and decreases the data to form a correct result, and transfers it back to the Hadoop server

Chapter 6: Experimental Results

6.1 Simulation setup

In our work, we created a Hadoop environment for Smart grid, and we have mapped and reduced the data using MapReduce technique. Following are the steps in which we execute our work:

1. **Creation of Hadoop environment:** We created a Hadoop environment and we have configured the Hadoop services. The generator unit, solar panel unit, processor unit, industrial plant, central power plant and wind turbines are deployed in the network. All the data coming to these units are directly stored to the cloud (see figure 7). Here, we can see all the data transactions happening between all the units.

2. **Implementation of Hadoop distributed file system (HDFS):** In HDFS, we can access and store the log files of each unit that are mentioned previously. In each unit, we can see the number of sent packets as well as receive packets. It shows how many packets are sent or received in bytes (packet size) in accordance to time (ms). In every unit, specific node location and along with its data transfer can be seen. The snapshot is shown in figure 8.

3. **Implementation of MapReduce algorithm:** The data (log files) generated from the smart grid are mapped and reduced, and are stored in the cloud. The files that has generated packet sizes on the right column corresponding to their node ID are shown in figures 7 and 8. Figure 8 shows input to the MapReduce algorithm i.e. audit log files (Big Data) generated in smart grids.

4. **Performance analysis with Hadoop configuration:** Latency, Packet delivery ratio, Throughput, total energy consumption and residual energy for Smart grids with HADOOP configuration with respect to packet size (in bytes).

6.2 Simulation Studies

8	40
8	40
7	40
7	40
4	40
4	40
3	40
3	40
0	40
0	40
17	40
17	40
2	40
2	40
13	40
13	40
10	40
10	40
11	40
11	40
12	40
12	40
1	40
1	40
14	40
14	40
9	40
9	40
16	40
16	40

Figure 7: Input for the MapReduce Algorithm

0	2
1	2
10	2
11	2
12	2
13	2
14	2
15	209345
16	2
17	2
2	2
3	2
4	2
40	209379
5	4
7	2
8	2
9	2

Figure 8: Output of the MapReduce Algorithm

Figure 7 shows input to the MapReduce algorithm i.e. audit log files (Big Data) generated in smart grids. In Figure 8, the output of the MapReduce algorithm is represented where the log files are mapped and reduced, thus solving the storage limitations issue.

Performance Parameters:

A. Latency

Latency can be defined as delay that occurred in data communication over the network. In other words, it can be also defined as the time taken by the packets to reach from source (sender) to destination (receiver).

B. Packet Delivery Ratio

It is defined as the ratio of the amount of all packets arriving at the destination (receiver) to the amount of all packets departed by the source (sender). . It can be formulated as: Packet Delivery Ratio = (Σ No. of packet arrived) / (Σ No.of packet departed).

C. Throughput (bits/bytes per second)

Throughput can be defined as how many data packets received by a receiver within data communication time or successful data communication executed within a time period. Throughput is the average rate of successfully data packet delivered from source to destination node.

Chapter 7: Conclusion

We are in the generation of Big Data. MapReduce is a universal programming configuration for big data study. In our work, we have created a Hadoop environment. We have also implemented MapReduce algorithm that reduces large clusters consisting of multiple machines and to solve large number computational problems encountered in the Smart Grids. The log files (Big Data) that generated in Smart Grids are efficiently mapped and reduced using MapReduce technique. The MapReduce data is successfully stored on the Hadoop cluster. The Gigabytes of data (log files) are diminished to a few Kilobytes or Megabytes. We have explained how HDFS generates numerous replicas of data streams and circulates them on compute nodes with all the clusters to implement reliable, extremely breakneck computations. The performance parameters mentioned in Chapter 6 (Section 6.2) can be used to evaluate the performance of the model suggested in this paper.

References

[1]Velaga, R. and Kumar, A. 2012. Techno-economic evaluation of the feasibility of a smart street system: A case study of rural India. Procedia Social and Behavioral Sciences. 62,1220-1224.

[2] Benhaddou, Driss, et al. "BIG DATA PROCESSING FOR SMART GRIDS."

[3] Inforchimps, “Cloud::hadoop,” Available:

http://www.infochimps.com/infochimpscloud/cloudservices/cloudhadoop/ Accessed in Nov.2014.

[4] Ubarhande, Vrushali, Alina-Madalina Popescu, and Horacio Gonzalez-Velez. "NovelData-Distribution Technique for Hadoop in Heterogeneous Cloud Environments." Complex,Intelligent, and Software Intensive Systems (CISIS), 2015 Ninth International Conference on. IEEE, 2015.

[5]Available: [6] M. Centenaro, L. Vangelista, A. Zanella, and M. Zorzi, “Long-range communications in unlicensed bands: The rising stars in the iot and smart city scenarios,” IEEE Wireless Communications, vol. 23, no. 5, pp. 60–67, 2016

[7] Martha, Venkata Swamy, Weizhong Zhao, and Xiaowei Xu. "hMapReduce: a framework for workload balancing in MapReduce."Advanced Information Networking and Applications (AINA), 2013 IEEE 27th International Conference on.IEEE, 2013.

[8] Benhaddou, Driss, et al. "BIG DATA PROCESSING FOR SMART GRIDS."

[9] Dean, Jeffrey, and Sanjay Ghemawat. "MapReduce: simplified data processing on large clusters."Communications of the ACM51.1(2008): 107-113.