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

**Bachelor of Technology**

***in***

**Computer Science and Engineering**

***Submitted by***

**Asim Bera and Anupam Sen**

(Roll No.  **16CS8042 & 16CS8081**)

***Under the supervision of***

**Dr. Parag Kumar Guha Thakurta**

***(Assistant Professor)***



**Department of Computer Science and Engineering National Institute of Technology Durgapur**

**INDIA**

**July 2019**

# **National Institute of Technology, Durgapur** https://lh6.googleusercontent.com/9NwMbeXtsDygBbOOk5irWG4W4aZHVV57m2ILnEwd24uSBoMTuJuCKD5_D7GEQr9s87eN2ZxN6rNNJSUvH08sj_a6L8RrjZsGaAyBIUvPSk2NOhuUfeGMWZ4QLMjdnBsgXSpPOpRnuqCzC8fxyg

**CERTIFICATE**

It is certified that the work contained in the project 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 **Asim Bera (16CS8042) & Anupam Sen(16CS8081)** under the guidance of **Dr. Parag Kumar Guha Thakurta,** the data reported here is original and this work has not been submitted elsewhere for any other Degree or Diploma.

----------------------------------------

**Asim Bera**

**Roll No: 16CS8042**

----------------------------------------

**Anupam Sen**

**Roll No: 16CS8081**

Place: **Department of Computer Science and Engineering, NIT Durgapur**

Date: **16/07/2019**

This is to certify that the above declaration is true:

----------------------------------------

**Dr. Parag Kumar Guha Thakurta**

**Assistant Professor, Department of Computer Science and Engineering, NIT Durgapur**

Date: **16/07/2019**

# A **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 heartily thanks to all the respected professors of 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, batchmates, all those who has provided many helpful suggestions and also encouraged us from time to time in completion of this project.

-----------------------------------------------

#### Asim Bera

**Roll No: 16CS8042**

----------------------------------------

**Anupam Sen**

**Roll No:16CS8081**

**----------------------------------------**

**Department of Computer Science and Engineering National Institute of Technology Durgapur**

**West Bengal-713209, India**

Abstract

The steps of this project are as followed

Step 1:

To achieve the energy saving efficiency design an architecture of the smart street light system.

Step2:

For power supply and monitoring the using of smart grid.

Step3:

Data analysis in smart grid.

1. Collect the data (Big Data) from the system.
2. Analysis the data using Hadoop Environment
3. Function Performed after the data analysis by the smart grid.

Architecture of the Smart Light System

# **Step1:**

# Purpose:

Design an architecture of Smart Light System (SLS) to achieve energy saving efficiency.

# Proposed Model:

There are following requirements to implement our system-

• A modified switching center

• Street light with multi sensor (Optical, Motion & Speed,

1. A modified switching center, it controls the switching system of different areas.
2. Street light with multi sensor
3. Optical sensor
4. Motion sensor
5. Speed sensor
6. Visibility sensor for determining the visibility under different weather condition
7. Low power consuming and bright LED Light with fast
8. Low power consuming and bright LED Light with fast response time that is it can response instantly on switch on\off request.
9. Brightness controller
10. Pole equipped with solar panel and energy storage
11. \*\*A Smart Grid and database center, for supply power and monitoring the system.
12. Classification of area:

High way area: Where vehicle motion available.

Low frequency area of pedestrian

viii) \*\*Smart decision making algorithm

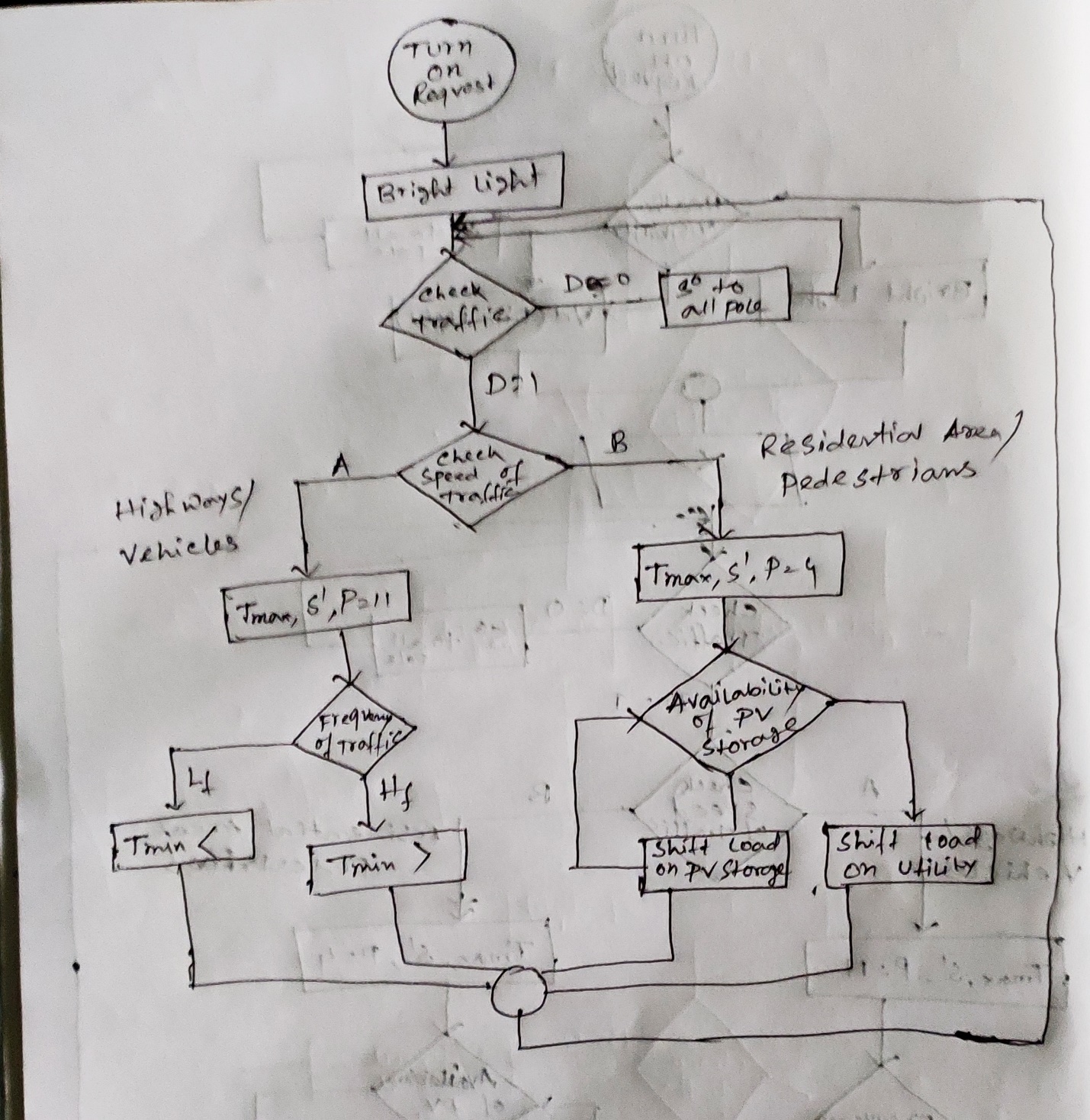
# Parameters:

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

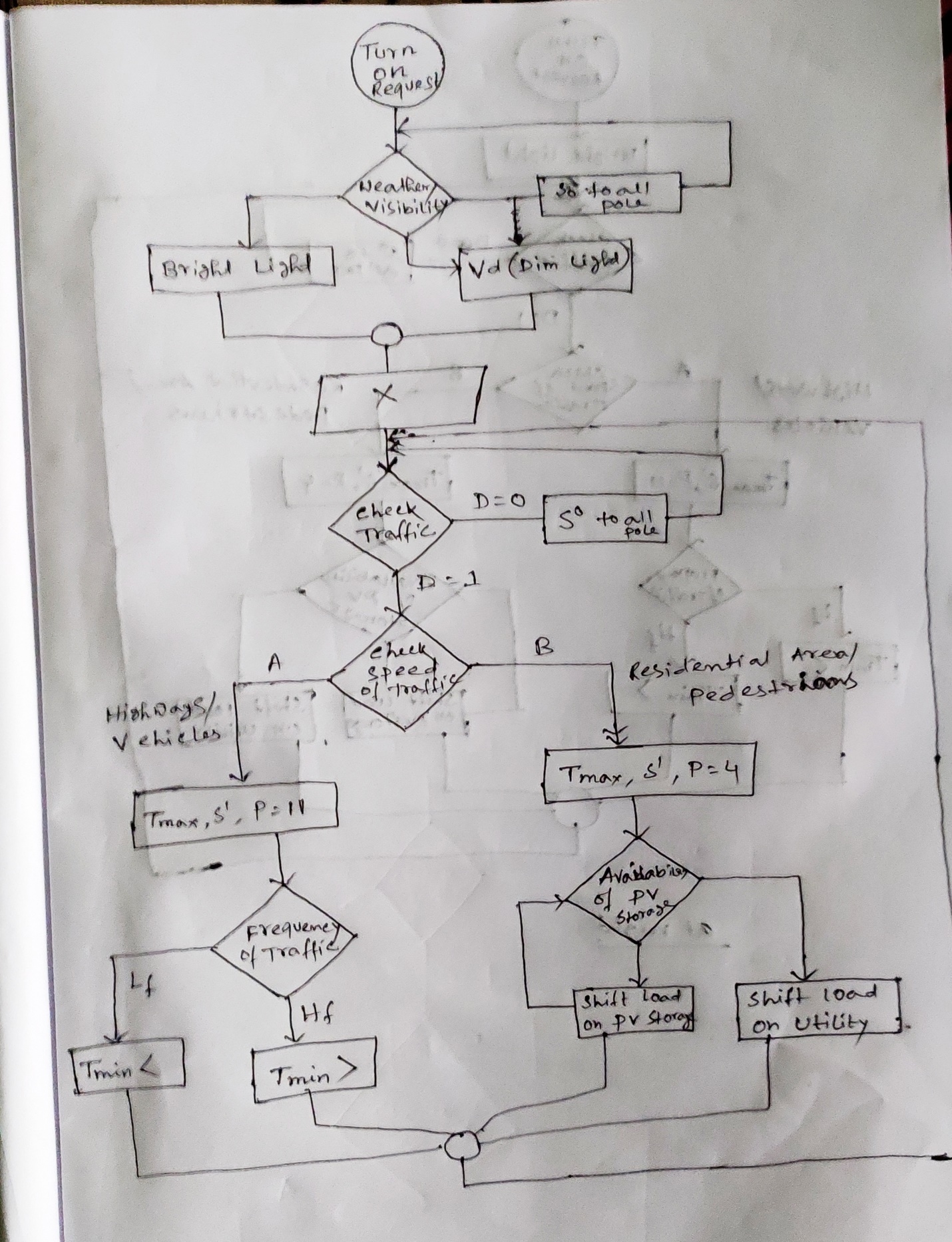
|  |  |
| --- | --- |
| Parameter | Description |
| V | Voltage of each light |
| Vd | Dim voltage |
| Tmin | Minimum turn on time of LED |
| Tmax | Maximum Turn on Time LED |
| S0 | Switch off |
| S1 | Switch on |
| P | Number of Poles |
| Hf | High frequency of traffic/ pedestrian |
| Lf | Low frequency of traffic/pedestrian |
| A | Fast moving traffic |
| B | Slow moving traffic |
| W | Total power consumption |
| D | Detection of traffic(0/1) |
|  |  |

Algorithm

# Flow Chart:



Flow Chart: During Night time



Flow Chart: During Day Time Under different Weather condition

Day time: When we implement the algorithm,

1. the sensor continuously check the weather and visibility condition. And the brightness of the LED will be adjusted according to the visibility levels. If weather condition is good and there is no Fog, the LED remains off.
2. In case of bad weather condition when LED is on it Checks whether there is any traffic in the area. If no traffic, turn off the switch (S0) of all poles in the area.
3. Now if there is motion available of traffic, it checks speed of traffic. During the peak traffic hours the lights will remain turned on. The algorithm perform different functions for Highways and pedestrian area.
4. In Highway when vehicle motion is detected, it turns on the switch of more number of contiguous LED poles (we assumed P = 11) for a predefined time duration (Tmax).
5. Now it checks the frequency of traffic and changes the turn on time accordingly.
6. Go the step 3 and repeat the process.

Now in case of pedestrian area it skips the step 4, 5 and jump to the step 7.

1. It turns on the switch of less number of contiguous LED poles( We assumed P = 4) for a predefined time duration.
2. Here, the less amount of power is consumed. So the control continuously checks the availability power storage and stores the extra power in the storage.
3. Then control goes to again step3 and repeat the process.

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

Analyzing Power Consumption

W1 = A\*Hf(Power consumption in highway area in high frequency vehicle),

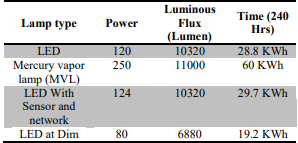
W2 = A\*Lf(Power consumption in highway area in low frequency vehicle),

W3 = B\*Hf( Pedestrian area in high frequency motion).

W4 = B\*Lf(Pedestrian area in low frequency motion).

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

Ԝ shows the energy consumption in overall system with different speed and rate of flow of traffic and pedestrians. Table describes the specification of lamps and energy consumption in residential areas. For the calculation of results we consider the street and roads of 1 kilometer. The height of lamp post is 8 meters and spacing between them is 24 meters. Similarly, table 3 defines the parameter for highways, where the heights of the towers are 12 meters with 36 meter spacing we also consider the 1Km length of the highway.



Step2: We are using smart grid to transfer power and monitor the previously discussed System.

Now the question should arise what is a smart grid?

# Smart grid:

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 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.

# Why not traditional grid:

There is a lot of deficiency in case of traditional grid system-

Compared with traditional power systems, the widespread application of distributed generators under the call of green energy resources is shaking the hegemony position of large-scale centralized power plants, which makes the conventional centralized control strategy less effective due to the unidirectional power flow. Connection of small-scale power generations (typically in the range of 3 kW to 10 kW) to the public distribution grid requires two-directional operation and control of distribution grids. Faced with the challenges of more complicated control and protection strategies, the conventional electro-mechanical electric grid is supposed to be enhanced with the help of innovations in the digital information and telecommunications network to overcome the cost from power outages and power quality disturbances as billions of dollars annually.

# 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 system 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 that 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 of electric demand.

Now, the information which are communicating between smart grid and 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 (Zikopoulos & Eaton, 2011). It is a relative concept instead of an absolute definition. The big data can be understood as amount of data beyond technology’s capability to store, manage and process efficiently in (Kaisler et al., 2012) as the data size increasing along with the evolvement of ICT technologies.

# Big Data Characteristics in Smart Grid:

The characteristics of big data in smart grid is also in accordance with the universal 5 V big data model in many researches as below:

# 1)Volume:

Volume refers to the vast amount of data generated, which makes data sets too large to store and analyze using traditional database technology. The possible solution to this problem is the distributed systems to store data in different locations, connect them by networks and bring them together by software. In the smart grid the widespread application of smart meter and advanced sensor technology connected with the lighting network provide huge amount of data.

# 2) Velocity:

Velocity refers to the speed at which new data is generated and the speed at which data moves around. In the smart lighting system in a large city the new data generation rate is very high.

# 3) Variety:

Variety refers to the types of data we can now use. In the past, we focus on structured data that neatly fits into tables or rational databases such as financial or meteorological data. With big data technology, we need to handle different types of unstructured data.

In our smart light system the different types of data are-

1. Message from sensor.
2. Information from meters connected in the network.
3. Status of the Equipments
4. Equipment parameters
5. Load control
6. Power Quality
7. Geographycal information
8. Meterological Information.

# 4) Veracity:

Veracity refers to the messiness or trustworthiness of the data. The quality and accuracy are less trustworthy with such large amount of big data, which challenge the outcome data analysis. Errors of measurements in smart grid may exist due to the imperfections in devices or mistakes in data transmission. The secure and efficient power system operation relays on the data assessment and state estimation.

# 5) Value:

Value refers to our ability to extract valuable information from the huge amount of data and derive a clear understanding of the value it brings. The larger the amount of data is, the lower the density of valuable information will be. With the improvement of intelligent devices adopted in smart grid, more and more value of big data analytics is revealed according to the various applications.

**Step3:** Data analysis in Smart Grid

# Data Analysis Techniques:

There are basically three ways to analyze 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 information communicating in the designed Street Light System.***

# Reason of not using other two approach:

There is some disadvantages of Relational Database and the Real-Time Database as well.

# Disadvantages of relational database:

### Data storage isolated and separated:

According to entity relationship model a data record is saved as a single line. If we use relational database to store the data, the increasing demand of accessing the large amount of data will causes pressure.

### 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 date are indexed into relational databases, which consumes large amount of system resources.

### Unable to meet the demand of real-time processing of data:

Time series data generated in real time must be loaded into the database and accessed via the database. In case of massive data, relational database can’t handle high speed loading and meet the demand.

# Disadvantages of Real time database approach:

### Unable to support management of large scale measuring points:

In case of national or provisional smart grid, they need to support the measuring point on the scale of hundred millions or even billions.

### Insufficient scalability:

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

### 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.

For the above discussed disadvantages of Relational Database and Real-Time database, we prefer to use Hadoop (Hbase).

Overview of Hadoop

Hadoop is an Apache open source framework written in java that allows distributed processing of large datasets across clusters of computers using simple programming models. The Hadoop framework application works in an environment that provides distributed storage and computation across clusters of computers. Hadoop is designed to scale up from single server to thousands of machines, each offering local computation and storage.

# Hadoop Architecture:

At its core, Hadoop has two major layers namely-

* Processing/Computation layer (Map-Reduce), and
* Storage layer (Hadoop Distributed File System).

# Map Reduce:

Map-Reduce is a parallel programming model for writing distributed applications devised at Google for efficient processing of large amounts of data (multi-terabyte data-sets), on large clusters (thousands of nodes) of commodity hardware in a reliable, fault-tolerant manner. The Map-Reduce program runs on Hadoop which is an Apache open-source framework.

# Hadoop Distributed File System:

The Hadoop Distributed File System (HDFS) is based on the Google File System (GFS) and provides a distributed file system that is designed to run on commodity hardware. It has many similarities with existing distributed file systems. However, the differences from other distributed file systems are significant. It is highly fault-tolerant and is designed to be deployed on low-cost hardware. It provides high throughput access to application data and is suitable for applications having large datasets.

Apart from the above-mentioned two core components, Hadoop framework also includes the following two 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.

# 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 −

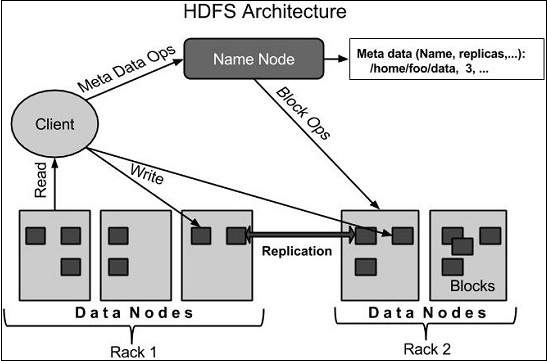
* Data is initially divided into directories and files. Files are divided into uniform sized blocks of 128M and 64M (preferably 128M).
* These files are then distributed across various cluster nodes for further processing.
* HDFS, being on top of the local file system, supervises the processing.
* Blocks 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 automatic 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.

# Namenode

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.

# 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.

# Block

Generally the user data is stored in the files of HDFS. The file in a file system will be divided into one or more segments and/or stored in individual data nodes. These file segments are called as blocks. In other words, the minimum amount of data that HDFS can read or write is called a Block. The default block size is 64MB, but it can be increased as per the need to change in HDFS configuration.

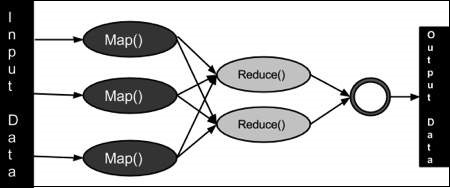
# What is MapReduce?

MapReduce is a processing technique and a program model for distributed computing based on java. The MapReduce algorithm contains two important tasks, namely Map and Reduce. Map takes a set of data and converts it into another set of data, where individual elements are broken down into tuples (key/value pairs). Secondly, reduce task, which takes the output from a map as an input and combines those data tuples into a smaller set of tuples. As the sequence of the name Map-Reduce implies, the reduce task is always performed after the map job.

The major advantage of Map-Reduce is that it is easy to scale data processing over multiple computing nodes. Under the Map-Reduce model, the data processing primitives are called mappers and reducers. Decomposing a data processing application into *mappers* and *reducers* is sometimes nontrivial. But, once we write an application in the Map-Reduce form, scaling the application to run over hundreds, thousands, or even tens of thousands of machines in a cluster is merely a configuration change. This simple scalability is what has attracted many programmers to use the Map-Reduce model.

# The Algorithm

* Generally Map-Reduce paradigm is based on sending the computer to where the data resides!
* Map-Reduce program executes in three stages, namely map stage, shuffle stage, and reduce stage.
  + **Map stage** − The map or mapper’s job is to process the input data. Generally the input data is in the form of file or directory and is stored in the Hadoop file system (HDFS). The input file is passed to the mapper function line by line. The mapper processes the data and creates several small chunks of data.
  + **Reduce stage** − This stage is the combination of the **Shuffle** stage and the **Reduce** stage. The Reducer’s job is to process the data that comes from the mapper. After processing, it produces a new set of output, which will be stored in the HDFS.
* During a Map-Reduce job, Hadoop sends the Map and Reduce tasks to the appropriate servers in the cluster.
* The framework manages all the details of data-passing such as issuing tasks, verifying task completion, and copying data around the cluster between the nodes.
* Most of the computing takes place on nodes with data on local disks that reduces the network traffic.
* After completion of the given tasks, the cluster collects and reduces the data to form an appropriate result, and sends it back to the Hadoop server.



Function Performed after data analysis

We have created a Hadoop environment to analyze the information communicating between smart grid and street lighting network. The data are stored in the Hadoop Distributed file system and we developed the map reduce algorithm in such way that the following operation are performed:

# Fault detection:

The information about any fault in the network or any equipment fault is send as data to the Hadoop distributed file system. These information is analyzed to locate the position of the fault in network.

# Distribution of Power:

This is a concept of smart grid which focuses on the operation and system reliability for distribution of power at different level of network. A successful smart grid has the capability to localize and isolate the units to which power must be distributed as requirement with a reduced functional time and an efficient way. Under this concept, increasing volume of operational data have been collected from advanced metering infrastructure (AMI).

# Electric device Life-Cycle Management:

A failure in any device in the network may cause catastrophic blackouts in the system. Therefore, the life-cycle management of electric devices based on an accurate estimation has been included in this stable and reliable smart power grid. The existing diagnosis methods mainly focus on limited state parameters with the threshold-based diagnosis. To take information of system operation and meteorological conditions into state estimation analysis, three classical algorithms for association rule mining are discussed in (Sheng et al., [2018](https://energyinformatics.springeropen.com/articles/10.1186/s42162-018-0007-5#CR89)), namely, Apriori, AprioriTid and AprioriHybrid. The rule mining methods are combined with probabilistic graphical model for potential failure prediction.

# Power quality monitoring:

As a worldwide issue, Electric power quality (PQ) refers to the magnitude, frequency and waveform of voltage and current in power system and highly

Map Reduce Algorithm

# 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;

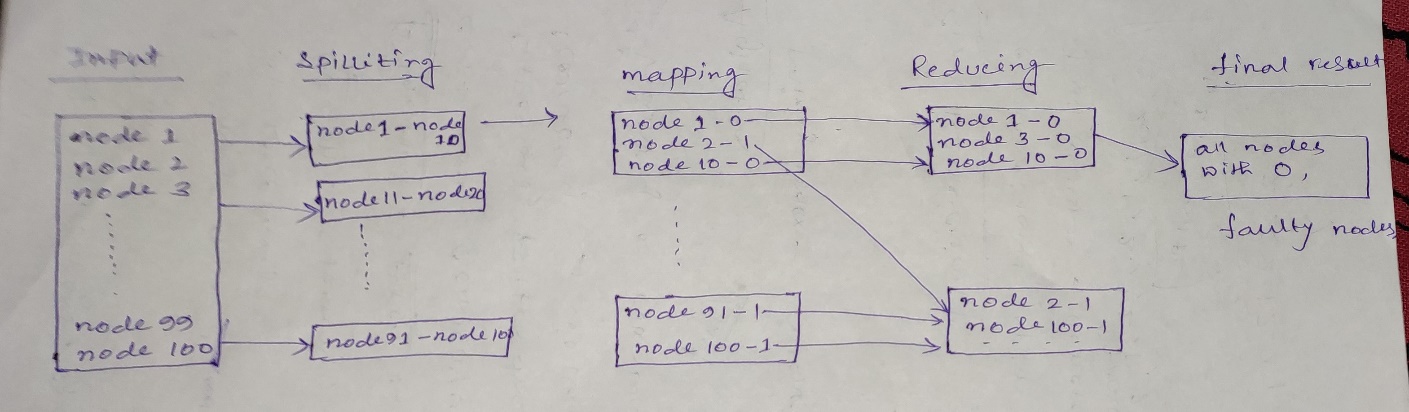
Current Range;

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 huge amount of data is splitted into small pieces and processing will be performed in parallel. This small pieces will be assigned in multiple number of nodes.

Now the operation is to check 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 the 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 final result.