Solar Energy Analytics Using Internet of Things

B. Vikas Reddy  
Department of ECE, SRM University, Chennai, Tamil Nadu, India.

Sai Preetham Sata  
Department of Mechatronics, SRM University, Chennai, Tamil Nadu, India.

Sateesh Kumar Reddy  
Department of ECE, SRM University, Chennai, Tamil Nadu, India.

Bandi Jaswanth Babu  
Department of ECE, SRM University, Chennai, Tamil Nadu, India.

Abstract  
The current scenario of today’s solar energy ecosystem is that, it is highly unstructured and localized. There are about 50 solar power plants in India but none of them are connected in a manner that there would be a method to perform analytical analysis of the solar energy produced. Today, with the advancements in sensor technology it is a very viable option to connect the solar energy systems to the cloud (internet) with the help of Internet of Things. Once these systems are connected to the cloud, the analysis of the performance, productivity and efficiency can be calculated very easily. With the software-technology of Big Data it is also possible to predict possible problems and failures with ease. Once, all the known solar energy systems are connected to the internet it can then be used to monitor these systems at a global level. This paper aims at finding a possible and viable method to connect the Solar based systems to the Cloud and perform analytical operations to increase efficiency of Solar Energy.

Keywords: Solar Energy, Cloud Services, Internet of Things (IoT).

Introduction  
As of 2011 the world’s energy consumption is estimated to be 10 terawatts (TW) per year, and by the year 2050, it is expected to be about 30 TW [1]. With such high demand for energy it is imperative that Solar Energy is going to be a major player in the energy race. We are going to build more and more solar farms and we are going to connect them to the grids. With global solar alliance of around 120 countries in the COT 21 Paris summit there is going to be a lot of Solar based projects that are going to come up [2]. As of now more than 12.67 MW of solar based energy have so far been installed for voltage support of weak grids, and for peak load saving and as diesel saving and Solar device based industrial production has touched a level of 7 MW/year[3].

With such an increase in solar based technology, a system has to be developed that can be used to monitor and analyse the entire solar infrastructure so as to increase efficiency and profitability of the Solar Energy. Around 14% of solar energy systems face a major fault every year and stop working all together leading to more than half of all residential solar systems a significant performance problem [4]. Now if we are able to ascertain these kinds of failures or at least figure out the trends that lead to such failures then we can obliviously make effort to building more robust and adaptable infrastructure. Taking into consideration that storage batteries are not 100% efficient and can lose almost 30-40% energy as heat while charging and discharging it is the best option to use all the produced energy in real time and not resort to energy storage, but as of today these kind of operation is very difficult for the lack of real-time analytical data about where energy is surplus and where it is deficient. The latest development in the field of micro-electronics and Internet of Things gives us the ability to connect all of the infrastructure to the internet at a very low power consumption and cheap price.

Data Collection  
The next step to the puzzle of an efficient analytical engine to perform analytics on the solar system is the speedy, reliable and energy efficient mode of collection of data. The different parameter that are essential include, but not limited to:

- Solar Panel tilt angle(Irradiance)
- Wind Factors
- Ambient Temperature
- Location

So for a basic analytical setup of a solar analytics system we need to obtain information about the voltage and current...
which is being generated by the solar panels. Along with this data we would also be requiring the information about the wind direction, the ambient temperature sensor, tilt angle sensors.

We use sensors like the LTC2990 - Quad I2C Voltage, Current and Temperature Monitor14-Bit ADC sensor which acts as our primary sensor unit which is connected to a very low cost and power efficient (~110mW) Broadcom BCM2835 application processor (Raspberry Pi Zero) using a simple universal protocol like the I²C. The ARM processor is the then used to acquire the sensor data and then convert it to the required format for transfer over to the cloud.

One of the best high performance IaaS (Infrastructure as a Service), Google Compute Engine is used as the cloud end point which is responsible for connecting to all the data source nodes using a light weight sensor messaging protocol MQTT (MQ Telemetry Transport). By adopting this kind of network system we can efficiently collect a lot of data and generate a data-set large enough for performing analytics and compute predictive algorithms.

Cloud and Big-Data

The amount of data which will be generated from a fully functional system is huge and it’s impossible for any conventional database to store such huge amount of data. For a 500MW power plant with each solar panel generating around 200W we get around 25,00,000 solar panels.

MQTT has smallest packet footprint of around 60 bytes and along with payload data it would amount to around 200 bytes [5]. If the latency of the system is set to 1s then the total amount data that is produced amounts to 8 Gb/s.

The solution to such a problem comes from cloud services like the Google Cloud BigTable, which is fast, fully managed, massively scalable NoSQL database service which is optimized for web, mobile, and IoT applications and which involve terabytes to petabytes of data. A Bigtable is a sparse, distributed, persistent multidimensional sorted map. The map is indexed by a row key, column key, and a timestamp; each value in the map is an uninterpreted array of bytes. Each random read involving the transfer of a 64 KB block over the network from Google File System to a tablet server, only used asingle 1000-byte value [7].

(row:string, column:string, time:int64) → string

The BigTable uses the above storage pattern to store its data. The row is a unique string which identifies each data-set, the column contain the different parameter data along with the timestamp. The timestamp is used for versioning of the data based on time.

This intrinsic property of BigTable is apt for storing Solar data in a meaningful manner under a large datasets as the data is sorted lexicologically by the row key, it has a method of storing large number of parametric columns and supports time indexing. The three way row, column, and time based indexing along with load balancing makes it very efficient to deal with data generated from IoT sensors and devices as they generate data at a steady state over a long period of time along which are intrinsically multi-parametric.

UsingBigTable we should able to:

- Calculate the energy that is being generated in all the connected solar energy systems.
- Analysis of energy generation pattern
- Fault/ Problem detection
- Real-time visualization of the solar systems
- Implement Google Prediction API

Figure 1: Data Acquisition Architecture

Data Model

The format in which data is sent across the network and the model in which it is stored and handled play as major part in building a scalable application. Taking into consideration the use case of solar energy system, JSON (Javascript Object Notation) can be chosen as one of the best format for communication between the nodes and cloud. As the system is very versatile and contains a lot of variables, what we have tried to do is to formulate a schema that can be used as a standard to communicate between the nodes and the server in a solar analytics system.

Referring to Figure 2, the model comprises of JavaScript objects which store information in a concise key-value pair hash table form. Every data packet which is sent by the unconstrained node has these entities

- uid: A unique-id which identifies the particular node over the entire network. This is can be generated by the universal uuid4 algorithm.
- gid: A group id which will be used to combine together multiple node end points to a single logical entity.
- os: This contains the information about the Operating Software that is responsible to connecting the nodes to the Cloud and the Sensors
- sensor: It contains an array of sensor objects which associate to each of the sensor that is connected to the node
  - id- Id of the sensor
  - type – current/voltage/temperature
  - value – sensor value
  - unit – the metric unit of the value
  - timestamp – the time when the sensor reading was taken
The next step in the process is to define a BigTable structure that will be able to store the data received in the efficient manner. Figure 4 shows the manner in which the data will be stored. The row name corresponds to the uid of the end point node. The contents column family contains the different properties namely, the gid, os, sensor array, meta and location. The contents columns will have multiple versions, at timestamps T1 and T2 and so on.

The way in which the data store works is that whenever a new node is connected to the system a new row is created and inserted into the table along with all the columns. And whenever an existing node sends new data, then it is stored into the existing node with a new time stamp. Using this kind of a database we don't lose any data about the state and can store time based information in absolute manner decreasing the reading and writing time.

**Analytics**

The last and final piece to the entire system is to perform analytics on the data collected. For the process we are using the MapReduce algorithm which is an associated implementation for processing and generating large data sets with a parallel, distributed algorithm on a cluster. The map reduce algorithms operate in 2 steps.

- Map function is applied to every row in the BigTable data set. This produces a new key-value pair for each row based on the function then combines all the items with the same key and then groups them together.
- Reduce function is then applied in parallel to each of the group formed from the previous step and then generated a collection of value in same domain.

Now to implement the basics of our analytic engine for calculating the energy that the solar panels are producing we use MapReduce algorithm to first extract the voltage and current reading from the data set. Then, using the Map function we find out the instantaneous power and finally by we use reduce function to calculate the average power. So in this manner we would be able to find the energy consumed by the system to a very high degree of accuracy.

```
function map is
  input: integer K1 between 0 and T, representing the time
  input: uid, the node unique id
  for each data in the BT(UID,K1) do
    let POW is data.voltage*data.current
    produce one output record (POW,s)
  end function

function Reduce is
  for each input record (POW,s) do
    Accumulate in S the sum of POW*C
    Accumulate in C the sum of C
    repeat
      let A be S/C new
      produce one output record (A, new)
    end function
```

**Figure 3:** BigTable Storage Model

**Figure 4:** MapReduce for Power

We have designed the following pseudocode for application for calculation of energy over a stipulated period of time.
instantaneous power after the application of the MapReduce Function. Now, after this computation is over, what we can do is multiply T to the average power obtained hence obtaining the energy consumes.

The other applications like real time monitoring and Failure detection can hence also be computed by deploying further machine algorithms in proper manner.

**Conclusion**

Solar panels and Solar energy has been prevailing from a decade along with their shortcomings. In the recent years, the booms in micro-electronics has made a huge impact in increasing computational power and cost of embedded electronics. With the development of ARM based platforms like Raspberry Pi and Intel Galileo, it has become very easy to induce intelligence to things. Keeping this in mind we have tried to find a probable model for connecting the Solar Energy system with these micro-electronic systems to give birth to the Internet of Things. We have designed an architecture for connecting the individual solar units to the internet along with providing them with sensors that can be used to measure their efficiency. As an overall design these panels become the part of a huge network of panels that can talk to each other and behave in an intelligent fashion. This would lead to real-time knowledge about the operations and detect failures in an early stage.

With the formulation of a standard data schema we should be able to make more and more devices that communicate with cloud services without worrying about any proprietary protocol. The Schema would help us concentrate what on the data is about more that how to send the data. In this manner that Cloud servers also cares less about what format the data is going to come in and dedicate more processing power towards the analysis and data and learn from the data. With cloud specific BigData algorithms like the MapReduce would help us to analyse huge amount of data with ease and at a very high speed. Further more and more algorithms can be formulated to make sense of the data so collected and help in increasing the efficiency of the solar energy system.

Finally we would state the if the motioned procedure is followed to setup an analytic system then a very efficient Solar analytics system could be built at a very low cost and at a very high efficiency rate.

**References**