AI based Smart Meter for Future Homes

Aditya Mantri, Sarthak Jain, Nishant Jani, Nachiket Trivedi
Department of Computer Engineering,San Jose State University
San Jose,California

***Abstract*—As we know that power consumption is increasing day by day so we need to create technologies which can help us in utilizing our resources wisely. Here we have proposed a technique using the particle sworm algorithm through which one can monitor the power consumption of each of their devices based on which it provides the optimal schedule to utilize that device keeping in mind all the devices installed at a particular home. The overall objective is to schedule multiple devices of the entire house in such a way that the overall power consumption of the house in the peak hours is minimized. The user would get charged minimal amount for their electricity consumption.**

***Index Terms*—particle sworm algorithm**

I. Introduction

In today's world the consumption of electricity has been increasing very quickly and the resources which are utilized for it's production are depleting quickly. Thus due to this gap the charges per unit of electricity are increasing every year. Many large electricity producing companies find it difficult to cope up with this huge demand of electricity. We have suggested an approach which helps in reducing the overall electricity consumption of a particular home during peak hours by scheduling some devices to run during non-peak hours. The reason behind this is that during the peak hours the charge of electricity per unit is maximum while that during the non-peak hours is significantly low. Thus if a user wants to utilize an appliance but it's not that urgent to use then he/she can schedule it to run later on so that the cost if running the same device would reduce significantly than when it would run during peak hours. For example:- Washing machine is a very common appliance which everyone uses in their day to day use. It is an appliance which utilizes quite a lot of power in order to work so running that appliance during the peak hours when the rate are maximum would charge the user highly as compared to utilizing the same appliance during non-peak hours. Our algorithm provides the user with this result where in they can get to schedule multiple appliances in such a way that the overall cost occuring to the user gets reduced.

II. Methodology Followed

We will be providing the user with the smart IOT based devices like: smart plugs and a tablet. The smart plugs would be controlling whether a particular device should be enabled or disabled as per their schedule computed based on the algorithm. The tablet would act as an interface to the user where in the user would be able to add the list of devices in his home and then that data would be computed on our algorithm in the cloud. Then the results would displayed to the user on their device and once the user enables scheduling mode for that device the system would automatically schedule the device based on the time which the algorithm had provided it.

A. Technology and tools used

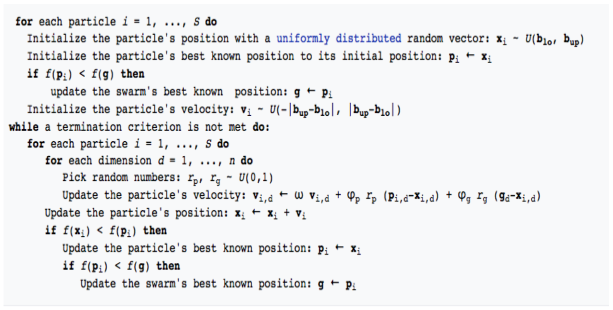
- IBM Cloud
- Particle Swarm Algorithm
- Apache Kafka
- Docker (containerization of application)
- Python Flask
- ReactJS and NodeJS (web application)
- MongoDB (Database)

III. What is Particle Swarm Algorithm?

Particle swarm is an optimization algorithm which performs iterative computations on a given problem equation which it internally creates and then based on that equation it provides the final result as to which parameters selection would lead to the most optimal answer. It approaches to a problem by considering various solutions of a given problem which are here considered to be particles and then moving these particles around the search space in order to find out solution to a problem based on a mathematical equation with particle's position and velocity as parameters. The search space can vary and so would be it's effect on the final output of the result obtained.
Each particle has it's own search space which would provide a local optimal solution while in the bigger picture we have multiple particles so we should consider global search space in order to reach a global optimum answer. Here we have the problem of finding the optimal time schedule for each of the devices which are used by a particular user. Thus we consider various devices as particles and then we find global optimal answer which would provide us the minimum cost for operation of each devices based on the schedule the algorithm has computed. Thus we frame the mathematical equation here in such a way that the total cost of operation of the devices turns out to be minimum by varying various parameters like different schedules which in turn would be having different cost per 1 unit of electricity. So the algorithm would perform various combinations of all the schedules and then come to a solution which would deliver the minimum electricity bill.

A. Algorithm

As we had previously stated that various potential solutions to out problem act as particle and then based on the mathematical equation of the particle's position and velocity optimization is performed through an iterative process. The movements of these particles is around their local best search space along with that of the global search space so when improved positions are found then the particles movement manages to move that way.



```
for each particle $i$ = 1, ..., $S$ do
  Initialize the particle's position with a uniformly distributed random vector: $\mathbf{x}_i \sim U(\mathbf{b}_{lo}, \mathbf{b}_{up})$
  Initialize the particle's best known position to its initial position: $\mathbf{p}_i \leftarrow \mathbf{x}_i$
  if $f(\mathbf{p}_i) < f(\mathbf{g})$ then
      update the swarm's best known  position: $\mathbf{g} \leftarrow \mathbf{p}_i$
  Initialize the particle's velocity: $\mathbf{v}_i \sim U(-|\mathbf{b}_{up}-\mathbf{b}_{lo}|, |\mathbf{b}_{up}-\mathbf{b}_{lo}|)$
while a termination criterion is not met do:
  for each particle $i$ = 1, ..., $S$ do
     for each dimension $d$ = 1, ..., $n$ do
        Pick random numbers: $r_p, r_g \sim U(0,1)$
        Update the particle's velocity: $\mathbf{v}_{i,d} \leftarrow \omega\, \mathbf{v}_{i,d} + \varphi_p r_p (\mathbf{p}_{i,d}-\mathbf{x}_{i,d}) + \varphi_g r_g (\mathbf{g}_d-\mathbf{x}_{i,d})$
     Update the particle's position: $\mathbf{x}_i \leftarrow \mathbf{x}_i + \mathbf{v}_i$
     if $f(\mathbf{x}_i) < f(\mathbf{p}_i)$ then
        Update the particle's best known position: $\mathbf{p}_i \leftarrow \mathbf{x}_i$
        if $f(\mathbf{p}_i) < f(\mathbf{g})$ then
           Update the swarm's best known position: $\mathbf{g} \leftarrow \mathbf{p}_i$
```

Fig. 1. Brief Overview of the Algorithm [1]

The above figure gives a brief overview regarding how the particle swarm algorithm works and how it comes to an optimal solution. So each particle would be having 2 search spaces a local one and another global one. Based on how each of these search spaces better performs the particle moves towards that search space. So as each and every particle align themselves towards their best position where they can lead to an optimal solution all these particles start converging towards a particular area. Then later on they all merge into a single point.
Here instead of having position of a point as a parameter we have timing/schedule for running a device as a parameter for the mathematical equation of the problem which needs to be optimized to get the final solution.

B. Convergence

The algorithm would converge to a single point when we are close to obtaining an optimal solution to our problem.

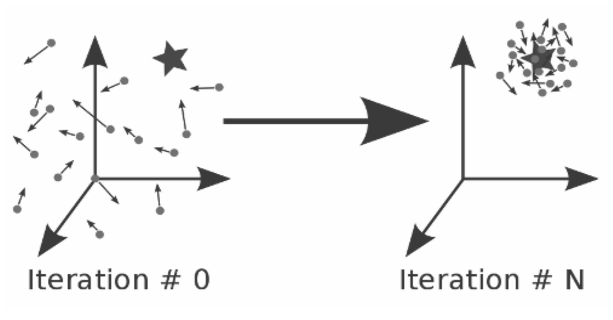
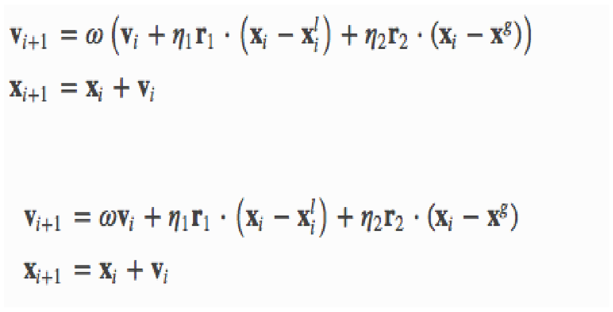


Fig. 2. Convergence of particles [2]

As we can see in the figure stated below that how scattered the particles are initially and how they gradually converge as we perform iterations over the mathematical equation of the problem. Initially all the particles are totally separate and in different directions having different velocity and position. But as each particle starts searching their search space they get to know where they can find good solution so then iteratively each particle goes towards a common position of the problem. Thus one needs to perform many iterations in order to reach to that point.



$$\mathbf{v}_{i+1} = \omega\left(\mathbf{v}_i + \eta_1 \mathbf{r}_1 \cdot \left(\mathbf{x}_i - \mathbf{x}_i^l\right) + \eta_2 \mathbf{r}_2 \cdot (\mathbf{x}_i - \mathbf{x}^g)\right)$$
$$\mathbf{x}_{i+1} = \mathbf{x}_i + \mathbf{v}_i$$

$$\mathbf{v}_{i+1} = \omega \mathbf{v}_i + \eta_1 \mathbf{r}_1 \cdot \left(\mathbf{x}_i - \mathbf{x}_i^l\right) + \eta_2 \mathbf{r}_2 \cdot (\mathbf{x}_i - \mathbf{x}^g)$$
$$\mathbf{x}_{i+1} = \mathbf{x}_i + \mathbf{v}_i$$

Fig. 3. Standard Mathematical equation used [3]

The above equations represent the standard mathematical equations which are used during the particle swarm algorithm. Here each xi represents the ith particle while the v represents the velocity of that particle. Here we consider time/schedule as a parameter in order to find an optimal solution so for each device we check in different time slots what is it's power consumption amount generated.

IV. Apache Kafka

- It is an open source stream processing software and is mainly written in Java and Scala.
- It has an immutable commit log from which you can subscribe to it and publish data to any number of systems or applications. It is highly scalable, fault tolerant system which is widely used for applications which have a large number of users.

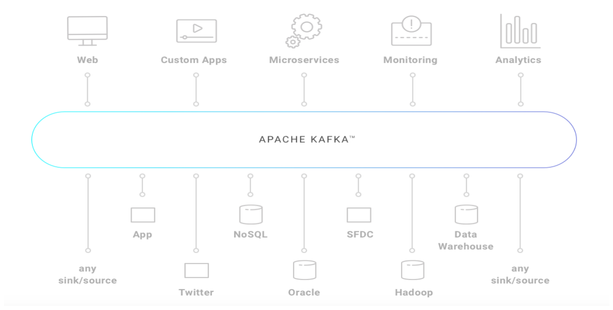


Fig. 4. Overview of uses of Kafka [3]

Above image shows the application of Apache Kafka in various areas. Apache kafka consists of mainly 4 API's as follows:

- Producer: Gives access to an application to publish streams of records.
- Consumer: Gives access to subscribe to topics and process the streams of records.

- Stream: Converts input stream to output to produce results.
- Connector: Helps in establishing connection between the consumer of the data stream and then one who produces it. In our project we have made the API's distributed and robust by integrating Apache Kafka into our backend server.

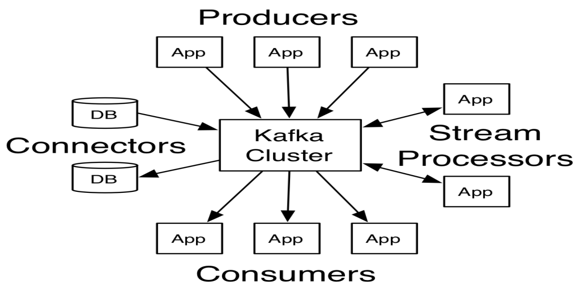


Fig. 5. View of how Kafka working [4]

V. IBM Cloud

IBM Cloud is the cloud computing services offered by IBM. It includes Infrastructure as a service(Iaas), Platform as a service(Paas) and Software as a service(Saas) accessible through different cloud models like: public, private or hybrid cloud.

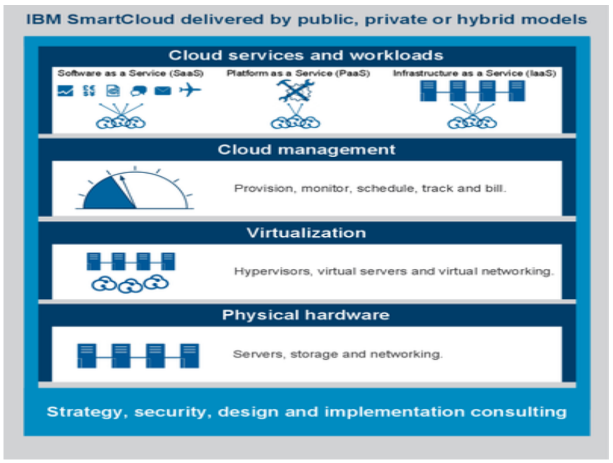

IBM SmartCloud delivered by public, private or hybrid models
Cloud services and workloads
Software as a Service (SaaS)
Platform as a Service (PaaS)
Infrastructure as a Service (IaaS)
Cloud management
Provision, monitor, schedule, track and bill.
Virtualization
Hypervisors, virtual servers and virtual networking.
Physical hardware
Servers, storage and networking.
Strategy, security, design and implementation consulting

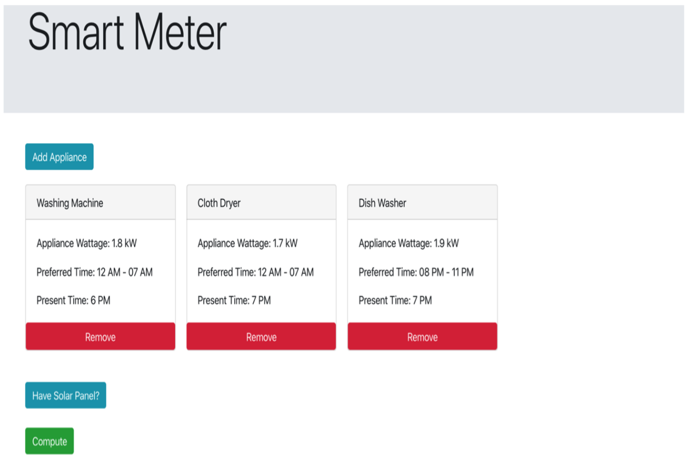
Fig. 6. Brief Overview of the Algorithm [5]

In the above figure we can see the various levels of the IBM cloud. At the top most level we have: Cloud services and workloads like: Saas,Iaas and Paas. Then we have cloud management which undertakes the task of resource provisioning, monitoring and billing. Then there is a virtualization layer where there are various hypervisors, virtual servers and virtual networks. Then at the bottom most level we have the physical hardware which includes the servers, storage and various networking layers.

A. Application

We have created web application which is hosted on the cloud so that it can easily be accessed from anywhere and even can handle large number of client requests. We have used NodeJS and ReactJS for creating the web application. The application server where in the computation using the particle swarm algorithm takes place is hosted on IBM cloud. The web application is also hosted on web application.
The web application has it's database as MongoDB and the application is made to suppport distributed systems as Apache kafka has been integrated in it's internal architecture.

The user when lands on to the home page of the application the user would be able to see all the services provided by us. The user would then need to provide the device which they have installed in their house. By selecting that device the user would get redirected to their dashboard. So if a user has already been logged into their account they would directly get redirected to their dashboard page else they would be redirected to login into their account first. If the user is a first time user then they need to register themselves first and then login into their account. Once the user gets into their dashboard page they would be able to see all their devices which they have added. Once they click on the compute now button on the page their request gets redirected to the cloud server where our particle swarm algorithm is running. The algorithm would perform computation from that device details sent to it by the application and then after on it returns the results back to the application. The result would show the optimal time at which a device must be run in order to get minimum operational cost for that particular device.


Smart Meter

Add Appliance

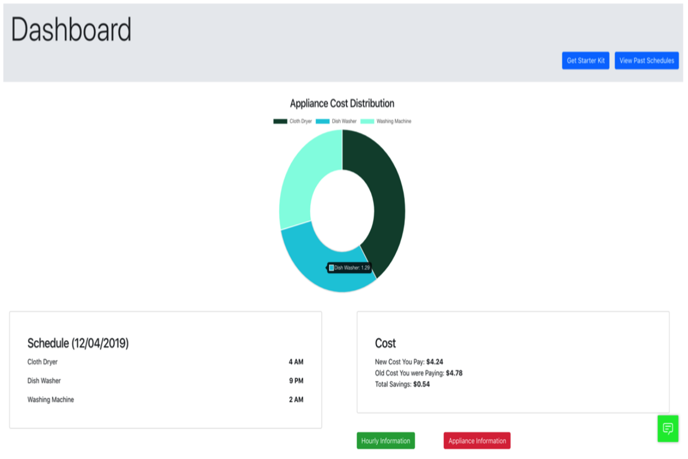
Washing Machine	Cloth Dryer	Dish Washer
Appliance Wattage: 1.8 kW	Appliance Wattage: 1.7 kW	Appliance Wattage: 1.9 kW
Preferred Time: 12 AM - 07 AM	Preferred Time: 12 AM - 07 AM	Preferred Time: 08 PM - 11 PM
Present Time: 6 PM	Present Time: 7 PM	Present Time: 7 PM
Remove	Remove	Remove

Have Solar Panel?

Compute

Fig. 7. Adding an appliance

The above photo shows the way user would be able to see their devices. Here the term device refers to the appliances which the user wants to schedule using our optimization algorithm.
The user dashboard is shown in the above image. The user would be able to see their devices which he had decided to


Dashboard

Get Starter Kit | View Past Schedules

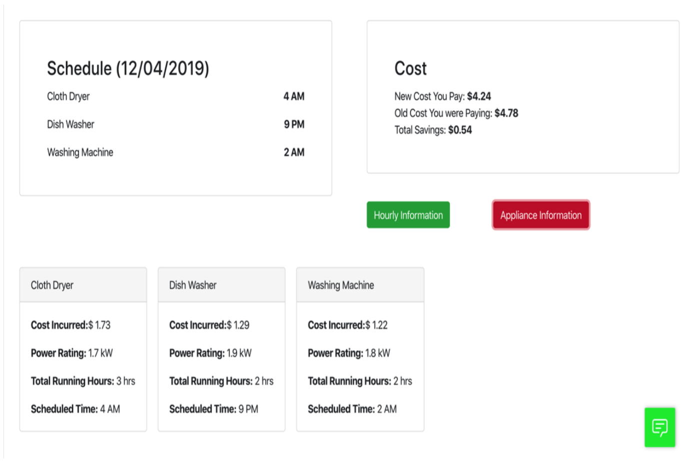
Schedule (12/04/2019)	
Cloth Dryer	4 AM
Dish Washer	9 PM
Washing Machine	2 AM

Cost
New Cost You Pay: **$4.24**
Old Cost You were Paying: **$4.78**
Total Savings: **$0.54**

Hourly Information | Appliance Information

Fig. 8. User dashboard

schedule. The dashboard provides the scheduling time of each of their device along with the total cost incurred by running of those devices. It also shows past schedules along with hourly cost information. It shows the current total cost of running those devices also the previous cost of running those same devices. Thus the user can know how much savings does he get.The user can also request a starter kit for their use. The user is also provided with appliance information which shows how much the particular appliance would consume power and that particular appliance's cost for running it.


Schedule (12/04/2019)	
Cloth Dryer	4 AM
Dish Washer	9 PM
Washing Machine	2 AM

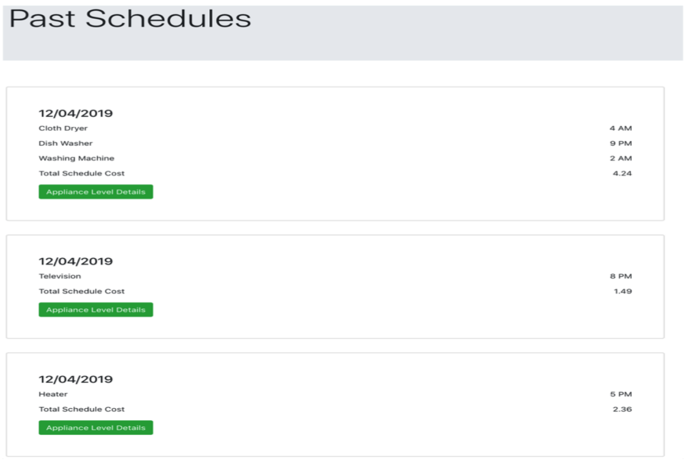
Cost
New Cost You Pay: **$4.24**
Old Cost You were Paying: **$4.78**
Total Savings: **$0.54**

Hourly Information | Appliance Information

Cloth Dryer	Dish Washer	Washing Machine
Cost Incurred:$ 1.73	**Cost Incurred:**$ 1.29	**Cost Incurred:**$ 1.22
Power Rating: 1.7 kW	**Power Rating:** 1.9 kW	**Power Rating:** 1.8 kW
Total Running Hours: 3 hrs	**Total Running Hours:** 2 hrs	**Total Running Hours:** 2 hrs
Scheduled Time: 4 AM	**Scheduled Time:** 9 PM	**Scheduled Time:** 2 AM

Fig. 9. User dashboard appliance info

The above image shows the detailed information regarding each and every device. Here we can see 3 devices which are namely: cloth dryer, dish washer and washing machine. The detailed information provides the cost incurred for running that device along with the device's power consumption, running hours and the scheduled time for that particular device.This detailed view can provide the user with more detailed information regarding their devices. This can help the user check on which time which particular device is being used so that if a user has priority for a particular device then they can plan accordingly.


Past Schedules

12/04/2019	
Cloth Dryer	4 AM
Dish Washer	9 PM
Washing Machine	2 AM
Total Schedule Cost	4.24

Appliance Level Details

12/04/2019	
Television	8 PM
Total Schedule Cost	1.49

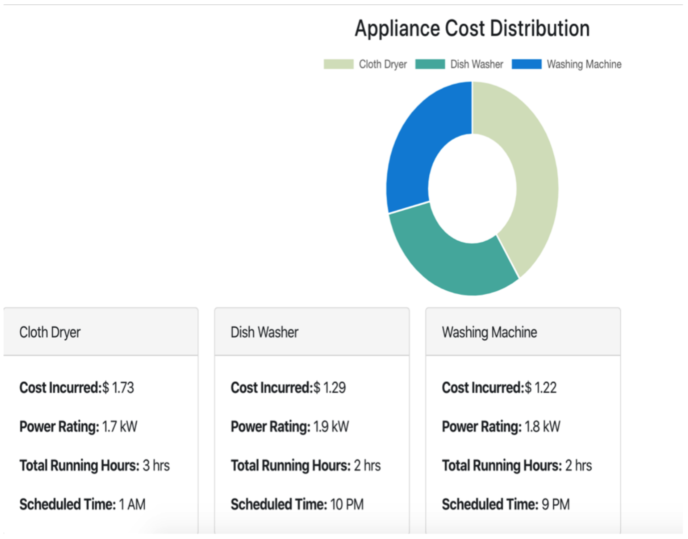
Appliance Level Details

12/04/2019	
Heater	5 PM
Total Schedule Cost	2.36

Appliance Level Details

Fig. 10. User past schedules information

The above image shows the list of all the past schedules of the user. So that if a user is planning to schedule their devices which they had previously scheduled together they can get an estimate of amount of savings they would receive so that they can plan accordingly.



Cloth Dryer	Dish Washer	Washing Machine
Cost Incurred:$ 1.73	**Cost Incurred:**$ 1.29	**Cost Incurred:**$ 1.22
Power Rating: 1.7 kW	**Power Rating:** 1.9 kW	**Power Rating:** 1.8 kW
Total Running Hours: 3 hrs	**Total Running Hours:** 2 hrs	**Total Running Hours:** 2 hrs
Scheduled Time: 1 AM	**Scheduled Time:** 10 PM	**Scheduled Time:** 9 PM

Fig. 11. Detailed past schedules information

The above image shows the detailed cost estimation of the appliances in the past. It shows for how much time did those devices were on and what was their previous schedule along with the cost incurring at that time. So that when a user is scheduling a particular device they can know that what would be its power consumption along with it what would be the amount of cost savings which would take place when they would be scheduling it. The detailed past schedule information can be quite helpful to the end user and give statistical information to them.

The image below shows the present location of all the users of our application. It gives the admin an overview regarding

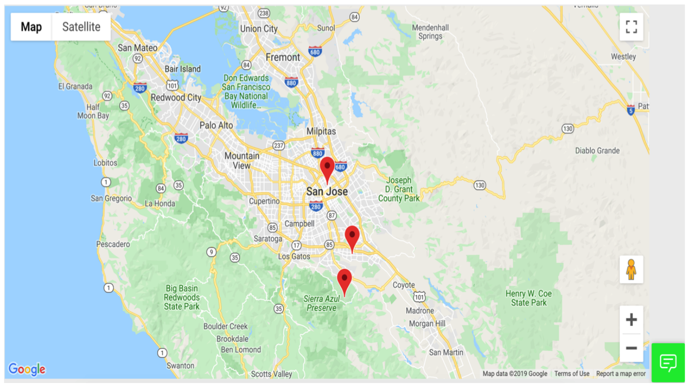
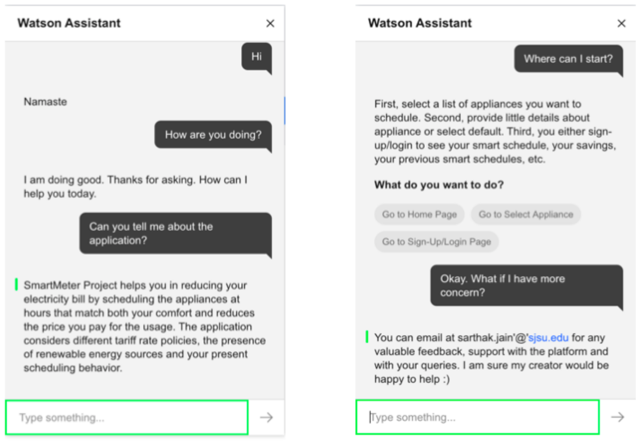


Fig. 12. User Location Information

the places where their end users are located.


Watson Assistant

Hi

Namaste

How are you doing?

I am doing good. Thanks for asking. How can I help you today.

Can you tell me about the application?

SmartMeter Project helps you in reducing your electricity bill by scheduling the appliances at hours that match both your comfort and reduces the price you pay for the usage. The application considers different tariff rate policies, the presence of renewable energy sources and your present scheduling behavior.

Type something...

Watson Assistant

Where can I start?

First, select a list of appliances you want to schedule. Second, provide little details about appliance or select default. Third, you either sign-up/login to see your smart schedule, your savings, your previous smart schedules, etc.

What do you want to do?

Go to Home Page | Go to Select Appliance | Go to Sign-Up/Login Page

Okay. What if I have more concern?

You can email at sarthak.jain'@'sjsu.edu for any valuable feedback, support with the platform and with your queries. I am sure my creator would be happy to help :)

Type something...

Fig. 13. IBM Watson AI assistant

The above image shows the AI based chatbot assistant which we have integrated in our project. We have made use of IBM Watson for this purpose. It has some predefined answers to some questions so that user can get answers for their questions at any time of the day. The smart chatbot is quite responsive and efficient in providing answers to almost all questions of the users.

ACKNOWLEDGMENT

REFERENCES

[1] https://en.wikipedia.org/wiki/Particle$_{s}warm_{o}ptimization$
[2] https://esa.github.io/pagmo2/docs/cpp/algorithms/pso.html
[3] https://www.confluent.io/what-is-apache-kafka/
[4] https://kafka.apache.org/intro
[5] https://en.wikipedia.org/wiki/IBM$_{c}loud_{c}omputing/media/File$: $SmartCloud_{m}odel.png$