**Project Charter**

Professors Alberto Lamadrid, **Shalinee Kishore**, Shamim Pakzad, Larry Snyder: Simulation of Electric Transit

| **Project Team:** | Ryanne Cox, Matt Culbertson, Aarushi Singh |
| --- | --- |
| **Date Last Revised:** | 5/5/2023 |
|  |  |

1. **Executive Summary**

The state of California has mandated that public transit agencies must have zero emissions bus fleets by 2040. This poses new challenges for transportation agencies that are used to planning around gas powered vehicles. In particular, the Valley Transit Authority (VTA) in Santa Clara has purchased ten electric buses, but is unsure of how best to implement them.

Professor Kishore, along with other faculty and students, are using data analysis and modeling to determine how to best transition to electric buses. Currently they are modeling the transit routes in the area with SUMO software, however the platform does not support controlling the state of charge of the busses. Prior data analysis found on sparse data sets struggled to find a correlation between energy composition and weather, although it did determine that the driver has a large impact.

Moving forward, our group hopes to build on this research. Importantly, new, higher resolution, data is available. We will be using simPy for simulating the buses, which will allow the simulation to model charging patterns and energy consumption. Various python libraries, including numPy will be used for data analysis. Regression models such as linear regression and machine learning among others will be used to further draw conclusions from the data.

1. **Business background**

The Valley Transit Authority (VTA) needs to deploy a fleet of zero emission buses by 2040 in order to align with new state laws. Currently, they are testing with ten electric buses, but are unsure how to best utilize them. Throughout the day, the cost of energy changes, so the VTA is seeking a way to charge the buses when electricity is cheapest while still meeting the community's needs. The strategy the VTA is following is to track and forecast how much energy buses are using and to create a simulation of the bus route involving the charging station. The end goal for the VTA is to minimize energy and cost wastes.

1. **Current Solution**

Currently, SUMO is being used to simulate the routes of the buses. However, SUMO does not support simulating charging or state of charge data. For this reason, the VTA is not using any information from the simulations. This limits the use of their 10 electric buses to only 3-5 running per day.

The current results of the data analysis took into account the daily temperature and the driver of the bus. The data used was somewhat sparse and only included the state of charge at the start and end of the day for the bus. The temperature analysis did not find a strong conclusion and revealed that the bus operator had more of an impact on discharge rates.

1. **Proposed Solution** 
   1. **Technical Architecture**

Simulation - The goal of the simulation portion of the project is to develop a simPy module that considers the many variables that affect energy use on the buses. The simulation will be slowly deployed in order to properly develop more involved features as the simulation advances. The ultimate goal of the simulation is to be able to predict the amount of charge a bus will come back with if assigned to a particular route.

Data Analysis: The goal of the data analysis portion of the project is to create a model to predict the amount of energy buses use.

* 1. **Software Design Description (SDD) (a list of major modules in the solution and what each one will do)**

**Data Analysis and Machine Learning**

* **Graphs and charts**
  + Code was written in Jupyter notebook files to create various graphs, charts and animations that would help initially analyze the data
* **Aggregate and combine data**
  + Code that combined data from various sources into one data structure to be used for graphing and modeling
* **Models**
  + Multiple regression models to predict the consumption of a bus. Code also including methods to tune the models, like filtering and feature selection

**Simulation – there are folders that have been created in this process.**

* **Driver\_analysis**
  + The driver\_analysis.py file took instanced where operator ID was recorded, averaged a unique operator’s kwh usage, and created a plot showing the operator versus their average kwh usage. There are two visuals created from this file, one with the operator ID on the x-axis and one with kwh on the x-axis. Both images show the range of kwh and how they are connected to the operators.
* **Kwh\_analysis**
  + Finding\_kwh\_dist.py determined the best distribution to model the kwh usage. The fitter found that a double gamma distribution was the most accurate distribution. The double gamma distribution was then used in the bat\_drain.py file to determine from the distribution what the kwh usage should be for a particular bus. The kwh\_ploy.py file created a double gamma distribution image, which is also saved in the folder.
* **Other\_dist**
  + The files in the other\_dist folder have functions similar to finding\_kwh\_dist.py, just with different values (miles and battery percent used). The output distribution corresponding to the miles distribution is ised to inform the overall simulation.
* **Other .py files in the bcs folder of the repository**
  + Charge\_time.py determines the relationship between charge time and change in charge. I found the relationship to be somewhat linear, as seen in the charge\_vs\_time\_cleaned.png.
  + Bat\_drain.py is the main deliverable for the simulation. It incorporates elements from many of the files mentioned above and uses them to inform the bus attributes. bat\_drain.py creates 10 bus instances and assigns them routes, drivers, start and end times, and start and end charges based on the distributions created by the data.
  1. **Development Architecture**

Python will be used extensively. For the simulation, simPy will be used. Eventually, matplotlib and seaborn may be used to create a visual representation of the data. For the data analysis, numPy and pandas will be used. Additionally, scikit learn and pyTorch may be used for machine learning.

Jupyter notebooks – helps to create a story, can put images in

* 1. **Exclusions and Limitations** 
     1. Traffic and real time route info website api integration
     2. Traffic effect on energy consumption
     3. GPS prediction
     4. Solar LSTM with probabilistic forecasting
     5. Viriciti will depend on what the team finds out
     6. Simulation Exclusions:
        1. Need to charge, but may all be busy (queuing system), amount of energy in local system, drive for x amount of time (time of day, traffic), variance on battery drainage based on temp, weather etc.
        2. Solar availability throughout the day, collective solar charges battery, electricity grid and charging
        3. Testing how the optimization works against a different way of chaging

Currently, these elements of the project are beyond our scope. As we progress through the project, some of these may be added to the scope of our project.

* 1. **Business Value**

This project is contributing to Prof. Kishore’s and others’ research for the transit department (VTA) in Santa Clara County, California. The value the VTA will get from our work is more efficient operation and charging of their buses. VTA needs to be compliant with the mandate, and this project is supposed to help the VTA learn how to reach compliance. Additionally, the VTA is hoping to be able to use the new zero emissions buses to benefit disadvantaged communities.

1. **Methodology and Schedule**
   1. **Methodology**

We will use scrum to communicate progress with one another due to the largely individual nature of our project work. Scrum will also be used to crowdsource solutions to points where team members are stuck. Our progress is also visible to other team members via github.

* 1. **Schedule**

We will be meeting every Tuesday and Thursday during class time with just the 3 of us where we will discuss our progress and plan out what we should do as a team next. Then on Friday from 3-4 with the whole team and show our progress to our sponsors and plan for the future.

1. **Risk Management**
   1. **Testing**

Comparing data analysis to simulation results will be the primary testing tool.

Simulation is the testing for the theoretical optimization frame work. The simulation tests the results of the data analysis and can be compared to the raw data.

Testing the simulation involves running the simulation several times in order to determine trends when comparing the simulation to the raw data.

* 1. **Security**

We will all login to the Viriciti data with Nick’s email and password. To aid security, this login information will be used only to download the needed data, all interaction with the data will happen with the local copies. Basic precautions will be taken to keep the Viriciti data private.

All team documents have been shared in a google drive that was created with a shared email [ebcapstone23@gmail.com](mailto:ebcapstone23@gmail.com) with a password that all team members know. This ensures that no one person will lose access to important documents.

The simulation files and some of the data analysis work have been saved in a github repository. Only the three team members have edit access for the repository.

* 1. **Other risks**

Loss of progress – To do prevent losing progress we will use github to be able to access old versions of our code in case anything happens. The github also prevents anyone from losing data in the event that a computer crashes. We also used Jupyter Notebooks which allowed us to keep track of all the progress of our code. We have separate Jupyter Notebooks for each part of our project like Getting Rid of Outliers, Regression Models RMSE with KFold, PGBM Models with and without weather data, ext. This overall made our code more organized and prevents losing any progress as it saves automatically.

1. **Deliverables**

For the simulation, the deliverables are in the bcs folder in our github repository, linked here: [link to bcs folder](https://github.com/rec224/ElectricTransitSim/tree/main/bcs). In the folder, there are several files that I have run. Finding\_kwh\_dist.py determined the best distribution to model the kwh usage. The fitter found that a double gamma distribution was the most accurate distribution. The double gamma distribution was then used in the bat\_drain.py file to determine from the distribution what the kwh usage should be for a particular bus. The kwh\_ploy.py file created a double gamma distribution image, which is also saved in the repository. The driver\_analysis.py file took instanced where operator ID was recorded, averaged a unique operator’s kwh usage, and created a plot showing the operator versus their average kwh usage. There are two visuals created from this file, one with the operator ID on the x-axis and one with kwh on the x-axis. Both images show the range of kwh and how they are connected to the operators. The files in the other\_dist folder have functions similar to finding\_kwh\_dist.py, just with different values (miles and battery percent used). The output distribution corresponding to the miles distribution is used to inform the overall simulation. Charge\_time.py determines the relationship between charge time and change in charge. I found the relationship to be somewhat linear, as seen in the charge\_vs\_time\_cleaned.png.

Bat\_drain.py is the main deliverable for the simulation. It incorporates elements from many of the files mentioned above and uses them to inform the bus attributes. bat\_drain.py creates 10 bus instances and assigns them routes, drivers, start and end times, and start and end charges based on the distributions created by the data.

The primary deliverable from the data analysis will be the conclusions drawn, not specific code. These conclusions can be accompanied by charts and the code used to generate them. The priority of the factors to analyze will be guided by the sponsors.

1. **Appendix**

The scope of the project will expand if/when tasks are completed.