

ENERGY SYSTEMS AND CONTROL

BOOK OF PROJECTS | SPRING 2019

Abstract

We are extremely proud to present the book or projects for "Energy Systems and Controls". This course pursues an aggressive objective. It introduces fundamentals of systems & control and optimization within the context of energy systems applications.

The course content includes: (i) mathematical models, (ii) state estimation, (iii) optimization, (iv) machine learning, and (v) optimal control (e.g. dynamic programming).

The students engage in team projects, in which they must apply one or more fundamental areas to an energy systems application.

We organized this project book into thematic areas. We wish you a pleasant reading.

Prof. Scott Moura & GSI Dong Zhang
smoura@berkeley.edu; dongzhr@berkeley.edu

Table of Contents

Part I: Decarbonized Electric Grids

- Integrated Hybrid Energy Generation and Storage Optimization of Cost and Carbon for Berkeley, CA (pp. 3 – 20)
- Portfolio Optimization for City Emission Goals (pp. 21 – 29)

Part II: Electric Vehicles

- Electric Vehicle Charging Stations Demand and Placement in New York City (pp. 30 – 44)
- Optimal Pricing of an Electric Vehicle Charging Station Based on a Flexible Charging Schedule (pp. 45 – 60)
- Optimal spatial and temporal schedule of EV fleet charging and discharging in the Bay Area (pp. 61 – 74)

Part III: Traffic, Trains, Planes, and Evacuation

- Traffic Dynamics Estimation Based on the Intelligent Driver Model (pp. 75 – 102)
- Control and Optimization of Electric Long-Haul Freight Trains based upon Route Timing, Power Consumption and Topography (pp. 103 – 116)
- Flight Path Optimization (pp. 117 – 132)
- Newport Beach Tsunami Evacuation Optimization (pp. 133 – 158)

Part IV: Green Buildings & Storage

- Comparative Analysis of Data-Driven Modeling for HVAC Systems (pp. 159 – 171)
- Optimizing Storage and Usage of Distributed Energy from Individual Households Through Machine Learning (pp. 172 – 195)
- Battery State of Charge Estimation (pp. 196 – 211)

INTEGRATED HYBRID ENERGY GENERATION AND STORAGE OPTIMIZATION OF COST AND CARBON FOR BERKELEY, CA

Bavisha Kalyan, Carteret Lawrence, Chandra Vogt, Christina Hyland and
Joseph Willenborg

I. ABSTRACT

The development of hybrid renewable electric power systems will be needed to meet the 2016 Paris Agreement climate goals. In this study, a convex optimization program is developed to integrate renewable electricity generation with distributed storage, while minimizing cost and carbon emissions from electric power. The model is applied to Berkeley, California to assess the feasibility of a grid independent system using solar, batteries and backup power. A second order cone program (SOCP) applies uncertain solar power and demand loads to determine the minimal economic cost of the system. The program accounts for capital and operational costs as well as a carbon tax to identify trade-offs between cost and emissions. The program is simulated over one representative day, using energy demand and climate data for Berkeley. A verification analysis is performed to assess the system's robustness to seasonal variation in load and solar inputs. From our analysis, we've determined that an stand-alone hybrid electric power system is a viable option to reduce costs and carbon emissions, with an approximate optimal cost of \$40,000 per day for the Berkeley case study.

II. INTRODUCTION

(a) Motivation & Background

In the United States, the electric power sector contributed 1.744 billion metric tons of CO₂ emissions in 2017, amounting to 34% of total US energy related CO₂ emissions. Currently, electricity consumption from the residential and commercial sectors accounts for 70% of total electricity related emissions in the United States.¹ As renewable generation and storage technologies become cheaper, a hybrid renewable generation and distributed storage system will be a viable option that can significantly reduce carbon emissions from electricity while bypassing costs of transmission infrastructure. However, increased integration of intermittent renewables in the energy supply yields an unprecedented challenge of maintaining reliability while still meeting peak demands.

Our goal is to model an independent stand-alone electric power system that minimizes the cost and carbon footprint of electricity while maintaining a reliable electric power supply for a sample population. In a stand-alone system, the operational capacity is matched to the demand, and storage or backup energy generation is used during peak demand periods. Stand-alone systems produce power independently of the utility grid, and are more suitable for remote locations where the grid cannot penetrate and there is no other source of energy.² Additionally, stand-alone systems are more suitable for local renewable energy generation, and have the potential to reduce carbon emissions from electricity production over time.

Our optimization program is developed to understand the feasibility, constraints, costs and emissions associated with a stand-alone system. It models intermittent power generation sources with storage which meets the demand of a defined population. We apply our model at the municipal scale for the City of Berkeley where 1,452 solar PV panels were installed between 2000 and 2014, with a cumulative capacity of 6,115 kW and a production of 10 million kWh/year. These solar PV installations have been estimated to offset over 5,000 metric tons of GHG emissions annually.³ While this is a good start, the city will have to do more to fully decarbonize its energy supply to meet clean energy goals and maintain energy security. In 2013, total residential energy demand for Berkeley, which has a population of approximately 120,000 people, was about 675 million kWh/year with 140,000 metric tons of GHG emissions annually.⁴ Thus, this project is motivated by Berkeley's goal to reach 100% renewables by 2035, as well as California's goal to reach zero-carbon electricity by 2045.⁵

(b) Relevant Literature

Hybrid energy systems have been investigated in the literature as a means of providing energy services, especially for isolated areas. This project expands upon previous literature, including a paper that optimizes an isolated hybrid energy system for a rural community in India.⁶ The analysis provides an assessment of optimal hybrid energy components in order to minimize life cycle cost. Optimization methods in this paper were used to initially develop the linear program for this project's optimization approach, which was then expanded into a second order cone program (SOCP) that included a carbon cost. Other studies have looked at optimizing multiple factors, including cost and emissions, including one applied to an electrical energy storage system on the island of Pantelleria.⁷ In this research, the authors utilize a multi-objective optimization algorithm to minimize grid energy losses, total electricity generation cost, and greenhouse gas emissions. Another paper also proposes a distributed optimization framework to assist planning decisions for multiple types of energy generation and storage technologies in stand-alone power systems.⁸ An additional resource used was a microgrid case study for the Oakland EcoBlock.⁹ The optimization program for Berkeley was based off of the SOCP used in the Oakland case study, which represents a grid connected scenario.

The relevance of renewable-integration and independent energy systems in our current climate situation is growing, and this is reflected in numerous studies looking at applications for different scales, regions, and associated challenges.^{10,11,12} A gap in the literature that we are hoping to address is using a SOCP with chance constraints related to the uncertainty of load and power generation to minimize total system cost that includes a cost of carbon. Additionally, this project looks to expand upon the past literature, which has focused on the role of hybrid energy systems in isolated, rural settings, and attempt to apply it to a more urban setting, that has the option of grid connectivity. This approach seeks to more robustly describe our energy system and make it valid in a real-world system with renewable generation and loads that are uncertain with time.

(c) Focus of this Study

This study investigates the viability of a combined renewable energy generation and storage system for Berkeley, CA. A convex SOCP with chance constraints will be used to minimize economic cost of the system as well as carbon dioxide emissions, while ensuring reliable power supply. The SOCP will output the composition of renewable energy generation and storage required and enables this program to be applied to future case studies.

(III) TECHNICAL DESCRIPTION

(a) Data Collection

Electricity load for Berkeley was determined using data from California Independent System Operator (ISO).¹³ Electricity load statistics for the state of California were scaled down to represent Berkeley based on population statistics as well as open data from the City of Berkeley.¹⁴ Electricity load was recorded for every day of a calendar year, on a half-hour time step, and the average value and standard deviation at each time-step was also determined and plotted, to use in the optimization program.

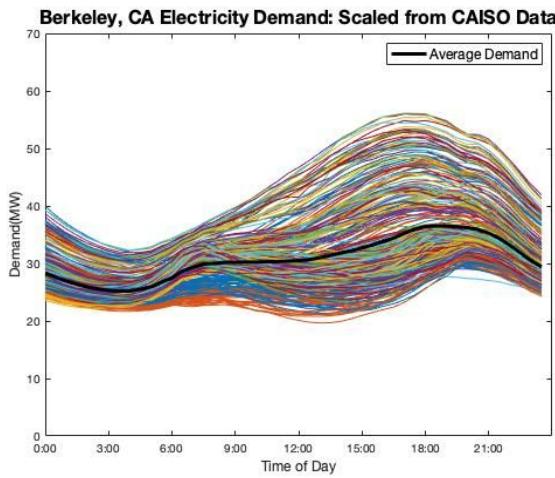


Figure 1: Total electricity demand for Berkeley, CA (April 2018-2019).¹²

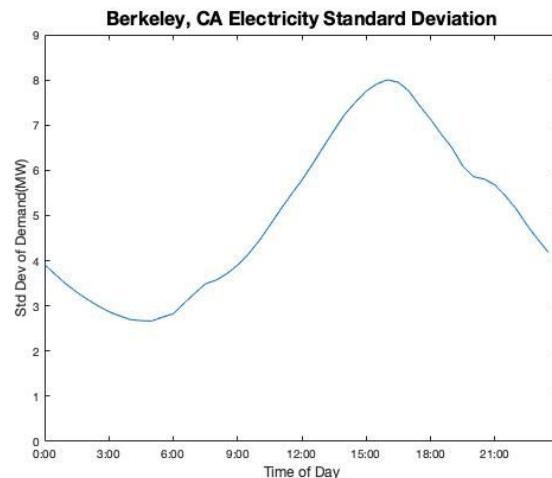


Figure 2: Standard deviation of electricity load for Berkeley, CA

Solar potential for Berkeley was determined using solar irradiance data from 1998-2015 provided by the National Renewable Energy Laboratory (NREL).¹⁵ An average solar irradiance for each day over this eighteen year period was determined, on a half-hour time step, and then a total average and standard deviation over the course of a single day was determined.

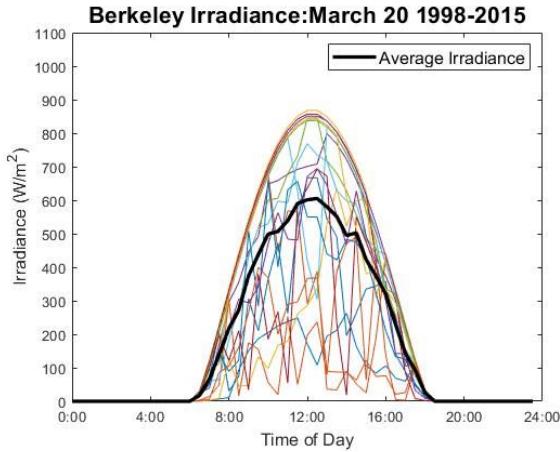


Figure 3: Berkeley solar irradiance for a single day over 18 years

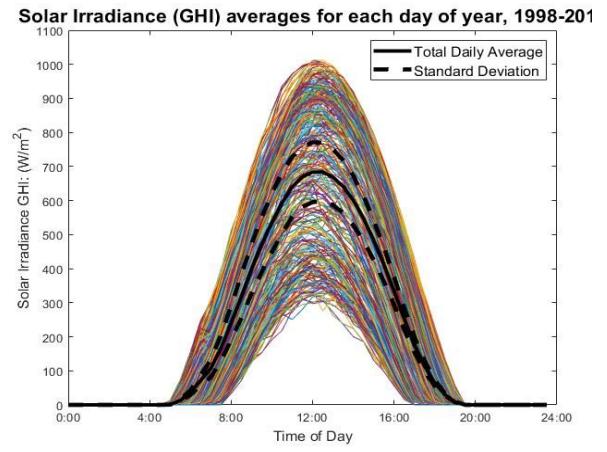


Figure 4: Total average for one year with bounds determined from the standard deviation.¹⁴

(b) Optimization Program Formulation

A convex SOCP with chance constraints was designed to find the most economical utilization of solar power generation, including solar curtailment, battery energy storage and backup power, as needed. Additionally, carbon dioxide emissions were taken into account through the use of a carbon tax. A simplified linear program was created to act as an initial verification of the setup (see Results section (e)). Here we will explain the SOCP used to account for the uncertainty associated with the solar generation and electricity load. Selected variables and their description are listed in Table 1. A comprehensive list of inputs and variables as well as associated values can be found in Appendix A.

s, b	[-]	Scale factors for solar PV and batteries
sc	[kW]	Curtailed solar generation
$S(k), L(k)$	[kW]	Solar generation and power load
$G(k)$	[kW]	Backup Power (Natural Gas)
$B_d(t), B_c(t)$	[kW]	Battery discharge/charge power
$E(k)$	[kWh]	Battery energy level
c_b, c_s	[\$/kWh], [\$/kW]	Cost of scaling battery and solar
$c_G(k)$	[\$/kW]	Time-of-use cost of natural gas or grid import

Table 1: Selected variables and parameters. Optimization variables in blue and uncertain variables in red.

While the solar generation and power load are uncertain random variables, the SOCP utilizes known characteristics of their probability distributions, explicitly the average and standard

deviation computed above. The SOCP formulation, with objective function and constraints is as follows:⁹

$$\begin{aligned} \text{minimize: } & C_s \cdot s + C_b \cdot b + \sum_{k=0}^N C_G [\bar{L}(k) - s \cdot \bar{S}(k) - B_d(k) + B_c(k)] \\ & + CO2_s \cdot s + CO2_b \cdot b + \sum_{k=0}^N CO2_G [\bar{L}(k) - s \cdot \bar{S}(k) + B_d(k) + B_c(k)] \end{aligned} \quad (1)$$

$$\text{subject to: } E(k+1) = E(k) + [\eta_c B_c(k) - (1/\eta_d) B_d(k)]\Delta t \quad \textit{battery dynamics} \quad (2)$$

$$0 \leq E(k) \leq b \cdot E_{max}, \quad E(0) = E_0 \leq E(N) \quad \textit{battery energy limits} \quad (3)$$

$$0 \leq B_c(k) \leq b \cdot B_{max}, \quad 0 \leq B_d(k) \leq b \cdot B_{max} \quad \textit{battery power limits} \quad (4)$$

$$s_{min} \leq s \leq s_{max} \quad \textit{solar scale limits} \quad (5)$$

$$b_{min} \leq b \leq b_{max} \quad \textit{battery scale limits} \quad (6)$$

$$0 \leq sc(k) \leq \bar{S}(k) - \bar{L}(k) \quad \textit{curtailment limits} \quad (7)$$

Upper bound chance constraint on G(k)

$$\sqrt{(\sigma^2_s(k) \cdot s^2 + \sigma^2_L(k))} \leq \frac{1}{\phi^{-1}(\alpha)} (s \cdot \bar{S}(k) - sc + B_d(k) - B_c(k) - \bar{L}(k) + G_{max}) \quad (8)$$

lower bound constraint on G(k)

$$\bar{L}(k) - s \cdot \bar{S}(k) - B_d(k) + B_c(k) \geq 0 \quad (9)$$

The objective function in equation (1) minimizes the daily capital and operating costs of the solar, battery (modeled as lithium ion), and backup power units in the first three terms, respectively. It also minimizes the daily carbon costs associated with each of these units in the final three terms. The backup power is calculated as the residual of the power balance equation, balancing the other load and generation terms for each half-hour timestep k from 0 to N, where N is 48, corresponding to every half hour over a single day. The solar and battery costs are taken as amortized daily costs of installation and operation over an expected lifetime and scaled by the optimized capacity. The backup power cost includes both generation cost and amortized capital and maintenance costs. Carbon costs were calculated from a carbon tax value, chosen as \$50/ton CO₂ for the initial analysis, and corresponds to life cycle emission factors for each unit.

Equations (2), (3), and (4) outline typical battery dynamics and constraints, taking into account energy losses during charging and discharging and specifying an initial and final energy level per day. Equations (5) and (6) correspond to the minimum and maximum capacities for the solar and battery units. Equation (7) specifies that solar curtailment is non-negative and less than the average solar generation minus the average load at each timestep, which specifies the curtailment

of excess generation. Equation (8) represents the relaxed chance constraint for the upper bound limit on backup power generation. It requires that the backup power satisfy the power balance, based on the probability distributions of load demand and solar input, at some high frequency, α , here chosen to be 95%. This could easily be increased for even higher system reliability. Φ denotes the cumulative distribution function. Note that solar curtailment is included in the power balance as it reduces total solar generation. It does not, however, affect the total cost of solar generation as calculated in the objective function. Equation (9) specifies a non-negative backup power generation.

(c) Isolated System Case Study Results

The SOCP program is solved using cvx in MATLAB. Note that solar curtailment is not included in the program for this analysis, but is included in the verification analysis. The program yields 1 and 0.25 as the optimal value for b and s , respectively, indicating that the battery unit is at full capacity, and the system is limited by storage. The corresponding solar and battery capacities, estimated from the scale factors, are shown in Table 2. With this 13 MW solar capacity, equivalent to about 5,000 square meters of solar panels, the economically optimized system provides 34% of Berkeley's power demand from solar generation at a cost of \$40,000 per day. Considering that Berkeley already has over 6 MW of solar installed, such a system is feasible to develop, especially with the decreasing cost of solar power. The equivalent 5,000 square meters of solar panels is less than 1% of the total surface area of Berkeley, and thus solar can be optimally placed in areas with high solar irradiance. The battery capacity of 8,000 kWh is also feasible and could easily be supplied in the future by parked electric vehicles connected to the power infrastructure. For context, a Tesla Model S has a battery capacity of 100 kWh.

b	s	Solar Capacity (m^2)	Solar Capacity (MW)	Cost (\$/day)	Battery Capacity (kWh)
1	0.25	5,000	13	40,000	8,000

Table 2: Results and optimization variables.

Solar generation and load are shown in Figure 5. Figure 6 shows the backup power generation over the course of the day, which drops to zero during the afternoon when solar generation is plentiful and peaks during the evening demand.

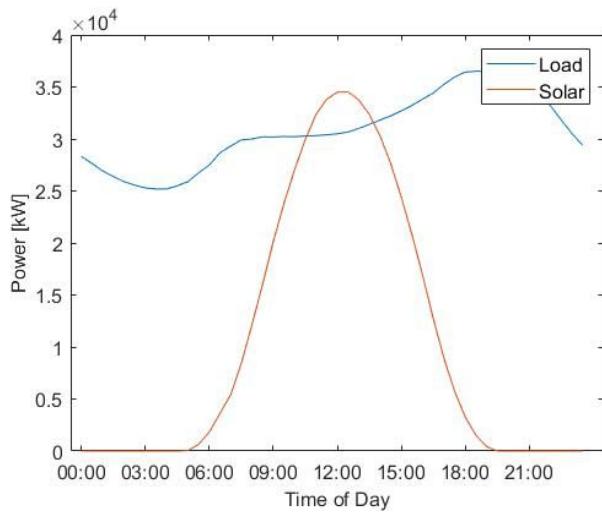


Figure 5: State of charge of the battery with time

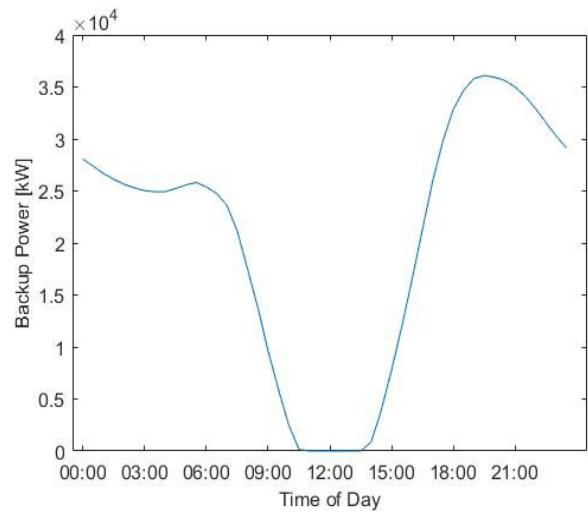


Figure 6: Battery charge and discharge with time

The battery dynamics are extracted to view the state of charge of the battery over the day. Figure 7 and 8 show the correlation between increase in load and battery discharge, as well as battery charging during the day. The battery energy capacity reaches its maximum charge at around 2pm when average solar generation remains relatively high. This indicates that increasing battery capacity could improve the reliability and sustainability of the system.

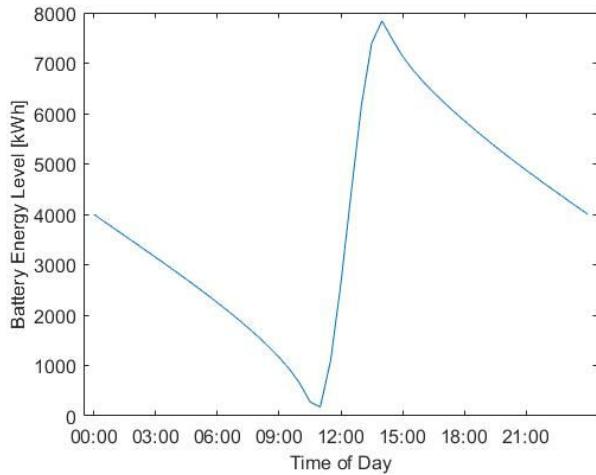


Figure 7: State of charge of the battery throughout the day for the grid isolated system

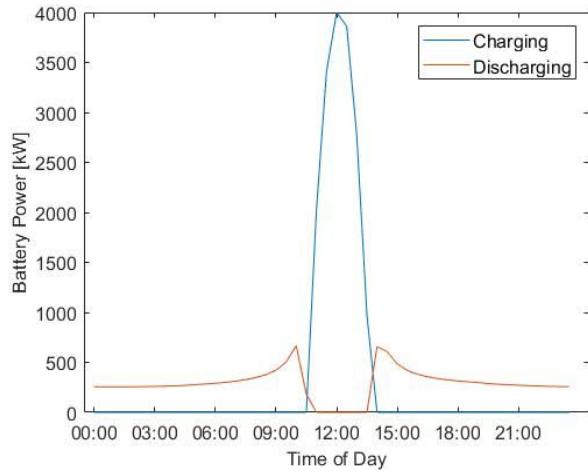


Figure 8: Battery charging and discharging times

Figure 9 shows the portion of solar generation curtailed throughout the day. Curtailing the excess solar power allows the system to turn off a portion of solar when there is excess generated.

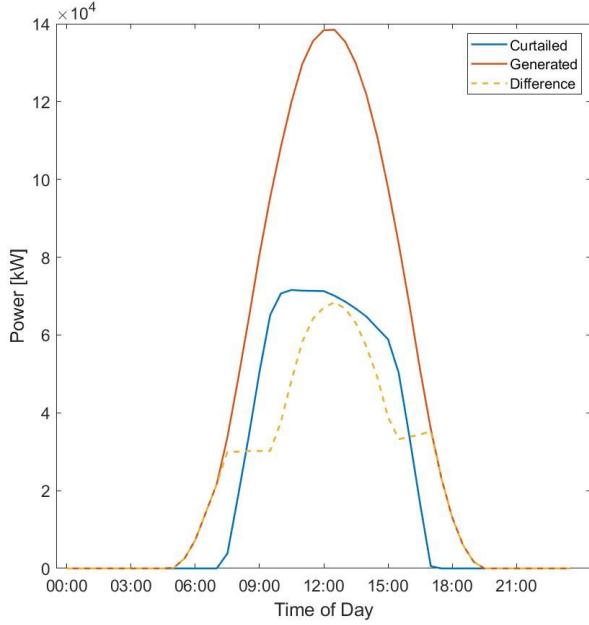


Figure 9: Effect of solar curtailment. The red line represents the total solar generation, the blue line represents the curtailed amount, and the dotted line is the net solar generation.

(d) Comparison to simplified non-Second Order Cone Program

The chance constraints described in equation (8), which account for uncertainty associated with solar generation and demand loads, are a defining characteristic of SOCP. This SOCP model is based off a simplified program, with the following objective function and constraints.⁹

$$\text{minimize: } C_s \cdot s + C_b \cdot b + \sum_{k=0}^N C_G \cdot G(k) + C02_b \cdot b + C02_s \cdot s + \sum_{k=0}^N C02_G \cdot G(k) \quad \text{CAPEX \& OPEX} \quad (10)$$

$$\text{subject to: } s \cdot S(k) - s_c(k) + B_d(k) - B_c(k) + G(k) = L(k) \quad \text{power balance} \quad (11)$$

$$E(k+1) = E(k) + [\eta_c B_c(k) - (1/\eta_d) B_d(k)] \Delta t \quad \text{battery dynamics} \quad (12)$$

$$0 \leq E(k) \leq b \cdot E_{max} \quad \text{battery energy limits} \quad (13)$$

$$0 \leq B_c(k) \leq b \cdot B_{max}, \quad 0 \leq B_d(k) \leq b \cdot B_{max} \quad \text{battery power limits} \quad (14)$$

$$s_{min} \leq s \leq s_{max} \quad \text{solar scale limits} \quad (15)$$

$$b_{min} \leq b \leq b_{max} \quad \text{battery scale limits} \quad (16)$$

$$0 \leq G(k) \leq G_{max} \quad \text{backup power limits} \quad (17)$$

The derivation of our SOCP model from the simplified model is found in Appendix C. In the simplified case, the backup power generation is an explicit variable and the power balance must be satisfied at all times. This formulation also requires known forecasted load and solar data, which is not feasible in practice, making the simplified model less robust than the SOCP. This program was run with the same parameters as the SOCP program, with simplified constraints, and yielded results listed in Table 3.

b	s	Solar Capacity (m ²)	Solar Capacity (MW)	Cost (\$/day)	Battery Capacity (kWh)
1	1	20,000	52	66,000	8,000

Table 3: Results and selected optimization variables for the non-SOCP formulation.

With this formulation, both the battery and solar capacities are maxed, and the solar capacity increased four times from the SOCP case. As a result, the cost increased by 65% to \$66,000 per day. This analysis shows the benefit of the SOCP formulation using statistical distributions and the more flexible chance constraint. Sacrificing a small margin of reliability (95% rather than 100% reliability) provides a significant economic savings. No model can be 100% reliable in real applications, so such a constraint is not practical. In addition, the isolated system can be combined with demand response to further mitigate the risk of power outages without having to increase system cost and infrastructure. The effect of increasing reliability closer to 100% would be an insightful future analysis.

(e) Verification through different Scenario Analyses

To test the robustness of this optimal system, the optimization program was applied to four specific dates of the year. The results are summarized in Table 4. The four dates chosen correspond to the summer and winter solstices and the spring and fall equinoxes, to best represent annual variability in available solar.

Date	b	s	Cost (\$/day)	Fraction Solar
Average	1	0.25	41,300	0.34
March 22	1	0.26	37,400	0.34
June 22	1	0.19	42,100	0.39
Sept. 22	1	0.24	47,800	0.32
Dec. 22	1	0.47	44,700	0.26

Table 4: Results of verification analysis of optimal system for different season scenarios. Fraction solar represents the percentage of the load that is being satisfied by solar generation.

The verification analysis reveals battery storage is the limiting factor. The recommended course of action would be to increase the amount of battery storage available to improve efficiency. With more storage, the system can utilize more of its potential solar capacity which is under utilized, ranging from 0.19 to 0.47. The highest cost is observed in September, as this date has the highest demand and also the smallest fraction of the load being supplied by solar generation.

(IV) DISCUSSION

(a) General Discussion

The findings of this report show the optimal cost and parameters for a grid independence for a community the size of Berkeley, CA. These costs take into account capital and operational costs as well as the cost of carbon, based on the hypothesis that a carbon tax will be utilized as a climate mitigation strategy. While the scope of this project is limited to Berkeley, with generation coming from solar and storage from batteries, the aim is to provide a framework that can be utilized on any scale and with any energy generation and storage options. In the coming decades, a shift towards a resilient and environmentally sustainable form of energy provision is necessary, and this program assess the viability of achieving climate goals.

The program additionally yields a tradeoff between CAPEX and OPEX costs and the cost of emissions. The costs of emissions increase proportionally with the tax as expected however, the CAPEX and OPEX costs decrease as the scale factors (s and b) decrease. However, at around a tax of \$180/ton CO₂e, the costs of emissions steeply increase while the CAPEX and OPEX marginal changes decreases. While Figure 10 does not reveal any information about the quantity of emissions, the relationship between emissions tax and CAPEX and OPEX shows the minimal obtainable CAPEX and OPEX costs to satisfy the demand. After this point, an increase in tax will only increase total cost and not change in the energy mixture. Further analysis of this relationship is needed to understand the optimal carbon tax.

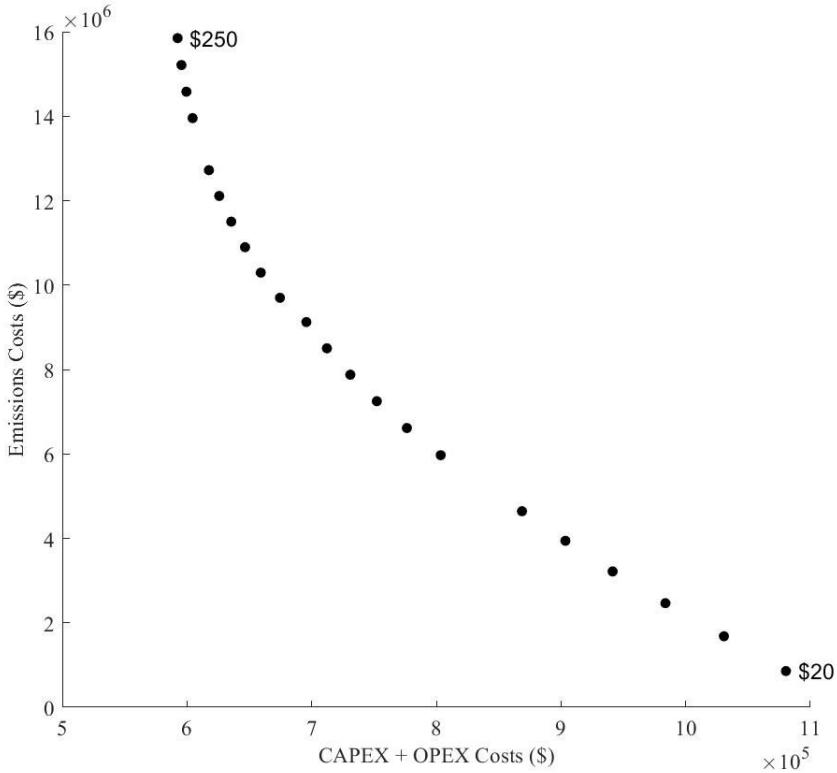


Figure 10: Cost comparison of emissions cost versus CAPEX and OPEX costs where each point is output of the optimization program for a ranging carbon tax (\$/ton CO₂e) from \$0-\$250 in increments of 10.

This project is a classic optimization problem, and presents an opportunity to apply a SOCP, to tackle problems associated with variable renewable energy generation and demand loads. The results comparing the SOCP with the non-SOCP demonstrates that for this case study, incorporating SOCP and chance constraints can lead to significant economic savings. This approach allows for a more robust program that can account for fluctuations in future power supply and demands. A prediction of optimal costs of energy generation and storage combinations is particularly important in scenarios where grid independence is necessitated, such as in remote, isolated, or island locations. Optimal cost predictions can also assist decision making for grid connected systems employing renewable energy and storage technologies.

(b) Future Work

One component of this project that can be expanded upon is the incorporation of carbon emissions into the model. A chief motivation of this project was to optimize the energy supply and demand with an emphasis on minimizing carbon emissions from energy generation and storage. However, our model limits an analysis of emissions to the use of a carbon tax. Therefore, future development would aim to investigate carbon emissions in more depth and dually optimize system cost and environmental impact and asses the tradeoffs.

Testing scenarios with different inputs and parameters, such as additional geographic locations, can aid in an analysis of our model's robustness. Our scenario analysis attempts to test temporal hardness and shows that optimal values vary depending on time of year and corresponding load and generation. The range of optimal values indicates a need to improve model robustness for temporal variability.

Additionally, because short term forecasting for both load and generation is more accurate than using historic averages, formatting this system into a dynamic program would allow for a higher accuracy than with the SOCP. By allowing the system to change its course of action depending on the current situation, it may more accurately manage the variable load and solar generation, in order to optimally meet demand and storage goals.

(V) SUMMARY

Our program is developed to analyze an isolated electric power system with renewable generators, distributed storage, and backup natural gas power generators, as isolated systems provide a viable solution for renewable energy integration. In this study, our model is applied to a sample population, with inputs of solar generators, lithium-ion batteries, and backup gas generators for Berkeley, CA. A simple linear optimization problem was transformed to a SOCP to account for the uncertainty in solar generation and demand loads for Berkeley. The optimal cost of \$40,000 and scale factors were found for this system and compared over a range of carbon tax values. The problem was also solved using inputs from different times of the year to assess the variation in the scale factors. This program is developed with the intention of being adaptable to any scenario, with respective energy generation, storage parameters and demands.

ACKNOWLEDGEMENTS:

We would like to thank Professor Scott Moura for all of his guidance and advice during this project. In addition, thank you to our GSI Dong Zhang for all of his assistance during this course and to the rest of the CE 295 class for suggestions and helping to create an encouraging learning environment.

REFERENCES

1. "Environment Baseline, Volume 1: Greenhouse Gas Emissions from the U.S. Power Sector." June 2016. U.S. Department of Energy.
<https://www.energy.gov/sites/prod/files/2017/01/f34/Environment%20Baseline%20Vol.%201--Greenhouse%20Gas%20Emissions%20from%20the%20U.S.%20Power%20Sector.pdf>
2. Kaundinya, D., Ravindranath, N.H, and Balachandra, P. "Grid-connected versus stand-alone energy systems for decentralized power - A review of literature." *Renewable and Sustainable Energy Reviews*, 13 (2009) 2041-2050.
3. "Energy Storage Exchange." (2019). *Energystorageexchange.org*,
<http://www.energystorageexchange.org/> (Feb. 15, 2019).
4. Berkeley Climate Action Plan: Tracking our Progress Building Energy Use – Solar PV
https://www.cityofberkeley.info/uploadedFiles/Planning_and_Development/Level_3_-_Energy_and_Sustainable_Development/Solar%20PV%20See-It.pdf (2015)
5. Roberts, D. (2019). "California Gov. Jerry Brown casually unveils history's most ambitious climate target". *Vox*, <https://www.vox.com/energy-and-environment/2018/9/11/17844896/california-jerry-brown-carbon-neutral-2045-climate-change> (Feb. 15, 2019).
6. Ashok, S. "Optimised Model for Community-Based Hybrid Energy System." *Renewable Energy*. 32. (2007): 1155-1164. <https://doi.org/10.1016/j.renene.2006.04.008>.
7. Ippolito, M. G. et al. "Multi-Objective Optimized Management of Electrical Energy Storage Systems in an Islanded Network with Renewable Energy Sources under Different Design Scenarios." *Energy*. 64 (January 1, 2014): 648–62.
<https://doi.org/10.1016/j.energy.2013.11.065>.
8. Yang, P., and A. Nehorai. "Joint Optimization of Hybrid Energy Storage and Generation Capacity With Renewable Energy." *IEEE Transactions on Smart Grid*. 5, no. 4 (July 2014): 1566–74. <https://doi.org/10.1109/TSG.2014.2313724>. (WAS 14)
9. Moura S., "Lec 05: Case Study: Microgrid Planning & Control." *ENE 2XX: Renewable Energy Systems and Control Lecture*. (2017).
10. Bagen and, R. Billinton. "Reliability Considerations in the Utilization of Wind Energy, Solar Energy and Energy Storage in Electric Power Systems." In *2006 International Conference on Probabilistic Methods Applied to Power Systems*, 1–6, 2006.
<https://doi.org/10.1109/PMAPS.2006.360204>.
11. Bernal-Agustín, José L., and Rodolfo Dufo-López. "Simulation and Optimization of Stand-Alone Hybrid Renewable Energy Systems." *Renewable and Sustainable Energy Reviews*. 13, no. 8 (October 1, 2009): 2111–18. <https://doi.org/10.1016/j.rser.2009.01.010>.
12. Krajačić, Goran. et al. "Planning for a 100% Independent Energy System Based on Smart Energy Storage for Integration of Renewables and CO₂ Emissions Reduction." *Applied Thermal Engineering*, Selected Papers from the 13th Conference on Process Integration, Modelling and Optimisation for Energy Saving and Pollution Reduction, 31, no. 13 (September 1, 2011): 2073–83.

13. California ISO: Current and Forecasted Demand.
<http://www.caiso.com/TodaysOutlook/Pages/default.aspx>. (2019). (WAS 16/6)
14. City of Berkeley Open Data: Community Energy.
<https://data.cityofberkeley.info/Natural-Resources/Community-Energy/jfw3-qg7m>. (2019).
15. NSRDB. National Renewable Energy Laboratory.
<https://maps.nrel.gov/nsrdb-viewer/?aL=UdPEX9%255Bv%255D%3Dt%26f69KzE%5Bv%255D%3Dt%26f69KzE%255Bd%255D%3D1&bL=clight&cE=0&lR=0&mC=3531736632923788%2C-73.125&zL=2>. (2019)
16. Mostert, C., Ostrander, B., Bringezu, S. and Kneiske, Tanja M., *Comparing Electrical Energy Storage Technologies Regarding Their Material and Carbon Footprint*, Energies 2018, 11, 3386; doi:10.3390/en11123386.
17. Turconi R., Boldrin A., Astrup T. (2013). “Life Cycle Assessment (LCA) of electricity generation technologies: Overview, comparability and limitations.” *Renewable and Sustainable Energy Reviews*. 28. (2013). 555-565.
18. Ran Fu, David Feldman, Robert Margolis, Mike Woodhouse, and Kristen Ardani, *U.S. Solar Photovoltaic System Cost Benchmark: Q1 (2017)* National Renewable Energy Laboratory (NREL) Technical Report NREL/TP-6A20-68925 September 2017.

APPENDIX A: Complete List of Variables and Parameters Used

Variable	Value	Units	Description
s, b	optimize	[-]	Scale factors for solar and battery size
$S(t)$	data provided	[kW]	Power generated from solar (NREL Data). ¹⁴ Incorporated a 15% efficiency for solar generation and 0.26 kW/m ² factor estimated for northern California solar resource of 5.5kWh/m ² day
$B_d(t), B_c(t)$	optimize	[kW]	Battery discharge/charge power
$G(t)$	optimize	[kW]	Backup Power (natural gas)
$L(t)$	data provided	[kW]	Power load of Berkeley (CAISO Data). ¹³
$E(t)$	optimize	[kWh]	Energy Level of Battery
c_b	0.00217	[\$/kWh]	Cost of scaling battery over a 20 year lifetime. ¹⁸
c_s	0.00317	[\$/kWh]	Cost of scaling solar over a 20 year lifetime. ¹⁸
c_G	data provided	[\$/kW]	Cost of back-up natural gas power
Δt	0.5	[hr]	Time Step
η_c, η_d	0.9	[-]	Battery charge/discharge efficiency
E_{max}	100,000	[kWh]	Battery energy capacity (varied by case study)
B_{max}	100,000	[kW]	Battery power capacity (varied by case study)
G_{max}	50,000	[kW]	Maximum backup power
s_{min}, s_{max}	[0,1]	[-]	Solar scale limits
b_{min}, b_{max}	[0,1]	[-]	Battery scale limits
CO_{2T}	50	[\$/ton]	Carbon Tax (unless varied)
CO_{2s}	2.8e-5*CO ₂ Tax	[tonsC _(e) /kW]	Solar CO ₂ lifecycle emissions. ¹⁶
CO_{2b}	10e-5*CO ₂ Tax	[tonsC _(e) /kWh]	Battery CO ₂ lifecycle emissions. ¹⁶
CO_{2G}	2.8e-5*CO ₂ Tax	[tonsC _(e) /kWh]	Backup power (natural gas) emissions
α	0.95	[-]	System reliability factor (fraction of time where power balance is exactly met)
$\bar{L}(t), \bar{S}(t)$	Data provided	[kW]	Average values of load and solar at half-hourly timestamps from NREL and CAISO
$\sigma_L(t), \sigma_S(t)$	Data provided	[kW]	Standard deviations of load and solar at half-hourly timestamps from NREL and CAISO

Table A1: List of variables, adapted and changed to fit the scope of this project, and descriptions.

APPENDIX B: Cost and Emissions Values

Storage Technology (2 MW capacity & 20 year life)	Kg CO ₂ e/MWh	LiI Energy Duration (hrs) (60 MW Capacity)	Cost \$/kWh
Lithium Ion (LiI) Battery	11		895
Second Life (LiI) battery	9		601
Compressed Air (CA) Storage	27		454
Power to gas (H ₂) Storage	47		380
Vanadium Redox Flow (VRF) Battery	53		Solar PV System Costs (\$/kW)
Power to gas Synthetic Natural Gas (SNG) Storage	92		
Sodium-Sulphur (NaS) battery	~160		

Table A2: Cost and carbon emissions of selected energy storage technologies.¹⁶

Carbon Emissions Comparison:

Electricity Source	Life-Cycle GHG Emissions (gC(e)/kWh)
Coal	218-355
Oil	145-245
Natural Gas	104-273
Nuclear	1-10
Biomass	2-35
Hydropower	1-5
Solar photovoltaic	4-52
Wind	1-11

Table A3: Carbon emissions of common electricity generating technologies, from an LCA standpoint.¹⁷

APPENDIX C: SOCP reformulation

Our model is developed to handle random solar and load inputs with chance constraints based on Gaussian distribution of independent random variables $L(k)$, $S(k)$:

$$L(k) \sim N(\bar{L}(k), \sigma_L^2(k)) \quad (\text{A1})$$

$$S(k) \sim N(\bar{S}(k), \sigma_S^2(k)) \quad (\text{A2})$$

We reduce our program by eliminating $G(k)$ from the objective function and backup power limits using $G(k) = L(k) - s \cdot S(k) - B_d(k) + B_c(k)$, and relax $-G_{max} \leq L(k) - s \cdot S(k) - B_d(k) + B_c(k) \leq G_{max}$ into chance constraint:

$$\Pr(L(k) - s \cdot S(k) \leq B_d(k) - B_c(k) + G_{max}) \geq \alpha \quad (\text{A3})$$

To express this as a SOCP constraint, we use $u = L(k) - s \cdot S(k)$ with mean $\bar{u} = \bar{L}(k) - s \cdot \bar{S}(k)$ and variance $\sigma_u^2 = \sigma_L^2(k) + s^2 \sigma_S^2(k)$ to rewrite (A3) as:

$$\Pr\left(\frac{u - \bar{u}}{\sigma_u} \leq \frac{B_d(k) - B_c(k) + G_{max} - \bar{u}}{\sigma_u}\right) \geq \alpha \quad (\text{A4})$$

The probability of (A4) is given by the normal cumulative distribution function (CDF):

$$\phi\left(\frac{B_d(k) - B_c(k) + G_{max} - \bar{u}}{\sigma_u}\right) \geq \alpha, \text{ where } \phi(z) = \frac{-1}{\sqrt{2\pi}} \int_{-\infty}^z e^{-t^2/2} dt \quad (\text{A5})$$

Consequently, (A4) can be expressed as:

$$\bar{u} + \phi^{-1}(\alpha) \cdot \sigma_u \leq B_d(k) - B_c(k) + G_{max} \quad (\text{A6})$$

Replacing mean $\bar{u} = \bar{L}(k) - s \cdot \bar{S}(k)$ and variance $\sigma_u^2 = \sigma_L^2(k) + s^2 \sigma_S^2(k)$ yields $\bar{L}(k) - s \cdot \bar{S}(k) + \phi^{-1}(\alpha) \cdot \sqrt{\sigma_L^2(k) + s^2 \sigma_S^2(k)} \leq B_d(k) - B_c(k) + G_{max}$ (A7)

Thus (A7) provides an upper bound constraint on the backup power generation.

Portfolio Optimization for City Emission Goals

Callie R. Clark, Salma Elmallah, Christian G. Miller, Peter Worley, and Hilary Yu

Abstract—Reducing emissions associated with the electricity usage of city operations, such as in city-owned and operated buildings, can help city governments meet their carbon reduction goals. The portfolio of options available to achieve these reductions include investments in renewable energy resources, such as solar PV, as well as energy efficiency upgrades. Deciding whether and how to invest in these technologies involves weighing a number of tradeoffs, particularly when the distribution utility is also city-owned. We develop an optimization tool to assist cities with municipal utilities in determining how much and where to invest in each technology by minimizing the cost to meet a desired level of emissions reduction. In determining the optimal portfolio, the tool considers generation and savings profiles and uncertainty in those profiles, generation and transmission contracts, tariffs, and feeder capacities. We demonstrate the utility of the tool by operationalizing it for the city of Columbus, Ohio.

Index Terms—Distributed power generation, Economics, Energy management, Optimization, Planning

I. INTRODUCTION

A. Motivation and Background

Taking a more central role in climate-forward policies, city governments across the United States have started to establish carbon goals targeted at reducing emissions associated with city operations. In a 2015 joint report, the World Wildlife Fund and Local Governments for Sustainability (ICLEI) listed 37 cities, including New York, Chicago, and Atlanta, as having already set goals to reduce emissions by 80 percent or more by 2050 [1]. Since June

Manuscript received May 10, 2019. This work benefited from discussions with Professor Scott Moura and feedback and comments from the CE295 Spring 2019 Energy Systems and Control class at the University of California, Berkeley.

Callie R. Clark is a graduating senior in the Department of Civil and Environmental Engineering, University of California, Berkeley, CA 94720 USA (e-mail: crclark@berkeley.edu).

Salma Elmallah is a PhD student in the Energy and Resources Group, University of California, Berkeley (e-mail: salmae@berkeley.edu).

Christian Miller is a graduating MS/MPP student in the Energy and Resources Group and the Goldman School of Public Policy, University of California, Berkeley (e-mail: cgmiller@berkeley.edu).

Peter Worley is a graduating MS student in the Energy and Resources Group, UC Berkeley (e-mail: peter_worley@berkeley.edu).

Hilary Yu is a PhD student in the Energy and Resources Group, UC Berkeley (e-mail: hilary.yu@berkeley.edu).

2017, more than 230 cities in the United States have had their mayors commit to the Paris Agreement goals in spite of the country's national-level withdrawal [2]. One approach that city governments (i.e. 'cities') may follow to achieve these goals include reducing emissions associated with municipal electricity usage, such as by streetlights or by the buildings they own and operate, including City Hall and school buildings. Such reductions can be achieved with various technologies, including solar photovoltaics and energy-efficient lighting. However, in deciding among these options, a common constraint is cost—that is, cities typically seek to implement the portfolio of technologies with the lowest cost, and often the lowest lifetime cost. When a city government also owns the local distribution utility, i.e. a municipal utility, calculating lifetime costs can be complicated by the fact that some technology investments can concurrently reduce costs for the distribution utility—for example, by enabling deferments of feeder capacity upgrades. For these cities, the decision to invest thus involves determining not only the magnitude of investment, but also of deciding where to place these technologies on the distribution grid.

The decision that cities face in these scenarios can be structured as a portfolio optimization problem: cities need to decide how much of each technology to invest in (MW of capacity) and where to install them (that is, which distribution feeder). By modeling the decision that cities face as an optimization problem, we formulate a generalizable portfolio optimization tool. The utility of developing such a tool for city planners is motivated by the reality that many cities lack the budget to pay for extensive consultations or to carry out a modeling approach when deciding how to achieve their emissions reduction goal. To demonstrate the utility of our tool, we operationalize it in the context of Columbus, Ohio, a city whose government has already set a goal to reduce yearly emissions by 40% by 2030, relative to a 2005 baseline. In developing strategies to achieve this goal, the city of Columbus is currently seeking to understand different possible approaches to reducing emissions associated with electricity usage.

B. Relevant Literature

To understand how our work will fit in and contribute to the existing literature, we review papers in the area of portfolio and emissions reduction optimization. This broad category of literature includes papers that consider the cost,

reliability, and environmental impact of hybrid power generation systems, capacity expansion plans, rooftop PV placement, and demand response [3]. In our review of this literature, we focus on which resources are considered, the methods and model formulations used, the broader technical or socio-economic impacts that are incorporated, representations of uncertainty, and the treatment of spatial differences.

1) Contexts and Resources Considered: Cai et al. [4] formulate a model for community-scale renewable energy planning, and motivate their work as serving communities that are moving off grid or expanding access to energy. As a result, the resources considered are energy generating resources, and Cai et al. formulate a generic model that can take the fixed and variable costs of user-specific generating technologies as inputs to their objective function, with constraints for electricity demand, energy-generating resource availability, and capacity limits [4]. Notably, although Cai et al. [4] consider a range of renewable resources in their model, they do not place a constraint on emissions or explicitly incorporate emissions into the model formulation. Moura et al. [5] also focus on renewable resources, but add demand-side management (DSM) and demand response (DR) as potential investments and look at a larger, national scale. While their model minimizes cost, it concurrently minimizes the peak load, and places constraints on generating capacity and yearly growth in capacity [5]. Li et al. [6] differ from other work in that they place a constraint on emissions, and also consider non-renewable resources as candidates for capacity expansion. Hashim et al. [7] present a similar motivation to our work, but in a different context. They aim to reduce the emissions from an existing fleet of generators, which leads them to consider increasing power plant efficiency, fuel switching, and deploying renewable energy as candidate resources [7].

We also looked specifically at work that optimized PV sizing and location based on feeder characteristics, looking to Kucuksari et al. [8], whose work optimizes the net profit of rooftop PV installations on a university campus while considering voltage characteristics of the distribution network. In this study, distribution grid impacts are considered after the optimization is performed by simulating the power flow once PV is installed to identify potential violations of feeder constraints, and then reallocating PV placement and sizing until the system adheres to distribution system specifications [8].

2) Methods and Model Formulations: The optimization methods selected by Cai et al. [4] were dictated partially by their treatment of uncertainty. Integer linear programming (ILP), selected because uncertainty in cost was represented through intervals, and chance-constrained programming (CCP), selected to support PDF representations of uncertainty, was integrated with mixed-integer linear

programming (MILP) [4]. Some work formulates multi-objective optimization problems - for instance, seeking to minimize system cost, peak load, and intermittence [5], while other papers focus on reducing system cost as the single objective, employing MILP [6]. Multiple objectives - for instance, minimizing cost and emissions - can also be represented in the same objective function, deploying a CO₂ emissions cost coefficient to tune tradeoffs between cost and emissions [7].

With respect to incorporating other socio-economic considerations, the model formulated by Cai et al. [4] presents a tradeoff between total system cost and energy security, offering the end-user a tunable parameter that will allow them to balance the two factors.

3) Representations of Uncertainty: Cai et al. [4] focus on the uncertainty inherent in energy management by expressing uncertainty both as probability density functions (PDFs) and as interval values. They note that uncertainty in generation profiles for wind and solar plants, for example, can be represented as PDFs, but factors such as capital investment costs or energy prices are better represented through confidence intervals [4]. Using PDFs to represent uncertainty in generation, particularly for intermittent resources, is seen across the literature [5] [6] [8]. The use of intervals is also present in multiple studies - Li et al. represent efficiencies and capacities of non-renewable resources using fuzzy lower and upper bounds [6]. In some cases, rather than building uncertainty into the model, uncertainty is explored through sensitivity analyses[6], [7].

4) Spatial Representation: The dynamic nature of community energy planning is recognized by Cai et al. [4], who note that social, economic, political, and resource considerations will vary over time and space in energy planning contexts. One approach to spatial considerations in energy system planning is presented by Kucuksari et al. [8], who use geographic information system (GIS) data to determine the suitability of buildings for solar PV, and use distribution feeder locations and characteristics to determine the impact of rooftop PV on feeder voltage profiles [8].

5) Takeaways and Opportunities: A survey of the academic literature on regional energy planning, capacity expansion, and emissions reduction is summarized in Table 1 of the Appendix. The surveyed literature provides guidance for representing uncertainty in optimization models, particularly through PDFs, and for incorporating planning periods and their constraints into models, both of which are directly applicable to this project. This survey also presents a number of opportunities and open questions that our work seeks to address. That is, the context that we are studying - minimizing the cost of a city's emissions reduction plan - enables us to consider a unique mix of resources that can be applied to an existing building stock. This particular combination of resources appears to be understudied in existing work. Furthermore, while many

studies are motivated by the need for emissions reductions, emissions are generally (with the exception of [6]) not placed as a constraint on the model, raising the question of what the optimal mix of emissions-reducing resources will be under emissions constraints. Finally, many optimization models do not consider distribution impacts of resource deployment, offering us an opportunity to build on existing work that considers rooftop PV and feeder limitations [8] alongside investments in energy efficiency.

C. Focus of this Study

As city governments are beginning to establish carbon reduction goals for the electricity usage of city operations, we create a portfolio optimization tool that cities with municipal utilities can use to determine the least-cost way to meet these goals. The tool incorporates the locational value of emission-reducing technologies to the distribution grid and considers reduction uncertainties due to variability in realized generation or demand savings. Although the tool is flexible to different locations and portfolio technologies, we operationalize it for the case of Columbus, Ohio and consider the following technology options: wind and solar transmission contracts, distributed solar generation, and energy efficient commercial and street lighting.

II. TECHNICAL DESCRIPTION

The optimization tool we develop determines the portfolio of technologies and locational placement that is expected to provide a city with a desired level of emissions reduction at least cost, whilst considering uncertainties. The results are unique to a city given their distinct feeder load profiles, generation and transmission contracts, solar insolation, customer makeup (residential, commercial, industrial, street light), and the load and hosting capacity headrooms on their feeders. We make the model more robust by considering uncertainties with the amount of emissions that can be reduced by each technology. For example, the amount of generation by solar photovoltaics has some uncertainty associated with it as a result of weather variability.

A. Model Formulation

1) *Model Objective:* In sum, the objective of our model is to minimize the lifetime costs (investment costs less operational savings) that a city government with a municipal utility incurs, when reducing emissions associated with the electricity used by city operations, in order to achieve a greenhouse gas reduction goal.

2) *Objective Function:* Given the model objective described above, we formulate a second order cone program (SOCP) with the following objective function, where i indexes the type of technology considered, j indexes the feeder, and h indexes the month-day-hour of a year:

$$\begin{aligned} \min_{X_{ij}} & C_i^T \cdot \sum_{j=1}^N X_{ij} && \text{(investment costs)} \\ & - \sum_{h=1}^N (y_h H_h^T \times \sum_{j=1}^N (\overline{S}_{hi} \times X_{ij})) && \text{(energy savings)} \\ & - (DMR \times CP) && \text{(demand savings)} \end{aligned} \quad (1)$$

$C_i \in R^{i \times 1}$ [\$/MW-year] is a vector of the annualized cost to purchase a certain savings capacity for each technology. These cost are all annualized over a twenty year lifetime (with replacement if necessary) at 9% interest and variable salvage value. The decision variable, $X_{ij} \in R^{i \times j}$ [MW], is the amount of savings capacity to install for a given type of technology, by feeder. $y_h \in R^{h \times 1}$ [\$/MWh] is a vector of the time-of-use energy prices from the grid, for each month-day-hour (h). $H_h \in R^{h \times 1}$ [hours/year] is a vector of scaling factors to account for simulations that use less than 8,760 hours in order to not under count energy charge savings. For example, if a simulation uses one representative week per month ($h = 2016$), the scale factor for Monday 3 p.m. 2019 would be four, since there are four Mondays in January 2019. If a simulation uses a complete 8,760 hours, then H_h is simply an 8760×1 vector of ones. $S_{hi} \in R^{h \times i}$ is the hourly (h) demand savings profile for each technology (i), normalized to the MW-savings capacity of the technology. In this case, energy efficiency is easily thought of as a demand savings technology, but we utilize the same concept for solar and wind as they too decrease the amount of demand that would need to be met by the grid. DMR [MW], a scalar value, is the demand peak reduction, which is determined by constraint (3) below. CP [\$/MW-year], also a scalar value, is the coincident peak charge over the course of a year across all feeders levied on the municipal utility by a transmission organization.

3) *Constraints:* The objective function is optimized subject to the following constraints:

$$\|\sum_{j=1}^{1/2} X_{ij}\|_2 \leq \frac{1}{\Phi^{-1}(\eta)} (\varepsilon_{ij,1}^T \cdot X_{ij,1} - G) \quad (\text{SOC constraint for achieving a given greenhouse gas reduction target}) \quad (2)$$

$$[\max(\sum_{h=1}^N D_{hj} - \max(\sum_{h=1}^N (D_{hj} - \overline{S}_{hi} \times X_{ij})))] \geq DMR \quad (\text{determination of peak demand reduction variable, DMR}) \quad (3)$$

$$D_{hj} - S_{hi} \times X_{ij} \geq 0 \quad (\text{no back-feeding at feeders}) \quad (4)$$

$$X_{ij} \geq 0 \quad (\text{positive values}) \quad (5)$$

$$X_{ij} \leq L_{ij} \quad (\text{resource limits}) \quad (6)$$

$E_{h,i} \in R^{h \times i}$ [lbs CO₂-equivalent] is the hourly grid emission profile $\varepsilon_{ij,1} = E_{h,i}^T \cdot S_{hi} \in R^{i \times 1}$ [lbs CO₂-equivalent] is the expected emission reduction per technology (i) replicated

by the number of feeders (j). $\sum \in R^{ij \times ij}$ is the covariance matrix for emissions reduction per technology (i) replicated by the number of feeders (j). G [lbs CO₂-equivalent] is the city's target emissions reduction goal. $D_{hj} \in R^{h \times j}$ [MW] is a matrix containing the original ('gross') demand for every month-day-hour (h) at each feeder (j). $L_{ij} \in R^{i \times j}$ [MW] is a matrix of the resource limit (feasible maximum MW) for each technology (i) at each feeder (j).

B. Alternative Deterministic Approach

In our results, we compare our preferred model formulation (above) with a deterministic version which does not consider uncertainty in the technology savings profiles S_{hi} . To amend our formulation above, we replace \bar{S}_{hi} in the objective function with S_{hi} . Additionally, we replace the second order cone (SOC) constraint (2) with the following inequality constraint:

$$\epsilon_{ij,1}^T \cdot X_{ij,1} \geq G \\ \text{(constraint for achieving a given greenhouse gas reduction target) (7)}$$

C. Methods for Implementation in Columbus, Ohio.

We provide an example for operationalizing the model in the context of Columbus, Ohio. The current goal for the city of Columbus is to reduce electricity emissions from city-owned/operated buildings and street lights by at least 40 percent by the year 2030, relative to a 2005 baseline. Our model was not created to optimize investment decisions over time, that is, to consider what year is the best to install a technology. Instead, it was created to determine the optimal investment in a given year. Hence, for our case study, we assume a scenario that the city hopes to achieve a 5 percent reduction from the baseline in the first year as a milestone - this translates to a target of approximately 5 million lbs of CO₂ equivalent emissions (or more specifically $G=4,715,150$ lbs of CO₂-e/yr) [9]. When running the SOCP, we choose $\eta=0.95$, to meet this emissions reduction target with 95% reliability.

The general service territory covered by the city's municipal utility - the Columbus Division of Power - is provided in Figure 1. In our operationalization, we consider 5 technology options ($i=5$), 5 feeders ($j=5$), and a representative week of month-day-hours for each month in a year (i.e. 24 hours per day for 7 days per week for 1 representative week per month, for 12 months of a year, such that $h=2016$). Given a lack of publicly-available feeder data for the Columbus Division of Power, we make a rough assumption, dividing the utility service area into 5 'feeders' as shown in Figure 1. As we assume the city government can only claim emissions reductions from changes in the electricity usage of city-owned/operated buildings and street lights, when applicable, we consider the technology options

in the context of these feeders. We use publicly available information identifying city buildings [10] and schools [11], and obtain geographic coordinates using street addresses and Google Maps. These city buildings and schools are plotted in Figure 1. Additional model inputs, parameters, and data are described in further detail below.

1) General Model Inputs & Parameters: Non-technology specific data needed to operationalize the tool include demand data, grid emissions, and energy and demand charges.

a) Demand: Demand profiles were constructed from publicly available load data from the Electric Power Research Institute (EPRI) [12]. EPRI provides load data for commercial, residential and industrial sectors for weekends and weekdays during peak and off-peak seasons. Using information on the energy consumption by sector from the U.S. Energy Information Administration (EIA) for Columbus, Ohio, we determined the percentage of demand that corresponded to each sector. The energy consumption was 31.3% residential, 25.3% commercial and 43.3% industrial [13]. Using the load shapes and the determined weights, we constructed a normalized demand load profile that takes into account season and day of week. Due to the accessibility of EPRI, the construction of the demand profile data is easily reproducible for any city in the U.S. Using the total power sold by the Columbus Division of Power, 787,427 MWh and a load factor of 0.4, we calculated a peak power of 225 MW [14]. Then we distributed total power over the five feeders, and applied the normalized demand load profile to end up with the load curve at each feeder.

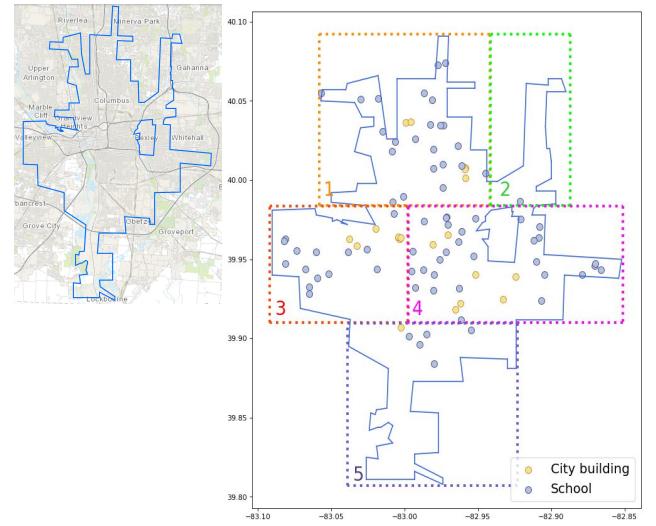


Figure 1. Left: Columbus Division of Power service area outlined in blue. Right: Allocation of city buildings and schools to 'feeders' ($j=5$).

b) Grid emissions: We have average grid emissions for each month of 2016, which was sourced from PJM, the regional transmission organization [15].

c) *Energy charge*: Columbus was unable to disclose their energy rates to us. In the interest of analyzing a time-of-use rate structure, we use one from Toledo Edison utility for their transmission customers. This data was sourced from Open EI Utility Rate Database, under the rate class unique id: 5ba02bee5457a3ee0dd845cc [16].

d) *Demand charge*: The demand charge is the annual one coincident peak with the transmission territory. For Columbus that is the AEP Ohio territory. Its charge was \$70,154/MW-year for 2018, which is the most recent year for which this information was available. [17].

2) *Portfolio technologies*: In our demonstration of the portfolio optimization tool for Columbus, Ohio, we consider 5 technologies: 1) utility-scale wind; 2) utility-scale solar; 3) distribution-level rooftop solar; 4) energy-efficient LED commercial lighting; and 5) energy-efficient LED streetlights.

a) *Utility-scale wind*: Utility-scale wind generation profiles are based on the National Renewable Energy Laboratory (NREL) Eastern Wind Integration Data Set [18], which provides power output values for simulated wind power plants. One year of power output data is extracted for 14 simulated power plants in Ohio at 3 combinations of hub heights and wind classes, giving 42 possible generation curves for wind power in Ohio. Those values are then normalized to the rated capacity of the individual power plants. The mean normalized power value is then determined to populate the technology savings profile S_{hi} , while the variance across datasets is used to populate Σ . Utility-scale wind costs are determined from Lawrence Berkeley Lab's Wind Technologies Market Report [19]. Historic power purchase agreement (PPA) prices are filtered for the Great Lakes region for most recent 3 years of data to approximate recent costs of wind PPAs in Ohio [20], and are combined with the average wind farm capacity factor for the year in which the plant is built, the rated capacity of the plant and an assumed lifetime of 20 years to obtain an estimate of PPA cost per MW installed.

b) *Utility-scale and rooftop solar*: Utility-scale and rooftop solar both use the same generation profiles based on the NREL National Solar Radiation Database's 1991-2010 Update [21]. Specifically, 2010 METSTAT-modeled global horizontal hourly class I data from the Columbus Port Columbus International Airport (Columbus Port Columbus Intl A) were used. This data provided hourly solar insolation data for an entire year. We then assumed a standard solar photovoltaic solar insolation to electric power conversion efficiency of 17%. These values are averaged and then normalized to the max solar insolation throughout the year, producing a power value for every one of the single month-day-hour combinations ($h=2016$). These values populate the technology savings profile S_{hi} , while the variance across these month-day-hours (e.g. there are four Thursdays at noon for the month of February) is

used to populate Σ . Utility-scale and rooftop solar costs are determined from Lazard's Levelized Cost of Electricity Analysis [22]. Solar resource limits were approximated using Google's 'solar calculator', Project Sunroof [23], for the city buildings and schools depicted in Figure 1 (right). In converting from square meters of available rooftop space for solar panels, as provided by [24], to an upper resource limit, we use the maximum month-day-hour insolation from the solar data (1.67×10^{-4} MW/m²).

c) *Commercial lighting*: Savings profiles from a switch to energy-efficient lighting are modelled using available load shapes from EPRI [12], for commercial, internal lighting. As shown in Figure 2, reductions in electricity use - which translate to avoided emissions - are largely realized when an energy-efficient technology is being used, and for commercial lighting, this is primarily during normal working/school hours. We approximate uncertainty in realizing these savings using a sample of real-world savings profiles for commercial and industrial lighting from [25]. To calculate the resource limit for commercial lighting upgrades, we first assume 1) that buildings currently use linear fluorescent lighting, specifically the Philips 32-watt T8 linear fluorescent bulb, and 2) that they will switch to LED tube lighting, specifically an 18-watt LED tube light from Green Creative, which has a comparable output of lumens [26]. Assuming 50 lumens per T8 bulb [27] and the National Optical Astronomy Observatory's (NOAO) recommendation [28] of 500 lumens per square meter for normal office and computer work, we calculate (and broadly assume) that there are 3.2 T8 bulbs/m². Given comparative light output of the replacement LED lighting [26], we further assume a 1:1 replacement of light bulbs. We calculate building area by scaling rooftop area by 1.2 times, and sum the areas for all city and school buildings within a feeder, to get a value for total building square area by feeder. We then use the number of bulbs to be replaced and the differences in the amount of power drawn by each bulb to calculate the resource limit for commercial lighting, i.e. the maximum number of MWs per feeder that can be saved from commercial lighting upgrades. Finally, we calculate annualized costs assuming installation and capital costs of \$30 per light [26].

d) *Street lighting*: The last technology option we include is energy-efficient street lighting, a technology that is currently being considered for implementation by the Columbus city government [29]. According to [30], Columbus, Ohio has approximately 1400 miles of lighted streets, which translates to approximately 53,000 street lights [31].

Savings profiles from a switch to energy-efficient street lights are modelled using available load shapes from EPRI [12], for commercial, external lighting. As shown in Figure 3, reductions in electricity use, and consequently avoided

emissions, are largely realized when street lights are on, which is typically overnight.

At present, the city of Columbus uses High Pressure Sodium (HPS) street lights [32]. A common HPS street light uses about 150 W [33]. The city government is currently discussing a switch to LED street lights, which use less electricity (and can be programmed, although we do not consider that functionality here); according to existing studies, a common LED street light uses approximately 105 W [34]. To calculate the resource limit (maximum MWs saved) for street light upgrades, we take into consideration the 45 W of savings per upgraded street light, per the above assumptions, and use a map of the population density across Columbus Ohio as a rough proxy to distribute the city's 53,000 HPS street lights across our assumed 5 feeders [35][36]. Considering a number of values from existing studies, we calculate annualized costs assuming installation and capital costs of \$250 per street light [37][38][39].

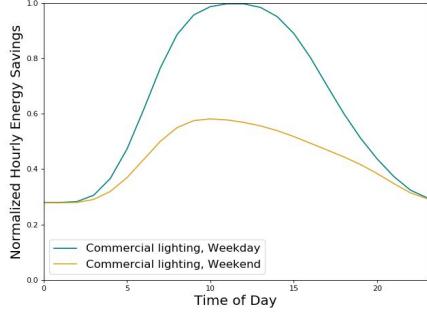


Figure 2. Savings profile for commercial lighting for an average weekday and average weekend during the peak season (normalized).

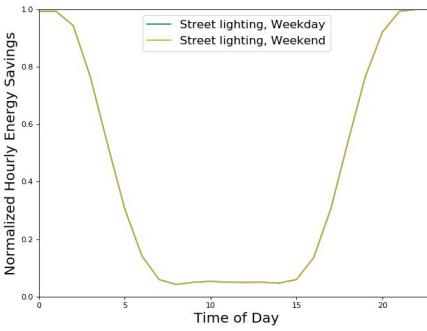


Figure 3. Savings profile for street lighting for an average weekday and average weekend during the peak season (normalized).

3) *Implementation:* We implement our model by coding it in Python 3 and solve it using cvxpy, with ECOS as the solver.

III. DISCUSSION

A. Results

We evaluated our model using two formulations: 1) with a second order cone constraint for achieving the greenhouse gas emissions reduction target, G , and 2) with a deterministic constraint for achieving that target. The results, which are displayed in Figure 5, show that the SOCP model specifies the installation of utility-scale solar, a small amount of commercial lighting, and an even smaller amount of utility-scale wind, whilst the model with deterministic constraints specifies the installation of utility-scale wind, utility-scale solar, and some commercial lighting. Table 2 provides the emissions reduction and annual cost associated with these two formulations.

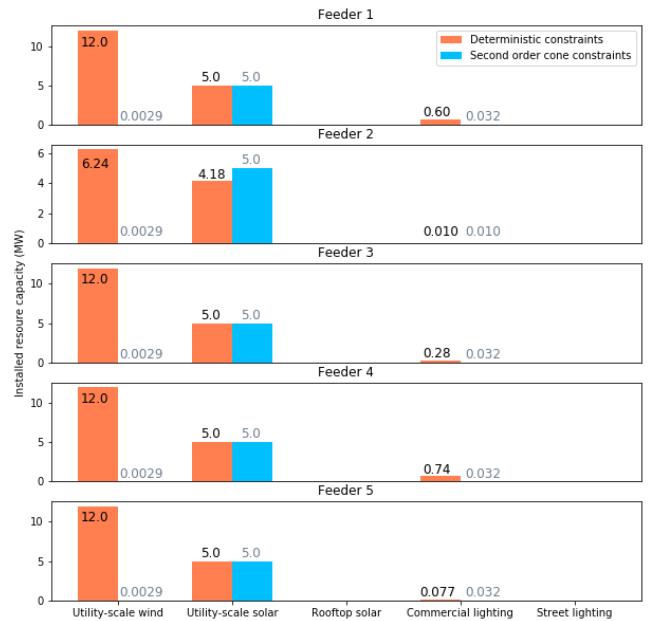


Figure 5. Contrasting results for the model output varied by use of deterministic or chance constraints for achieving the emissions reduction goal.

Unlike the SOCP, the formulation with the deterministic constraint simply selects the optimal resources based on the lowest cost of installation - which results in a prioritization of utility-scale wind, then utility-scale solar, and finally commercial lighting. With this formulation, it appears that a large reduction in emissions with a substantial amount of savings is achievable (however, the magnitude is in part because we do not include a constraint for the maximum amount of an initial investment and as such, the formulation provides the minimum-cost solution for the maximum emissions reduction and maximum savings that can be achieved, within specified resource limits). In contrast, the formulation with the SOC constraint selects technologies with lower variances, such as utility-scale solar and commercial lighting. That is, although utility-scale wind is in relative terms the cheapest technology in our portfolio

set, it is also the most variable. Thus, the SOCP model mitigates risk by taking into account the variance of the hourly demand savings profile for each technology and selecting a portfolio that will more reliably achieve our target emissions reduction. The comparison of these two formulations cautions against the naive implementation that uses a deterministic constraint - although the results of the ‘naive’ optimization apparently suggests that the target emissions reduction can be achieved, the real-world results of implementing such a portfolio could lead to substantially different results.

Model run	Annual emissions reduction (lb CO ₂ e)	Annual cost (\$/year)
Deterministic constraints	38,470,404	-2,558,467
Second order cone constraints	8,097,915	-483,987

Table 2. Optimization results for the two model runs. Negative values denote that savings/year are greater than the annualized costs.

B. Discussion of Results for the Preferred Formulation

Given the above, our preferred formulation is the one with the SOC constraint. As previously noted, under the formulation with the SOC constraint, the optimal solution selects utility-scale solar, a small amount of commercial lighting, and an even smaller amount of utility-scale wind. Heeding the caveat that our implementation is conducted with significant assumptions and less than perfect data, we nevertheless note that it is interesting that neither rooftop solar nor street light upgrades are selected as technologies to invest in. This appears to be a curious result, given the popularity of rooftop solar and street light upgrades in the news, particularly as programs that cities are interested in supporting and carrying out. One reason that our model may select utility-scale solar and not rooftop solar is because in our implementation, the initial cost of utility solar is lower and operational savings are the same. We have the same operational savings because we did not incorporate how rooftop solar can also reduce distribution losses and because we do not include feeder capacity deferral in our model. Furthermore, we assume in our model that utility solar is connected at the substation of the distribution utility, which reduces peak demand. However, if the utility solar was upstream of the substation, then there would be no peak demand savings and the rooftop solar would have higher operational savings, and would be more likely to be chosen.

Another technology where we see no capacity installation in our optimization model is street lighting. Given our formulation, this is largely due to the fact that we consider demand savings, and as shown in Figure 6, the savings realized by street lights are not coincident with the peak of our demand. Looking at Figure 6, we can see that commercial lighting and solar (utility and rooftop) are the most coincident with our peak power demand - these technologies thus have the highest combined potential for reducing emissions, avoiding time of use charges, and reducing peak demand charges.

Finally, there may be other reasons that rooftop solar investments and street light upgrades are popular programs that cities support and implement, and that our model does not capture. In particular, we focus on economic costs in our model. However, there may also be political and social pressures that inform the decision-making environment that city governments are in. For one, city planners may consider the visible, public promotion of renewable energy to be valuable, and rooftop solar panels and upgraded street lights are investments that have substantially higher visibility. Such visibility may also be desirable, or be perceived more positively for being additional, when a city is an active participant in a climate-forward agenda. Additionally, as infrastructure ages, such as High Pressure Sodium (HPS) street lights, and the cost of energy-efficient technologies drop, there may be a natural time for infrastructure replacements. Furthermore, we may also miss certain non-monetary costs associated, for example, with a municipal utility having to negotiate power purchase agreements for utility-scale solar or wind generation.

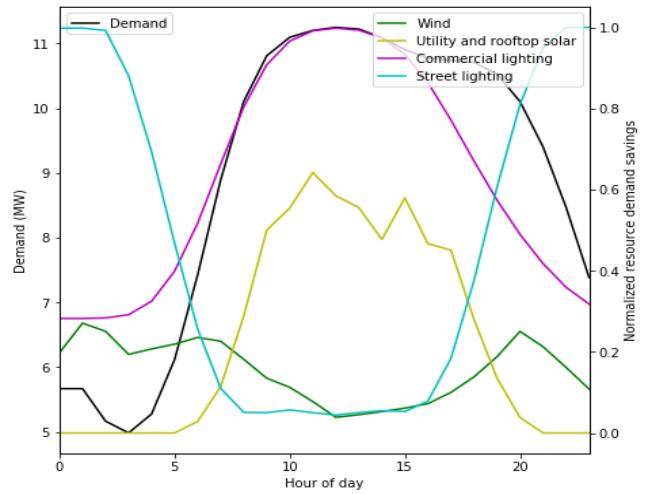


Figure 6. Load (left axis) and resource (right axis) profiles for a representative weekday in May.

B. Limitations and Opportunities

There are several limitations to our approach, which present opportunities for further work. A notable caveat is that our example implementation of the optimization tool for

Columbus, Ohio is done primarily for illustrative purposes - that is, to demonstrate how someone with data for a given context of interest would be able to use the model we have formulated. That is, the results that we find in our example should be taken with the consideration that real-world data for Columbus were not publicly available for our use.

Another limitation of our model is that we do not consider deferral of feeder capacity upgrades. Although we were initially interested in modelling this, we instead prioritized the formulation of a model using a second order cone constraint, as both were challenging features of our model. We look forward to implementing feeder deferral in the next iteration of this model as it will make the spatial component of this model more robust.

Furthermore, we hope to consider additional portfolio technologies that cities are likely to consider in their quest to achieve emissions reduction targets. One such technology we initially considered was energy-efficient HVAC upgrades. We did not ultimately implement HVAC efficiency measures in this model iteration due to the lack of access to robust building data. To increase HVAC efficiency there are many approaches one can take, including: implementing building management systems (BMS) and upgrading equipment, and within these technologies there is a wide range of possible implementations. The cost of implementation and energy savings on both of these approaches are highly subjective to the specific building application. The lack of scalability from specific buildings to all buildings is a common challenge in building analysis. The implementation of SOCP constraints in our model provides a unique opportunity to create a robust model to take into account the variance across buildings. Thus, the future work we anticipate also includes adding HVAC efficiency measures to the portfolio of options, when improved building data are available and we believe we have enough information to produce meaningful results.

IV. SUMMARY

In a time when decarbonization approaches are of paramount importance, cities are increasingly taking the lead and implementing ways to achieve lower greenhouse gas emissions. As substantial opportunities exist for reducing emissions from electricity-related activities, a relevant lever for city-level decision-makers, particularly those with a municipal distribution utility, is to reduce electricity-associated emissions. To support these cities, which are often budget-constrained, we develop a flexible optimization tool that determines an appropriate portfolio of electricity emission reduction technologies to implement at least-cost.

The tool considers utility-scale wind and solar, rooftop solar PV, and energy efficient commercial and street lighting as the suite of available technologies, and

incorporates energy and demand savings, as well as uncertainty in forecasted emission reductions. We demonstrate the utility of this tool in the context of Columbus Ohio, where we find the least-cost portfolio to achieve a 5% reduction of emissions from a 2005 baseline, in one year. Whilst future opportunities for improving the utility of this tool include obtaining real-world data relevant to the decision-making environment of city governments, we also note that city governments may be concerned about non-monetary costs that we do not consider in our model, such as political and social pressures.

REFERENCES

- [1] "Measuring Up 2015: How US Cities Are Accelerating Progress Toward National Climate Goals | Publications | WWF," *World Wildlife Fund*. [Online]. Available: <https://www.worldwildlife.org/publications/measuring-up-2015-how-us-cities-are-accelerating-progress-toward-national-climate-goals>. [Accessed: 07-May-2019]
- [2] "We Are Still In Statement on Bloomberg American Cities Climate Challenge | We Are Still In." [Online]. Available: <https://www.wearestillin.com/news/we-are-still-statement-bloomberg-american-cities-climate-challenge>. [Accessed: 08-May-2019]
- [3] R. Baños, F. Manzano-Agugliaro, F. G. Montoya, C. Gil, A. Alcayde, and J. Gómez, "Optimization methods applied to renewable and sustainable energy: A review," *Renewable and Sustainable Energy Reviews*, vol. 15, no. 4, pp. 1753–1766, 2011 [Online]. Available: <http://dx.doi.org/10.1016/j.rser.2010.12.008>
- [4] Y. P. Cai, G. H. Huang, Z. F. Yang, Q. G. Lin, and Q. Tan, "Community-scale renewable energy systems planning under uncertainty—An interval chance-constrained programming approach," *Renewable and Sustainable Energy Reviews*, vol. 13, no. 4, pp. 721–735, 2009 [Online]. Available: <http://dx.doi.org/10.1016/j.rser.2008.01.008>
- [5] P. S. Moura and A. T. de Almeida, "Multi-objective optimization of a mixed renewable system with demand-side management," *Renewable and Sustainable Energy Reviews*, vol. 14, no. 5, pp. 1461–1468, 2010 [Online]. Available: <http://dx.doi.org/10.1016/j.rser.2010.01.004>
- [6] Y. P. Li, G. H. Huang, and X. Chen, "Planning regional energy system in association with greenhouse gas mitigation under uncertainty," *Applied Energy*, vol. 88, no. 3, pp. 599–611, 2011 [Online]. Available: <http://dx.doi.org/10.1016/j.apenergy.2010.07.037>
- [7] H. Hashim, P. Douglas, A. Elkamel, and E. Croiset, "Optimization Model for Energy Planning with CO₂Emission Considerations," *Industrial & Engineering Chemistry Research*, vol. 44, no. 4, pp. 879–890, 2005 [Online]. Available: <http://dx.doi.org/10.1021/ie049766o>
- [8] S. Kucuksari *et al.*, "An Integrated GIS, optimization and simulation framework for optimal PV size and location in campus area environments," *Applied Energy*, vol. 113, pp. 1601–1613, 2014 [Online]. Available: <http://dx.doi.org/10.1016/j.apenergy.2013.09.002>
- [9] "[No title]." [Online]. Available: https://www.columbus.gov/uploadedFiles/Columbus/Programs/Get_Green/Survey/The%20Columbus%20Green%20Community%20Plan%20FINAL.pdf. [Accessed: 10-May-2019]
- [10] "Government Facilities." [Online]. Available: http://opendata.columbus.gov/datasets/6f59a6f71ebe4c9c819941b0ee0fe9c9_2. [Accessed: 10-May-2019]
- [11] "School Directory." [Online]. Available: <http://www.cesoh.us/site/default.aspx?DomainID=173>. [Accessed: 10-May-2019]
- [12] "End Use Load Shapes." [Online]. Available: <http://loadshape.epri.com/enduse>. [Accessed: 10-May-2019]
- [13] "Ohio - State Energy Profile Overview - U.S. Energy Information Administration (EIA)." [Online]. Available: <https://www.eia.gov/state/?sid=OH#tabs-2>. [Accessed: 10-May-2019]
- [14] "Annual Electric Power Industry Report, Form EIA-861 detailed data files." [Online]. Available: <https://www.eia.gov/electricity/data/eia861/>. [Accessed: 10-May-2019]
- [15] "[No title]." [Online]. Available: <https://www.pjm.com/~media/library/reports-notices/special-reports/20170317-2016-emissions-report.ashx>. [Accessed: 10-May-2019]
- [16] "U.S. Utility Rate Database." [Online]. Available: https://openei.org/apps/USURDB/rate/view/5ba02bee5457a3ee0dd845cc#1_Basic_Information. [Accessed: 10-May-2019]
- [17] "Transmission Cost Management | Commercial | AEP Energy," *AEP Energy*, 08-Mar-2018. [Online]. Available: <https://www.aepenergy.com/2018/03/08/february-2018-edition/>. [Accessed: 10-May-2019]
- [18] "Eastern Wind Data Set | Grid Modernization | NREL." [Online]. Available: <https://www.nrel.gov/grid/eastern-wind-data.html>. [Accessed: 10-May-2019]
- [19] R. Wiser and M. Bolinger, "Wind Technologies Market Report," Lawrence Berkeley Lab, 2017 [Online]. Available: https://www.energy.gov/sites/prod/files/2018/08/f54/2017_wind_technologies_market_report_8.15.18.v2.pdf. [Accessed: 10-May-2019]
- [20] "Wind Power Purchase Agreement (PPA) Prices | Electricity Markets and Policy Group." [Online]. Available: <https://emp.lbl.gov/wind-power-purchase-agreement-ppa-prices>. [Accessed: 10-May-2019]
- [21] "National Solar Radiation Data Base: 1991- 2010 Update." [Online]. Available: https://rredc.nrel.gov/solar/old_data/nsrdb/1991-2010/. [Accessed: 10-May-2019]
- [22] "Lazard's Levelized Cost of Energy—Version 11.0." [Online]. Available: <https://www.lazard.com/media/450337/lazard-levelized-cost-of-energy-version-11.0.pdf>. [Accessed: 10-May-2019]
- [23] "Project Sunroof." [Online]. Available: <https://www.google.com/get/sunroof>. [Accessed: 10-May-2019]
- [24] "Project Sunroof." [Online]. Available: <https://www.google.com/get/sunroof>. [Accessed: 10-May-2019]
- [25] "[No title]." [Online]. Available: <https://e2e.haas.berkeley.edu/pdf/workingpapers/WP023.pdf>. [Accessed: 10-May-2019]
- [26] "[No title]." [Online]. Available: <https://digitalcommons.wpi.edu/cgi/viewcontent.cgi?article=2678&context=mqp-all>. [Accessed: 10-May-2019]
- [27] "How efficacious are T8 fluorescent lamps? | T8 Fluorescent Lamps | Lighting Answers | NLPIP." [Online]. Available: <https://www.lrc.rpi.edu/programs/NLPIP/lightingAnswers/t8/04-t8-efficacy.asp>. [Accessed: 10-May-2019]
- [28] "[No title]." [Online]. Available: https://www.noao.edu/education/QLTkit/ACTIVITY_Documents/Safety/LightLevels_outdoor+indoor.pdf. [Accessed: 10-May-2019]
- [29] "[No title]." [Online]. Available: <https://www.bizjournals.com/columbus/news/2018/12/03/columbus-plans-smart-streetlights-for-linden.html>. [Accessed: 10-May-2019]
- [30] "Columbus Street Lighting Upgrade Has Opportunities and Potential Drawbacks," *Sierra Club*, 12-Dec-2016. [Online]. Available: <https://www.sierraclub.org/ohio/central-ohio/columbus-street-lighting-upgrade-has-opportunities-and-potential-drawbacks>. [Accessed: 10-May-2019]
- [31] "[No title]." [Online]. Available: <http://www.eei.org/about/affiliates/nationalkeyaccounts/Documents/Smart%20Communities%20In%20Focus/Columbus,%20OH.pdf>. [Accessed: 10-May-2019]
- [32] "[No title]." [Online]. Available: <https://www.bizjournals.com/columbus/news/2018/12/03/columbus-plans-smart-streetlights-for-linden.html>. [Accessed: 10-May-2019]
- [33] "LED Street Lights Life-cycle Cost Analysis." [Online]. Available: <http://www.u-tron.com/en/4-lucas-ii-84-led-life-cycle-cost-analysis.html>. [Accessed: 10-May-2019]
- [34] "LED Street Lights Life-cycle Cost Analysis." [Online]. Available: <http://www.u-tron.com/en/4-lucas-ii-84-led-life-cycle-cost-analysis.html>. [Accessed: 10-May-2019]
- [35] Jenna, "Where Are They Going? Population Growth in Franklin County, Ohio | Urban Decision Group." [Online]. Available: <http://urbandecisiongroup.com/where-are-they-going-population-grow-th-in-franklin-county-ohio/>. [Accessed: 10-May-2019]
- [36] "Website." [Online]. Available: https://www.researchgate.net/figure/Household-population-density-of-Columbus-Ohio-by-zip-code-area-By-author_fig6_267203122. [Accessed: 10-May-2019]
- [37] S. B. O'Brien, "Convert City Lights to LEDs? Not so Fast. - Business Journal Daily," *Business Journal Daily*, 11-Apr-2019. [Online]. Available: <https://businessjournaldaily.com/converting-leds-lighting-not-easy/>. [Accessed: 10-May-2019]
- [38] "[No title]." [Online]. Available: https://kb.osu.edu/bitstream/handle/1811/81628/ENRAEDE4567_Sm

- artStreetLightsOSU_au2017.pdf?sequence=1. [Accessed: 10-May-2019]
- [39] “[No title].” [Online]. Available: https://www.henniker.org/sites/default/files/fileattachments/General/page/6823/led_street_light_conversion_report_2018_final.pdf. [Accessed: 10-May-2019]

the location, size, and installation period			capacity		geographic factors; impact of PV on feeders is simulated
---	--	--	----------	--	--

ACKNOWLEDGEMENT

We thank Professor Scott Moura for helpful guidance at key junctures of this project, as well as the CE295 Spring 2019 Energy Systems and Control class at the University of California, Berkeley for their feedback and comments.

APPENDIX

Note from the authors: The final results presented here differ from those in the class presentation, given final updates to the data sources used. Additionally, we caught an error in the way we normalized across energy efficiency measures and renewable energy technologies.

Table 1. Summary of literature review

	Objective	Programming approach	Constraints	Uncertainty	Spatial factors
[4]	Minimize cost and maximize energy security in capacity expansion planning	ILP, CCP, MILP	Electricity demand, generation resource availability, capacity limits	Generation profiles (PDF) Costs (intervals)	N/A
[5]	Minimize cost, peak load,& intermittence of national electricity system	Multi-objective modeling	Generating capacity, yearly growth in capacity	Generation profiles (PDF)	N/A
[6]	Minimize system cost of electricity-generation system	MILP	Resource availability, peak demand, capacity growth, GHG emissions	Energy demand (PDF) Generator efficiency and capacity (fuzzy intervals)	N/A
[7]	Minimize cost and emissions of a power generation fleet	MILP	Capacity, capacity factor	Cost (sensitivity analysis)	N/A
[8]	Maximize profit of rooftop PV by optimizing	MILP	PV generation in each period, building	PV power production (PDF)	Suitability of rooftops is determined based on

Additional Data Notes:

[EPRI Data] For the demand profile data and internal commercial lighting data pulled from EPRI’s Load Shape Library: End Use Load Shapes, the region chosen is SERC/STV, given the absence of an option for the appropriate FERC region covering Ohio. The external commercial lighting data from EPRI is only available at the level of “All Regions.” For energy-efficient lighting savings profiles, we use the average weekday and average weekend off-peak season load shapes to represent the summer months, and the average weekday and average weekend peak season load shapes to represent the winter months.

Electric Vehicle Charging Stations Demand and Placement in New York City

Andualem Girma, Arthur Aumont, Tanwei Chen, Zachary Carroll

Department of Civil and Environmental Engineering, University of California, Berkeley

Abstract

Battery-electric vehicles offer a great opportunity to reduce carbon emissions from the transportation sector, but necessitate the installation of additional charging stations to satisfy their need. The number and location of these charging stations must be designed optimally in order to maintain high utilization rates and satisfy the needs of the customers, while accounting for the impact imposed on the local electrical distribution network. Furthermore, different standards for charging stations have been implemented all around the world which calls for a unified standard that account for charging speeds and the convenience of the user, while minimizing costs in grid upgrades and addition infrastructures. In this project, we predicted the future power demand and energy depleted of electric vehicles system under consideration using the Average, AutoRegressive with eXogenous inputs (ARX) and Neural Network models. And computed their accuracy and reliability for future use. We analyzed the effect of the charging time and frequency of the electric vehicles on the power demand and battery capacity requirement. The placement of electric vehicle charging stations to maximize utilization rate while keeping in mind the costs incurred by the electric vehicle owners and the city administration in charge of constructing the stations. We use the same model-based approach to identify a standard charging times best suited to meeting driver needs while avoiding grid overload. Our results provide insight on determining parking facilities suitable for implementing charging infrastructure as well as insight on charging demand induced from a hypothetical fleet of shared, autonomous, electric vehicles (SAEVs) in New York City. Our findings show a direct tradeoff between charger infrastructure costs and vehicle battery costs, as well as more complex interactions between the frequency of vehicle charging and the burden on the electrical grid.

Introduction

Motivation and Background

Electric vehicles (EVs) are one of the major technologies that dominated the modernization of the transportation sector. The increasing popularity of battery-electric vehicles offers an opportunity to reduce emissions produced by the transportation sector. It also ensures higher energy efficiency and reduced user cost, making it the focus of many researchers. One of the key issues in the utilization of electric vehicles is the availability and accessibility of charging stations. Thus, the optimization of electric vehicle charging stations is a very attractive subject. In addition to the importance of the problem, this topic serves as an intersection between infrastructure-scale energy systems and personal transportation. Thus, this topic allows each

group member to practice what has been taught in class while making use of skills cultivated in other classes and projects.

Charging is a much slower process than gas filling. Furthermore, the size of batteries in EVs have been reduced to avoid excessive weight and cost, limiting battery capacity. This can cause charging delay and discomfort for EV drivers. As a result, much work is needed in the design and implementation of charging stations.

The placement of these stations is critical to electric vehicles gaining wider acceptance among drivers. Despite the increasing range of newer battery-electric vehicles (BEVs), “range anxiety” is still a concern for potential buyers. Furthermore, large numbers of charging electric vehicles could risk overloading the local electrical grid. While this risk can be mitigated through assigning dedicated distribution lines or by upgrading local infrastructure, both of these solutions add to the cost of the charging station. Proper placement of electric vehicle chargers must account for both of these issues, while accommodating local and commuting users. Finally, competing standards for fast-chargers also create obstacles for would-be drivers. Faster chargers allow drivers to charge faster but impose a greater load on the electrical grid.

To be considered satisfactory, the design and installation of these charging stations must answer the following questions:

1. What is the number and location of charging stations needed to address the demand? This must also consider differences in battery capacity between EV models.
2. What are the effects of the charging stop times on the power demand and battery capacity of electric vehicles? Which limiting factors that will govern the system? A few models will be used to predict the power demand are and energy depleted so the investigation of the analysis of which model is the best one for a given purpose is also a very important aspect of this project.

This report provides a solution for the above questions and also gives an insight for the effects of vehicle battery capacity and rated power of chargers on the placement of the charging stations.

Literature Review

The topic of understanding EV charging demand and planning infrastructure has been actively researched over the past decade. Public fleets such as buses and taxis are likely to adopt vehicle electrification early.¹ For example, the city of Shenzhen completely electrified its public buses before the end of 2017 and 99 percent of taxis in the city are powered by batteries as of early 2019.² Several works model fleets of Shared Autonomous Electric Vehicles (SAEVs)^{3,4}, while others model traditional taxi fleets with either Battery-Electric Vehicles (BEVs)⁵ or Plug-in Hybrid Electric Vehicles (PHEVs)⁶. Our study assumes an SAEV fleet for modeling charging demand.

Two main approaches are typically used in modeling charging demand: computational geometry-based approach^{5,6,7} and origin-destination flow-based approach^{8,9}. With the computational geometry-based approach, charging demand can be estimated using real vehicle trajectory, simulated vehicle trajectory, or statistics from population density data and travel surveys. Origin-destination flow-based approaches consider EVs’ driving range constraints in transportation networks.

Integration of public charging infrastructure typically has two forms: gas-station-based⁶ and parking-facility-based⁴. Arguments can be made that gas-station-based infrastructure has

benefits in that it is a familiar concept, may help reduce “range anxiety”, and as EV integration increases, the decrease in Internal Combustion Engine vehicles (ICEV) at gas-stations can be offset by EV charging, thus keeping utilization rates of these public infrastructure resources at a reasonable level.⁶ The downside to gas-station-based charging infrastructure is that charging may take up to hours and customers can’t be expected to wait this long at a non-ideal location. Parking-facility-based public charging infrastructure has the benefit that charging can be done while a car is parked somewhere it may have been regardless, though parking fees may cost more than that of electricity used for charging.⁶

Focus of the study

In this project, we will use several models to determine the time and location of the charging demand. Using this output, we will optimize the placement of electric vehicle charging stations to increase their use, while maintaining realistic constraints. Finally, we will predict the impact of this network of charging station on the local electrical distribution grid and vehicle battery sizes.

Data Processing

In order to train and test our models we used an open source dataset on the famous yellow taxis of NYC, aggregating more than 15 million trips by 13,500 vehicles over a month. We were able to individually identify the behavior of each car by grouping the dataset by car .

The issue we ran into is that unlike our autonomous electric vehicle fleet, a traditional taxi does not work 24/7 and has to go home once in a while. Luckily, the dataset only compiles the trips where a fee was paid. Therefore, the trip to head home is not present to bias the data and we just see that the car stopped serving customers for 8 to 10 hours, sometimes more. The trips present in the dataset are therefore still representative of the behavior of taxi customers in NYC (places and times of departure and arrival). The problem is that we are looking at stop times and these halts in activity might disrupt our models.

In order to reconcile this we made an assumption moving forward that can only be applied to cities like NYC: a developed nightlife and the fact that a taxi driver can be found day and night shows two things. First, although drivers have to sleep, the taxi service overall is always delivered 24/7. Secondly, the taxi market knows its customers well and a decrease in service indicates a corresponding decrease in demand. Otherwise, an unmet demand would be supplied by enterprising taxi drivers. Thus, that these stops of 8 to 10 hours or more are justified by a fall in demand and if aggregated the number of taxis in service at any given time would correspond to the demand for the service.

After cleaning up the dataset of quirks and GPS errors we built 4 datasets based on this one depending on the duration of a vehicle stop. In order to study the impact on the electrical grid we need to know how often cars charge and that depends a lot on the cars used. We modelled with the following assumption: cars return to an optimal charging station (close to local demand) after dropping off each passenger. If the car reaches the station it would start charging immediately while waiting for an order. If it gets an order before that it would check if it had

enough battery to make the trip, accept the offer and change course without charging. Cars with smaller batteries would be cheaper for the operator to buy but expensive for the city; more charging stations would need to be built because the cars would have to charge more often.

Therefore, we decided to simulate different battery capacity by modelling the fleet behavior if it had to charge every stop longer than 10 minutes, 30 minutes, 1 hour and 3 hours. For all the stops that go beyond the threshold we would aggregate the total distance traveled before that to estimate how much charge a EV with a similar behavior would have lost and therefore needed to charge. Using the stop times and the aggregated distances we were able to move forward with our models. Figure 1 shows a heat map of a sample of taxi stops greater than 10 minutes.

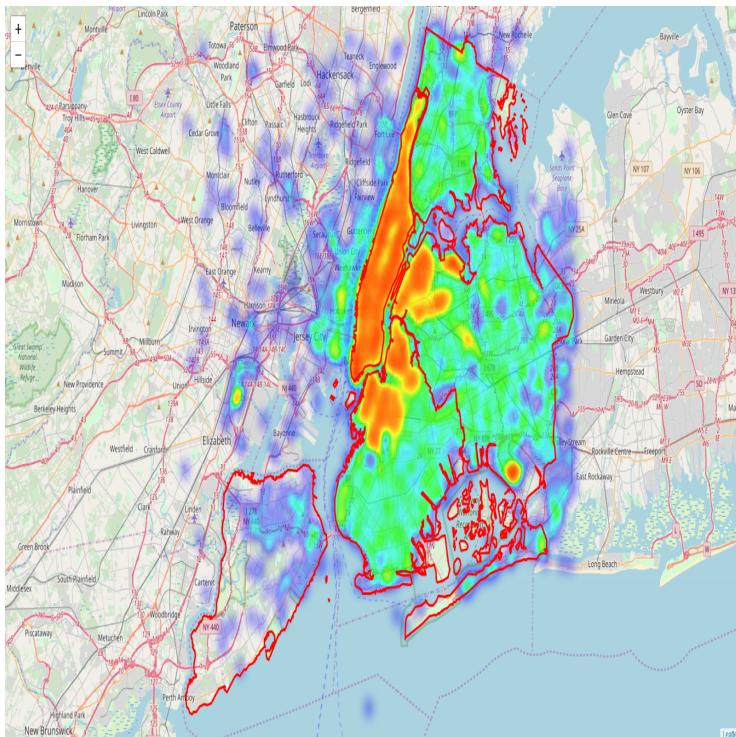


Figure 1. Sample data representing the location the cars stopped more than 10 minutes.

Data on parking facilities in NYC was obtained from NYC Open Data. Included in this dataset are all 1,912 registered parking facilities, their addresses, facility type, and number of spaces. Geocoding was used to find geographic coordinates of each facility and cleaning misidentified facilities from the set resulted in a total of 1,846 parking facilities throughout NYC.

Models

The Average Model

The average model was trained on the first three weeks of data, and tested on the fourth week. For each weekday and hour of day, the values for each week are averaged to produce the average model.

Mathematically, this can be accomplished through the following matrices:

$$\hat{P}_{avg} = \sum_{wd=1}^7 \sum_{h=1}^{24} \bar{P}_{wd,h} U_{wd,h}(k)$$

Matrix \bar{P} shows the average values corresponding to each combination of weekday and hour of day, and matrix U is a matrix of ones and zeros where ones correspond to the weekday and hour-of-day indices of interest.

In code, this was instead implemented by iterating through each weekday and hour index with nested for-loops. This produced an average model for each weekday.

The ARX Model

Similar to the other models, for this model we took the first three weeks as training data and the fourth week is used as the sample data for comparison purposes. First, the data is converted into a matrix which is easy to use.

Table 1. Data before converted in to matrix form.

car_ID	stop_lon	stop_lat	start_stop_d	start_stop_tod	end_stop_d	end_stop_tod	prev_trip_dist	prev_trip_t	stop_t	accumulated_dist	energy_depleted	power_demand	weeks	day_of_week	hour_of_day
1	-73.925	40.744	1	9342	1	10248	3.2	707	906.0	14.9	0.864	3.452200	1	1	3
1	-73.919	40.747	1	10755	1	11392	2.9	506	637.0	14.6	0.783	4.460100	1	1	3
1	-73.913	40.748	1	13159	1	14183	5.1	883	1024.0	19.0	1.377	4.864800	1	1	4
1	-73.738	40.731	1	17372	1	42791	16.5	1684	25419.0	37.5	4.455	0.631070	1	1	5
1	-73.862	40.768	1	43806	1	47408	9.6	1015	3602.0	30.6	2.592	2.594200	1	1	13
1	-73.962	40.720	1	62602	1	63304	37.0	1082	702.0	100.8	9.990	51.598000	1	1	18
1	-73.872	40.774	1	64265	1	66519	7.7	960	2254.0	71.5	2.079	3.327900	1	1	18
1	-73.972	40.677	1	67780	1	69900	10.8	1260	2120.0	74.6	2.916	4.963400	1	1	19

The matrix constructed for each week. For example the sum of energy depleted matrix produced can be seen in Table 2.

Table 2. Summed values of the Energy depleted data for every hour of the week days.

HOD	1	2	3	4	5	6	7
0	3157.2261	6753.5449	7669.3144	9488.5398	11006.3637	8732.5128	7836.9741
1	5830.8116	3904.0677	5542.6815	6674.1489	10989.6314	10441.5129	6249.2367
2	7477.7336	2704.9221	3531.5406	4370.6817	9725.2542	11253.4543	4682.7720
3	9902.2284	1786.8897	2281.2273	3201.0444	8561.1330	10914.7230	3693.2706
4	10370.9045	2880.5085	2808.2673	3265.4043	7173.9671	9406.3056	3907.7267
5	8171.2125	4046.5332	3592.5930	3558.0894	4463.8695	5188.4929	3715.0407
6	6939.7830	5578.6158	5125.6233	4987.6291	4552.0947	4404.0109	4950.6957
7	6624.6714	7108.9218	6956.7174	6617.0304	4763.8341	4390.6752	6131.0358
8	5886.1944	7636.4532	7793.9285	6606.4221	5814.4014	4782.9717	6672.9096
9	6010.6807	9622.1470	8633.4147	7972.1523	6206.4306	5379.7932	7110.7794
10	6093.2922	9552.6948	8527.5477	8024.5051	6865.8219	6152.0364	7220.4969
11	6623.5455	8900.4471	8036.1965	7397.1414	7343.1494	6737.9822	7398.8586
12	6662.2365	8603.0559	8050.2017	6828.5160	7254.8865	6868.1250	7019.0901
13	7185.5801	8752.2499	8093.1636	6925.4001	7527.4461	7441.7913	7778.2572
14	8419.6236	9684.4896	9087.4035	7865.1351	8405.8099	7783.8084	8233.6851
15	8597.5911	10095.1863	9141.8736	8746.8259	7944.5421	8616.9123	8839.0224
16	7691.8061	8698.5721	7851.6702	7616.4758	6405.6633	8040.1792	8558.3848
17	6350.8133	6777.0113	5917.0743	5854.2778	5655.2445	7115.8851	8137.1737
18	7188.2424	8543.3913	7264.9737	7221.1530	7380.9248	6915.7285	8558.8677
19	8221.6863	10544.2634	8921.4696	7602.6870	7197.1902	8290.2366	9514.3869
20	8265.6667	10101.9582	10392.4080	9767.3090	8814.3579	8002.7025	9632.2230
21	7650.9630	10556.6058	10797.6024	10161.5366	8118.8109	6704.3322	10147.7153
22	8193.7656	11084.1402	11282.5058	9461.9367	7902.2358	7875.9810	10118.8010
23	7412.3053	9510.6312	10421.3166	10240.7220	8256.8025	8277.1122	9800.4165

These values in the matrix, for the first two weeks, are used as “Phi” values to train the model. The ARX model is applied after sorting dates in to a proper ‘Phi’ matrix. This means our ‘phi’ matrix contains two columns of data which represents data from week one and data from week two of the same day, respectively.

$$Y = \begin{bmatrix} y(1) \\ y(2) \\ \vdots \\ y(k+1) \\ \vdots \\ y(K) \end{bmatrix} \quad \Phi^T = \begin{bmatrix} --\phi^T(0)-- \\ --\phi^T(1)-- \\ \vdots \\ --\phi^T(k)-- \\ \vdots \\ --\phi^T(K-1)-- \end{bmatrix}$$

In this case, the Y Matrix is the data obtained from the third week of the same day.

We know that:

$$Y = \Phi^* \theta$$

The **Theta** values are calculated using the formula:

After solving for **Theta** and using it to produce graph of the model, and further comparing it to the data from the fourth week we get the following graph. This model produced an output which is very close the test data with small Mean absolute error which is going to be discussed in the following sections.

Neural Network Model

The simplified neural network model consisted of a single neuron, which uses a weighted function of the previous three hourly values to predict the value for the upcoming hour. The model is described through the following equations:

$$y = f(\sum_i w_i x_i) = f(w^T x)$$

$$\frac{\partial J}{\partial \delta} = \delta, \quad \frac{\partial \delta}{\partial f} = -1, \quad \frac{\partial f}{\partial z} = f'(z) = 1 - \tanh^2(z), \quad \frac{\partial z}{\partial w} = x \in \mathbb{R}^3$$

$$w^{k+1} = w^k - \gamma \sum_{i=1}^m \delta^{(i)} \cdot (-1) \cdot f'(z^{(i)}) \cdot x^{(i)}$$

The model is then trained on three weeks of training data, where for each hour, the model uses gradient descent to find the optimal values of w . To improve applicability, the model is designed to be weekday-agnostic; the same values of w are used for each day of the week. For the sum of energy depleted at 10-minute minimum stop time, these weights were as follows: $w_1 = 2.2359$, $w_2 = -0.8024$, $w_3 = -0.0930$.

Locating Charging Infrastructure

For developing a method on locating charging infrastructure, the taxi dataset was filtered to include only idling instances between 1 and 3 hours. The day with the highest number of these instances was considered for analysis. An algorithm was developed to loop through each idling instance and determine the nearest parking facility. The logic behind this is that a ranking of parking facilities can be determined which may indicate facilities that are the best candidates for implementing charging infrastructure. Table 3 shows a sample of facilities and the number of parking instances nearest a given parking facility. A total of 22,520 idling instances between 1 and 3 hours occurred on the day analyzed.

Table 3. Sample of parking facilities in order of most nearby idling instances between 1-3 hours

	Lat	Long	Spaces	Instances	Type	Location
257	40.6413	-73.7781	225	1694.0	Garage - Parking Lot Combo	PARKING COMPANY OF AMERICA AIRPORTS, LLC 130 S...
88	40.7671	-73.8671	259	367.0	Garage - Parking Lot Combo	FIELD LAGUARDIA ASSOCIATES INC DITMARS BOULEVA...
1164	40.7708	-73.8705	139	350.0	Parking Lot	AIR PARK LGA INC DITMARS BOULEVARD 11369
1438	40.7769	-73.874	323	269.0	Parking Lot	PARKING COMPANY OF AMERICA AIRPORTS, LLC GRAND...
242	40.7689	-73.8676	410	190.0	Parking Garage	LAGUARDIA MARRIOTT HOTEL CORP DITMARS BOULEVAR...
828	40.7574	-73.9754	116	180.0	Parking Garage	SWEETS PARKING I, INC. MADISON AVENUE 10022
1477	40.7369	-74.1694	200	176.0	Parking Lot	WELCOME PARKING LIMITED LIABILITY COMPANY 45 A...
812	40.7736	-73.9134	40	137.0	Parking Garage	SYLVAN ELM GARAGE, LLC 59 AVENUE 11373
1359	40.7221	-73.9871	182	111.0	Parking Lot	EDISON NY PARKING LLC LUDLOW STREET 10002
888	40.7133	-73.9584	6	111.0	Parking Lot	INGENITO, FRANK ROEBLING STREET 11211
1716	40.7536	-73.9781	145	105.0	Parking Garage	CENTRAL PARKING SYSTEM OF NEW YORK, INC MADISO...
35	40.7487	-73.9917	210	102.0	Parking Garage	CENTRAL PARKING SYSTEM OF NEW YORK INC 7 AVENU...
1789	40.7518	-73.9755	11	95.0	Parking Lot	MARAND REALTY COMPANY LLC EAST BURNSIDE AVENUE...
1107	40.7613	-73.9255	175	92.0	Parking Lot	T R PARKING STATION INC 31 STREET 11106
640	40.7281	-73.9805	90	92.0	Parking Garage	CITY PARKING LLC EAST 11 STREET 10009
1299	40.7591	-73.9123	50	91.0	Parking Lot	FOIRE, GEORGE M 31 AVENUE 11103
643	40.7604	-73.9767	90	91.0	Parking Garage	MODERN PARKING LLC 5 AVENUE 10019
423	40.7853	-73.9531	74	91.0	Parking Garage	MAJESTIC CAR PARK, LLC PARK AVENUE 10128
162	40.7771	-73.9769	176	91.0	Parking Garage	15 WEST 72ND ST CORP WEST 72 STREET 10023
682	40.7571	-73.9718	200	85.0	Parking Garage	METROPOLITAN 51 PARKING, LLC LEXINGTON AVENUE ...

Discussion

Comparison of Models

All three models were trained and tested on the same data: energy depleted before each stop, for the case where vehicles begin charging at every stop 10 minutes or longer. The energy for each vehicle that stopped in the same hour was aggregated, producing hourly data for the total amount of energy that needed to be recharged during stops that hour. The mean absolute error (MAE) for each day of the week is shown in Table 4, along with plots from Sunday in Figure 2.

Table 4. MAE for each day of the week

Weekday	Sun	Mon	Tues	Wed	Thurs	Fri	Sat	Week MAE
Average Model	0.0303	0.0513	0.0635	0.0614	0.0809	0.0475	0.0710	0.0512
ARX Model	0.0421	0.0517	0.0315	0.0401	0.0804	0.0540	0.0811	0.0544
Neural Network	0.0612	0.0638	0.0776	0.0774	0.0734	0.0648	0.0739	0.0703

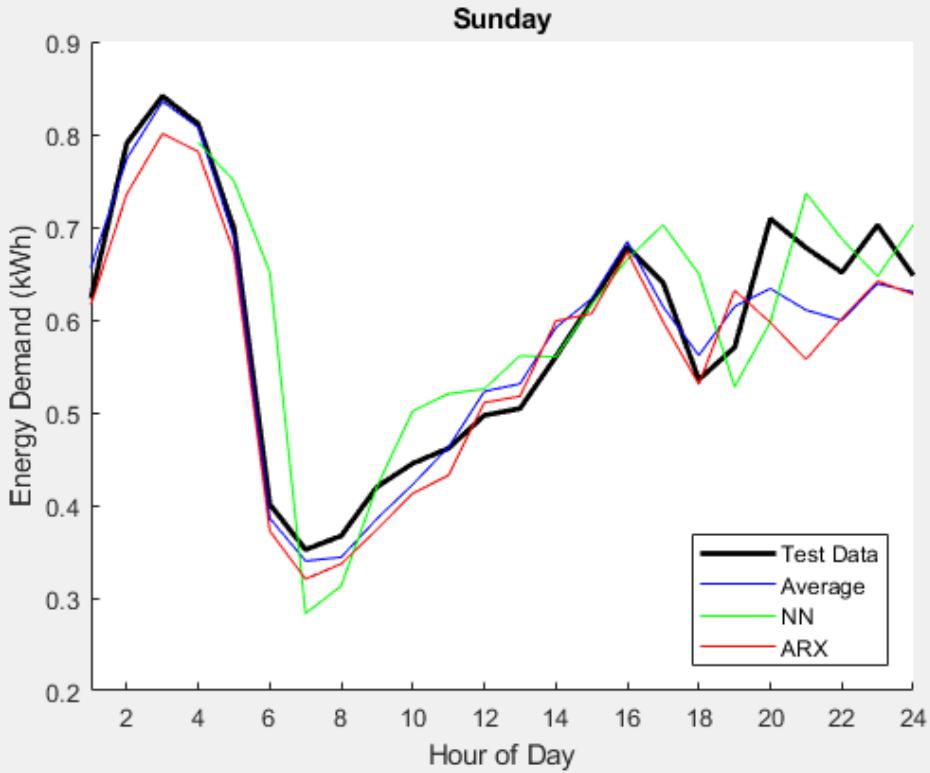


Figure 2. Plot of all model MAE for Sunday

Overall, the average model performed best by MAE, followed by the ARX model. The plot for Sunday serves as a good explanation for this phenomenon: the average and ARX models tend to follow the test data fairly closely, but the neural network tends to lag behind changes in the test data.

Given these trends, the average model is best suited for making long-term projections, such as the total energy demand for a day. It requires less training data, and the relative regularity and “cleanliness” of the data itself helps to increase its accuracy. The ARX model was generally better at following the contours of the model, making it more suited for situations where the magnitude of individual peaks may matter more. For example, the ARX model could

be employed for next-day forecasts of maximum charging energy demand. Both the ARX and neural network model could benefit significantly from access to additional training data; this is a good opportunity for future exploration.

Implications for fleet and grid

The length of charging stops has a significant impact on the vehicle composition of the taxi fleet. Vehicles are able to find more frequent opportunities to stop for 10 minutes or longer; by charging more often, they can complete their short trips between charges with smaller battery capacity. Conversely, stops of longer duration occur less frequently; vehicles which can only charge during longer stops require larger batteries to complete the longer stretches between charging stops.

By sorting the energy consumed between individual charging stops for each dataset and plotting, the relationship between energy consumed and number of trips becomes visible; most trips consume less than a set amount of energy, with a few outliers requiring large amounts of energy to complete the trip. These results are plotted in Figure 3, with dashed lines showing battery capacities which would allow for the completion of the vast majority of individual trips in that dataset. These dashed lines were visually placed to capture the vast majority of trips without wasting battery capacity trying to capture all of the outliers. Almost all trips which occur between 10-minute charging stops can be completed with a 10 kWh battery capacity, trips between 30-minute charging stops can be completed with a 15 kWh battery capacity, trips between 1-hour charging stops can be completed with a 25 kWh battery pack, and trips between 3-hour charging stops can be completed with a 40 kWh battery pack. Increasing battery capacity beyond these values will help capture additional outliers, but will grant vastly diminishing returns; outlier trip lengths may be best captured by a few internal combustion engine vehicles.

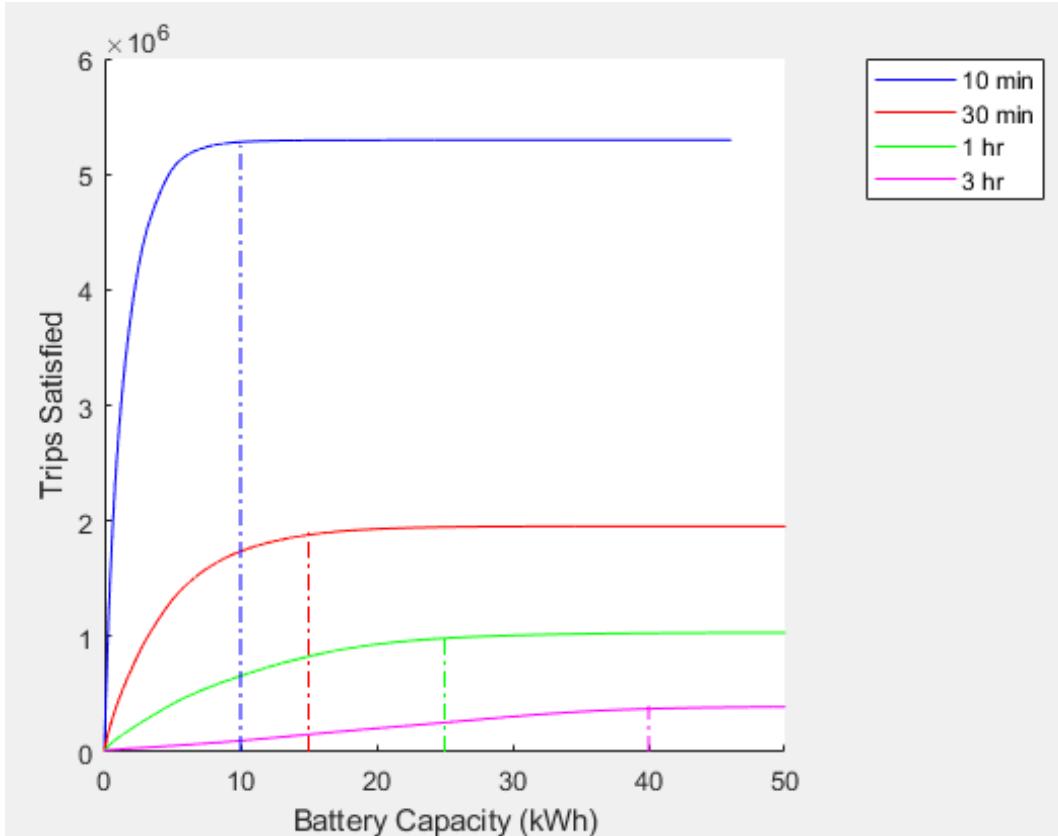


Figure 3. Battery capacity requirements

This also presents requirements for the chargers installed. Our problem setup assumed that the charging stops would be sufficient to keep battery SOC at near-unity. Given the 5-minute non-charging delay built into every charging stop and the amount of energy that must be rechargeable during the stop to allow for continuous driving until the next stop, we can calculate the maximum charger power needed to satisfy these conditions from dividing maximum energy demand by minimum charging time. This is summarized in Table 5.

Table 5. Charger power needed to satisfy assumptions

Minimum Stopping Time	10 minutes	30 minutes	1 hour	3 hours
Maximum Energy Demand	10 kWh	15 kWh	25 kWh	40 kWh
Minimum Charging Time	5 minutes	25 minutes	55 minutes	175 minutes
Maximum Charger Power	120 kW	36 kW	27.3 kW	13.7 kW

Furthermore, the total energy demands on the city's power grid can be visualized by aggregating the energy demanded for every stop in that hour. This results in a matrix of total energy consumption per hour, for each hour of day, weekday, and week. Tuesday of Week 4 has been plotted in Figure 4 as a demonstration.

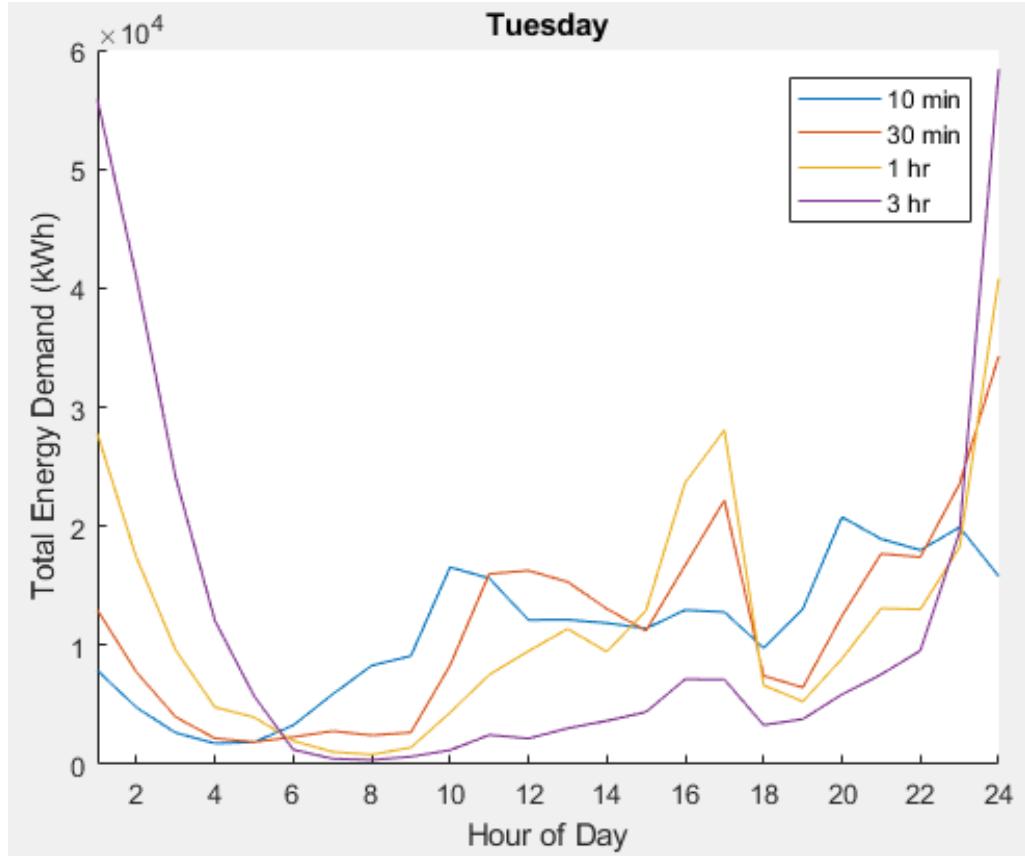


Figure 4.

The total energy demanded over the course of the day is consistent between each of the 4 datasets, because the total distance driven between stops is the same. There is a 5% max difference between the integrals of total energy demand over the whole day, likely due to small errors in aggregation. However, the energy demand shapes are different for each length of charging stop, and are dictated by the natural stopping times of taxis. Taxis naturally perform more 10-minute (or longer) stops during the day and early evening, thus they will demand more energy during those times by charging during their short stops. Conversely, taxis rarely stop for 3 hours (or longer) except overnight; this is illustrated by their energy demand, which would peak overnight if taxis were only permitted to charge while stopped for 3 hours or longer.

Finally, the total number of charging stops in each hour can be aggregated and plotted in Figure 5 to visualize the number of chargers that must be installed to meet demand.

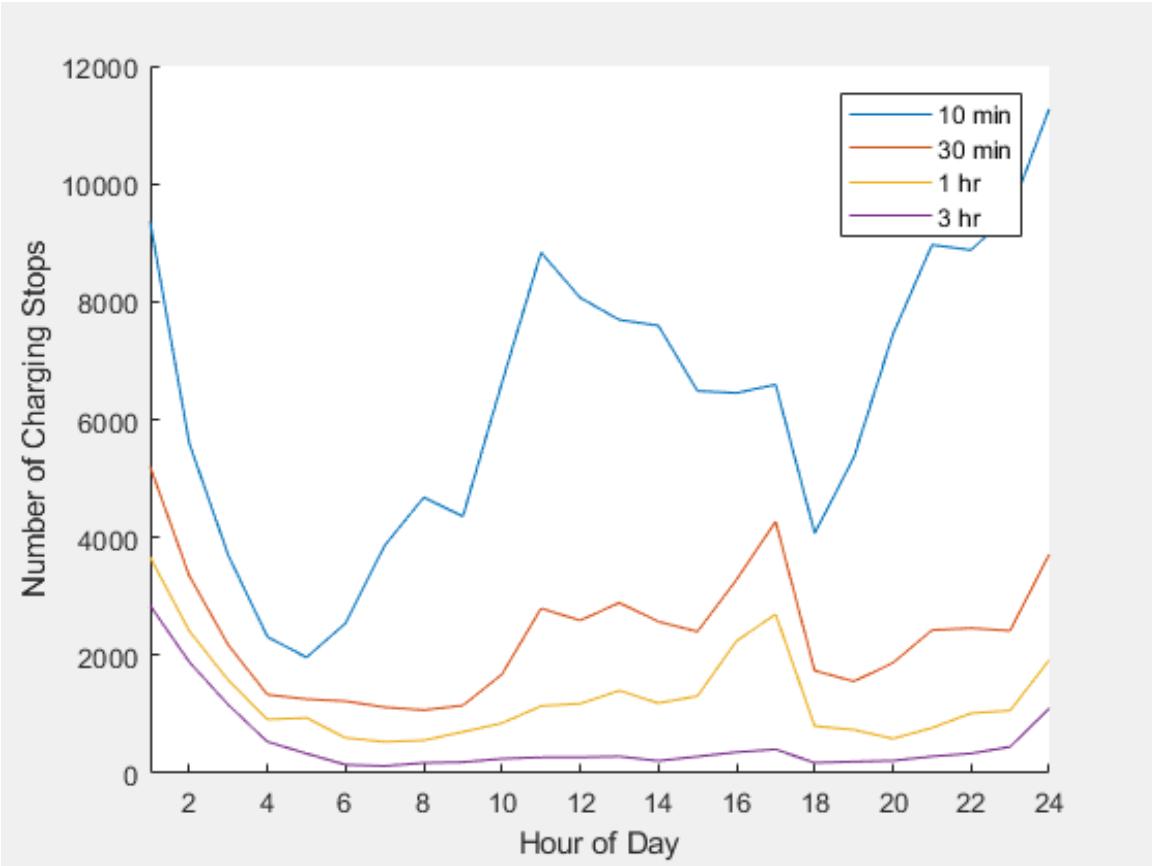


Figure 5.

The information in Figures 3, 4, and 5, and Table 5 allow us to generalize four options for the city's management of fleet charging, with trade-offs for each option. The first option is to install sufficient chargers to allow taxis to recharge every time they stop for 10 minutes or longer. Figure 5 and Table 5 show that this requires the city to install the highest quantity and fastest type of chargers to satisfy demand, which increases infrastructure costs for the city. However, this reduces the size of battery required for the fleet vehicles to complete their relatively short drives in between charging stops, which reduces costs for the fleet operator. Finally, Figure 4 shows that the energy demand shape caused by charging vehicles will be relatively flat, which may reduce line loads and ramping costs for the utility grid.

On the opposite end of the spectrum, the city can install a fewer number of slow chargers, such that fleet vehicles can only realistically charge when stopped for a period of 3 hours or longer. This combination of smaller quantity and lower charger power demand means that the city can spend the least on chargers. However, the more infrequent charging stops requires the fleet to invest in vehicles with larger batteries to endure the longer trips between charging stops, increasing vehicle costs for the fleet operator. Stops of 3 hours or longer primarily occur overnight; charging only during these stops would create the most extreme peak seen in Figure 4. However, the overnight timing of this peak may be advantageous for grid ramping costs if it does not coincide with daytime peaks caused by other users of electricity. The other options (enough chargers to charge when stopped for 30 minutes or 1 hour) result in costs between these extremes.

With these options, the city has enough information to perform cost-benefit analyses between charger infrastructure costs, electrical grid ramping costs, and burden on the vehicle operator. Ultimately, this will allow the city to make a more informed decision in charger installation and therefore facilitate the use of battery-powered vehicle fleets in reducing greenhouse gas emissions.

Locating Charging Infrastructure

There are several issues with the methods used in ranking parking facilities for their potential to implement charging facilities and with the idea of parking-facility-based charging infrastructure in general. The issue of costliness of parking in facilities is not considered - NYC has the highest parking costs in the US, without considering the added cost of charging¹⁰. Ideally, a future fleet of EV taxis may have priority to park in facilities at low or no parking cost, but rather the main cost incurred would come from charging fees. Another issue with parking-facility-based charging is that the availability of spaces will be difficult to guarantee in private parking facilities. For this analysis, the issue of who is using these facilities and when is not considered.

As for issues in the facility ranking model, by only considering one day of data, a bias may occur if this day had any special event which placed parking instances in a certain area. This bias can be avoided by expanding the analysis over many more days. Another issue that may result from the methods used is that parking facilities that have many nearby taxi idling instances may be favored by the algorithm, and other facilities just slightly further away may be disregarded even though they can be just as suitable. To address this problem, the algorithm for ranking facilities must be improved, perhaps to increase the rank of all facilities within a specified distance of an idling instance.

The reason 1 to 3 hours of idling instances are considered is that this is assumed to be a typical amount of time that EV taxis may spend charging. This is only an assumption and the model may improve if other lengths of stops are considered. The algorithm developed for parking facility ranking is just a framework for developing methods for siting charging infrastructure. The idea is that parking facilities nearby popular idling locations may be suitable candidates for integrating charging infrastructure, and the goal is to gain insight on where infrastructure may best be located in facilities that are already built and designed for storing vehicles.

Future Work

Our analysis does not look into the environmental impacts of implementing charging infrastructure, nor do we quantify the costs associated. Further work on these issues will greatly benefit this analysis. Secondly, the charging infrastructure location modeling was not directly influenced by our charging demand modeling. Future works can be improved by integrating charging demand found in our models with a spatial aspect and parking facility capacities.

Furthermore, the accuracy of our ARX and neural network models was limited by the small amount of available training data. Processing more weeks of data would take an excessive amount of computing time; future works could instead process a representative or random sample of the cars to include more weeks of data without lengthening the processing time.

Finally, future projects could model the number of simultaneous stops to calculate peak power demand.

Summary

The results of our project suggest that the increase in frequency and length of the charging stops greatly affects the power demand and battery capacity requirement of our vehicles. When the frequency of charging is increased the vehicles consume a lot less power leading to smaller battery sizes and when we have lesser frequency of charging the vehicles demand a lot higher power to compete a trip of approximately equal length. Thus, they require larger battery capacity and longer charging times. The trends captured while processing the data also suggest that the outlier areas are best captured by Vehicles using higher battery capacity or using internal combustion engine. This also alternatively can be solved by constructing more charging stations in those areas.

Furthermore, more frequent charging creates flatter energy demand curves, though their daytime placement may coincide with demand peaks caused by non-EV users. Very infrequent charging results in an extremely high overnight peak, which could be beneficial if offset from daytime peaks caused by non-EV users. As a result, the net cost or benefit for the electrical grid requires further analysis of electricity demand by non-EV users, or time-of-use electricity rates for NYC.

The inverse relationship between charger availability and required vehicle battery capacity represents a direct tradeoff in costs between city charger infrastructure and fleet vehicle ownership. Combined with the unclear net impact on the grid of flatter daytime energy demand vs. extreme overnight peak demand, the results of this report allow a city to begin considering trade-offs while suggesting avenues for further exploration. Ultimately, this will help cities like New York City optimize the use of electric vehicles in reducing emissions.

References

- (1) Krieger, Axel, Radtke, Philipp, and Wang, Larry. Recharging China's electric-vehicle aspirations: A perspective on revitalizing China's electric vehicle industry. *McKinsey & Company Report*. (2012).
- (2) Liao, Rita. First buses, now Shenzhen has turned its taxis electric in green push. <https://techcrunch.com/2019/01/04/shenzhen-electric-taxis-push/> (accessed May 10, 2019)
- (3) Chen, T. Donna, Kara M. Kockelman, and Josiah P. Hanna. "Operations of a shared, autonomous, electric vehicle fleet: Implications of vehicle & charging infrastructure decisions." *Transportation Research Part A: Policy and Practice* 94 (2016): 243-254.
- (4) Bauer, Gordon S., Jeffery B. Greenblatt, and Brian F. Gerke. "Cost, energy, and environmental impact of automated electric taxi fleets in Manhattan." *Environmental Science & Technology* 52.8 (2018): 4920-4928.
- (5) Cavadas, Joana, Gonçalo Homem de Almeida Correia, and Joao Gouveia. "A MIP model for locating slow-charging stations for electric vehicles in urban areas accounting for driver tours." *Transportation Research Part E: Logistics and Transportation Review* 75 (2015): 188-201.

- (6) Cai, Hua, et al. "Siting public electric vehicle charging stations in Beijing using big-data informed travel patterns of the taxi fleet." *Transportation Research Part D: Transport and Environment* 33 (2014): 39-46.
- (7) Mak, Ho-Yin, Ying Rong, and Zuo-Jun Max Shen. "Infrastructure planning for electric vehicles with battery swapping." *Management Science* 59.7 (2013): 1557-1575.
- (8) Shahraki, Narges, et al. "Optimal locations of electric public charging stations using real world vehicle travel patterns." *Transportation Research Part D: Transport and Environment* 41 (2015): 165-176.
- (9) Chung, Sung Hoon, and Changhyun Kwon. "Multi-period planning for electric car charging station locations: A case of Korean Expressways." *European Journal of Operational Research* 242.2 (2015): 677-687.
- (10) Nichols, Adam. See how much more you pay to park in NYC than other cities.
<https://patch.com/new-york/new-york-city/see-how-much-more-you-pay-park-nyc-other-cities>
(accessed May 10, 2019)

Optimal Pricing of an Electric Vehicle Charging Station Based on a Flexible Charging Schedule

Katie Lee, Arthur Tseng, Anqi Zhang, Simon Aldebert, Vincent Fighiera

Abstract

This paper considers a parking garage with shared charging points for electric vehicles (EVs), for which we consider the introduction of a dynamic charging tariff that aims to minimize operating costs and maximize revenue, while satisfying technical constraints. To minimize cost, EVs will be charged at times of lowest electricity rates. From this cost minimization problem, we then maximize facility operator's (FO) revenue by adjusting the profit margin on the customer-facing price. By using flexible charging at the cheapest hours, the FO is able to profit while still charging a reasonable price to EV owners. The ultimate goal of this study is to encourage investments in EV charging stations by quantifying their financial returns.

Index Terms

Electric Vehicles (EV), Optimization, Flexible Charging, Behavior Modeling, Sustainable Energy System

I. INTRODUCTION

A. Motivation & Background

With ever increasing renewable penetration on the grid, increasing electric vehicle (EV) adoption can result in drastically lower global carbon dioxide emissions, as people retire their internal combustion engine vehicles (ICE). However, there are a few factors that make people reluctant to buy an EV. One of the main reasons is a lack of charging infrastructure to keep up with the number of EVs on the road. A study by the US Department of Energy found that only 40% of US homes have garages capable of charger installation [1]. Essentially, this means that 60% of people are unlikely to purchase an EV, unless public infrastructure improves. Our project seeks to improve upon one potential solution to this: workplace EV charging.

Installing chargers at each individual home is expensive and exacerbates the duck curve because people charge in the evenings when they get home from work, as seen in Figure 1. Flexible workplace charging can ease the duck curve by charging in the day time when prices and demand are low. For the purposes of this paper, "flexible charging" refers to a situation when EV owners plug in to a charger and do not demand power instantaneously, but instead are flexible with the timing of the charge delivered to their EV, based off lowest cost electricity time periods. The amount of range added will be the same, whether 'flexible' or not.

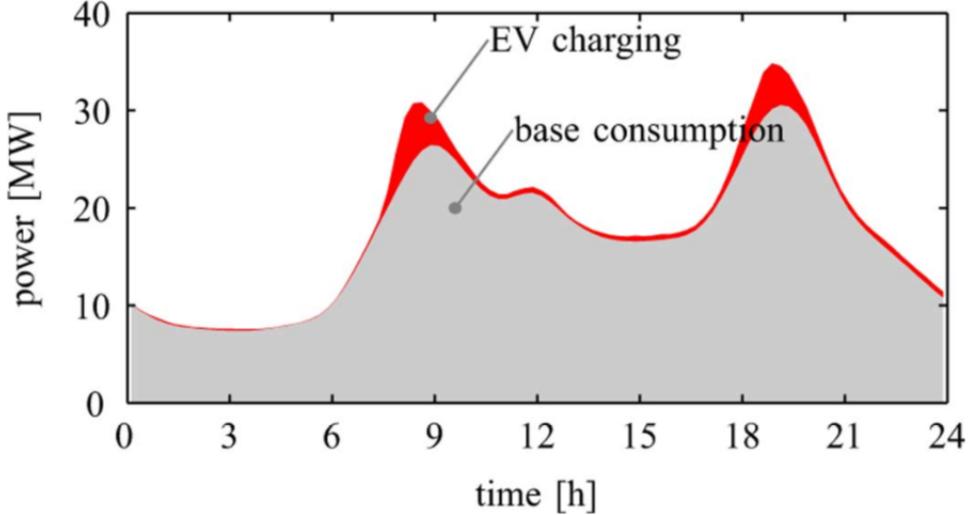


Figure 1. EV charging occurring simultaneously with peak demand [2]

At the moment, however, there is little incentive for a facility operator (FO) to install chargers in their parking area, due to zero profit potential. Thus, we propose a model to promote sustainable EV charging systems, incentivizing FOs to install chargers with profits guaranteed.

Our model will allow: (1) FOs to reduce their marginal cost by charging at the cheapest electricity hours, and then calculate an ideal margin on top of the base price to keep as profits. (2) EV owners to maintain low charging prices. (3) The grid to be more balanced by using the model to ease the duck curve and charge at low net demand periods. The constraints of our model include the maximum power capacity of the facility and the State of Charge (SOC) target of each user. Inputs to the optimization include Time of Use (TOU) electricity prices, maximum charging capacities of the parking lot, number of individual chargers, arrival and departure distributions of EVs, and the binary stochastic choice of the user to accept the price given or leave.

There are plenty of benefits this model can bring to society. First, it will solve the cost minimizing optimization problem based off electricity prices throughout the day. Then, based off the new control schedule, we adjust the FOs margin to maximize their profit while still keeping low enough prices for EV owners to accept the charge. Lastly, we investigated the effect of minimizing carbon dioxide emissions on our model, though our shift in results after adding this feature was not dramatic.

A major challenge to this energy system is that user choices are stochastic variables in the optimization model. This paper overcomes this by projecting people's choices using Monte Carlo simulation based on behavior modeling. We created logistic utility functions to model EV owner's choice to accept the price or decline. More elaboration can be found in the Technical Description part.

As for our group, we all have similar interests and skills that led us to collaborate on this project. Arthur had a previous project where he investigated the embedded carbon in transforming

Singapore into an EV only city. Katie is currently thinking about how a pilot optimal-charging project would work at UC Berkeley, and going to be studying electric mobility trends after graduation. Anqi finds it is interesting to develop infrastructure to support emerging mobility technology to achieve future sustainable communities. Simon is very interested in electric vehicles chargers and the corresponding challenge for the electricity grid. Vincent is passionate about human mobility in which EVs will surely play a key role in the upcoming years. The team is currently mentored by Bertrand Travacca and Sangjae Bae, both PhD students at UC Berkeley.

B. Literature Review

There is a great deal of literature on optimizing EV charging. We found papers on charging pricing strategies, behavioral modeling methodologies, boundary setting, and parameter references. The literature listed below is categorized by these topics, highlighting the aspects that have not been previously written about, that we will focus on in this paper.

1) Parameters: The first, by Alonso, et al. discusses the optimal demand schedule for residential parking in Spain to smooth their duck curve, which provides a baseline for valuable behavior modeling methodology, charging station choices, and boundary studies for us to refer [3]. Then, the paper discussed impact from on-site chargers to the grid, charging time, and GHG emission, can vary with different AC voltage and AC current. Valuable parameters of on-site voltage and current can be referred to from their studies. The way they set their boundaries for SOC, power flow, charging, and technical limits was helpful when defining our model.

2) Charging: Another, by Sundstrom and Binding, is focused on forecasting state of charge (SOC) depletion from a trip in order to optimize charging [2]. They do this by collecting time of departure, time of arrival, beginning SOC, ending SOC, and ending location. The charging optimization schedule is then based on providing enough energy for the predicted future trips. Our focus, on the other hand, will consider one specific parking structure, not optimizing based off of trip length but instead on the electric grid demand.

3) User Choices and Behavior: A paper by Samadi, et al. delves into user preferences and behavior based off electricity prices in a household. They use similar constraints, limited total electricity consumption to the system capacity of one electricity provider [4]. Each user is considered independently, and their consumption choices depend on price of electricity, time of day, and weather conditions, modeled using a quadratic utility function with linear decreasing marginal benefit. Samadi, attempts to maximize the sum of user welfare functions, subtracting the cost of the facility owner. For the utility functions they take into account power consumption as well as other factors that affect a user's decision such as time of day. We will do the something similar, for EV charging rather than household electricity consumption.

A paper by Ding et al [5] introduces two different types of customers which are the price-responsive commercial charging customers and contracted charging fleets to capture the characteristics of different consumers. They take into account individual PHEV users that have flexibility on charging time and who can make their charging decisions based on the charging price and their sensitivity to the price change. The paper by Alonso, et al. also mentioned about methodology of user behavior studying that could be used to update our behavioral model [3].

4) Pricing Strategies: A paper by Cao, et al [6] looked into the potential of EVs to shift loads from peak to valleys of the load curves. The paper minimized costs that EV users pay by looking at the times that the EVs would be parked, and the TOU prices during those hours. By controlling when the cars start and end charging (i.e. not start immediately when the car is connected to the charger), the paper was able to shift demands drastically to off-peak times. This “intelligent” way of charging is one way of meeting demand response requirements of a grid, as well as lowering EV ownership cost, and relieving congestion in the electric grid. The paper by Ding et al [5] introduces a stochastic resource-planning scheme for PHEV charging station to jointly optimize its energy supply side and demand side. In the proposed scheme, the supply-side decisions for forward and spot markets procurement and internal resources scheduling are co-optimized with demand-side decisions for charging service pricing and controllable load allocation. In this way, the optimal resource-planning decision portfolio for PHEV charging station can be achieved by considering all those variables simultaneously. Our model closely relates to their article and we drew inspiration from their case study to build our own cost-minimizing simulations.

5) Parking: A paper by Zeng, et al [7] focused on the issue of overstay in EV charging station. They introduced a new way of management by adding the concept of interchange. The goal is to minimize the infrastructure cost and operating cost of charging station by switching the power of a spot to another car when the battery of the first car parked is full. Similar to our approach, they model the customer behavior, using stochastic variables and applying a chance constrained optimization to deal with those stochastic variables. This new way of management reduces the total cost by almost 17%. In our project the approach of the customer behavior is different, the stochastic variables will be generated by a Monte Carlo simulation.

C. Focus of the study

This study focuses on developing a pricing strategy for FOs to maximize profit, reducing electricity costs by charging at the minimal TOU price hours and then adding an optimal margin on top of the base price, without increasing the price so much that EV drivers choose not to charge. The goal of this study is to help developers determine where to install chargers, and how to operate with financial sustainability, while also working towards environmental sustainability.

II. TECHNICAL DESCRIPTION

A. Modeling Steps

1) Programming Mechanism: We started by formulating the optimization problem that the hypothetical FO aims to solve. We found that it is intractable to solve using classic Dynamic Programming. Even if we consider the simple case of having to choose between charging and not charging, and apply 1 tariff for all charging stations (i.e. we do not optimize tariff), 10 charging stations, and a grid of 10 elements for the state of charge, the size of the grid that needs to be considered for the optimization variables and occupancy is already $2^{20} \times 10^{10}$, which is way too large. This is an illustration of the curse of dimensionality for Dynamic Programming: the size of the grid is exponential with respect to the dimension of the state. Nevertheless, the problem (SP) is separable between stations if not for the coupling constraint (C1).

Thus, instead of using Dynamic Programming, we use a Monte-Carlo scenario based approach. We argue that although this method is an approximation of the initial problem, it provides a

tractable and good enough solution.

2) Math Modeling Mechanism: We started by defining the stochastic optimization problem that we aim to solve. The detailed constraints and inputs can be found in the following subsection. Our assumptions are as follows:

- Only grid to vehicle is possible: an EV cannot discharge power to the grid
- The car time-of-arrival distribution depends on a dual-peak probability density function proposed by Tulpule [1]. We redesigned the peak values for those two peaks to relate to our proposed workplace EV parking scenario. We set the two peaks at 8:30am and 12pm and with probability of arrival of 0.9 and 0.4 respectively.
- The car time-of-departure distribution is uniformly distributed between 2 hours after the arrival time and 7pm. A random noise of 1 hour is added to this stated time to account for users' uncertainty.
- The EV owner has to decide whether to accept the tariff proposed by the parking manager. If the EV rejects the tariff, it leaves immediately, opening the space for another vehicle. Whether the car accept the price is depended on a logistic function where the midpoint is set at 33 cents per kWh (equivalent to \$3 per gallon of gas).
- Each parking spot is modelled separately in the model, which means no interactive impacts are considered among parking spots.
- When an EV parks, it is required to provide an estimated time of departure, current SOC, and desired final SOC. The FO then guarantees the additional charge for the car based on the given time of departure.

3) Modeling Mechanism: As for building the model, we started with one parking spot in our model to understand the price sorting mechanism. Then, we extend the number of parking spots to 10 to understand the human behavior on people's time-of-arrival and occupancy. After gathering the human behavior data from our probability function, we designed a parking lot with n ($n=10$) EV charging stations and applied the optimization objective to be minimizing the charging cost of the charging stations, from the FO's point of view, on 15-minute interval basis for one week (7 days and 24 hours per day). At the end, we tried to implement more sustainable approach to our parking lot by sorting price with carbon emissions for flexible charging.

B. Notation

The variables that we used in the model are defined in Table 1, for your reference. We index the charging points by $i \in \{1, \dots, n\}$. A variable z_i^t refers to the value of z at charging point i and time t . Capital letters will generally denote random variables (R.Vs), small letters denote an occurrence of a random variable. $\mathbb{E}_X[f(X)]$ denotes the expectation of the R.V $f(X)$. $\mathbb{P}(A)$ denotes the probability for an event A to occur. Each variable is explained further below.

1) Parameters [Θ]:

- TOU^t [\$/kWh] is the time-of-use price at time t
- \bar{w}_i [kW] denotes the maximum charging rate of charging point i (this is determined by the charger type/level). For our analysis, assumed to be 10kW
- $\beta \in \mathbb{R}$ is the profit margin we apply to TOU prices.

Table I
VARIABLE DEFINITION

Symbol	Description	Units	Type
t	Time step	hours	index
i	Station	number	index
N_i^t	Count of EVs arrived	number	R.V.
A_i^t	Event of a car arrival	{1,0}	R.V.
D_i^t	Event of car departure	{1,0}	R.V.
$T_i(N_i^t)$	Stated time of departure by N^{th} car	hours	R.V.
TOU^t	TOU cost of electricity at time t	\$/kWh	parameter
β	Profit margin	%	parameter
$\bar{S}_i(N_i^t)$	Desired energy of N^{th} car	kWh	R.V.
$C_i(N_i^t)$	Choice of N^{th} car	{1,0}	input
O_i^t	Occupancy	{1,0}	input
U_i^t	Charging rate	kW	optim.
$\Lambda(N_i^t)$	Price displayed to EV	\$/kWh	optim.

2) Exogenous Random Variables [W variables]:

- $T_i(N_i^t) \in \{1, \dots, h\}$ is the expected/stated time of departure for EV N_i^t
- $\bar{S}_i(N_i^t)$ [kWh] is the total charging energy required by EV N_i^t before stated departure time.
- $A_i^t \in \{0, 1\}$ is the event of a car arriving at station i at time t , although considered exogenous here, we will only model conditioned on the occupancy of station i , which is an endogenous R.V.
- $D_i^t \in \{0, 1\}$ is the event of a car departing from station i at time t , we consider this event to be conditional on $T_i(N_i^t)$. Note that we assume only one departure or arrival event is permitted during a given time interval t .
- $N_i^t \in \mathbb{N}$ (\mathbb{N} is the positive integer set) is the count of EVs that arrived at charging point i from time 1 to t . A given EV might refuse the tariff proposed by the FO, this event still count as $N_i^t \leftarrow N_i^t + 1$, It is a monotonically increasing step function that takes integers values. We can see this as allowing to have an identification number for a given EV during the time horizon. The use of this variable is made necessary by the fact that EVs are out of the system of definition: some EVs come and go, but we don't know which, EVs are not part of the system of definition.

We will refer to these exogenous variables as W_i for a given station. We assume that W_i independently is and identically distributed (i.i.d)

3) Endogenous/controllable Random Variables [E variables]:

- $C_i(N_i^t) \in \{0, 1\}$ is the choice made by EV N_i^t . Where 0 corresponds to choosing not to charge and park at a charging station and 1 to accept the tariff and charge at a station. We will assume that we already have a behavioral model,

$$\mathbb{P}(C = 1 | \lambda) = \sigma(\beta \lambda)$$

Where σ is the logistic function

$$\sigma(\Lambda) = \frac{1}{500(1 + \exp(\Lambda - 0.33))}, \quad x \in \mathbb{R} \quad (1)$$

0.33 corresponds to the price that makes charging a car equivalent to paying \$3/gallon in gasoline prices to drive an equivalent distance[8]. And "500" squeezes the function to match

the shape of how we anticipate human behaviors would follow.

The decision to use gasoline prices as a proxy for decision making for consumers was made because the general public is much more conscious about gasoline prices than electricity prices, so we can present the equivalent gasoline price for them to decide whether to charge or not. A \$3/gallon value is still cheaper than gasoline prices (at \$4/gallon at the time of writing) and is still attractive to EV owners to charge at. Over this amount, charging gets closer to gasoline prices and EV users presumably won't be willing to pay. To conclude, $C_i(N_i^t) \sim \mathcal{B}(\sigma(\beta\lambda_i^t(N_i^t)))$, where \mathcal{B} denotes a Bernoulli distribution with parameter $p := \sigma(\beta\lambda_i^t(N_i^t))$.

- $O_i^t \in \{0, 1\}$ is the occupancy of charger i at time t

4) **Optimization variables [X variables]:** Note that all optimization variables are also random variables because before a realization of W or E

- U_i^t [kW] is the charging rate at charging point i and time t ,
- $\Lambda_i(N_i^t)$ [\$/kWh] $\in \mathbb{R}$ is the proposed tariff to EV N_i^t . Note that it is a fixed total tariff for EV N_i^t , it does not vary with time contrary to TOU, although this fixed price will depend on the arrival, departure time and total energy required.

C. Objective function

Our objective function f is to minimize FO's charging cost on a 15-minute time interval basis per week given TOU cost, charging rate, and event of a car arrival.

$$f = \min \mathbb{E}_{W,E} \left[\sum_{t=1}^h \sum_{i=1}^n U_i^t (TOU^t - \Lambda_i(N_i^t)) \right]$$

D. Constraints

1) **Technical constraints:** At the aggregate level, we assume that there exist a maximum charging rate \bar{u}_Σ , and at an individual level, a maximum charging rate \bar{u}_i , such that:

$$\sum_{i=1}^n U_i^t \leq \bar{u}_\Sigma, \quad \forall t \tag{C1}$$

$$U_i^t \leq \bar{u}_i, \quad \forall t \tag{C2-i}$$

We would also need make sure an EV reaches its desired SOC between its arrival and departure times.

$$S_i^f - S_i^0 = \sum_{t=A(N_i)}^{D(N_i)} U_i^t \tag{2}$$

2) *Controllable RVs dynamics:* We have the EV count dynamic at a station i given by

$$N_i^{t+1} = N_i^t + A_i^t$$

where we count all EVs even if they choose not to accept the tariff and leave the station. The occupancy dynamic is given by

$$O_i^{t+1} = O_i^t - D_i^t + A_i^t C_i(N_i^t) \quad (\text{C3-i})$$

This equation is central because it is here that the choice of the user has an impact. If one parking spot is free and a EV arrives, the charging station will be occupied in the next time step ($O_i^{t+1} = 1$) if the user accepts the tariff.

3) *Non technical constraints:* The tariff to charge vehicles are tied to TOU price at that time. The β value is our profit margin.

$$\Lambda_i(N_i^t) = (1 + \beta)TOU^t \quad (\text{C4-i})$$

stochastic optimization problem can be written compactly as

$$\begin{aligned} \min_X \quad & \mathbb{E}_{W,E} \left[\sum_{t=1}^h \sum_{i=1}^n U_i^t (TOU^t - \Lambda_i(N_i^t)) \right] \\ \text{s.to} \quad & C1 \text{ and } (C2 - i, \dots, C4 - i), \forall i \end{aligned} \quad (\text{SP})$$

An visualization of our model can be seen in Figure 2, below.

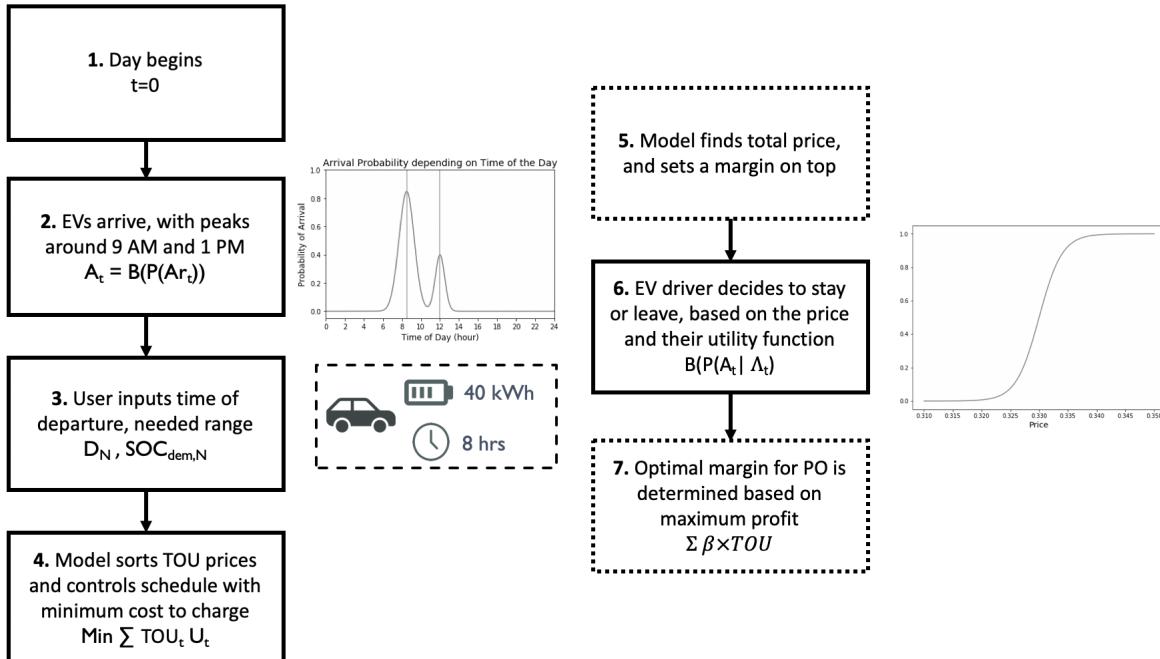


Figure 2. Model Visualization

E. A Simplified Approach

The stochastic problem formulated in the previous section is not tractable to stochastic dynamic programming. To deal with this, we loosened the constraint under the assumption that a parking lot would not build more charging spots than it can actually service. For example, if a parking lot has a rated capacity of 100kW, and each parking spot requires 10kW, the parking lot would only build 10 parking spots. We realize that an operator may choose to build more so it can stagger its charging, i.e. while some cars are done charging, other cars can be charged. Our analysis therefore only applies to the first case.

To find the optimal pricing strategy, we simply ran Monte Carlo simulations using different β values to find the one that yields the highest profit for the FO. A β value at 0 would mean \$0 as profit (costs = revenue), but an extremely high β value would mean all approaching EVs will reject that price. There will be a sweet spot in the middle where the FO earns the highest profit, and EVs enjoy the benefit of having lower driving costs than gasoline. Our approach to this problem is therefore simply to iterate through values of β and finding the one that yields the maximum profit.

Our Monte Carlo simulation involved running a parking spot across 365 days and 24 hours per day. The mechanics of the model is as such:

We used the previously described probability curves to simulate arrival times, and once a car arrives, it tells the FO how much charge it needs, and how long it will park for based on uniform distributions. The FO then calculates the time needed to charge and spots the cheapest time slots to charge the car. Then it averages the prices at those hours and adds a margin β on top. The new price is then plugged into the logistics curve (Equation 1) to evaluate how likely an EV will accept that price. The profit ($\beta \times TOU$) is then tallied across a year. In our analysis, we simulated profits both for flexible charging and fast charging scenarios.

F. Preliminary Results

1) **Charging Price Reflection:** The determined price for charging example is shown in Figure 3. In this example, an EV arrives at 10:00am and states to stay parked until 7:00pm, with an energy need of 20 kWh. The price curve here shown in blue is the real time prices as discussed previously. The grey shade is the occupancy time period of the parking spot. During this period, the FO will spot the lowest prices of electricity and charge the car accordingly (during the green time intervals).

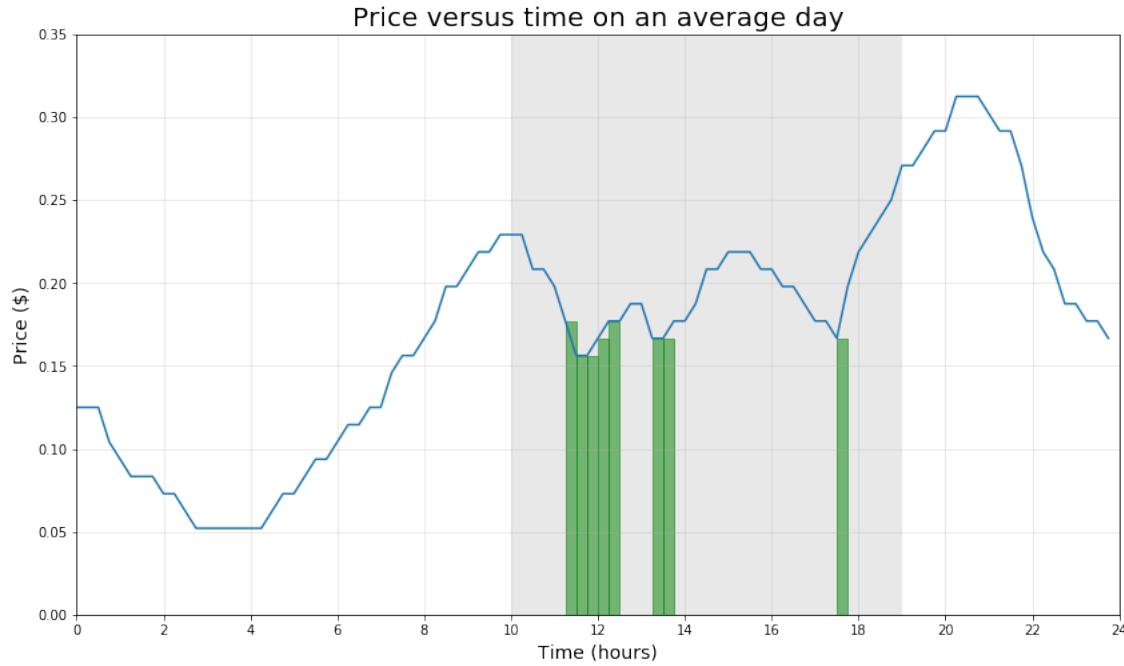


Figure 3. Flexible charging schedule and control

2) **Occupancy Simulation:** After determining our electricity curve, we increased the parking spots to be 10 and used the statistical distributions described previously to model the user's behavior to simulate a potential working week of occupancy. The results of the simulation is shown in Figure 4. In this figure, the black blocks are the occupied spots. The blue and orange lines represents the two peaks of the time-of-arrival density function. The red line represents the maximum time at which a user should have left based on the time-of-departure function. We can see that some EVs stay parked longer due to the random noise we imputed to the stated departure time to better approximate human behavior. By generating the random occupancy data, we are able to execute our optimization model to find the minimal charging cost for the FO.

3) **Optimization Charging Profits:** We investigated how to maximize a FOs profit on top of this base marginal cost. We used a utility function to represent an EV owner's choice to stay or go, based on price. The inflection point of the logistic function was based on prices people are willing to pay for gasoline in an ICE vehicle. Then, we developed a single parking spot after aggregate with the acceptance probability curve for both urgent charging and flexible charging. We chose the time horizon to be one year to smooth the curve and also because the profit over a year is more relevant from the point of view of future investors. The results are shown in Figure 5.

As we can see in Figure 5, a FO could charge a 66% margin on top of a flexible charging schedule, for a \$919 profit per charging spot over one year. In contrast, a FO at a conventional charging point would only be able to charge a 60% margin for a \$842 profit. This difference is due to the fact that fast charging would have higher costs, which leads to higher prices, resulting in fewer EV owners accepting the charge compared to rejecting the high price and leaving. This is encouraging because this quantifies a 6% increase in profits with the same end product delivered to consumers at no extra cost to the FO.

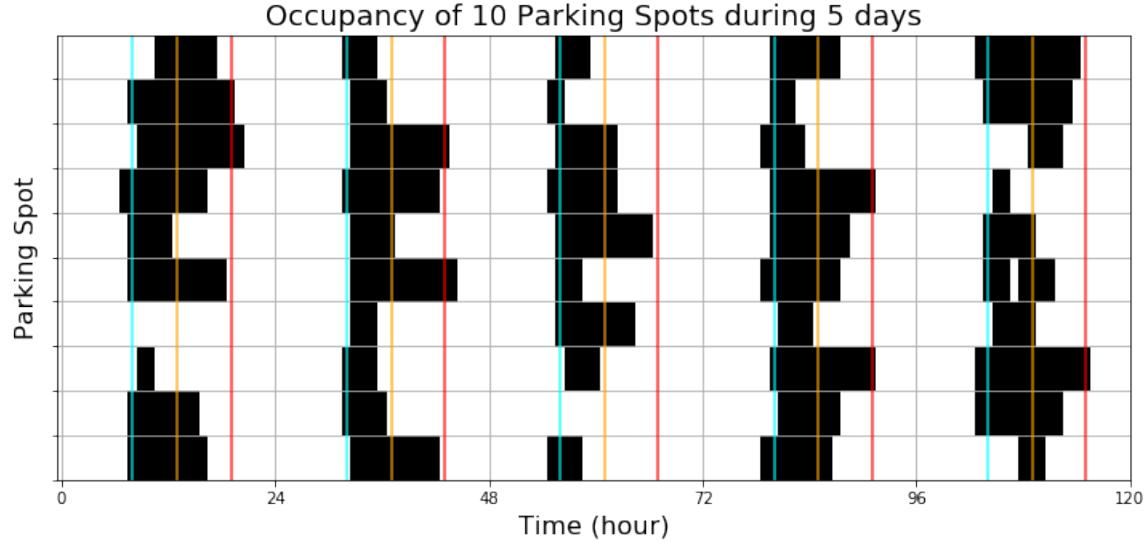


Figure 4. Simulation results of parking garage with 10 spots

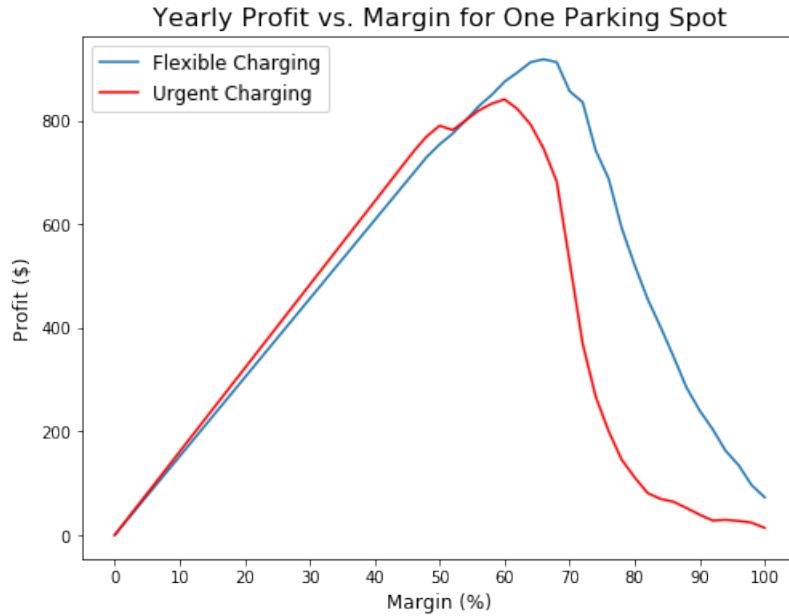


Figure 5. Profit and optimal margin for normal charging compared to flexible charging

III. DISCUSSION

A. Result Elaboration

1) **Charging Price Reflection:** As we assumed no interaction between parking spots, we first modeled only one parking spot for a week. Our first model had no significant differences between flexible charging and fast charging because the wholesale electricity data we used had little variation during the parking period (we used average Berkeley locational marginal price data from CAISO). Our second iteration used a model of real time prices from 295

class/homework, which had more significant changes in price during the daytime. Berkeley likely has low fluctuations due to its mild climate, but other cities TOU prices could reflect a curve more similar to our second model. Currently, actually, consumers do not typically see fluctuations in price over the course of the day, but our model follows projects of the future that wholesale prices will be passed off to consumers to shift demand as more and more renewable energy enters the grid.

2) Optimization Charging Profits: We found that flexible charging could save the FO an average 10% on their costs compared to normal charging occurring immediately after plug-in. The result is exciting because it is more likely to persuade FO to build flexible charging EV parking lot when it generates better profits. At the same time, flexible charging can help to smooth the duck curve, which also achieve sustainable energy system for the future transportation structure.

3) Return on Investments: Our analysis assumed FOs operate level 2 chargers in the parking lots (between 3.4kW to 19.2kW output). These chargers cost between \$400 and \$6,500, and on top requires an installation fee between \$600 to \$12,700, bringing a total cost to \$1,000 to \$19,200 [9]. Picking an average value at \$10,000, and using the \$919 that we predict a charger can earn in a year, it would take just over 10 years to recover the capital investment. The \$919 per year amount is only one part of the value added by a charger. For example, if a office building builds them, there is value in attracting tenants with EVs, and value in projecting an image that the building cares about sustainability.

4) Emission Adjustment: As mentioned in our introduction, we also thought about further developing our parking lot in a more sustainable way by reducing the carbon emission from the parking lots. Instead of sorting charging time with the lowest charging price, we decided to sort the charging time with the lowest emission hours, which in general are those hours with more renewable energy portions. However, after we tested our lower carbon emission option, we found that there is not much variation from the results we found from optimizing prices, because the parking periods are mostly located when there is high solar production and therefore low carbon emissions from the grid. The emission graph is shown in Figure 6, the range within the green shade is our proposed high occupancy hour while the red lines are the boundaries for the peak arrival hours.

However, because we set a typical work day as between 8:30AM and 7PM, the variance on carbon dioxide emissions during this period was low, and an optimization for lowest emissions had little effect on profits or cost.

B. Significance of Our Study

Utilizing a flexible charging software will advance sustainability in energy systems in two different ways. First by reducing the magnitudes of peak loads, which reduces utilities' need to build inefficient fossil fuel peaker plants. Second, because low cost hours are likely times of high renewable penetration, shifting charging to be during these hours will procure more energy from renewables than normal charging.

The main deliverable of our project is producing a tool that allow FOs to determine an optimal margin for EV charging to add on top of a base rate for electricity. By controlling the charging

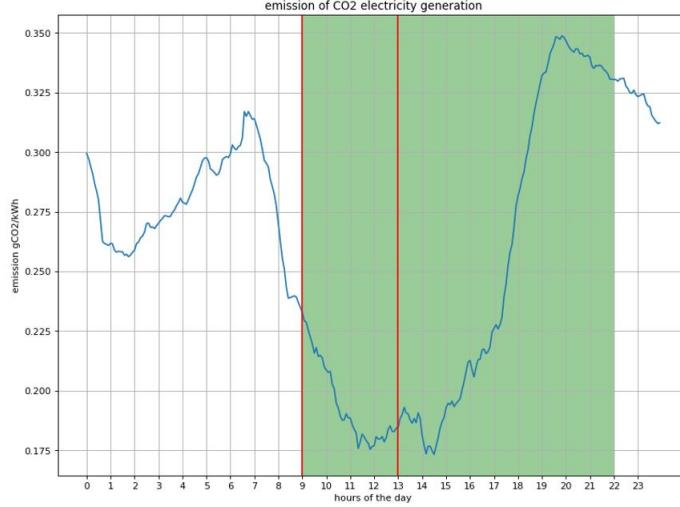


Figure 6. Carbon emission generation from different operating hours

to be during the cheapest hours, the FO can increase their profit without increasing price as much as they would have to without flexible charging. We therefore provide more sustainable charging options to both an FO and to EV drivers, to promote sustainability in energy systems. By innovatively charging during the day to avoid adding peak loads to the duck curve, we successfully found an easy implementation to contribute in energy sustainability. This task cannot be achieved without using system and control tools to pick out the cheapest charging hours and to calculate the optimal charging price in a short amount of time.

C. Future Studies

1) Parking Spot Interactions: Demand charge: In the future, other students should think of modeling the interactions among each parking spots to the designed parking lot. In reality, different parking spots have varying cumulative impact. For instance, an FO might want to stagger charges as to avoid a demand charge. A demand charge is a tariff imposed by a utility on a the highest hour of use for the month, in addition to the regular \$/kWh based system. The purpose is to incentivize consumers to reduce their peak loads and thus reduce the utility's need for peaker plants. Incorporating a demand charge into the model would be an interesting next step for future students, and would require considering the interplay between multiple parking spots, as well as some duality theory.

2) Parking Spot Interactions: Surge Pricing: Another interesting further topic could be to add another charge on top of the base rate charged to the consumer, that accounts for high demand and low supply of charge spots. Essentially similar to Uber or Lyft's 'surge pricing' algorithms, this would allow the FO owner to increase prices at times of high demand.

For example, with 5 open spots left on the parking lot, the price should be different when 1 car arrive versus 15 cars arrive. We proposed to build two metrics to solve this problem. For example, with 20 parking spots and assuming a maximum 50 EV demand, we can put the proposed different prices to each situation as described in the Figure 7.

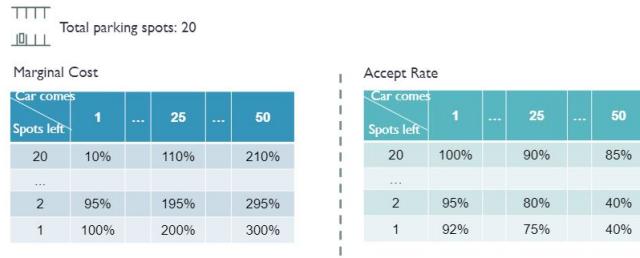


Figure 7. Additonal charge on demand and supply spot changes

As shown in Figure 7, after deciding the price for certain available parking spots and EV demand, we can have another table suggesting the human behavior on the proposed price. The number in the above Figure is just an example, without furthering human behaviour data support. In the future, a more realistic model should be explored to represent the human behaviour study, perhaps created using survey data.

IV. SUMMARY

The focus of this paper is to create a two-choice optimal pricing model to maximize a facility owner's profits. The two user choices are between cheaper grid-flexible charging and not charging at all. We used Monte Carlo simulation to generate scenarios and convert the stochastic variables into known variables and run the optimization model for every scenario. We found that by deploying flexible charging, a FO can increase their margin from 60% up to 66%, reduce costs by 10% and raise profits by 9% on one parking spot over a year when compared to the immediate charge alternative. We hope that our model provides a quantitative tool for FOs to calculate EV charging installation profitability.

REFERENCES

- [1] P. J. Tulpule, V. Marano, S. Yurkovich, and G. Rizzoni, "Economic and environmental impacts of a pv powered workplace parking garage charging station," *Applied Energy*, vol. 108, no. C, pp. 323–332, 2013. [Online]. Available: <https://EconPapers.repec.org/RePEc:eee:appene:v:108:y:2013:i:c:p:323-332>
- [2] O. Sundstrom and C. Binding, "Flexible charging optimization for electric vehicles considering distribution grid constraints," *IEEE Transactions on Smart Grid*, vol. 3, no. 1, pp. 26–37, 2012.
- [3] M. Alonso, H. Amaris, J. Germain, and J. Galan, "Optimal charging scheduling of electric vehicles in smart grids by heuristic algorithms," *Energies*, vol. 7, no. 4, pp. 2449–2475, 2014.
- [4] P. Samadi, A.-H. Mohsenian-Rad, R. Schober, V. W. Wong, and J. Jatskevich, "Optimal real-time pricing algorithm based on utility maximization for smart grid," in *2010 First IEEE International Conference on Smart Grid Communications*. IEEE, 2010, pp. 415–420.
- [5] Z. Ding, Y. Lu, L. Zhang, W. J. Lee, and D. Chen, "A Stochastic Resource-Planning Scheme for PHEV Charging Station Considering Energy Portfolio Optimization and Price-Responsive Demand," *IEEE Transactions on Industry Applications*, 2018.
- [6] Y. Cao, S. Tang, C. Li, P. Zhang, Y. Tan, Z. Zhang, and J. Li, "An optimized ev charging model considering tou price and soc curve," *IEEE Transactions on Smart Grid*, vol. 3, no. 1, pp. 388–393, 2012.
- [7] T. Zeng, H. Zhang, and S. Moura, "Solving overstay in phev charging station planning via chance constrained optimization," *arXiv preprint arXiv:1901.07110*, 2019.
- [8] D. of Energy, "egallon," <https://www.energy.gov/maps/egallon>, April 2019.
- [9] U. D. of Energy, "Costs associated with non-residential electric vehicle supply equipment," November 2015.

Optimal spatial and temporal schedule of EV fleet charging and discharging in the Bay Area

CE 295: Energy Systems and Control Final Project

Hugo Basset, Tristan Metz, Jasper Pakshong, Sierra Spencer, Jean-Paul Wallis



May 11, 2019

Abstract

This project accomplishes an economic optimization for a fleet of autonomous plug-in electric vehicles (PEVs). There is a large opportunity for PEVs to provide a necessary link in large-scale, sustainable transportation electrification in the form of both electrical load and storage. A convex optimization program is discretized using the Lax-Wendroff method. The program generates optimal times of providing mobility services (using simulated trajectories based on Uber usage data), discharging to the grid, and charging given the wholesale prices of electricity. The scheduling model works to minimize the cost of purchased electricity and maximize the revenue from providing mobility services. This project verifies the capability to integrate control systems technology with PEV fleet aggregation. Future work to improve the optimization model could incorporate electricity emission rates, grid ancillary services, and increased iteration density.

1 Motivation and Background

The transportation sector is undergoing rapid transformations and technological advances alongside the need for urgent climate action. PEVs have seen significant improvements in battery capacity and affordability, giving them the potential to play a significant role in the passenger vehicle sector. Ridesharing holds the potential to revolutionize modes of personal transportation in the next decade. Research and development into autonomous vehicles has also drastically increased, paving the way for self-driving passenger vehicles.

Together, these changes and advancements allow us to envision a society in which personal mobility is satisfied through a shared, autonomous, and electric fleet of vehicles. For our project, we aim to develop a charge/discharge and mobility schedule optimization program that could be viewed as a framework to be built upon in order to consider many of the potential benefits and challenges from a shared, autonomous, and electric vehicle fleet.

Rapid adoption of PEVs can significantly alter instantaneous electricity supply and demand. Although PEVs increase demand while charging, they can also provide load shifting and ancillary services to help equilibrate the grid through vehicle-to-grid (V2G) services. Perhaps most pertinently, PEVs can beneficially charge during periods of high renewable electricity penetration and discharge to the grid during demand ramps.

The usage of PEVs in a shared, autonomous transportation network adds complexity to the manner in which vehicles are operated. Under an autonomous and shared fleet, vehicles are no longer parked at a user's destination, but can continuously seek the opportunities to satisfy mobility demand in the most efficient way possible. With PEVs capable of V2G populating a shared and autonomous fleet, considerations of charging and discharging must be incorporated in determining how to best satisfy demand and operate the fleet.

While most models of PEV charging have focused on economically optimal charge/discharge patterns from the perspective of either the PEV user or the grid operator, significantly less attention has been given to finding the environmentally optimal charging/discharging schedule. The availability of renewable electricity resources has the potential to vary greatly over the course of a day. With limited storage, instantaneous charging is the primary current method of harvesting renewable resources when they are available. Additionally, many optimizations of PEV charging focus on individual personal passenger vehicles, and few studies consider a shared fleet, particularly an autonomous shared fleet.

Our team built a model to find the optimal charging and discharging schedules for a shared, autonomous, electric fleet to achieve the lowest operating cost. We take into consideration constraints such as the necessity to satisfy driver travel demand, battery range, and battery capacity. We determined the optimal charging and

discharging schedule during a day in July 2018 based on a data set of personal mobility throughout the San Francisco Area and available data on electricity prices.

2 Relevant Literature

A wealth of literature on PEV/PHEV charging optimization is available. Different optimization programs focus on different objectives such as minimizing charging costs and maximizing user satisfaction[1], minimizing wait times[2], and maximizing economic performance of bi-directional charging[3]. There are also studies that center the use of distributed algorithms across PEV charging stations to optimize charge and discharge schedules[4].

The products and findings from studies vary. Many studies aim to develop schedules for optimal charging[1][5][6]. This includes components such as choice of charging station and the amount of charge (or discharge, in some cases). Alonso et al. found that optimal charging is capable of flattening the load profile, reducing peak load, and minimizing aging of power system elements[5]. Several studies compared the optimal results to a baseline to assess benefits or costs from optimal managed charging[1][2].

It is important to recognize the constraints that exist in these optimization programs. Constraints such as driver behavior, grid loading, thermal limits, voltage limits, fleet size, network limitations, existing infrastructure, allowable range, converter efficiency, charge/discharge timing, autonomous fleets size, and comfort ratio all showcase the inherent complexity of the control of aggregated PEVs[1][5][6][3][7]. Some studies also considered environmental inputs or goals. Lombardi et al. used a requirement that 20% of energy annually must be from renewable sources[3]. Ahn et al. apply a two level optimization to reduce power generation costs and carbon emissions as well as minimize the usage of traditional ancillary service providers[8].

Data on driver behavior such as parking locations with correlated times of starting and stopping parking were used)[1]. Alonso et al. used transportation data from MOVILIA to determine parking patterns of Spanish EV drivers[5]. Yi et al. used data from Idaho National Laboratory's EV Project[6]. Chen et al. used NHTS data on trip distance and time-of-day distributions[6]. Iversen et al. used a Markov decision process fitted on real EV patterns to estimate over a day the temporal probability that a driver may use its car at a given time[9].

Fewer studies consider environmental inputs. Yao et al. used solar radiation data and Lombardi et al. used conventional and renewable generation plant types and sizes as inputs to their optimization models [1][3].

Nimalsiri provided a comprehensive survey of literature on distributed charge management of EVs[10]. While there are studies that separately consider autonomous EV fleet charge management and charge management of EVs to better utilize renewables, there lack studies that comprehensively optimize charging for an autonomous, public-serving fleet to simultaneously maximize its renewable utilization, meet travel demand, and also provide V2G services by discharging back to the grid.

3 Focus of Study

The goal of this study is to develop an algorithm for optimizing the charging, discharging, and movement profiles of a fleet of autonomous PEVs based on hour-ahead electricity prices and the travel behavior of users within the San Francisco Bay Area.

4 Model Description

4.1 Description of Data

Mobility Demand Data

The transportation demand for this optimization simulation is adapted from the Mobiliti dataset. The Mobiliti data is a simulated set of trajectories for a single day based on Uber trips within the Bay Area. The data set included the start and end latitudes and longitudes of each trip as well as the length of time and distance of each trip. Figure 1 shows a heat map of the associated departures (areas in red are the areas with the highest number of departures).

We then use an Uber traffic analysis zone (TAZ) file containing geospatial cells which each contain a population of 2000. By applying a k-means clustering algorithm weighted by population, we create 30 cells of interest in the area corresponding to the Mobiliti dataset. Figure 2 is an illustration of the clustering process.

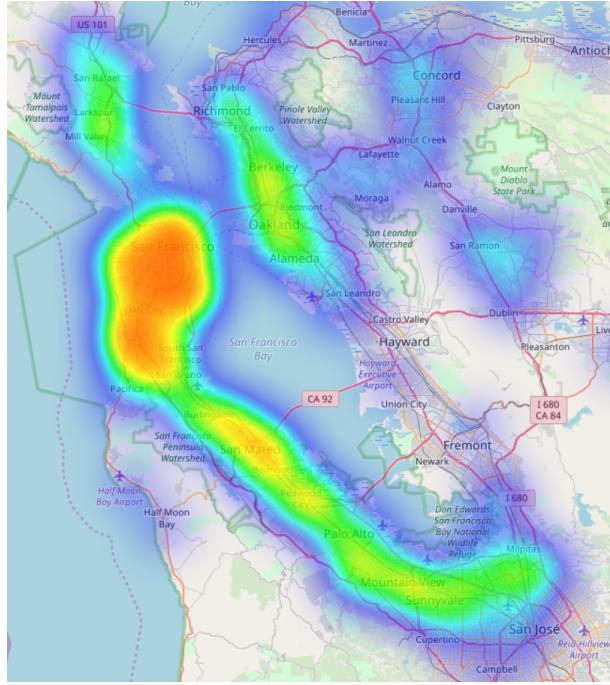


Figure 1: Heat map of Mobiliti departures at 5pm

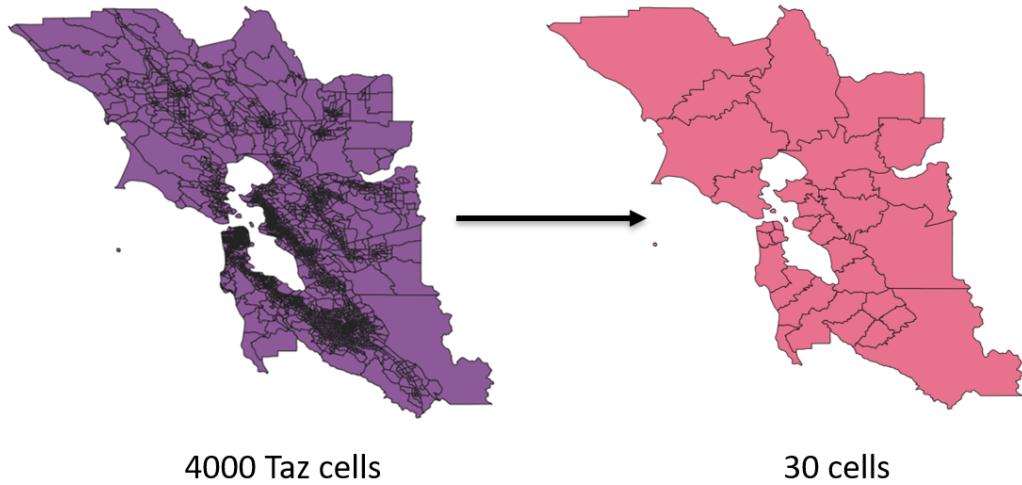


Figure 2: Clustering process with k-means clustering algorithm weighted by population density

Ridesharing Rate and Profit Data

The Uber Price Estimator formula was used to estimate a breakdown of rates charged and profits received by a mobility company. Rates included a base fare, booking fee, minimum fare, per-minute fare, and per-mile fare[11].

Electricity Price Data

Kevala API was used to determine the locational marginal prices (LMPs) for each of the nodes in the Bay Area encompassed by our study for sample day in July 2018. Data was broken down by hour, so prices were extrapolated to be continuous over a given hour to fit the 10-minute consideration of our model.

Determining costs

The state of charge (SOC) cost (in step units) to operate the vehicles was inferred from vehicle economy data from the US EPA for a Chevrolet Bolt EV. The time cost is calculated using average time of trips between

nodes and the electricity consumption per trip and mobility cost (e.g. from node 1 to 4) were calculated using average distances between nodes.

4.2 Model Formulation

Assumptions

Several assumptions are made to model the behavior of the cars:

- The geography is split in several spatial areas in which demand for mobility to the other cells is evaluated.
- The cars can belong to 4 states: charging, discharging, idle without a passenger and idle with a passenger.
- To avoid keeping track of the SOC of each car individually, we handle the cars populations of SOC distributions.
- A car must necessarily be in an idle state (1: with a passenger, 0: without a passenger, car relocation) to move from one cell to another.
- Flows between populations are constrained as we develop the model further below. For instance, a vehicle in a charging state can not take a passenger right away (i.e. move to the idle with passenger state) and must transfer to idle without passenger first.
- Distance and time to go to charging stations are neglected (compared to step size). By doing so, we assume that the charging stations infrastructure is not a constraining factor in term of locations or available power to be dispatched.

Problem Variables

All variables used throughout the formation of the problem are shown in Tables 1 and 2. Table 3 describes the parameters used for the discretization scheme.

Table 1: Model parameters

Notation	Units	Description
$N_{EV} = 50000$	cars	Number of EVs car in the fleet
$t \in [0, T], T = 24$	hour	time
$s \in [0, 1]$	unit	EV's SOC
$N_B = 30$	unit	Number of geographical bins
$(x_j)_{j \in [0, N_B]}$	-	Spatial bins partitioning the studied area
$S_{min} = 0.2$	unit	Minimal EV's SOC necessary to discharge or drive
Δt_{ji}	hour	Time required to go from bin j to i
Δs_{ji}	unit	SOC required to go from bin j to i
$p_{mob}(j, i)$	\$/trip	mobility price between bins j and i
$\Delta t_c = 0$	hour	Time to go to the closest charging station
$\Delta s_c = 0$	hour	SOC to go to the closest charging station
$q_C = 10$	kW	Instantaneous charging power at time t and SOC s
$q_D = -10$	kW	Instantaneous discharging power at time t and SOC s
$p_{elec}(t, j)$	\$/kWh	Electricity price at time t and bin j
$D(t, j, i)$	trips	Demand for mobility at time t between areas j and i

Table 2: Optimization variables

Notation	Units	Description
$u_j(s, t)$	cars	SOC distribution of charging cars at node j and time t
$v_j^0(s, t)$	cars	SOC distribution of idle cars without passenger at node j and time t
$v_j^1(s, t)$	cars	SOC distribution of idle cars with passenger at node j and time t
$w_j(s, t)$	cars	SOC distribution of discharging cars at node j and time t
$\sigma_{I_j^0 \rightarrow C_j}(t, s)$	cars/hour	Flow of vehicles from bin j switching from idle to charge at time t and SOC s
$\sigma_{C_j \rightarrow I_j^0}(t, s)$	cars/hour	Flow of vehicles from bin j switching from charge to idle at time t and SOC s
$\sigma_{I_j^0 \rightarrow D_j}(t, s)$	cars/hour	Flow of vehicles from bin j switching from idle to discharge at time t and SOC s
$\sigma_{D_j \rightarrow I_j^0}(t, s)$	cars/hour	Flow of vehicles from bin j switching from discharge to idle at time t and SOC s
$\sigma_{C_j \rightarrow D_j}(t, s)$	cars/hour	Flow of vehicles from bin j switching from charge to discharge at time t and SOC s
$\sigma_{D_j \rightarrow C_j}(t, s)$	cars/hour	Flow of vehicles from bin j switching from discharge to charge at time t and SOC s
$\sigma_{I_j^0 \rightarrow I_j^1}(t, s)$	cars/hour	Flow of idle vehicles from bin j accepting a passenger at time t and SOC s
$\sigma_{I_j^1 \rightarrow I_j^0}(t, s)$	cars/hour	Flow of idle vehicles from bin j dropping a passenger at time t and SOC s
$\sigma_{I_j^0 \rightarrow I_i^0}(t, s)$	cars/hour	Flow of idle vehicles from bin j going to bin i without a passenger at time t and SOC s
$\sigma_{I_j^1 \rightarrow I_i^1}(t, s)$	cars/hour	Flow of idle vehicles from bin j going to bin i with a passenger at time t and SOC s

Temporal evolution of car populations

$$\frac{\partial u_j}{\partial t}(t, s) = -q_c \frac{\partial u_j}{\partial s}(t, s) + \sigma_{I_j^0 \rightarrow C_j}(t, s) - \sigma_{C_j \rightarrow I_j^0}(t, s) + \sigma_{D_j \rightarrow C_j}(t, s) - \sigma_{C_j \rightarrow D_j}(t, s) \quad (1)$$

$$\begin{aligned} \frac{\partial v_j^0}{\partial t}(t, s) = & \sum_{i \in [1, N_B]} \sigma_{I_i^0 \rightarrow I_j^0}(t - \Delta t_{ji}, s + \Delta s_{ji}) - \sigma_{I_j^0 \rightarrow I_i^0}(t, s) + (\sigma_{I_j^1 \rightarrow I_j^0}(t, s) - \sigma_{I_j^0 \rightarrow I_j^1}(t, s)) \\ & + (\sigma_{D_j \rightarrow I_j^0}(t, s) - \sigma_{I_j^0 \rightarrow D_j}(t, s)) + (\sigma_{C_j \rightarrow I_j^0}(t, s) - \sigma_{I_j^0 \rightarrow C_j}(t, s)) \end{aligned} \quad (2)$$

$$\frac{\partial v_j^1}{\partial t}(t, s) = \sum_{i \in [1, N_B]} \sigma_{I_i^1 \rightarrow I_j^1}(t - \Delta t_{ji}, s + \Delta s_{ji}) - \sigma_{I_j^1 \rightarrow I_i^1}(t, s) + (\sigma_{I_j^0 \rightarrow I_j^1}(t, s) - \sigma_{I_j^1 \rightarrow I_j^0}(t, s)) \quad (3)$$

$$\frac{\partial w_j}{\partial t}(t, s) = -q_d \frac{\partial w_j}{\partial s}(t, s) + \sigma_{I_j^0 \rightarrow D_j}(t, s) - \sigma_{D_j \rightarrow I_j^0}(t, s) + \sigma_{C_j \rightarrow D_j}(t, s) - \sigma_{D_j \rightarrow C_j}(t, s) \quad (4)$$

Equation 1 (respectively 4) describes the variation of the number of charging (respectively discharging) cars in bin j at time t and SOC s. The first term corresponds to the vehicles currently charging (respectively discharging) and whose SOC is increasing beyond (decreasing below) their previous SOC at speed q_c . The second term correspond to the net number of cars currently idle without a passenger that are switching to charge (respectively discharging) at time t and SOC s. The last term corresponds to the number of cars that were discharging (respectively charging) and switching to charging (respectively discharging).

Equation 2 describes the variation of the number of idle cars without any passengers in bin j at time t and SOC s. The first term corresponds to the net flows of empty idle cars driving from all the other bins to the bin j. These vehicles left before the current time at some higher SOC. The second one represents the net number of cars that are letting a passenger go inside area j. The last term is the net number of cars switching from discharging (respectively charging) to idle without passenger.

Equation 3 describes the variation of the number of idle cars with passenger in bin j at time t and SOC s. The first term corresponds to the net flows of cars with passengers driving from all the other bins to the bin j. They left before the current time and had a higher SOC when they did. The second term is the net number of idle cars taking a passenger inside bin j.

The stocks and flows can be visualized in Figure 3.

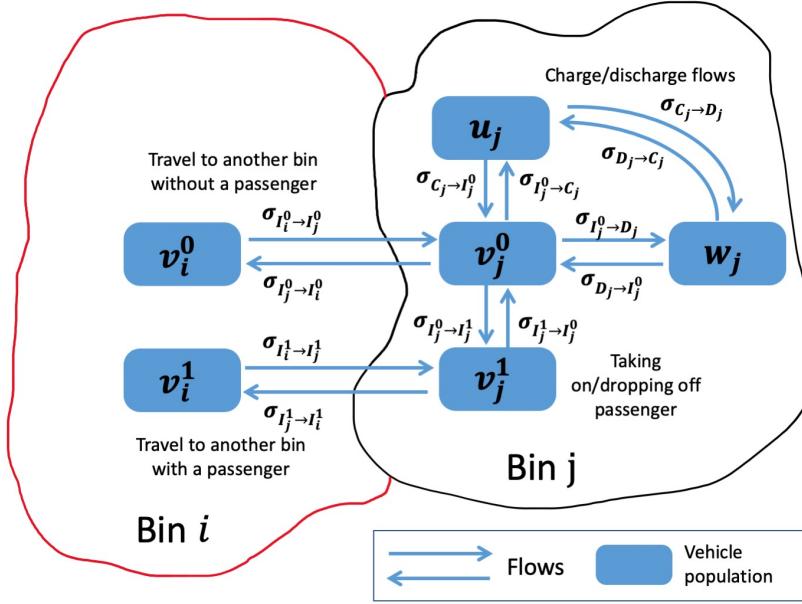


Figure 3: Visual representation of vehicle distributions and flows for a time step

Cost functions

$$Cost_{total} = Cost_{elec} + Cost_{mobility} \quad (5)$$

$$Cost_{elec} = \int_{t=0}^T \sum_{j \in [1, N_B]} \left(\int_{s=0}^1 (q_C * u_j(t, s) + q_D * w_j(t, s)) ds \right) * p_{elec,j}(t) dt \quad (6)$$

$$Cost_{mobility} = - \int_{t=0}^T \sum_{j \in [1, N_B]} \sum_{i \in [1, N_B]} p_{mob}(t, x_i, x_j) \left(\int_{s=0}^1 (\sigma_{I_j -> I_i}^1) ds \right) dt \quad (7)$$

Equation 6 describes the net cost resulting from the charging and discharging operation of the fleet where $u_j(t, s)$ is the number of charging cars and $w_j(t, s)$ the number of discharging cars. Since q_D is negative, the revenue from the V2G is a negative cost. Equation 7 describes the revenues from providing a mobility services where $\sigma_{I_j -> I_i}^1$ is the number of cars leaving the bin j with a passenger (it also works for inside trips).

Constraints on flows

$$0 \leq u_j - \Delta t * (\sigma_{C_j -> I_j^0} + \sigma_{C_j -> D_j}) \quad (8)$$

$$0 \leq w_j - \Delta t * (\sigma_{D_j -> I_j^0} + \sigma_{D_j -> C_j}) \quad (9)$$

$$0 \leq v_j^0 - \Delta t * \left[\left(\sum_{i \in [1, N_B]} \sigma_{I_j^0 -> I_i^0} \right) + \sigma_{I_j^0 -> I_j^1} + \sigma_{I_j^0 -> C_j} + \sigma_{I_j^0 -> D_j} \right] \quad (10)$$

$$0 \leq v_j^1 - \Delta t * \left[\left(\sum_{i \in [1, N_B]} \sigma_{I_j^1 -> I_i}^1 \right) + \sigma_{I_j^1 -> I_j^0} \right] \quad (11)$$

$$\sigma \geq 0 \quad \forall [\sigma] \quad (12)$$

$$\sigma_{I_j^0 -> I_j^1}(t, s) = \sum_{i \in [0, N_B]} \sigma_{I_j^1 -> I_i^1}(t, s) \quad (13)$$

Equation 8 makes sure that the number of charging cars leaving this state in cell j at time t and SOC s can not be higher than the number of charging cars in cell j at time t and SOC s .

Equation 9 makes sure that the number of discharging cars leaving this state in cell j at time t and SOC s can not be higher than the number of discharging cars in cell j at time t and SOC s .

Equation 10 makes sure that the number of idle cars without a passenger leaving this state in cell j at time t and SOC s can not be higher than the number of charging cars in cell j at time t and SOC s.

Equation 11 makes sure that the number of idle cars without a passenger leaving this state in cell j at time t and SOC s cannot be higher than the number of charging cars in cell j at time t and SOC s.

Equation 12 ensures that all flows are positive.

Equation 13 ensures that all cars taking a passenger in area j leaves the area right away.

Constraints on charge and discharge

$$u_j(t, 1) = \Delta t * (\sigma_{C_i -> I_i^0}(t, 1) + \sigma_{C_i -> D_i}(t, 1)) \quad (14)$$

$$\forall i \in [0, N_B], \forall s \leq S_{min} w_j(t, s) = \Delta t * \sigma_{D_i -> C_i}(t, s) \quad (15)$$

$$\forall i \in [0, N_B], \forall s \leq S_{min} v_j^0(t, s) = \Delta t * \sigma_{I_i^0 -> C_i}(t, s) \quad (16)$$

$$\forall i \in [0, N_B], \forall s \leq S_{min} v_j^1(t, s) = \Delta t * \sigma_{I_i^1 -> C_i}(t, s) \quad (17)$$

$$\forall i \in [0, N_B], \forall s \leq S_{min} + \Delta s_{ij}, \sigma_{I_j^0 -> I_i^0}(t, s) = \sigma_{I_j^1 -> I_i^1}(t, s) = 0 \quad (18)$$

Equation 14 makes sure that every car reaching a SOC of 1 does not stay in the charging population and switch to an other population.

Equation 15 to 17 force every cars in idle with or without passenger as well as currently discharging to switch to the charging population if their SOC is below the minimal SOC we authorize.

Equation 18 forbids every car to make a trip that will make it reach a SOC below the minimal SOC when it arrives.

Supply and demand constraints

$$\int_{s=0}^1 \sigma_{I_j^1 -> I_i^1}(t, s) ds \leq D(t, j, i) \quad (19)$$

$$\int_{s=0}^1 \sum_{i \in [1, N_B]} u_i(t, s) + v_i^0(t, s) + v_i^1(t, s) + w_i(t, s) = N_{fleet} \quad (20)$$

Equation 19 limits the number of cars leaving area j to area i with a passenger over all the SOC to be below the demand of trips from area j to area i.

Equation 21 forces the population of cars across all the populations, areas, and SOC to be constantly equal to the fleet size.

Initial conditions

$$\begin{aligned} u_i(., 0) &= u_{i0} = 0 \\ v_i^0(., 0) &= 1/N_B * N(\mu = 60\%_{SOC}, \sigma = 8\%_{SOC}) * N_{EV} \\ v_i^1(., 0) &= v_{i0}^1 = 0 \\ w_i(., 0) &= w_{i0} = 0 \end{aligned}$$

Initially, all the cars are in the idle without passengers state and are equally spread across all cells. Their SOC follow a normal distribution of mean 60% SOC and standard deviation 8%. There are no other cars in any other population.

5 Optimization program

The full optimization program is described mathematically in this section. The equality and inequality constraints are discretized and organized in the canonical format. We use a Lax-Wendroff scheme to discretize the transport equations and ensure that the stability condition for these scheme is satisfied i.e.

$$q_c * \Delta t / \Delta s \leq 1$$

Table 3: Set up for discretization

Notation	Units	Description
$\Delta t = 1/6$	hour	Time step for discretization
$N_T = 36$	step	Discretized time horizon
$k \in [0, N_T]$	step	Discretized time
$\Delta s = 0.033$	unit	EV's SOC discretization scheme
$N_s = 30$	unit	Discretized SOC horizon
$p \in [0, N_T]$	step	Discretized time
$(x_j)_{j \in [0, N_B]}$	-	Spatial bins partitioning an area
Δk_{ji}	steps	Discretized time required to go from bin j to i
$p_{min} = 6$	unit	Minimal EV's SOC necessary to discharge or drive
Δp_{ji}	steps	discretized SOC required to go from bin j to i

$$\min_{\sigma} \sum_{k \in [0, N_T - 1]} \sum_{j \in [1, N_B]} \sum_{p \in [0, N_S - 1]} (q_C * u_j(k, p) + q_D * w_j(k, p)) * \Delta t * \Delta s * p_{elec}(j, k) - \sum_{i \in [1, N_B]} p_{mob}(k, x_i, x_j) * \Delta t * \Delta s * \sigma_{I_j \rightarrow I_i}^1(k, p) \quad (21)$$

subject to

$$u_j(k+1, p) = u_j(k, p) - \frac{1}{2} * \frac{q_c \Delta t}{\Delta s} (u_j(k, p+1) - u_j(k, p-1)) + \frac{1}{2} * (\frac{q_c \Delta t}{\Delta s})^2 (u_j(k, p+1) - 2 * u_j(k, p) + u_j(k, p-1)) + \Delta t * (\sigma_{I_j^0 \rightarrow C_j}(k, p) - \sigma_{C_j \rightarrow I_j^0}(k, p) + \sigma_{D_j \rightarrow C_j}(k, p) - \sigma_{C_j \rightarrow D_j}(k, p)) \quad (22)$$

$$v_j^0(k+1, p) = v_j^0(k, p) + \Delta t * (\sum_{i \in [1, N_B]} \sigma_{I_i^0 \rightarrow I_j^0}(k - k_{ji}, p + p_{ji}) - \sigma_{I_j^0 \rightarrow I_i^0}(k, p)) + (\sigma_{I_j^1 \rightarrow I_j^0}(k, p) - \sigma_{I_j^0 \rightarrow I_j^1}(k, p)) + (\sigma_{D_j \rightarrow I_j^0}(k, p) - \sigma_{I_j^0 \rightarrow D_j}(k, p)) + (\sigma_{C_j \rightarrow I_j^0}(k, p) - \sigma_{I_j^0 \rightarrow C_j}(k, p)) \quad (23)$$

$$v_j^1(k+1, p) = v_j^1(k, p) + \Delta t * (\sum_{i \in [1, N_B]} \sigma_{I_i^1 \rightarrow I_j^1}(k - k_{ji}, p + p_{ji}) - \sigma_{I_j^1 \rightarrow I_i^1}(k, p)) + \Delta t * (\sigma_{I_j^0 \rightarrow I_j^1}(k, p) - \sigma_{I_j^1 \rightarrow I_j^0}(k, p)) \quad (24)$$

$$w_j(k+1, p) = w_j(k, p) - \frac{1}{2} * \frac{q_c \Delta t}{\Delta s} (w_j(k, p+1) - w_j(k, p-1)) + \frac{1}{2} * (\frac{q_c \Delta t}{\Delta s})^2 (w_j(k, p+1) - 2 * w_j(k, p) + w_j(k, p-1)) + \Delta t * (\sigma_{I_j^0 \rightarrow D_j}(k, p) - \sigma_{D_j \rightarrow I_j^0}(k, p) + \sigma_{C_j \rightarrow D_j}(k, p) - \sigma_{D_j \rightarrow C_j}(k, p)) \quad (25)$$

$$0 \leq u_j(k, p) - \Delta t * (\sigma_{C_j \rightarrow I_j^0}(k, p) + \sigma_{C_j \rightarrow D_j}(k, p)) \quad (26)$$

$$0 \leq w_j(k, p) - \Delta t * (\sigma_{D_j \rightarrow I_j^0}(k, p) + \sigma_{D_j \rightarrow C_j}(k, p)) \quad (27)$$

$$0 \leq v_j^0(k, p) - \Delta t * [(\sum_{i \in [1, N_B]} \sigma_{I_j^0 \rightarrow I_i^0}(k, p)) + \sigma_{I_j^0 \rightarrow I_j^1}(k, p) + \sigma_{I_j^0 \rightarrow C_j}(k, p) + \sigma_{I_j^0 \rightarrow D_j}(k, p)] \quad (28)$$

$$\Delta t * [(\sum_{i \in [1, N_B]} \sigma_{I_j^1 \rightarrow I_i^1}(k, p)) + \sigma_{I_j^1 \rightarrow I_j^0}(k, p) - v_j^1(k, p)] \leq 0 \quad (29)$$

$$u_j(k, N_s) = \Delta t * (\sigma_{C_i \rightarrow I_i^0}(k, N_s) + \sigma_{C_i \rightarrow D_i}(k, N_s)) \quad (30)$$

$$\forall i \in [0, N_B], \forall k \leq p_{min} w_j(k, p) = \Delta t * \sigma_{D_i \rightarrow C_i}(k, p) \quad (31)$$

$$\forall i \in [0, N_B], \forall p \leq p_{min} v_j^0(k, p) = \Delta t * \sigma_{I_i^0 \rightarrow C_i}(k, p) \quad (32)$$

$$\forall i \in [0, N_B], \forall p \leq p_{min} v_j^1(k, p) = \Delta t * \sigma_{I_i^1 -> C_i}(k, p) \quad (33)$$

$$\forall i \in [0, N_B], \forall p \leq p_{min} + \Delta p_{ij}, \sigma_{I_j^0 -> I_i^0}(k, p) = \sigma_{I_j^1 -> I_i^1}(k, p) = 0 \quad (34)$$

$$\sigma \geq 0 \quad \forall [\sigma] \quad (35)$$

$$\sigma_{I_j^0 -> I_j^1}(k, p) = \sum_{i \in [0, N_B]} \sigma_{I_j^1 -> I_i^1}(k, p) \quad (36)$$

$$\forall i \in [0, N_B], \sum_{p \in [0, N_s - 1]} \sigma_{I_j^1 -> I_i^1}(k, p) * \Delta s \leq D(k, j, i) \quad (37)$$

$$\sum_{p \in [0, N_s - 1]} \sum_{i \in [1, N_B]} (u_i(k, p) + v_i^0(k, p) + v_i^1(k, p) + w_i(k, p)) * \Delta s = N_{fleet} \quad (38)$$

Simplification

We had difficulty to make the entire program computationally feasible at the scale we initially wanted to handle. As a result, we limited our analysis to the first 36 time steps of our discretization scheme (6 hours from 3am to 9am) and to 4 geographical areas that correspond to the inner San Francisco area (see Figure 4). We also reduce the number of cars to 250 to match the demand at this time and in the studied geography. Although this value does not seem realistic for San Francisco and this time frame, this reduction is due to the Mobility data demand. The data set is a simulation of trips across a larger area and were not weighted by the population density of the local areas. As a result, the importance of San Francisco is undermined. We also reduce the time step and SOC step to reduce the complexity of the algorithm while still satisfying the condition for the Lax-Wendroff scheme to remain stable. While this is not completely satisfying, this could be improved by using external servers or parallel computing as the model itself does not lose its generality.

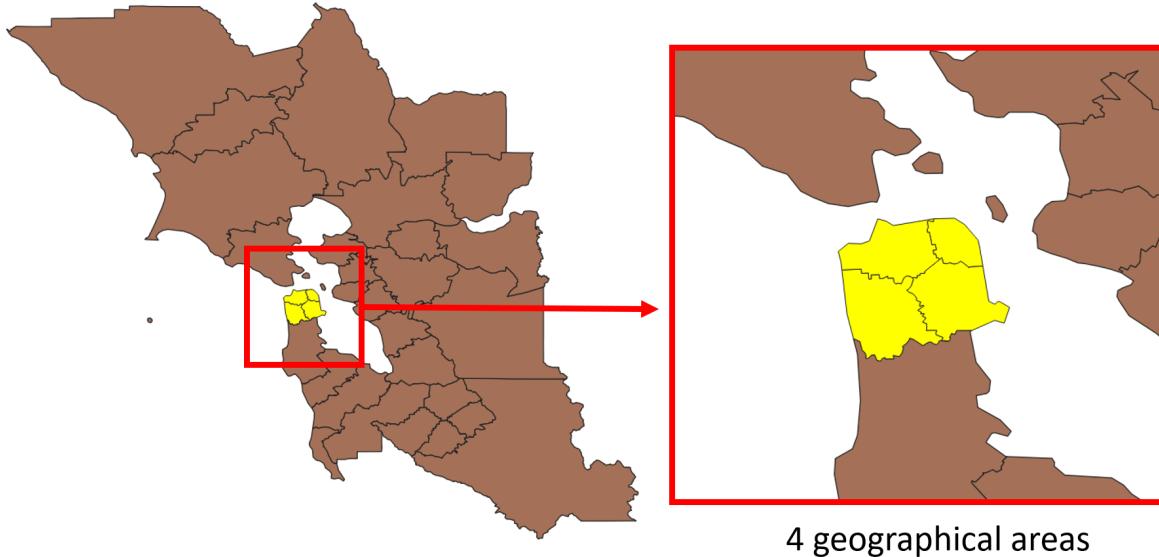


Figure 4: Limitation of our analysis to 4 geographical areas: inner San Francisco.

6 Results & Discussion

The results of the optimization program are the optimal flows of vehicles outlined above, which dictate the optimal distributions of vehicles in their respective populations - charging, discharging, idle without passenger, idle with passenger. This set of variables therefore define an optimal charge, discharge, and mobility satisfaction schedule subject to the constraints imposed. We find that satisfying mobility dominates the ability to earn revenue (negative cost). We observe a realistic scenario in which cars charge when demand is low, with minimal discharge, and satisfy demand when available.

The optimization program and input data were implemented into the CVXPY Python 3 package. Resulting outputs include the charging, discharging, and idle populations and all of the flows between populations shown

in Figure 3. The time-dependent distributions of the four populations for each spatial bin are shown in Figures 5 and 6. Initially, all cars are subject to the initial distribution placement into idle without passenger. Immediately, the majority of cars begin moving to a state of charging while a few move to a discharging state. Other vehicles are transferring between nodes and are not captured by the distribution of static populations. Figure 6 adds on to the details of Figure 5 through the addition of flows between the populations for the first time step.

To evaluate the behavior of trips being satisfied by the fleet, Figure 8 displays a sample evolution of the distribution of cars carrying passengers between bins for the simulated period. It shows a steep increase in the flow rate at time-step 24 (7AM). This corresponds to the time when demand for rides begins to increase and therefore suggests that the program optimizes for satisfying demand, which is expected to have a higher associated revenue than discharging. It is important to acknowledge that *each* population is a distribution of EVs at different states of charge. Figure 7 demonstrates an example charging distribution for the given spatial bin 3 and time step 3. The distribution of charging cars at this "early" time step follows expectations for the SOC level. Since the vehicles were placed normally distributed around 60% SOC, it follows that the charging population also sports a normally distributed shape. We view the interplay between charging, discharging, idling without a passenger, and idling with a passenger by plotting the evolution of these populations over time for a given spatial bin in Figure 5. All cars initially start as idle without passenger according to our initialization constraints. Initially cars only move to a state of charging. This charging population increases as electricity prices drop. However, as demand increases, the program shifts cars from charging to idle without passenger in preparation for satisfying demand, and then shift from idle without passenger to idle with passenger as demand is satisfied. Note that the transition towards supplying demand begins while electricity prices are still low, which implies that satisfying demand outweighs charging at low prices. By the time the average electricity price increases drastically, there are no cars charging.

To further illustrate the difference in magnitudes between costs associated with charge/discharge (V2G revenue) and satisfying mobility demand (mobility revenue), Figure 10 presents the cumulative revenue over time. An important note here is that the simulated time frame of 3AM to 9AM does not include the peak electricity prices usually seen in the afternoon, which would likely be the best time for a vehicle to earn revenue via discharge. For the given time frame we see a total earned revenue of \$4000. The interesting dynamic here is the constant steady loss in revenue from charging but the considerably larger gain in revenue once demand begins to be satisfied.

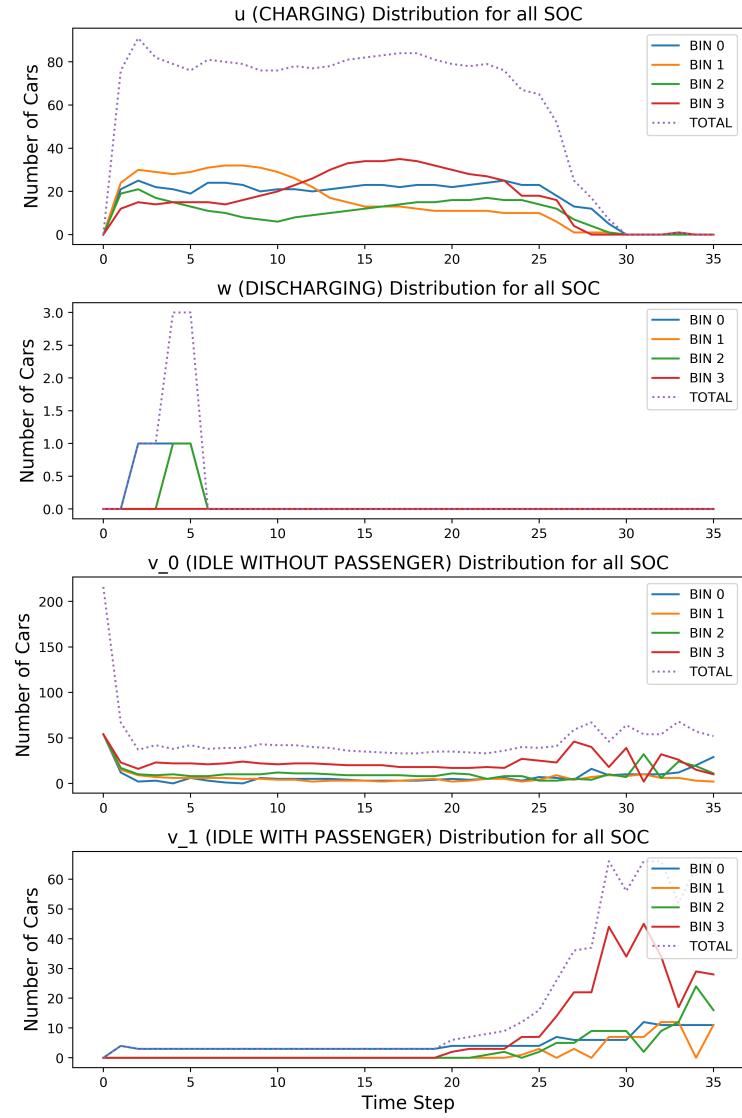


Figure 5: Vehicle distributions (idles, charge, and discharge) over time for each respective spatial node ("bin"). Note that these distributions only show the static populations and do not include the flows between nodes.

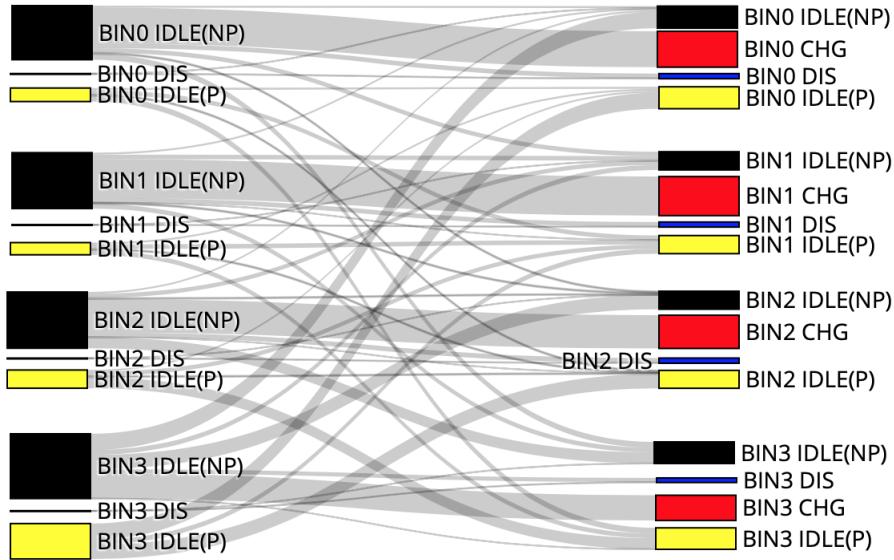


Figure 6: Sankey diagram of vehicle flows for first timestep. Note how idle populations move between bins, but discharge populations remain in their respective node.

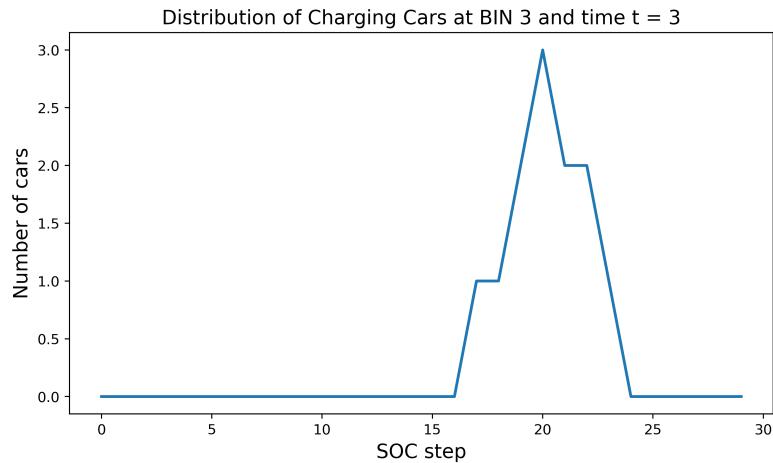


Figure 7: Example distribution of charging cars for a given node and time step. This distribution shows the number of cars at each SOC level.

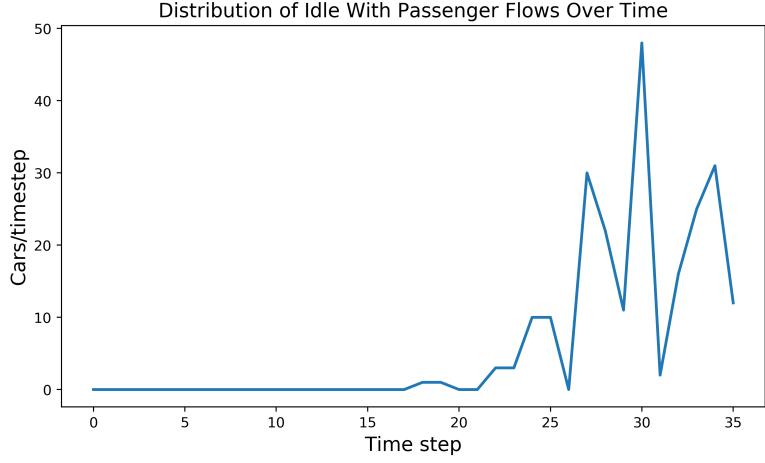


Figure 8: Idle with passenger flows over time. Beyond time step 20, more demand is simulated. The increase in idle with passenger flows shows trips being satisfied.

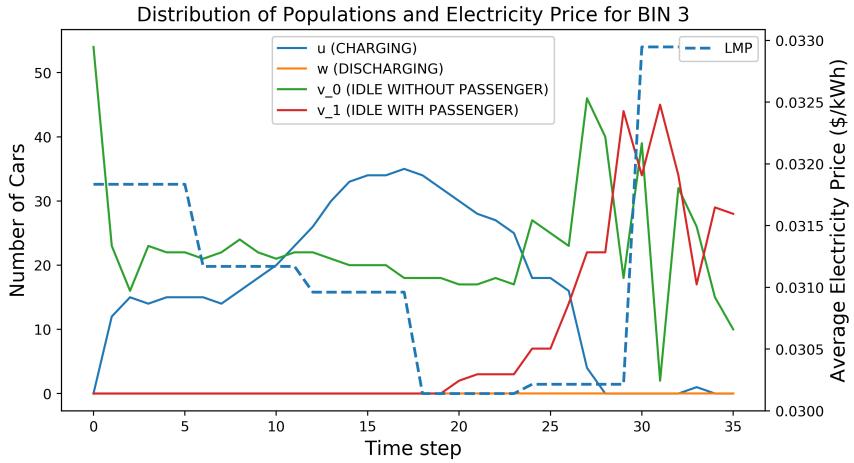


Figure 9: Evolution of charge, discharge, and idle populations over time for a given node. The average LMP for the node is also shown (dashed). Note how the number of cars charging decreases when demand is satisfied and when the electricity price rises.

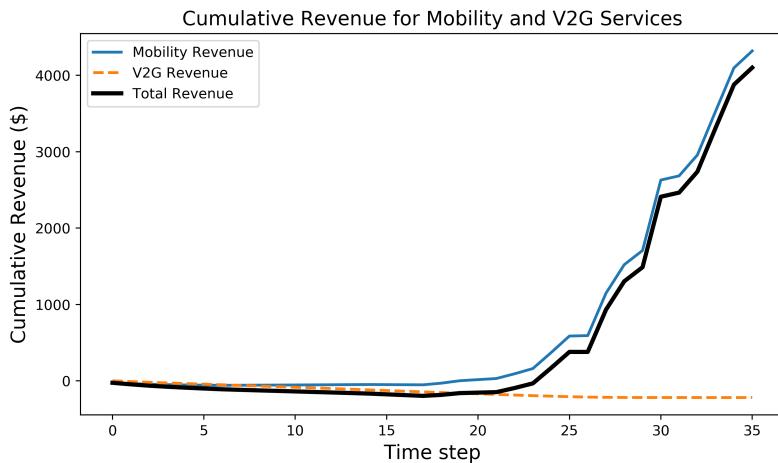


Figure 10: Cumulative revenue from mobility services and charge/discharge costs/revenues.

Future Work

We believe that the most important next step to take would be to test the optimization program for the unsimplified set of data. This would allow the model to simulate optimal PEV scheduling for many more trip distances, SOC requirements, and electricity prices. An in-depth simulation would also better represent the requirements for realistic implementation of the model by a fleet aggregator.

Future improvement of the model could integrate infrastructure systems as well as environmental costs. We also did not incorporate the distance or time required for a vehicle to reach a charging station. We also did not incorporate variable charging speeds which would likely vary with the charger type in a real system. We also did not include the value of any ancillary grid services that an aggregated fleet of batteries could provide. A sizeable PEV fleet could provide ancillary services such as frequency regulation (and, theoretically, voltage regulation). These subtle complexities would add realism and yield interesting conclusions on aggregated fleet behavior and economic potential. This connection with the surrounding infrastructure could significantly alter the optimal scheduling pathways while still maximizing aggregator revenue. Another valuable improvement to the model would integrate greenhouse gas emissions as a cost to minimize by itself. Electricity emissions per unit of electricity vary throughout the day. By setting a price on carbon, optimal scheduling would value electricity and emissions costs accordingly. This scheme could be part of an aggregator's sustainability mission as well as an effective climate change mitigation strategy in transportation. Using a price on carbon could create signals that would dictate a more optimal schedule in terms of flattening the duck curve and enabling renewable penetration.

7 Summary

The focus of this project was to develop a program that optimally schedules an aggregated mobility service fleet of autonomous PEVs based on realistic mobility data and real-time electricity prices in the San Francisco Bay Area. The model was formulated into a convex optimization program and discretized using the Lax-Wendroff method, subject to a stability condition, which was solved with CVXPY to calculate the optimal charging, discharging, and mobility service scheduling for a given period. Implementation of the model yielded results which demonstrated satisfied mobility demand and minimized cumulative cost of electricity through strategic charging. This project outlines a process that would be similarly used by an autonomous fleet aggregator in the future to optimally satisfy mobility demand. The application of optimization to an PEV fleet values both transportation sustainability and state-of-the-art autonomous vehicle controls technology.

References

- [1] Yao L. et al. “Optimal Charging and Discharging Scheduling for Electric Vehicles in a Parking Station with Photovoltaic System and Energy Storage System”. In: *Energies* 10.4 (2017).
- [2] Chen T. et al. “Operations of a shared, autonomous, electric vehicle fleet: Implications of vehicle charging infrastructure decisions”. In: *Transportation Research* 94 (2016).
- [3] Lombardi P. et al. “Plug-in electric vehicles as storage devices within an Autonomous power system. Optimization issue”. In: *In IEEE Bucharest PowerTech* (2009).
- [4] Vandael S. et al. “Decentralized demand side management of plug-in hybrid vehicles in a smart grid”. In: *Proc. 1st Int. Workshop Agent Technol. Energy Syst.* (2010).
- [5] Alonso M. et al. “Optimal Charging Scheduling of Electric Vehicles in Smart Grids by Heuristic Algorithms”. In: *Energies* 7.4 (2014).
- [6] Yi Z. et al. “Data-driven optimal charging decision making for connected and autonomous electric vehicles: A personal usage scenario”. In: *Transportation Research* 86.0 (2018).
- [7] Dong J. et al. “Stochastic Modeling of Battery Electric Vehicle Driver Behavior: Impact of Charging Infrastructure Deployment on the Feasibility of Battery Electric Vehicles”. In: *Transportation Research Record* 2454.1 (2014).
- [8] Ahn C. et al. “Optimal decentralized charging control algorithm for electrified vehicles connected to smart grid”. In: *Journal of Power Sources* 196.23 (2011).
- [9] Iversen E. et al. “Optimal charging of an electric vehicle using a Markov decision process”. In: *Applied Energy* 123 (2014).
- [10] Nimalsiri N. et al. “A Survey of Algorithms for Distributed Charging Control of Electric Vehicles in Smart Grid”. In: *IEEE Transactions on Intelligent Transportation Systems* (2018).
- [11] Uber. *Uber price estimator*. 2019.

Traffic Dynamics Estimation Based on the Intelligent Driver Model

Alben Rome Bagabaldo, Joy Carpio, Bingnan Li, Can Cao, Xinyu Hu

1 Abstract

Traffic congestion is a major problem across the world. There have been many initiatives to address this, and one of the solutions that is gaining attention in the transportation and control literature is the use of autonomous vehicles to improve traffic flow. Given that, this paper focuses on determining traffic dynamics with only one car's data input (velocity, displacement, and gap) which presumably is the autonomous car in a ring traffic. In doing such, we first used the classical intelligent driver model (IDM) and estimated the values of the parameters using the data from the Arizona Ring Experiment A by Stern, et al. conducted in 2016. The values of the parameters were determined using the autoregressive with exogenous input terms (ARX) model. Afterwards, state estimation using the Extended Kalman Filter was applied to show that predicting the movement of the successive vehicles is possible using just the data of one vehicle as the input and the fitted values to the IDM.

2 Introduction

2.1 Motivation and Background

According to EPA's 2017 annual report of primary sources of greenhouse gas (GHG) emissions, the transportation sector produces approximately 29% of GHG emissions, which is the largest contributor among all sources [1]. Furthermore, traffic congestion is responsible for a large amount of these emissions. The researchers at the Texas transportation institute published estimates on the effects of traffic congestion: In 2014, there were 3.1 billion gallons of fuel wasted due to traffic congestion. In addition, total wasted fuel as a result of congestion has been increasing since 1982 (except for 2008 due to the Great Recession) and is predicted to continuously increase in the future [2]. The increase in wasted fuel not only increases GHG and air pollutants emissions, but brings economic losses as well. Immediate reductions to these emissions are required to combat these issues primarily by addressing traffic congestion.

Many attempts were tried to provide solutions on congestion issue such as the implementation of policies for strict traffic regulations, building more road networks or even reduction in vehicle production. But due to the increase in demand for different transportation

means, this seems to be an unsolvable problem. Thus, understanding traffic dynamics is seen to be essential in smoothening traffic flow and mitigating congestion issues since it helps it is needed in redesigning infrastructures, and in guiding and controlling transportation.

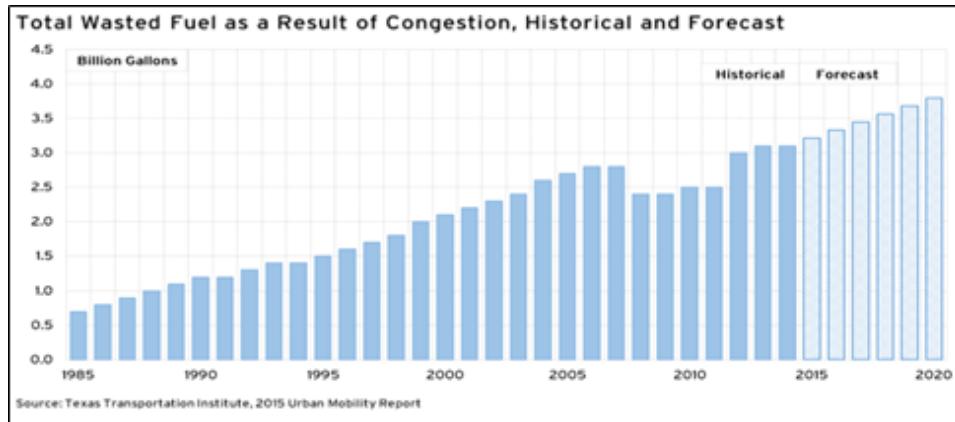


Figure 1. Total Wasted Fuel as a Result of Congestion, Historical and Forecast

One of the interesting studies to have better understanding of traffic dynamics is mitigating the occurrence of the so-called “stop-and-go waves” in traffic. In a recent study, these waves also form in a closed loop network called the ring experiment. It was found out that despite having a constant number of human-driven vehicles and constant circumference to be tracked, congestion still occurs, which they referred to as “phantom congestion” [3]. This gives an idea that human driving behaviors is among the important causes of congestion.

Sugiyama and Stern et al. found out that adding autonomous vehicles (even at small percentage) in mixed traffic will help dissipate these stop-and-go waves [4]. To better understand and control the system with autonomous vehicles, it's helpful to know the states of each vehicle in the traffic system. However, understanding those details can be complex and costly, with all the additional sensors and infrastructures required. Thus, we were encouraged to monitor the ring traffic dynamics with only one car's data input including velocity, displacement and gap.

2.2 Relevant Literature

In this section, we provided relevant literature of the project in the following topics: stop-and-go wave, car-following behavior, and traffic congestion experiment.

Stop-and-go wave: As the density of the vehicles increases, the dynamics of traffic flow is affected and becomes unstable, and small perturbations can amplify and grow into the so-called “stop-and-go waves” that propagates upstream against the flow of traffic and travel backward along the road. This phenomenon called “phantom traffic jams” or phantom congestion is experimentally reproducible and has actually been reproduced in different experiments[4]. The stop-and-go waves are composed of a sequence of different moving traffic

jams, where as a moving jam is spatially restricted by two sharp interfaces, the downstream jam front and the upstream jam front [5]. Within the downstream jam front, vehicles must accelerate and within the upstream jam front, vehicles must slow down [5]. Vehicle speed, flow rate and density varies sharply within different jam fronts. The book “The Physics of Traffic” by Kerner introduced a comprehensive review of reproducible empirical spatiotemporal congested pattern features and their engineering applications.

Car-following Behavior: A car-following model shows the traffic dynamics and this can represent the occurrence of stop and go mathematically. The model bridges environmental variables, like subject vehicle speeds, relative distances and relative speeds to a leading vehicle, and variables that are controllable, like acceleration or deceleration. Conventional approaches for representing car-following behaviors mainly include linear and non-linear models. In a previous paper [6], the motion of vehicles in a line obeying the following ‘traffic law’:

$$\frac{du_k(t)}{dt} = \lambda(u_{k-1}(t) - u_k(t)), (k = 1, 2, 3, , n) \quad (1)$$

In the differential equation, $u_k(t)$ and $u_{k-1}(t)$ are the velocity of the k th and $(k-1)$ th vehicle respectively. λ is a positive proportional constant. According to this equation, the acceleration of the k th vehicle is proportional to the difference of the velocities of the vehicle ahead and of its own. Chandler, Herman, and Montroll [7] modify the equation which gives a more realistic description of the motion of a line of vehicles by introducing a time lag Δ into the equation:

$$\frac{du_k(t + \Delta)}{dt} = \lambda(u_{k-1}(t) - u_k(t)), (k = 1, 2, 3, , n) \quad (2)$$

Tse-Sun Chow provides a solution to the linear difference differential equation with time lag, and considers a special problem [8]. Based on the linear model, researchers establish nonlinear models. The basic idea is that each driver of a vehicle responds to a given stimulus according to a relation:

$$\text{response} = \text{sensitivity} \times \text{stimulus} \quad (3)$$

The stimulus could be a functional of the positions of a number of cars and their time derivatives, and perhaps also some other parameters.

$$x_{n+1}(t + T) = \lambda x_n(t) - x_{n+1}(t), (k = 1, 2, 3, , n) \quad (4)$$

Several papers use different functions of $x_n(t)$ which are not constant values. Denos, Robert and Richard [9] generalize the functions as:

$$\lambda = ax_{n+1}^m(t + T)/[x_n(t) - x_{n+1}(t)]^l \quad (5)$$

However it is argued in a paper [10] that most existing car-following models are simplified to be parsimonious, which restricts model flexibility and accuracy. Additionally, conventional models are artificially developed that do not fully consider the memory effect or the prediction capability of drivers and only consider the instantaneous interaction between the specific vehicle and the one ahead. Therefore, deep learning based car-following models are introduced to learn driver behaviors from empirical car-following data collected in practice.

M. Treiber etc. provide a comprehensive and instructive coverage of vehicular traffic flow dynamics and modeling in a book[11]. The intelligent driver model (IDM) is a time-continuous car-following model for the simulation of freeway and urban traffic. It describes the dynamics of the positions and velocities of single vehicles.

$$a_{\text{IDM}} = \frac{dv_\alpha}{dt} = a \left[1 - \left(\frac{v_\alpha}{v_0} \right)^\delta - \left(\frac{s^*(v_\alpha, \Delta v_\alpha)}{s_\alpha} \right)^2 \right] \quad (6)$$

$$s^*(v_\alpha, \Delta v_\alpha) = s_0 + \max \left(0, v_\alpha T + \frac{v_\alpha \Delta v_\alpha}{2\sqrt{ab}} \right) \quad (7)$$

s_0 represents the minimum gap, T represents the safe time headway, a represents the acceleration, b represents the comfortable deceleration, and v_0 represents the desired speed. We will use the IDM model in our project. It takes the safety issue into consideration by adding the safe time headway parameter to calculate the gap between each vehicle. We will use the adapted equations from Zheng, et al[12]. for parameter estimation:

$$F(s_i(t), \dot{s}_i(t), v_i(t)) = a \left(1 - \left(\frac{v}{v_{\max}} \right)^4 - \left(\frac{s_{\text{st}} + T_{\text{gap}} v - \frac{\dot{s}v}{\sqrt{4ab}}}{s} \right)^2 \right) \quad (8)$$

$$s^* = \frac{s_{\text{st}} + T_{\text{gap}} v^*}{\sqrt{1 - \left(\frac{v^*}{v_{\max}} \right)^4}} \quad (9)$$

Traffic congestion experiment: The stop-and-go wave can best be shown in the work of Sugiyama et al., where they used a circuit experiment to show that the emergence of a traffic jam can happen in the absence of bottleneck [13]. The experiment is designed as 22 vehicles run on a large circuit, and the traffic condition of this flow is measured and analyzed. Researchers suggest that a jam is generated spontaneously only if the average vehicle density exceeds the

critical value. They consider jam formation is an effect of the collective motion in the physics of a non-equilibrium phase transition of a many-particle system.

Inspired by Sugiyama experiment, Stern et al. try to dampen the stop-and-go waves by using intelligent control of an autonomous vehicle(AV) with a small penetration rate [4] in Arizona. They use experiment on a circular track with over 20 vehicles (only 1 AV in the group) to demonstrate the traffic flow. The flow velocity, braking events and fuel economy are used as metrics of the experiment. Three control strategies are applied in three experiments:

1)The FollowerStopper controller, which is command exactly the desired velocity whenever sage but to command a suitable lower velocity whenever safety requires;

2) Human driver controller, which follows the same principle in Follwerstopper by the well-trained human driver;

3) The PI with saturation controller, which is based on that the AV may estimate the average speed of the vehicles in front, and then drive according to the average speed.

First two controller requires external input for the desired velocity, while the third one is a fully automatic control. The experiments results show that all of them can help to control the traffic flow under the low actuators rate (less than 5%).

2.3 Focus of this Study

In this study, we achieved the goal of determining the locations and velocities of the vehicles in a ring network given the information from an autonomous vehicle alone. Wherein using the data from the Arizona Ring Traffic Experiment A, we fitted the parameters to the intelligent driver model (IDM). And, we adopted these parameters in the state estimation of traffic dynamics of the vehicles in the system except for the autonomous vehicle (Vehicle No. 20) where we used its velocity as our data input to predict the states of the succeeding vehicles, specifically their respective displacements, velocities and corresponding gap from the vehicle in front of it. In short, from the available data of Vehicle No. 20 alone, we estimated the states of Vehicle No. 1, then using the estimated states of Vehicle No. 1, we estimated the states of Vehicle No. 2, and the same sequence continued until Vehicle No. 19 assuming that every vehicle follows the IDM.

3 Technical Description

The study considered a circular (ring) road network, specifically the set-up used in the Arizona experiment. There were two sections used to accomplish the objective of our study which are:

- Use of the ARX model to estimate the IDM parameters, and

- State estimation of the movement of the human-driven vehicles given the input from an autonomous vehicle.

Just like in the original setting based from the Arizona Ring Experiment A, 19 human-driven vehicles are accounted for, and one autonomous vehicle was used as an input to track the position and velocity of the other vehicles over time. The remaining human-driven vehicles are to move based on its coded dynamics, which is the IDM model based on equation (8). Basically, the vehicle dynamics is described by the velocity of the vehicle under consideration, the velocity and space headway of the leading vehicle, and the time headway between the vehicles. This can be translated into the dynamical equations of the states and output described below.

$$X_1 = \text{displacement of vehicle } (x_a)$$

$$X_2 = \text{velocity of vehicle } (v_a)$$

$$X_3 = \text{gap between cars } (s)$$

$$\begin{aligned}\dot{x}_1 &= X_2 \\ \dot{x}_2 &= a_{\text{IDM}} = F(s_i(t), \dot{s}_i(t), v_i(t)) = a \left(1 - \left(\frac{v}{v_{\max}} \right)^4 - \left(\frac{s_{st} + T_{\text{gap}}v - \frac{\dot{s}v}{\sqrt{4ab}}}{s} \right)^2 \right) \\ \dot{x}_3 &= \dot{s} = v_{a-1} - X_2\end{aligned}$$

$$\dot{x} = \begin{bmatrix} v_a \\ a[1 - \left(\frac{v_a}{v_0} \right)^4 - \left(\frac{s_{st} + T_{\text{gap}} \cdot v_a - \frac{\dot{s}v_a}{\sqrt{4ab}}}{s} \right)^2] \\ \dot{s} \end{bmatrix} = \begin{bmatrix} v_a \\ a[1 - \left(\frac{v_a}{v_0} \right)^4 - \left(\frac{s_{st} + T_{\text{gap}} \cdot v_a - \frac{\dot{s}v_a}{\sqrt{4ab}}}{s} \right)^2] \\ v_L - v_a \end{bmatrix} \quad (10)$$

Where :

$v_L(t)$: the velocity of the leading vehicle;

$v_a(t)$: the velocity of the specific vehicle;

s : the gap with the leading vehicle;

a : the maximum acceleration;

b : the comfortable deceleration;

$v_0(t)$: the desired velocity;

s_{st} : the minimum gap;

T_{gap} : the headway time constant.

Then, the input value in the system is the velocity of the autonomous vehicle; and, after the dynamics of the Vehicle No. 1 was estimated, the successive vehicles just depended on the initial states of the vehicle directly in front of it. This is illustrated in the figure below where each vehicle is represented as “ G .”

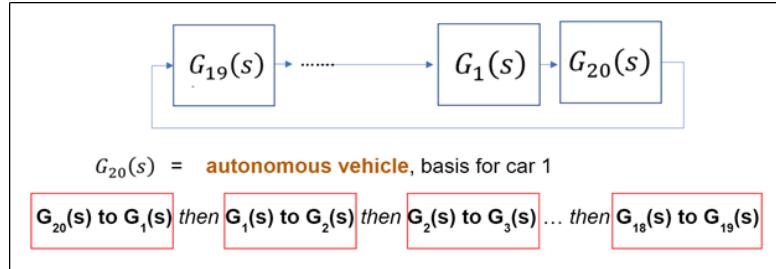


Figure 2. State Estimation Framework

For the output, since the gap (space headway) between the leading vehicle and succeeding vehicle can be measured using a sensor, we decided to set the gap as the output of the system. Thus, we have:

$$y(t) = s \quad (11)$$

Looking at the dynamics of the system, we can see a nonlinear expression, specifically the acceleration formula for IDM which is the second element of the states dynamics. On that note, we end up using the Extended Kalman Filter. Thus, the Jacobian equations are derived as shown below.

$$\frac{\partial f_1}{\partial X_1} = 0, \quad \frac{\partial f_1}{\partial X_2} = 1, \quad \frac{\partial f_1}{\partial X_3} = 0 \quad (12)$$

$$\frac{\partial f_2}{\partial X_1} = 0 \quad (13)$$

$$\frac{\partial f_2}{\partial X_2} = a \left[-4 \frac{(v_a)^3}{(v_a)^4} - 2 \left(\frac{s_{st} + T_{gap} v_a - \frac{\dot{s}v_a}{\sqrt{4ab}}}{s} \right) \left(\frac{T_{gap} - \frac{\dot{s}v_a}{\sqrt{4ab}}}{s} \right) \right] \quad (14)$$

$$\frac{\partial f_3}{\partial X_1} = 0, \quad \frac{\partial f_3}{\partial X_2} = -1, \quad \frac{\partial f_3}{\partial X_3} = 0$$

$$\frac{\partial f_2}{\partial X_3} = \frac{2a \left(s_{st} + T_{gap} v_a - \frac{\dot{s}v_a}{\sqrt{4ab}} \right)^2}{s^3} \quad (16)$$

$$F = \begin{bmatrix} 0 & 1 & 0 \\ 0 & a \left[-4 \frac{(v_a)^3}{(v_a)^4} - 2 \left(\frac{s_{st} + T_{gap} v_a - \frac{\dot{s}v_a}{\sqrt{4ab}}}{s} \right) \left(\frac{T_{gap} - \frac{\dot{s}v_a}{\sqrt{4ab}}}{s} \right) \right] & \frac{2a \left(s_{st} + T_{gap} v_a - \frac{\dot{s}v_a}{\sqrt{4ab}} \right)^2}{s^3} \\ 0 & -1 & 0 \end{bmatrix} \quad (17)$$

Since the output equation is linear with respect to the states, then nothing was changed in this equation. Therefore, we have:

$$y(t) = gap = [0 \ 0 \ 1]x(t) \quad (18)$$

$$H = [0 \ 0 \ 1] \quad (19)$$

For simplicity of our solution, we pre-assigned the values of the additional disturbance (W), output noise (N) and covariance matrix $Sigma$. Since we are uncertain how to penalize each state and the output, we decided to use low element values along the diagonal matrix W . The assigned values are as follows.

$$W = \begin{bmatrix} 0.001 & 0 & 0 \\ 0 & 0.001 & 0 \\ 0 & 0 & 0.001 \end{bmatrix} \quad (20)$$

$$N = 0.2 \quad (21)$$

$$Sigma = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (22)$$

Meanwhile, for the parameters used in the model, specifically \mathbf{a} , \mathbf{b} , v_0 , s_{st} and T_{gap} are estimated using the first half of the Arizona Experiment A dataset as training data from which we obtained the values of each parameters using the linearized equation of the acceleration adopted from the work of Zheng, et al., 2018, [12] as shown below.

$$\dot{\tilde{v}}_i = \alpha_1 \tilde{s}_i(t) - \alpha_2 \tilde{v}_i(t) + \alpha_3 \tilde{v}_{i-1}(t) \quad (23)$$

where:

$$\alpha_1 = 2a \cdot \frac{(s_{st} + T_{gap}v^*)^2}{(s^*)^3} \quad (24)$$

$$\alpha_2 = \sqrt{\frac{a}{b}} \cdot \frac{v^*(s_{st} + T_{gap}v^*)}{(s^*)^2} + 2a \left(\frac{2(v^*)^3}{v_{max}^4} + \frac{T_{gap}(s_{st} + T_{gap}v^*)}{(s^*)^2} \right) \quad (25)$$

$$\alpha_3 = \sqrt{\frac{a}{b}} \cdot \frac{v^*(s_{st} + T_{gap}v^*)}{(s^*)^2} \quad (26)$$

The parameter estimation method used in this study is the ARX model with least squares algorithm in which the theta, phi and Y used are as follows:

$$\theta = \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \end{bmatrix} \quad (27)$$

$$\phi = [s_a \quad v_a \quad v_L] \quad (28)$$

$$Y = \text{acceleration of vehicle } a \quad (29)$$

where the objective function is to minimize two-norm square of the difference between the product of phi and theta and Y, which is expressed mathematically in expression given below.

$$\min \|\phi\theta - Y\|_2^2 \quad (30)$$

Then, the second half of the data was used to test the model using the parameters obtained in the previous part. Finally, to obtain a measure of accuracy of the estimation, we used the mean absolute error as the criteria for the correctness of our predicted states.

4 Data

The data in the Arizona Ring Experiment A was used in our project. The said experiment was conducted by Stern et al. in July 2016 in Tucson, Arizona. In the setup, a 260m circumference of a closed ring (circular) road was used where there were 20 vehicles, 19 in which are human-driven and 1 is what they referred to as the Cognitive and Autonomous Test (CAT) Vehicle. In this paper, we simply called the CAT vehicle as the autonomous vehicle. The experiment A is the experiment with Follower stopper control strategy, as mentioned in the literature review part. It is also where the human drivers were an instruction to “safely follow the vehicle in front as if in rush hour traffic” making IDM applicable to the scenario because of its inherent safety feature within the equation. The data in the said experiment were collected by 360-degree cameras and then processed via computer vision techniques. [14]

5 Results and Discussion

5.1 Auto-Regressive with eXogenous input (ARX) Model Results

Using ARX, we estimated the values of alpha 1, 2 and 3 based on equation (23) then substituted to equations (24), (25), and (26) together with original IDM equation to determine

the corresponding values of the parameters. Fifty (50) percent of our dataset was used as the training dataset. And in the table below are our calculated values.

Table 1. Calculated Values of Alphas using ARX

Alpha	Calculated value
α_1	0.0004
α_2	-0.3847
α_3	0.3940

Then, using the previously mentioned equations, we calculated the values of the parameters for the IDM. In the table below are the values we found suitable in this scenario based on the Arizona Ring Experiment A dataset.

Table 2. Estimated Values of IDM Parameters

Parameter notation	Parameter name	Estimated values	Units
s_0	Minimum gap	5.4	meters (m)
T	Headway	1.0378	seconds (s)
a	Acceleration	0.6808	meters/sec ² (m/s ²)
b	Comfortable deceleration	1.211	meters/sec ² (m/s ²)
v_0	Desired speed	5.733	meters/sec (m/s)

The values above results to a mean absolute error (MAE) of 2.3342 m/s² when the predicted values of acceleration is calculated and compared to the actual recorded acceleration in the test dataset.

After fitting the parameters, we ran the simulation using the existing libraries in Flow [15] and see how the vehicles will move just based on these estimated parameters and using IDM alone. Screenshots of the simulation are shown below:

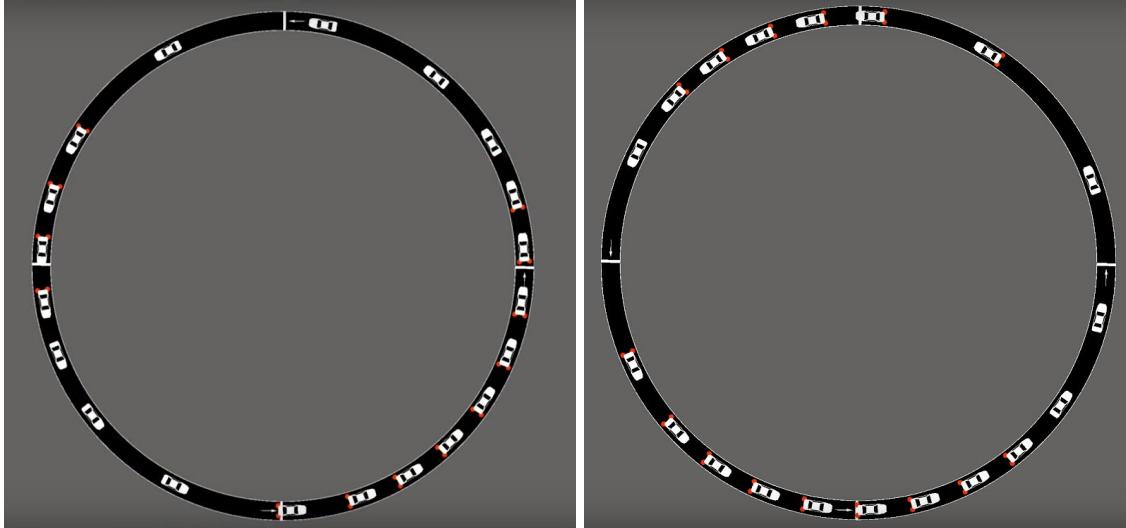


Figure 3. Screenshots taken from the simulation in SUMO using existing Flow libraries to simulate varying estimates of IDM parameters

The screenshots of the simulation shown above illustrate the occurrences of stop-and-go waves being formed after running it in the traffic microsimulator Simulation of Urban Mobility (SUMO) as generated by the existing library of codes in Flow. This result is contrary to the description of the said experiment where supposedly, only small traffic waves are expected. This simulation gives us a general sense of the resulting MAE of 2.3342 m/s^2 .

5.2 State Estimation with EKF Results

In the next part of our study, we adopted the results from the parameter fitting to the IDM using ARX and used state estimation with Extended Kalman Filter to predict the movement of all the vehicles in the ring road by only knowing the information of a single vehicle. The results below show the comparison of the calculated values of the states which are the displacement and velocity of each vehicle plotted against the actual values from the Arizona Ring Experiment A. Recall that for the state estimation, the idea is we only have the information from the first vehicle (the autonomous car which is Vehicle 20). The trajectories (displacement and velocity) of Vehicle 1 is estimated from this, and the trajectories of the following vehicles are determined using the estimated values of the vehicle in front of it (i.e. trajectories of Vehicle 2 is estimated from Vehicle 1, Vehicle 3 from Vehicle 2, and so on until Vehicle 19 from Vehicle 18).

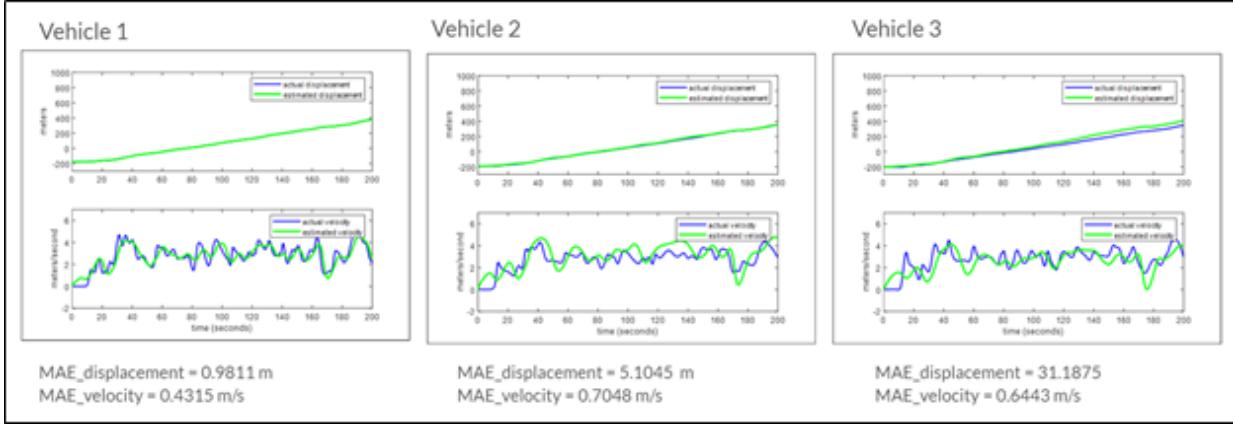


Figure 4. Comparison plots of the actual (in blue) and estimated (in green) displacement (top subplot for each vehicle) and velocity (bottom subplot for each vehicle) with the corresponding MAEs for Vehicles 1-3

From the figure above, it can be noticed that the estimated displacement of Vehicle 1 in the ring is almost the same as in the original data since the two plots (actual vs. estimated) overlaps; this is almost the same in terms of the estimated velocity, but the discrepancy is more noticeable in the plot.

In a general note, it shows that as we move from vehicle 1 to 3, we can see an increasing trend in the MAE values both for displacement and velocity. This trend continues almost all throughout the entire system until Vehicle 19 as shown in the figure below.

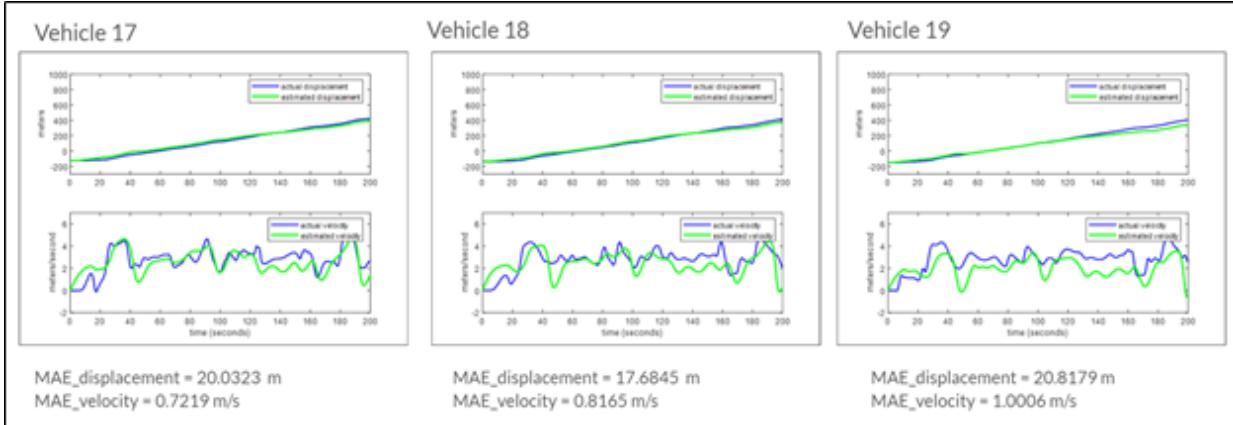


Figure 5. Comparison plots of the actual (in blue) and estimated (in green) displacement (top subplot for each vehicle) and velocity (bottom subplot for each vehicle) with the corresponding MAEs for Vehicles 17-19.

As seen above, the increasing trend also applies in the continuing number of vehicles, in this case is from vehicle no. 17 until vehicle 19. But what is not seen so far is that the increasing trend in MAE is disrupted after vehicle no. 8, as shown in the plot below.

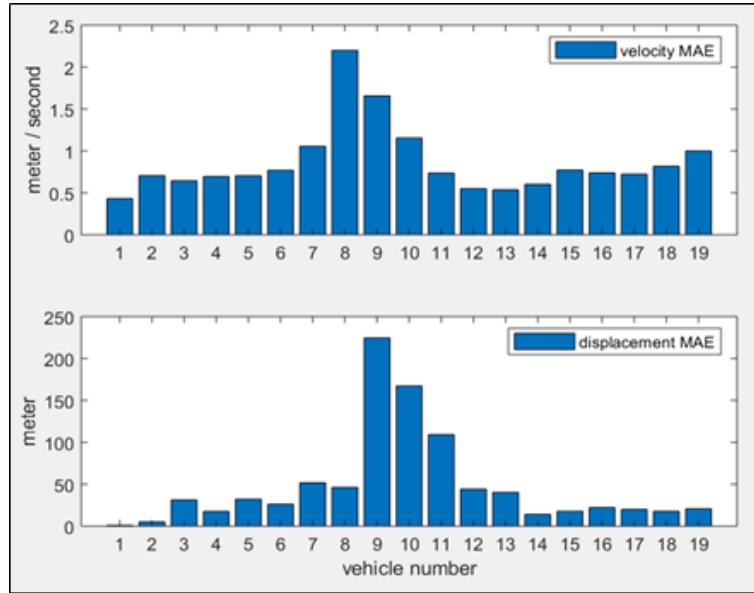


Figure 6. Plots of the Mean Absolute Error (MAE) in terms of velocity (*top*) and displacement (*bottom*) of each vehicle vs. the corresponding vehicle

Noticed in the plot above was a sudden increase in MAE in the velocity of vehicle no. 8 which then affected the approximated displacement of vehicle no. 9 which can also be explained by the state space dynamics. Then, the response of the following cars in terms of the estimated values are turning to be better than that of vehicle no. 8. The big leap in the MAE seems to be being absorbed by the succeeding vehicles, thus, creating the decreasing trend in the MAE.

In addition, we also showed the actual and estimated displacements of all the vehicles in a comparative plot shown below.

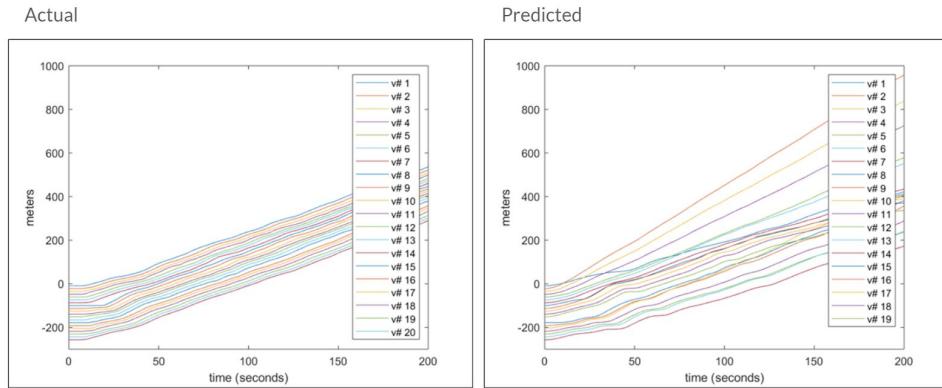


Figure 7. Comparative plots of the actual (*left*) and predicted (*right*) displacement of the vehicles with respect to time

From the summarized actual and predicted displacements of all the vehicles, the first seven vehicles follow the same trajectory similar to that of the autonomous vehicle (not shown in the plots above, but a small MAE in the estimated trajectory of vehicle 1 explains this), but because of the big MAE in the estimated velocity of vehicle 8 as previously illustrated, this original trajectory of the first vehicles is not followed by the remaining vehicles because they now follows the displacement of vehicle 9 as calculated from the velocity of vehicle 8.

Interestingly, because the results shown above illustrate variations in the MAE among the vehicles, we also tried changing the assigned values of the states disturbances (matrix W) to see if these values affect the results. We then assigned the following values to matrix W .

$$\begin{aligned} W = [& 0.1, 0, 0; \\ & 0, 0.1, 0; \\ & 0, 0, 0.1] \end{aligned} \quad (31)$$

Note that here, we simply increased the diagonal values from 0.01 to 0.1. This results in the following values of the MAE vs. the vehicles in the system.

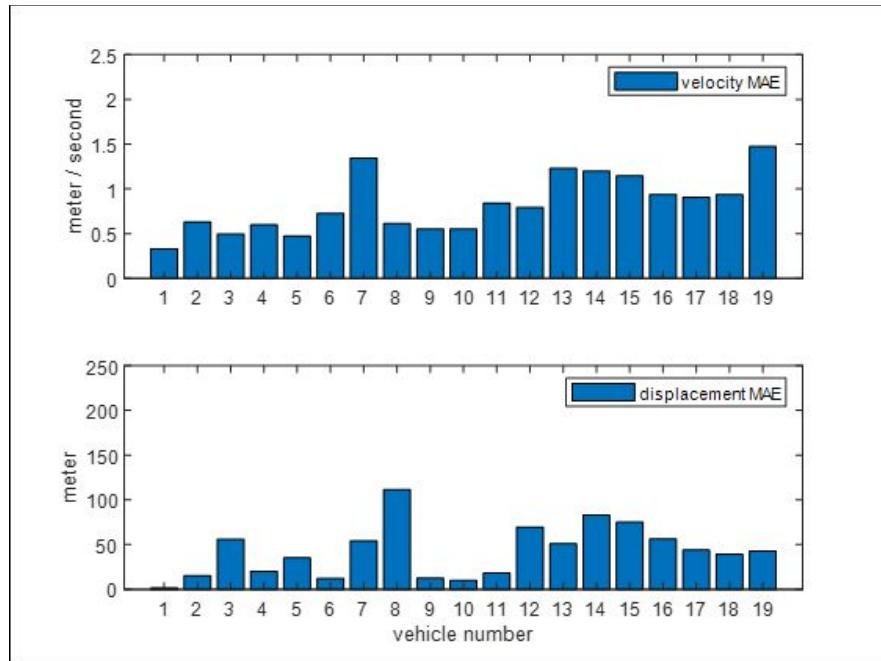


Figure 8. Plots of the Mean Absolute Error (MAE) in terms of velocity (*top*) and displacement (*bottom*) of each vehicle vs. the corresponding vehicle using a different set of values of matrix W

The figure above shows that there has been a significant change in the trend in the values of the MAE for both displacement and position. This noticeable when compared to the previous result as shown in Figure 6. And, this change is associated to simply changing the values assigned in our state disturbances (matrix W).

6 Summary

Since our project used ARX based on the least-squares algorithm to estimate the values of the parameters in the IDM, which is a black box, the estimated parameters introduce bias and overfitting. This resulted to some amount of errors in the state estimation process. This also creates a notion that there would be limited scenarios where the model developed could be applicable. In other words, we need specific training data to determine the values of the parameters for specific scenarios. We also find that the values of noises we assign will influence the MAE of state estimation. The MAE of the 8th vehicle decreases when we used larger values of noises. Additionally, there is an increasing trend for the error as the number of vehicles increases. So, with only one car's velocity, displacement, and gap data input, the scale of traffic that we can simulate will be limited by the desired accuracy.

Hence, our study can be used as the basis of the control of autonomous vehicles, which can help dampen the stop-and-go waves as mentioned before. The percentage of autonomous vehicles in a car platoon can also be decided based on the error occurrence condition in our project. To make the model more solid for the real world, following works can be done: 1) find a way to generalize IDM model parameters with more data to make it fit better in various environments/conditions; 2) try other car following models and conduct stability tests; 3) extend the scale of traffic dynamics simulation by increasing the input from more vehicles' data; 4) improve the accuracy of traffic dynamics simulation by having a periodical input (i.e. having inputs in the same interval) to correct the model continuously.

7 Acknowledgment

The group is thankful for the guidance and mentorship of Prof. Scott Moura throughout the CE295 class and this project. The team also wishes to thank Prof. Alexandre Bayen and members of the Mobile Sensing Lab at UC Berkeley for providing data and framework. Joy and Alben are also thankful to the Commission on Higher Education - Phillipine California Advanced Research Institutes (CHED-PCARI) and the Department of Science and Technology - Science Education Institute (DOST-SEI) for funding their studies at UC Berkeley. Alben is also grateful to Mapua University for the additional support for his studies.

8 References

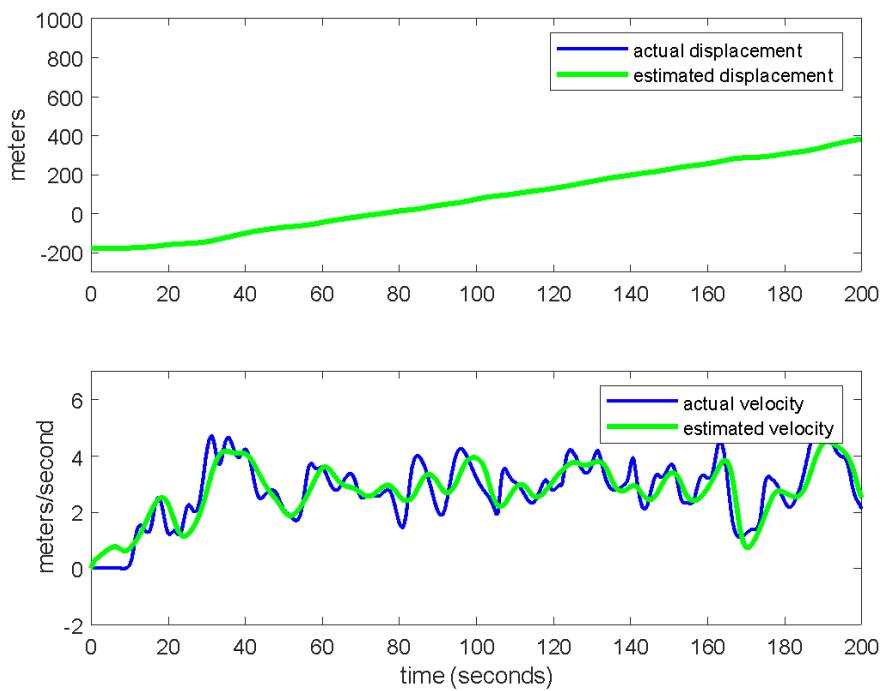
- [1] “Inventory of U.S. Greenhouse Gas Emissions and Sinks”, EPA, 2017.
<https://www.epa.gov/ghgemissions/inventor-us-greenhouse-gas-emissions-and-sinks>
- [2] “Fact #897 November 2, 2015 Fuel Wasted in Traffic Congestion”, Office of Energy Efficiency & Renewable Energy, 02 Nov. 2015.
<https://www.energy.gov/eere/vehicles/fact-897-november-2-2015-fuel-wasted-traffic-congestion>
- [3] A.R. Kreidieh et al., “Dissipation of stop and go waves in closed and open loop networks via deep reinforcement learning”, 21st International Conference on Intelligent Transportation (ITSC), Nov. 2018.
- [4] R. E. Stern et al., “Dissipation of stop-and-go waves via control of autonomous vehicles: Field experiments,” *Transportation Research Part C: Emerging Technologies*, vol. 89, pp. 205–221, Apr. 2018.
- [5] Kerner, B.S., 2012. *The Physics of Traffic: Empirical Freeway Pattern Features, Engineering Applications, and Theory*. Springer.
- [6] L. A. Pipes, “An operational analysis of traffic dynamics,” *J. Appl. Phys.*, vol. 24, no. 3, pp. 274–281, 1953.
- [7] R. E. Chandler, R. Herman, E. W. Montroll, “Traffic dynamics Studies in Car Following,” *Operations Research* 6, 165-184, 1958.
- [8] T. Chow, “Operational Analysis of a Traffic-Dynamics Problem,” *Operations Research*. vol. 6, No. 6 (Nov. - Dec., 1958), pp. 827-834, 1958.
- [9] D. C. Gazis, R. Herman, and R. W. Rothery, “Nonlinear follow-the- leader models of traffic flow,” *Oper. Res.*, vol. 9, no. 4, p. 545–567, 1961.
- [10] X. Wang, R. Jiang, L. Li, Y. Lin, X. Zheng, F. Wang, “Capturing Car-Following Behaviors by Deep Learning,” *Transaction on Intelligent Transportation Systems*, vol. 19, no. 3, March 2018.
- [11] M. Treiber and A. Kesting, “Traffic Flow Dynamics: Data, Models and Simulation,” Springer-Verlag Berlin Heidelberg, 2013.
- [12] Y. Zheng, J. Wang, K. Li, “Smoothing traffic flow via control of autonomous vehicles,” *Computer Science Mathematics*, Dec. 2018.

- [13] Y. Sugiyama et al., “Traffic jams without bottlenecks—experimental evidence for the physical mechanism of the formation of a jam,” *New Journal of Physics*, vol. 10, no. 3, p. 033001, Mar. 2008.
- [14] F. Wu, R. Stern, S. Cui, M. L. Delle Monache, R. Bhadanid, M. Bunting, M. Churchill, N. Hamilton, R. Haulcy, B. Piccoli, B. Seibold, J. Sprinkle, D. Work. “Tracking vehicle trajectories and fuel rates in oscillatory traffic.” *Transportation Research Part C: Emerging Technologies*. 2017
- [15] C. Wu, A. Kreidieh, K. Parvate, E. Vinitsky, and A. M. Bayen, “Flow: Architecture and Benchmarking for Reinforcement Learning in Traffic Control,” pp. 1–16, 2017.

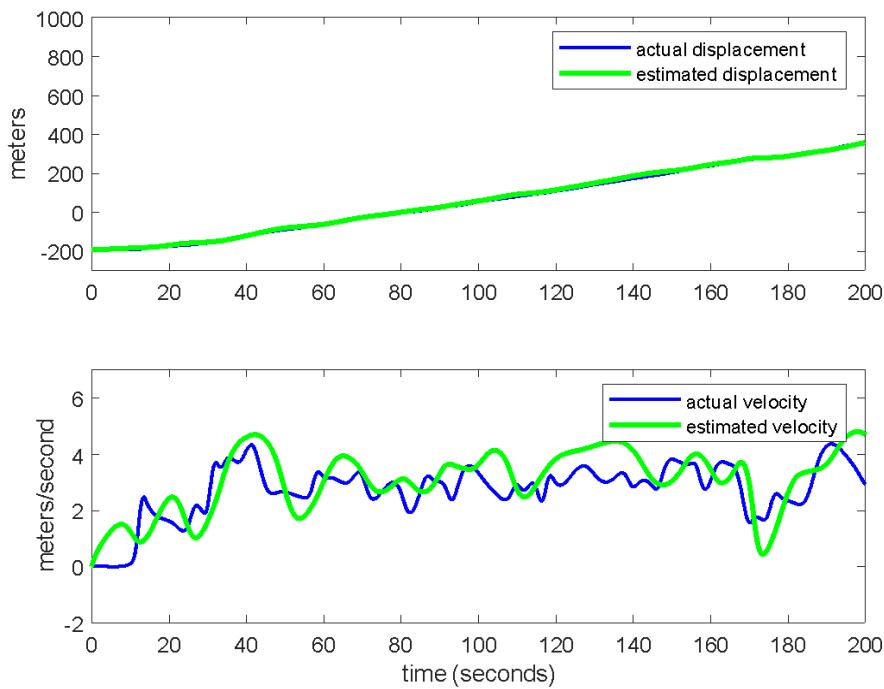
9 Appendix

Appendix 1: State estimation results for all 19 vehicles.

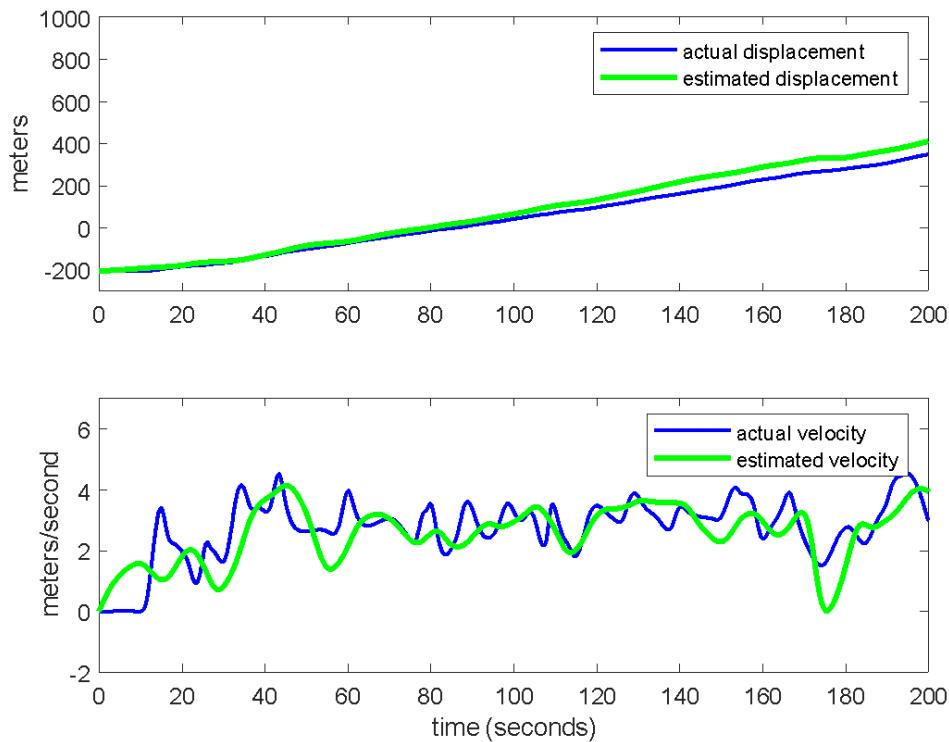
Vehicle #1



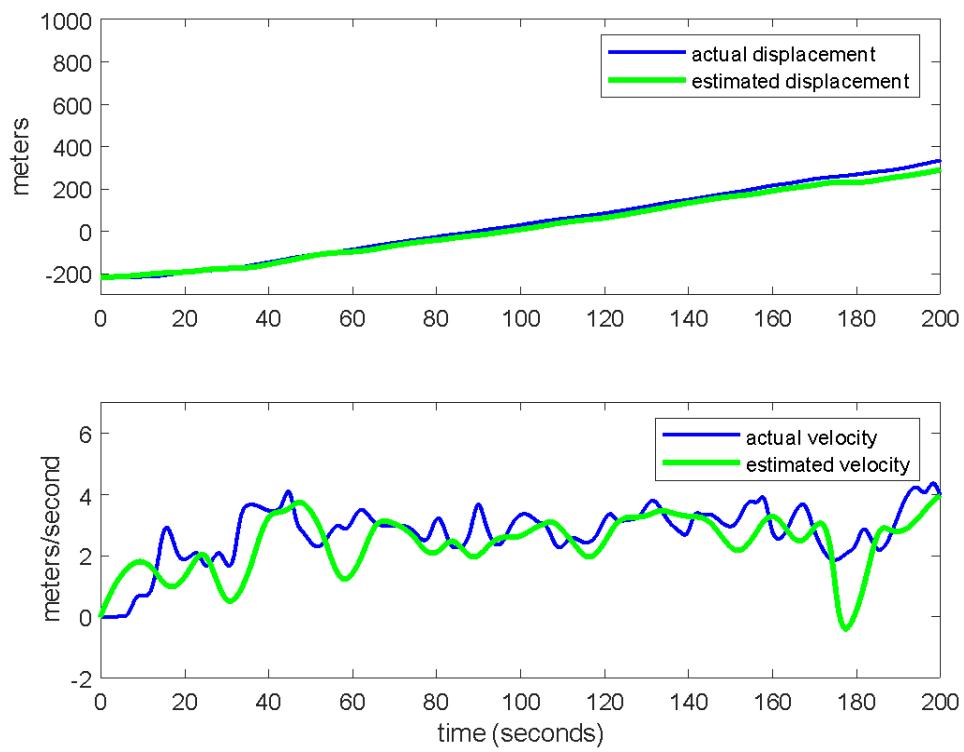
Vehicle #2



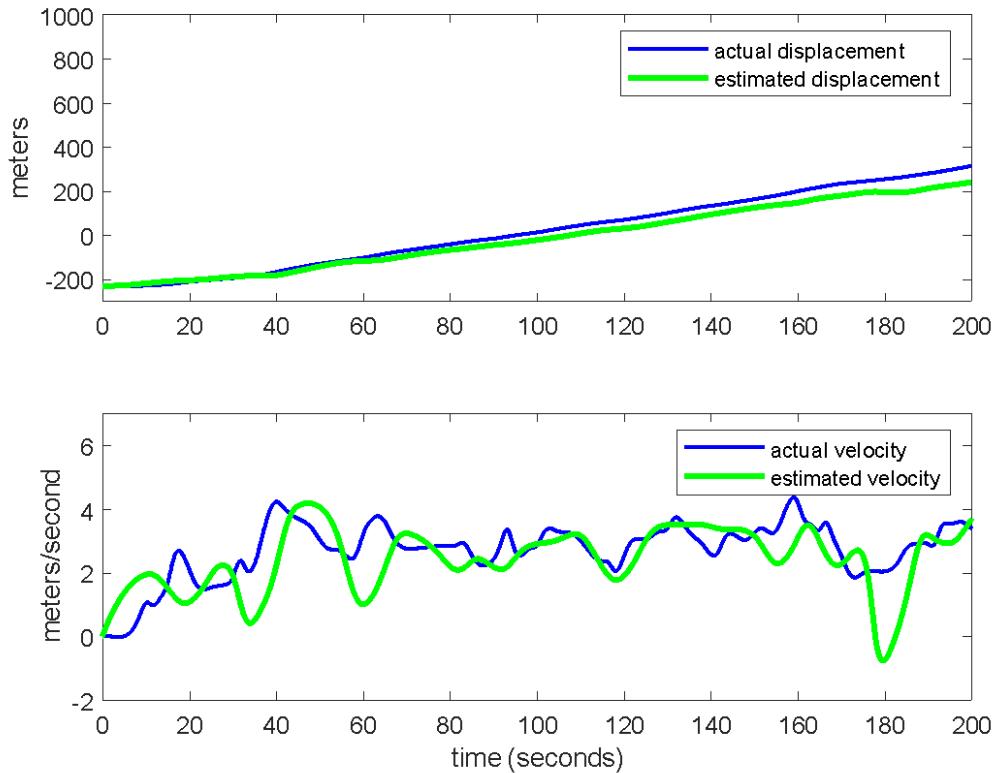
Vehicle #3



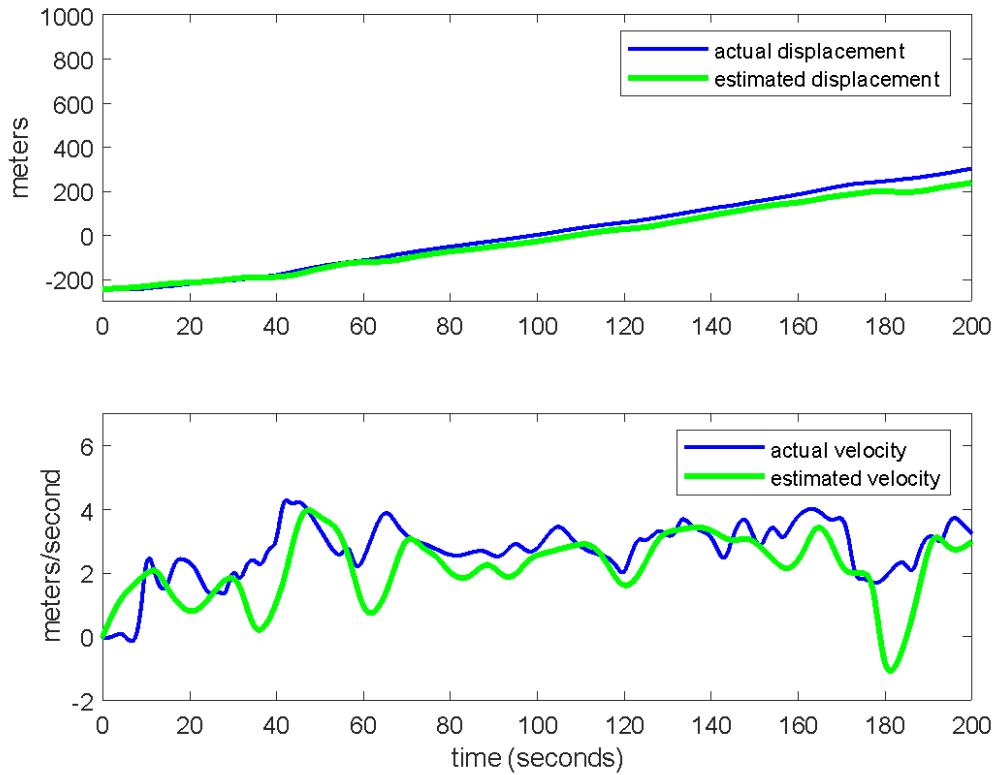
Vehicle #4



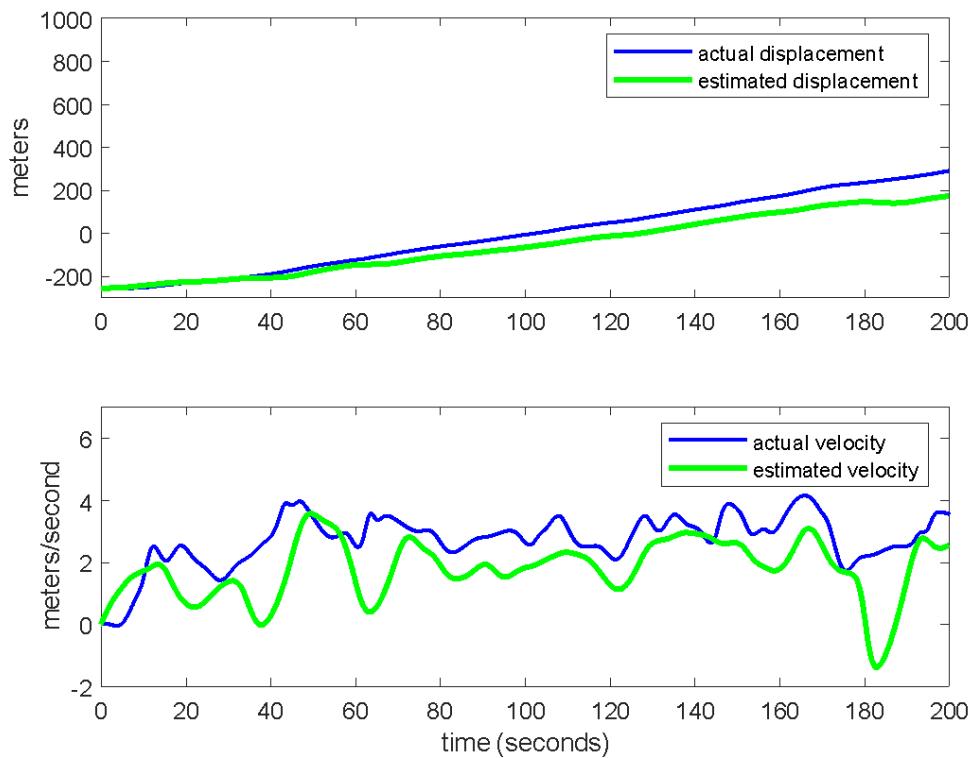
Vehicle #5



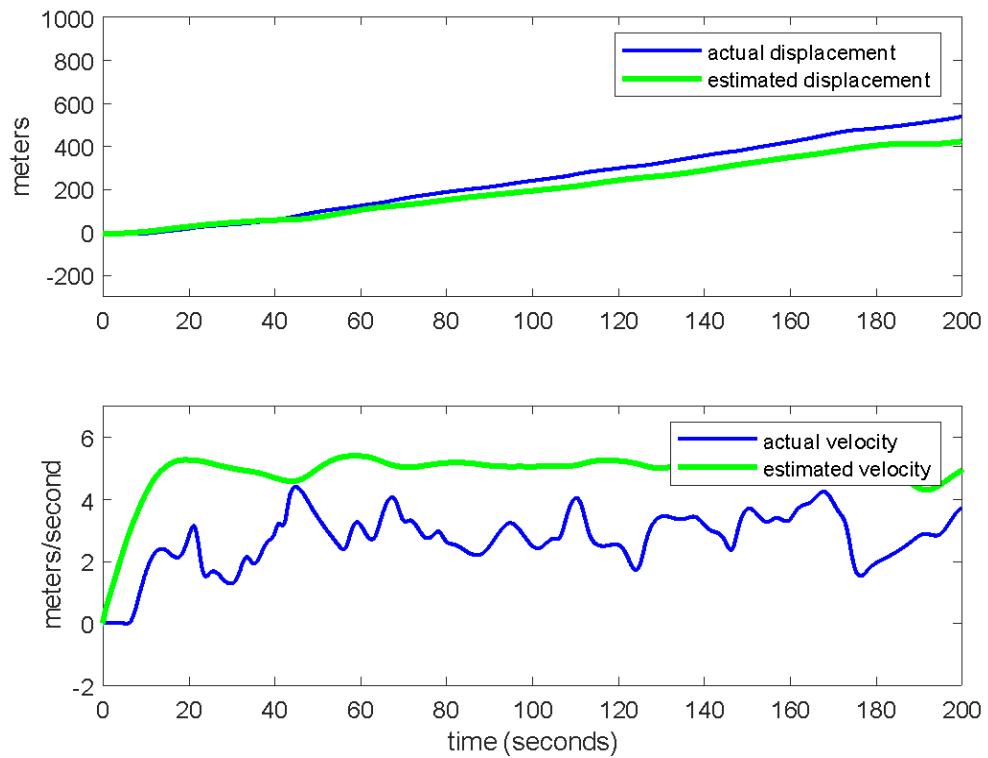
Vehicle #6



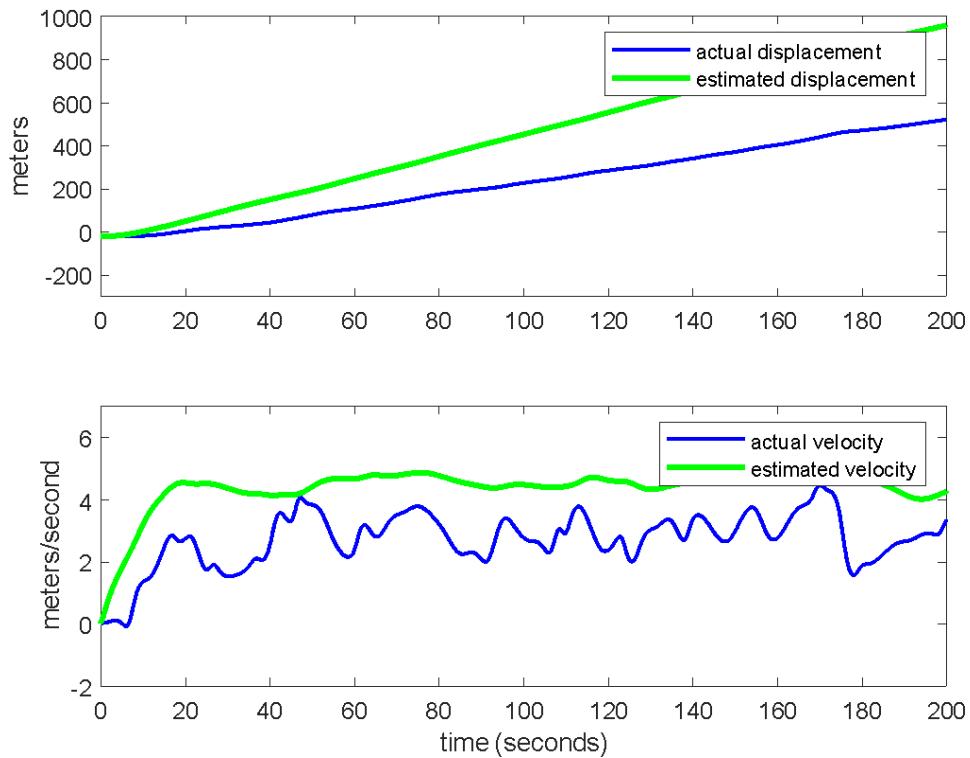
Vehicle #7



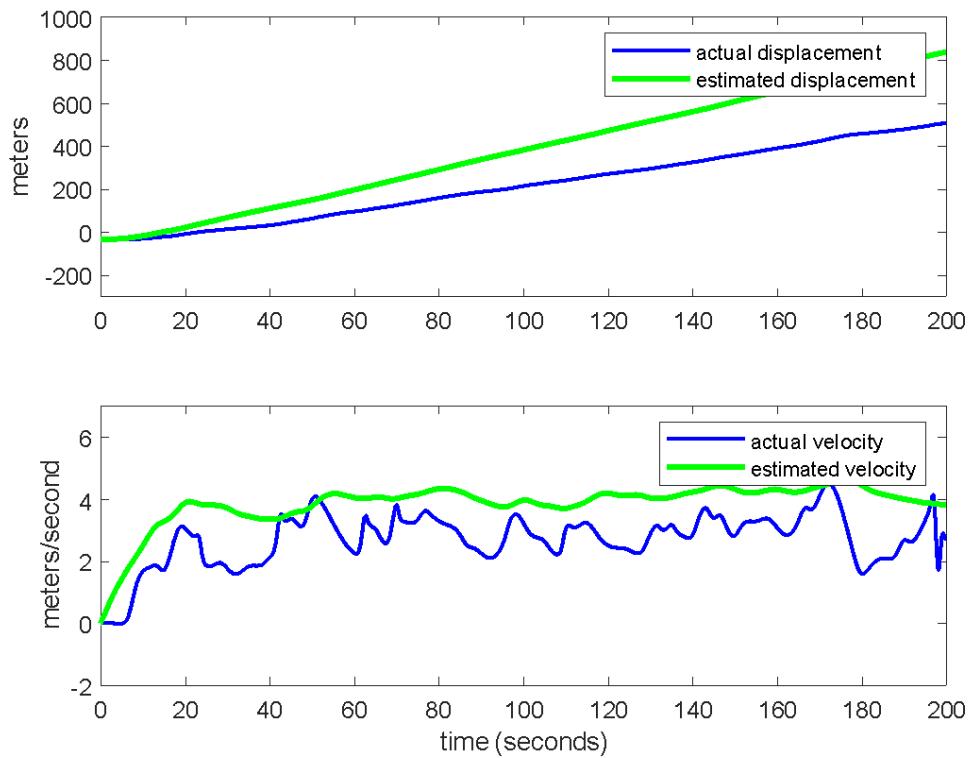
Vehicle #8



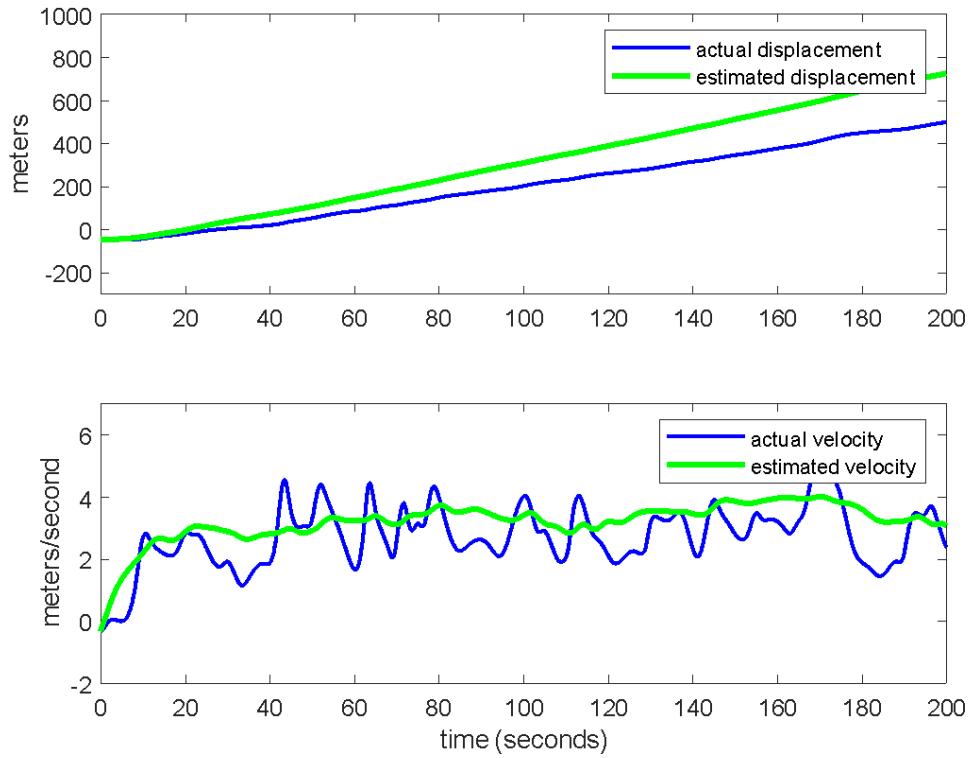
Vehicle #9



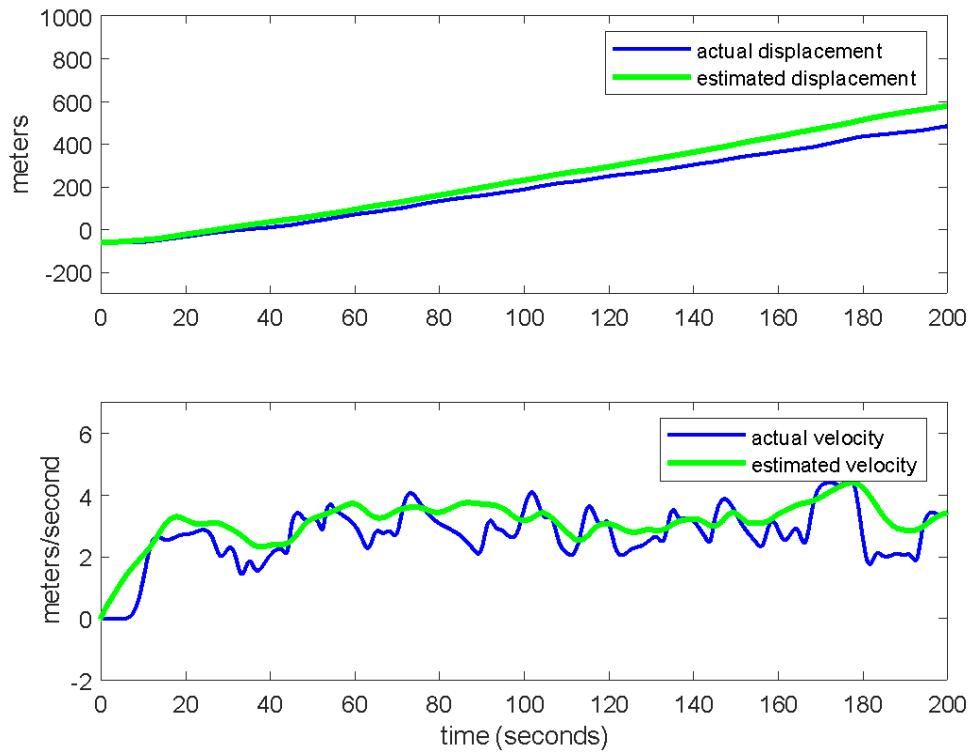
Vehicle #10



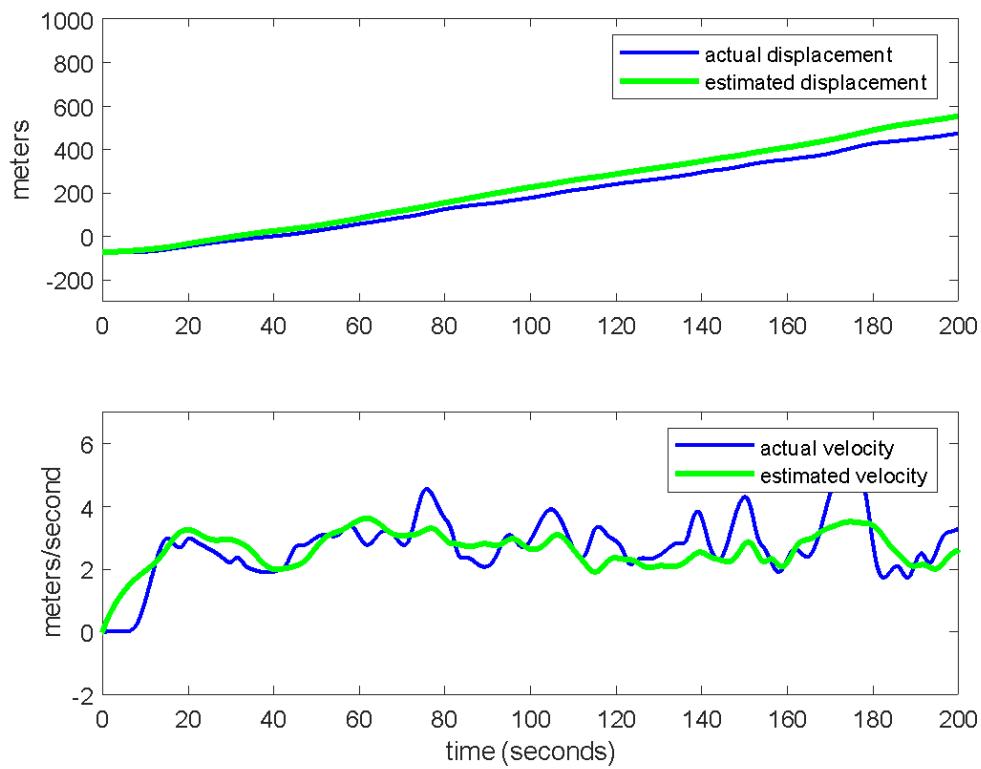
Vehicle #11



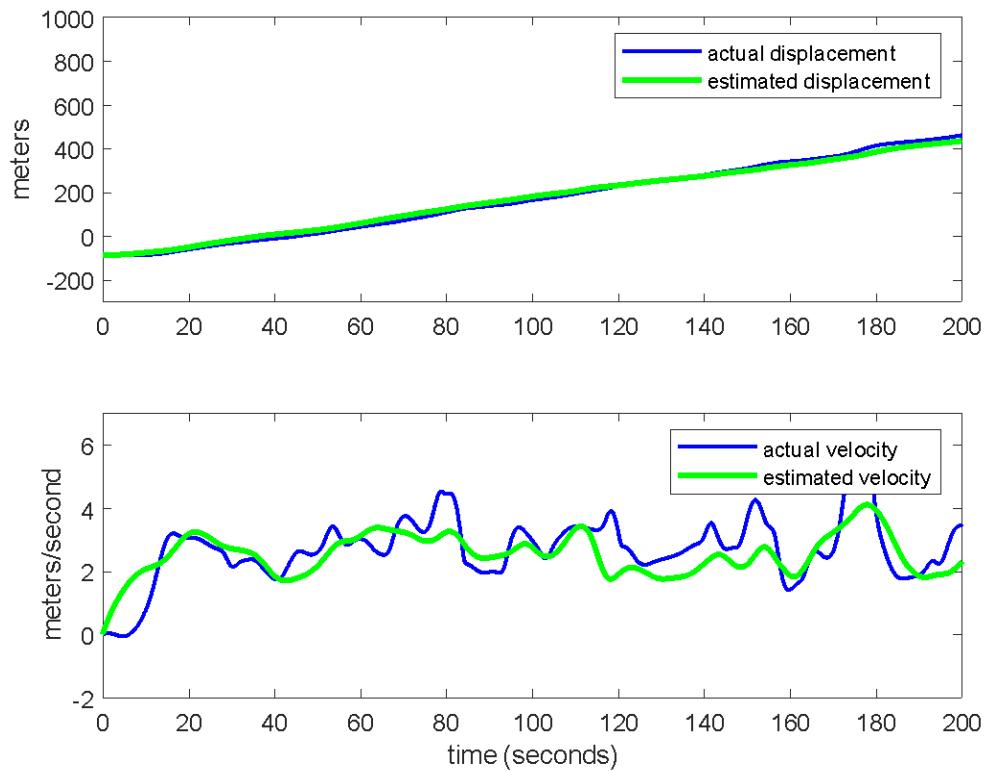
Vehicle #12



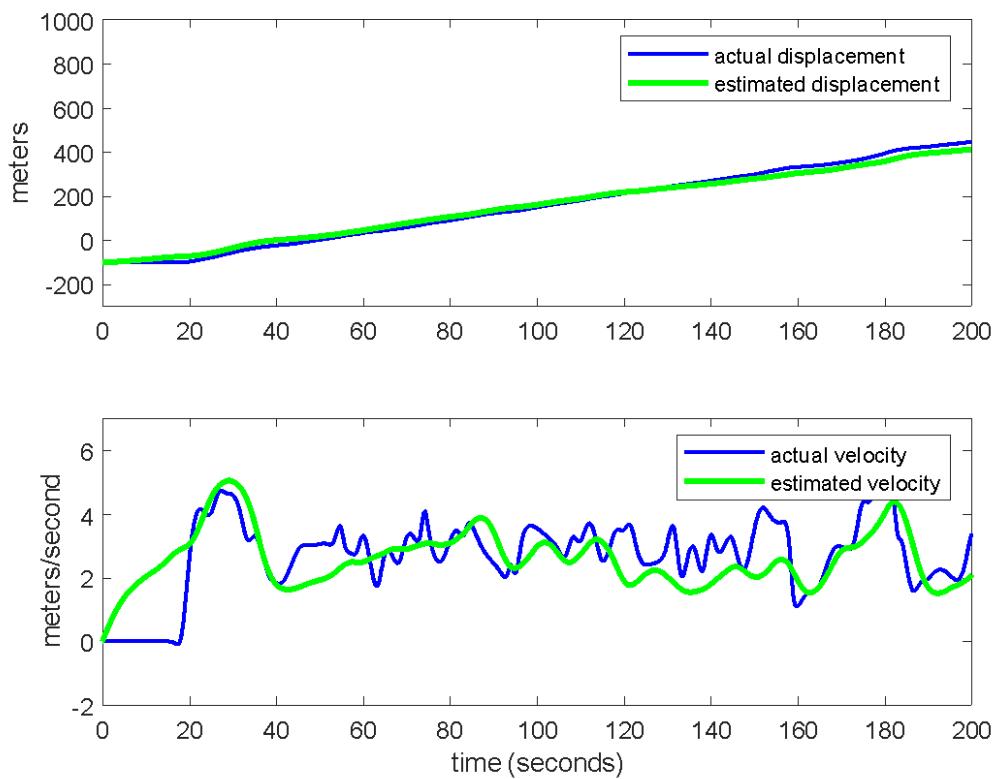
Vehicle #13



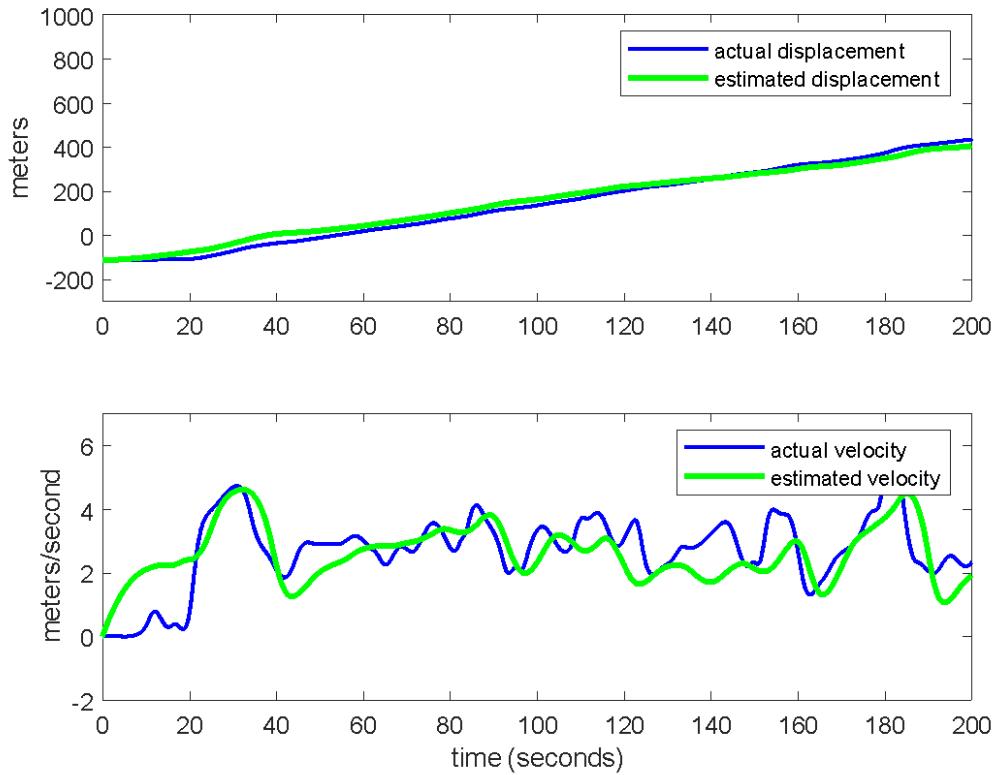
Vehicle #14



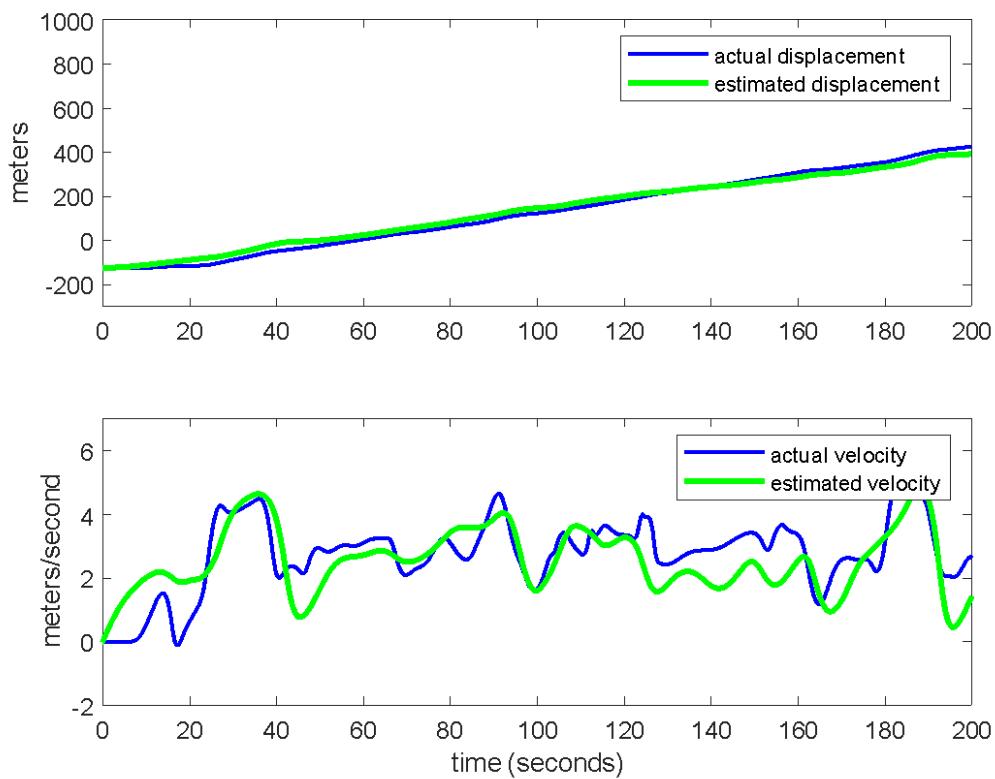
Vehicle #15



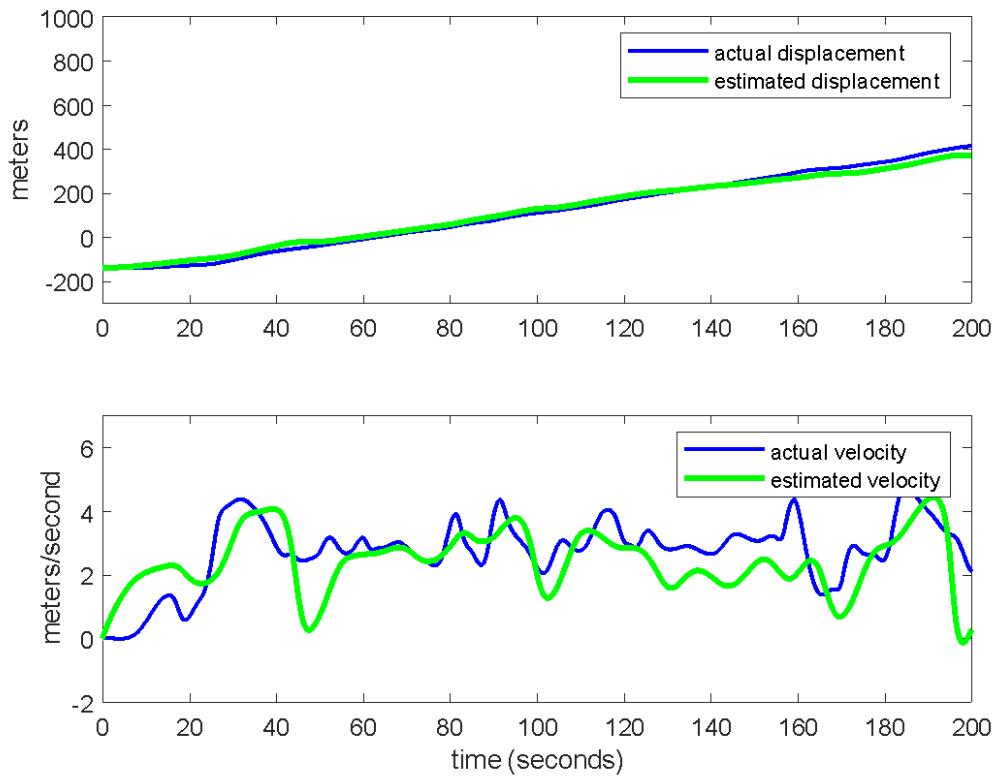
Vehicle #16



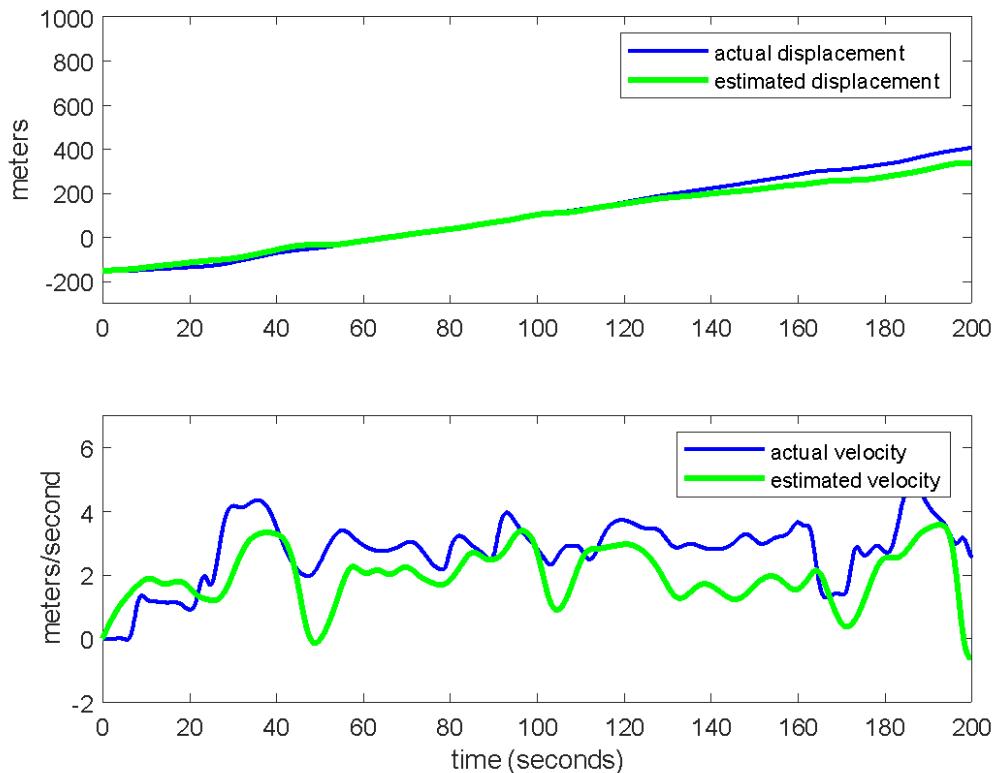
Vehicle #17



Vehicle #18



Vehicle #19



Appendix 2: Links for the codes, dataset, and simulation video

- If interested, our Matlab codes are available at:

https://drive.google.com/open?id=1mP68cPvrfXlZh33qbpjAAQiIozym_b1q

- The data (expA.csv) is made available publicly by F. Wu, R. Stern, S. Cui, M. L. Delle Monache, R. Bhadanid, M. Bunting, M. Churchill, N. Hamilton, R. Haulcy, B. Piccoli, B. Seibold, J. Sprinkle, D. Work. “Tracking vehicle trajectories and fuel rates in oscillatory traffic.” Transportation Research Part C: Emerging Technologies. 2017 and is available at:

The Arizona Ring Experiments Dataset (ARED)

<https://ir.vanderbilt.edu/handle/1803/9358>

- And our simulation video in SUMO using Flow libraries where we used estimated IDM parameters values is available at:

<https://youtu.be/7EYda3wrYns>

Then the idea of how the actual experiment worked is shown in this video:

<https://youtu.be/VRoBCfjXlXU>

CONTROL AND OPTIMIZATION OF ELECTRIC LONG-HAUL FREIGHT TRAINS BASED UPON ROUTE TIMING, POWER CONSUMPTION AND TOPOGRAPHY

Jacob Tenhoff, Edmond Yi, Morgan Wilder, Patrick Keyantuo, Peter Hubbard

ABSTRACT

The push to electrify transportation in the United States and around the world will eventually extend to the electrification of freight rail lines. This project will analyze a real long-distance freight line in the United States to simulate electricity demand and explore the relationship between reducing trip duration and increasing power demand. This work will use dynamic programming to create an optimal control policy for train velocities within a spatial domain along the track. Simulations will be run to determine the optimal velocity and torque profiles for various target velocities. These simulation outputs will be aggregated and analyzed to produce a final cost versus trip duration relationship curve. A final product of this report will be to provide guidance and reflection on the model generation process to assist organizations which would look to optimize their trip duration portfolio to maximize profits and operational efficiency.

INTRODUCTION

Motivation & Background. Rail freight transport moves 15.3% of all freight in the United States, totaling \$174B of goods per year [1]. While most freight railroads in the United States are not electrified, many in the rest of the world are, and with the increasing price of fuel, it is likely that many American Railroads will follow suit [2]. The electrification of these railroads presents new logistical and technological challenges. One such challenge is the optimization of electric power consumption through varying topography with differing train and engine characteristics. An example of this would be changing mass and engine size to maximize the benefits provided by regenerative braking and reduce overall energy demand for a given target speed. As the energy generated through regenerative braking becomes more accessible, either through on-board battery storage or through a direct connection to distribution lines, regenerative braking can be used at higher magnitudes and improve the overall energy efficiency of rail travel and lower energy consumption for a given trip managed by a rail service.

Literature Review. From an energy standpoint, rail freight is considered to be superior to car or truck freight, as the low frictional resistance between rails and wheels of the train creates lower losses per unit of mass moved by each mode [6],[7]. However, this benefit is balanced against high capital costs, long lead times for infrastructure development, and lower route flexibility compared to trucking, the dominant mode of freight in the United States [5].

Substantial literature has been developed for the dynamics of train motion and the power system dynamics of electrified rail systems [11]. Typically, the literature decomposes the forces acting on a train into three categories: tractive force, resistive forces, and the force due to gravity [5],[6],[7],[8]. These three forces encompass flows of energy to and from the gravitational and kinetic energy stocks of the train. Further literature uses models with a similar basic formulation to simulate train behavior. Using these simulations, various authors look to optimize speed profiles to minimize energy consumption [5], optimize for maximum energy regeneration through braking

[8], predict wheel slippage under various braking or acceleration conditions [7], or estimate and predict injection of harmonic current into the electric grid by the electrified rail [4].

From the literature, it is clear that the modeling and simulation of electrified rail lines can yield real benefits in their design and operation. It is also apparent that in order to take full advantage of the potential of these lines, they must be coupled to some extent with energy storage and regenerative braking [5],[8],[10],[11],[12]. To this date, large scale battery storage has been implemented to compensate for voltage drops in Japan's electrified railways. Flywheels and electric double-layer capacitors have also been implemented in Japan to store energy generated from regenerative braking on electrified rail lines. The use of flywheels in Japan was reported to have saved up to 12% of total energy, while capacitors were reported to be able to deliver up to 76% of regenerated energy back to the train [10]. Clearly, proper implementation of systems such as these, enabled by accurate modeling, simulation, and analysis, can yield energy savings.

However, electric energy can be stored through means other than batteries, flywheels, and capacitors. A concept proposed by the startup ARES, looks to store energy analogously to pumped hydroelectric storage, where electric trains with heavy masses transport the masses between two points featuring an elevation difference. This will allow the train to store energy when in excess and convert their stored potential energy to usable electricity through their motors when desired [11]. The concept presented by ARES does have the drawback that it requires the full utilization of an electric train and the adjoining infrastructure, without reaping the economic benefits that come from its intended use. However, it does inspire a related question, one that this report seeks to answer: Given the interconnection of an electrified rail to transmission and distribution systems, how can energy consumption by an electrified rail system be modeled and controlled to yield substantial benefits? Namely, an accurate representation of the marginal cost of reducing trip duration to a dispatcher of rail services, while fully optimizing the use of available resources and infrastructure.

In the process of answering this question, this report will incorporate power systems, energy storage, and a component of geospatial analysis that looks beyond the scope of the “traditional energy system problem,” in the hopes of contributing meaningfully to the base of knowledge that will be necessary for the modernization of the electric grid. The relevant literature falls into two categories, “Electrified Rail Modeling and Control,” [3],[4],[5],[6],[7],[8],[9],[13] and “Onboard Energy Storage and Regenerative Braking” [10],[11],[12].

Focus of Study. The purpose of this study is to develop a model that determines the optimal velocity profile required to minimize energy consumption of an electrified freight train traveling from the Port of Oakland to the Port of Long Beach, California. A cost versus trip time relationship will be developed by varying the minimum allowed speed of the train along the track to help stakeholders determine a desired trip duration.

TECHNICAL DESCRIPTION

The railway modeled in the project is the freight line from the Port of Oakland, CA to the Port of Long Beach, CA. This roughly 1,000 km route was chosen because it has an assortment of hills, avoids time zones changes and passes through a few populated areas, resulting in varying speeds.

Route data was obtained via interactive maps of railways, based upon CalTrans data [15]. The route was discretized into 197 data points. This data included the length of the track section,

speed limit and geographical coordinates. These coordinates were then run through a geographic information system (GIS) database to retrieve elevation data [16]. Figure 1 and Figure 2 illustrate the elevation and speed limit on the railroad versus distance.

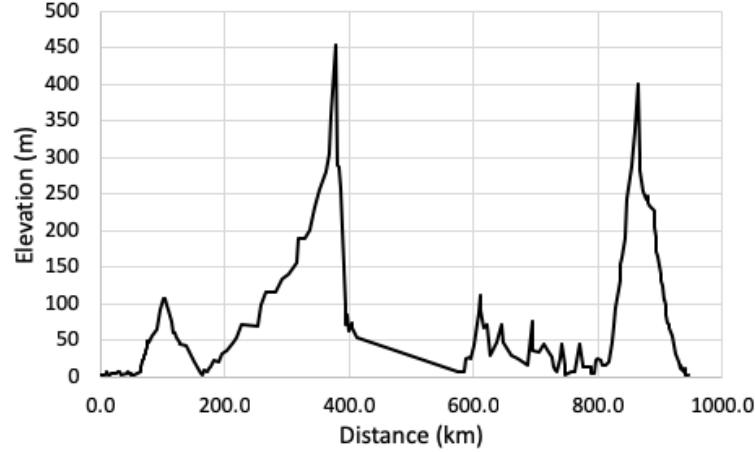


Figure 1. Elevation of rail line over the route.

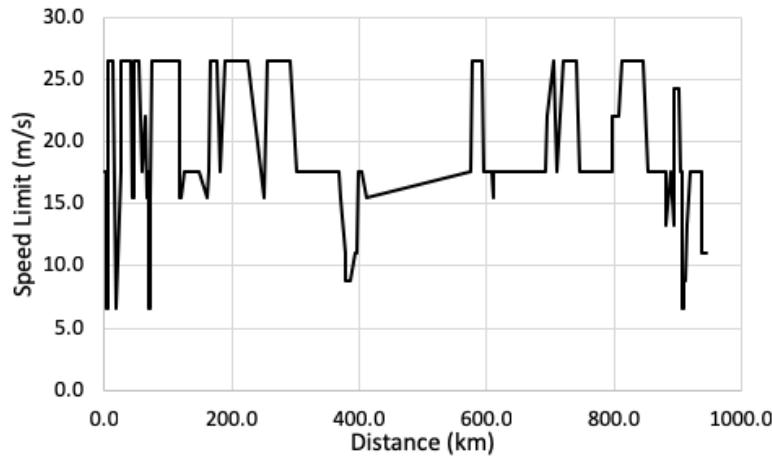


Figure 2. Speed limit along rail line.

DYNAMICAL MODEL AND OPTIMIZATION

Multi-Car Train Dynamics in Time. The rolling stock dynamical model equations are presented in Iwnicki *et al.* The general equations of motion assume that there is no lateral or vertical movement of the vehicle [18]. The diagram in Figure 3 illustrates the train model for which the general equations are derived.

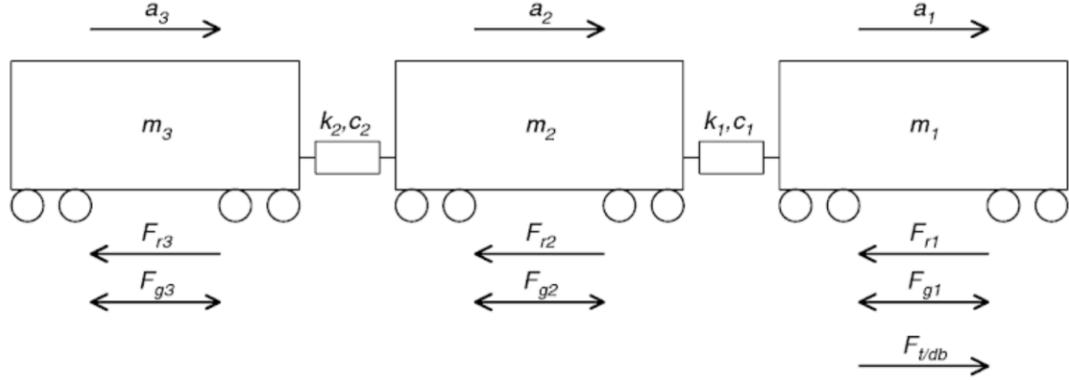


Figure 3: Three mass train model

This model considers the front wagon, m_1 , as the lead, the in-train wagon, m_2 and the last wagon, m_3 as the tail wagon. Thus, the in-train wagon, can represent a multiple number of vehicles, extending the equations for a generalized model which is can be written as follows:

$$\begin{aligned} m_1 a_1 + c_1(v_1 - v_2) + k_1(x_1 - x_2) &= F_{t/db_1} - F_{r_1} - F_{g_1} \\ m_i a_i + c_{i-1}(v_i - v_{i-1}) + c_i(v_i - v_{i+1}) + k_{-i}(x_i - x_{i-1}) + k_i(x_i - x_{i+1}) &= F_{t/db_i} - F_{r_i} - F_{g_i} \\ m_n a_n + c_{n-1}(v_n - v_{n-1}) + k_{n-1}(x_n - x_{n-1}) &= F_{t/db_n} - F_{r_n} - F_{g_n} \end{aligned}$$

Where the i index represents the i -th vehicle in between the lead and tail wagon, and the n index represents the n -th or last vehicle. The retarding force, F_r , contains the rolling resistance, curving resistance, air resistance and braking resistance. The grade force F_g , encodes the force required to overcome a gradient. The locomotive traction and dynamic braking term, $F_{t/db}$, encodes the traction and dynamic forces applied to the powered vehicles.

For the formulation implemented for this project, this model is complicated and contains parameters that are difficult to define. Also, it is formulated with respect to time. For ease of implementation and proof-of concept of the project objective, a simplified spatial formulation was developed.

Simplified Explicit Train Dynamics in Space. For simplicity in this study, the train motion dynamics have been be modelled as a point mass located at the train's center of gravity. Thus, the train motion dynamics will be formulated according to the free-body diagram in Figure 4.

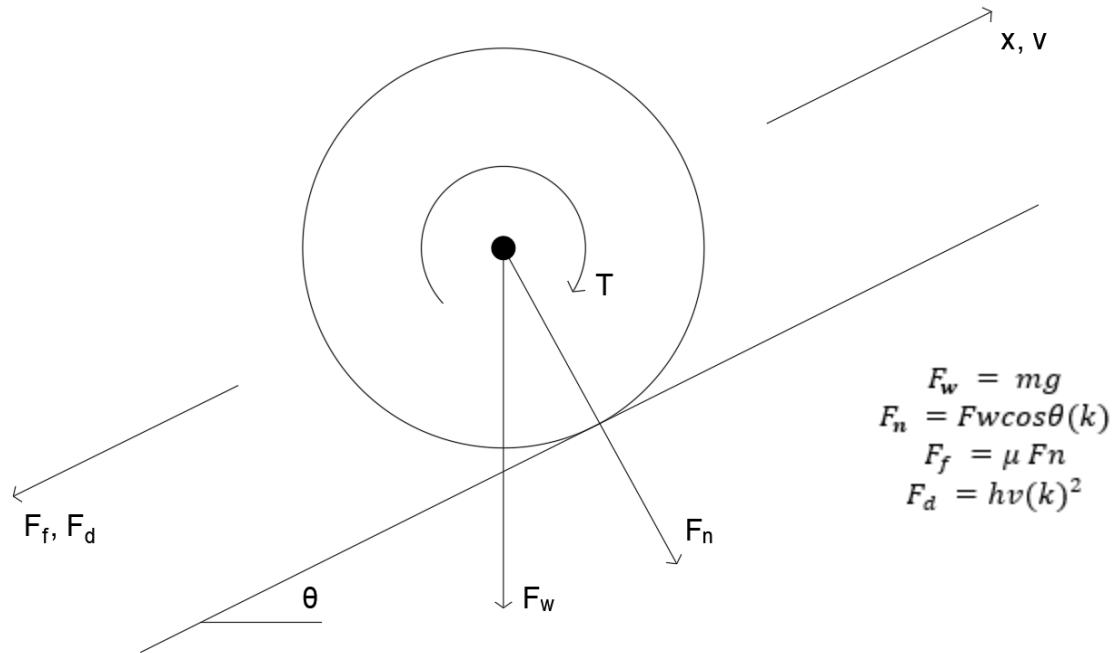


Figure 4: Point mass train model

The equations for the spatial formulation of the problem are shown below. The angle of track, θ , and the velocity, v , are modelled as discrete in space and are indexed by space-step k .

$$v(k+1) = v(k) + \frac{\Delta t}{m} \left[\frac{1}{R} T(k) - m g \sin\theta(k) - \mu m g \cos\theta(k) - h v(k)^2 \right]$$

$$\Delta t = \frac{x(k+1) - x(k)}{v(k)}$$

Where:

- $v(k)$ = velocity at space step k (m/s)
- Δt = calculated elapsed time for each space step k (s)
- R = train wheel radius (m)
- m = train mass (kg)
- g = gravitational acceleration (m/s^2)
- $T(k)$ = torque applied by the engine (N-m)
- μ = rolling coefficient of friction (-)
- h = coefficient of wind resistance against velocity squared (kg/m)
- θ = grade angle (rad)
- $x(k)$ = total distance from origin at the beginning of space index k (m)

Optimization Strategy. A dynamic programming approach will be used to develop an optimal energy reduction control policy for the engine torque applied by the simulated train on the rail line at each space index k . This control policy will depend on the topography, velocity, and engine

capacity of the rail line at each time step. This optimal control policy will be used to run a simulated train trip using parameters selected to be appropriate or as found in literature. Table 1 displays the parameters selected for this report. The modeling objective, control variable, state variable, objective function and cost function are stated below. The objective function is formulated as the minimization of the cost of power required by the electric motor over the spatial domain.

Modeling Objective: Minimize the total cost of power consumed by an electrified freight train at various minimum target velocities defined by the dispatcher. This objective is described by the objective function which minimizes cost, and the model constraints that set a policy for the minimum allowable speed.

Control Variable: Motor Torque $T(k)$

State Variable: Train velocity $v(k)$

Table 1. Model Parameters

Variable	Parameter	Model Value	Reference
R	train wheel radius	1.5 m	[19]
m	train mass	$7.0 \cdot 10^6$ kg	[20]
g	gravitational acceleration	9.81 m/s^2	-
μ	coefficient of rolling resistance	0.002	[21]
h	coefficient of wind resistance	125 kg/m	[6]
θ	grade angle	Input vector $\theta(k)$	[15],[16]
P_{max}	maximum engine power	20 MW	[22]
r_p	rate of purchased power	0.11 (USD/kWh)	[24]

$$\text{Objective Function: } J = \min \sum_{k=0}^{N+1} c(v(k), T(k))$$

Where:

$$c(v(k), T(k)) = \text{Cost of travel for space step k (USD)}$$

To simulate a model that adheres to the defined speed limits but can also model itself during abrupt changes in velocity limits, cost penalties were applied to deviations from the established limits. These penalties were imposed to inhibit the train from traveling at the minimum velocity allowed and also from traveling over the speed limit of the track. Their implementation serves as

a way to control the speed of the train within the allowable limits by assigning a monetary penalty. The cost function is defined with the following piecewise function:

Cost function for space step k:

$$c(v(k), T(k)) = \begin{cases} r_p \cdot T(k) \cdot \frac{v(k)}{R} \cdot \Delta t + 2.50 \cdot (v^{min}(k) - v(k)), & v^{min}(k) > v(k) \\ r_p \cdot T(k) \cdot \frac{v(k)}{R} \cdot \Delta t + 1000 \cdot (v(k) - v^{max}(k)), & v(k) > v^{max}(k) \\ r_p \cdot T(k) \cdot \frac{v(k)}{R} \cdot \Delta t, & \text{elsewhere} \end{cases}$$

Where:

$$\Delta t(\cdot) = \Delta t(x(k), x(k+1), v(k))$$

As shown in the cost function, $c(k)$, the first function represents the monetary penalty if the train is traveling below the minimum speed defined for a particular length of track. The cost is at a low value since the minimum speed is defined in terms of the maximum velocity and is implemented in order to prevent the train from remaining stationary. Without this limitation, the optimal way to save energy is to not travel down track. The second function implements a fine if the train is over the speed limit on a section of track. Since the maximum speed limit of the train is governed by a variety of factors such as grade, curvature and intermediate stations, the cost is increased to minimize the risk of violating these limits. Lastly, the third function defines the cost of the power required to propel the train if it adheres to the speed limits.

Modeling Constraints. Using the simplified train dynamics model, constraints can be formulated in terms of the torque applied by the engine.

Train Dynamics:

$$v(k+1) = v(k) + \frac{\Delta t}{m} \left[\frac{1}{R} T(k) - m g \sin \theta(k) - \mu m g \cos \theta(k) - h v(k)^2 \right]$$

We let $\xi(k)$ be the following expression:

$$\xi(k) = m g \sin \theta(k) + \mu m g \cos \theta(k) + k v(k)^2$$

$$\Delta v = \frac{\Delta t}{m} \left[\frac{T(k)}{R} + \xi(k) \right]$$

$$v^{min}(k) \leq v(k) + \Delta v \leq v^{max}(k)$$

$$v^{min}(k) - v(k) \leq \Delta v \leq v^{max}(k) - v(k)$$

$$v^{min}(k) - v(k) \leq \frac{\Delta t}{m} \left[\frac{T(k)}{R} + \xi(k) \right] \leq v^{max}(k) - v(k)$$

$$R[\frac{m}{\Delta t}(v^{min}(k) - v(k)) - \xi(k)] \leq T(k) \leq R[\frac{m}{\Delta t}(v^{max}(k) - v(k)) - \xi(k)]$$

Motor Capacity:

$$P_{motor} = T(k) \cdot \frac{v(k)}{R}$$

$$-P_{motor,max} \leq P_{motor} \leq P_{motor,max}$$

$$-P_{motor,max} \cdot \frac{R}{v(k)} \leq T(k) \leq P_{motor,max} \cdot \frac{R}{v(k)}$$

We let $V(v(k))$ denote the minimum cost required from space step k to the terminal space step N , where the train velocity in step k is $v(k)$. Note that in this formulation the capital letter V is used to denote the value function and the lowercase v is used to refer to the train velocity. Then the principle of optimality equations can be written as:

Principle of Optimality:

$$V(v(k)) = \min_{T(k)} [c(v(k), T(k)) + V_{k+1}(v(k+1))]$$

With the boundary condition that represents that the zero cost can be accumulated after the last time step.

Boundary Condition:

$$V(v(N)) = 0$$

RESULTS

In lieu of a two-state model, such as what is used by in research performed under the U.S. Department of Energy [23], a single state model over space was developed and utilized. This reduced the complexity of the dynamic program and in time is calculated as a byproduct of the analysis which is acceptable for the intended purpose. In a single run, the proposed model has no control over the desired trip time; however, due to the initial goal of creating a cost versus time curve, multiple simulations were run for a wide range of minimum speeds.

A single simulation run allows one to see how the train responds to the minimum and maximum velocity constraints imposed. As seen in Figure 5, the train velocity hugs the minimum allowed velocity in order to minimize energy consumption. However, due to the cost incentive to stay above the defined minimum speed, the train speeds up to avoid falling below the minimum speed in stretches of the route with smaller topographic changes. As seen between the 400 km and 600 km mark in Figure 5, the train coasts at this higher speed until it passes the stretch of track with an increased speed limit and until it reaches a steep grade. Comparing Figures 1 and 5 demonstrates this relationship. These trends seen for a minimum speed of 50% are consistent among the other possible minimum speeds.

As one would expect with the simulated velocity closely following the minimum speed, as the speed increases, the cost increase and time decreases. This is shown in Figure 6. At lower

minimum speeds, the impact of the cost penalties for speeding and going too slow are weighted stronger than at higher minimum speeds which causes jumps in the time versus minimum speed. With an increase in overall speed, the possibility of changing speed abruptly to accommodate the change in speed limits becomes less possible and results in a more dynamic velocity profile higher minimum speed. When the train is going to slow and is being penalized, the control determines the velocity should increase and when speeding, the control tells the train to slow down.

In the model, the cost is related to the torque that is required to maintain the desired speeds and when applicable, the difference between minimum and maximum velocity allowed. With all simulations starting at the same initial velocity, the lowest minimum speeds tend to coast throughout the duration of the trip, resulting in a very low power consumption and a long trip. Figure 6 demonstrates this point. Reducing the trip time starts having diminishing returns after trip duration falls below 30 hours and starts requiring a much higher cost to reduce the trip time much further. The minimum trip time for this route with current speed limits is just over 13 hours. This is not feasible for many loaded freight trains Due to the drastic speed changes that are required. The power required to produce these speed changes would most likely be beyond the capacity of a typical train engine.

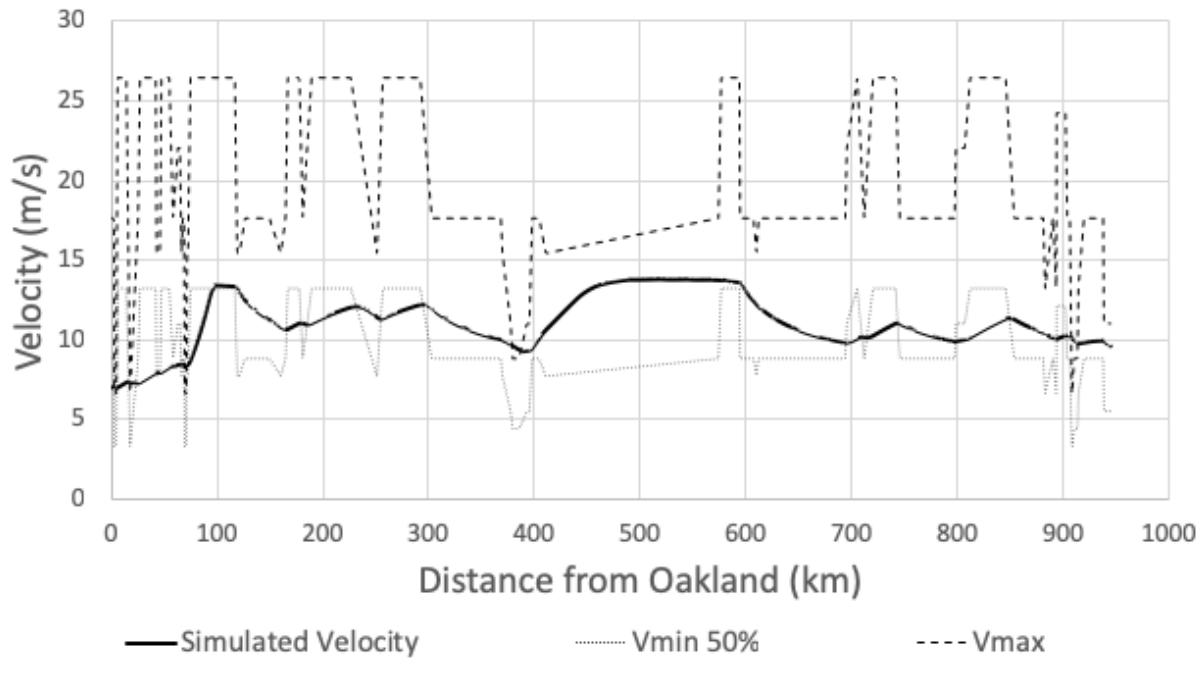


Figure 5. Simulated Velocity Profile, v^{min} 50% of v^{max}

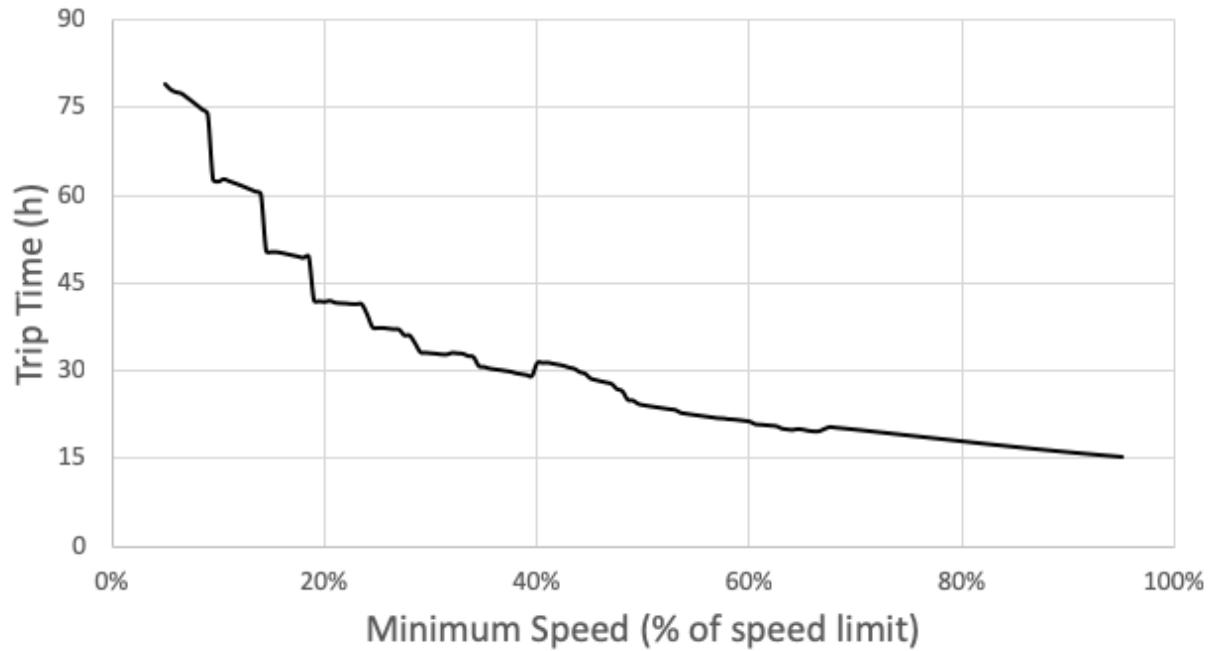


Figure 6. Trip time versus minimum speed (% of v^{max})

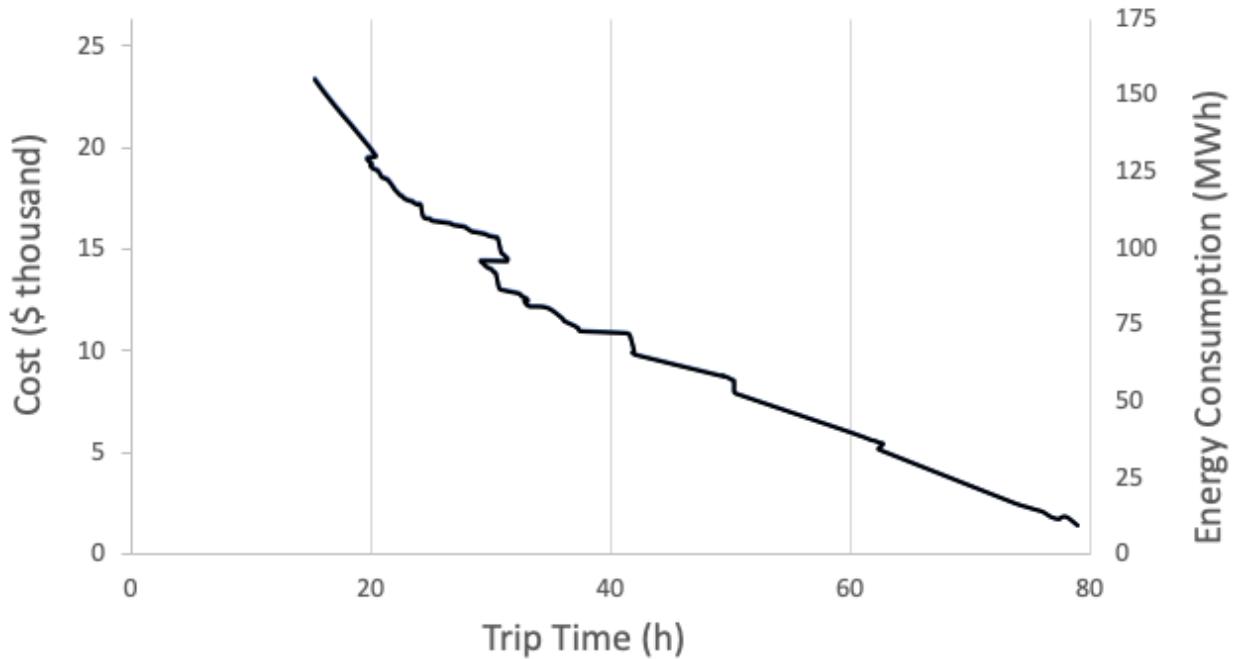


Figure 7. Cost and energy consumption profile versus time.

DISCUSSION

Figure 7 depicts the trade-off between speed and cost/energy consumption. The slope at any given point is the marginal cost in terms of energy consumption or dollars of an hour of trip time. Shorter trip times have greater opportunities to reduce consumption for a given amount of

time, whereas longer slower trips must increase trip time greatly to get a comparable cost reduction. This curve is not a true pareto curve due to each simulation adjusting the minimum speed parameter. This results in distinct optimal control profiles that has no control over time and can result in multiple trips of the same length costing different amounts, such as trip durations of around 30 hours. These likely occur due to the balance between cost to maintain the minimum speed and determination that speeding or going too slow is worth accepting the cost penalty.

In addition to reducing the model to a single state, additional physics-based simplifications were introduced to simplify the model. The model assumed a single train along the track, with no other trains affecting its journey. The model also simplified resistance by using a still air model for wind resistance, neglecting wind effects on the journey. In addition, the train was modeled as a point mass, and as such the length of the train being on different gradients was not considered. Regenerative braking and engine inefficiencies were also not considered.

With the initial goal of creating a cost versus time curve, the proposed dynamic program in the spatial domain with a single state (velocity) is sufficient. The model developed takes into consideration a minimum velocity that acts as a parameter the user would be able to adjust. Users of this model would able to influence the optimal control through the parameters of minimum speed, cost for going too slow, and cost for speeding. If a well-defined minimum speed is developed, the costs become less important; however, when speeds change drastically, and the train physically cannot accelerate to match the new speeds the cost penalties ensure the control system takes appropriate measures.

During these simulations, the train would occasionally need to exceed the maximum speed limit to optimize energy consumption. The places where this occurred corresponded to stations, where speed limits dropped to 10 miles per hour (4.47 m/s). The simulation heavily penalized speeding, but even so the simulation found that slowing down was too costly. This leads to the conclusion that bypass lanes in stations, where the speed limit was raised, would allow for reduction of energy costs by allowing a train to pass at higher speeds.

One of the weaknesses of a model in a spatial domain is an inability to account for time-based constraints. These constraints might include local regulations where speeds on tracks are reduced during night hours, or scheduling constraints where the train must be at a certain location at a certain time. The spatial domain model also does not actively constrain time; thus, many runs must be performed to produce trip duration versus energy consumption relationship. If a future author wanted to consider these constraints, they would have to implement a strategy where time is incorporated as a state, as seen in the literature for the control of autonomous vehicles in [23].

SUMMARY

This project simulated an electrified freight train journey from the port of Oakland to the port of Long Beach, California. The model was created using a simplified physics methodology in the spatial domain and incorporated spatial parameters associated with the route. Using dynamic programming with a single state variable, velocity, an optimal velocity profile was simulated to provide the minimum trip cost. With time being a byproduct of the optimal velocity profile, a time versus cost relationship was developed by varying the minimum speed parameter. It was found that the optimal way to reduce cost is to maintain a relatively stable velocity throughout the entire trip which would indicate a need for reconsidering the speed limits along the track. Additionally, for the chosen parameters, trip durations vary between 17 and 80 hours, and the trip cost varies between \$22,500 and \$2,000 respectively.

REFERENCES

- [1] "North American Transborder Freight Data," *Number of U.S. Aircraft, Vehicles, Vessels, and Other Conveyances / Bureau of Transportation Statistics*. [Online]. Available: <https://www.bts.gov/newsroom/2017-north-american-freight-numbers>. [Accessed: 15-Feb-2019].
- [2] "Analysis of Freight Rail Electrification in the SCAG Region ,," Apr-2012. .Available: <http://www.freightworks.org/DocumentLibrary/CRGMSAIS%20-%20Analysis%20of%20Freight%20Rail%20Electrification%20in%20the%20SCAG%20Region.pdf> [Accessed: 15-Feb-2019]
- [3] Li, Y., Qiang, S., Liao, H., & Xu, Y. L. (2005). *Dynamics of wind–rail vehicle–bridge systems*. *Journal of Wind Engineering and Industrial Aerodynamics*, 93(6), 483–507. doi:10.1016/j.jweia.2005.04.001
- [4] Chen, M., Roberts, C., Weston, P., Hillmansen, S., Zhao, N., and Han, X. (2017). "Harmonic modelling and prediction of high-speed electric train based on non-parametric confidence interval estimation method." *International Journal of Electrical Power & Energy Systems*, 87, 176–186.
- [5] Ghaviha, N., Bohlin, M., and Dahlquist, E. (2016). "Speed profile optimization of an electric train with on-board energy storage and continuous tractive effort." *2016 International Symposium on Power Electronics, Electrical Drives, Automation and Motion (SPEEDAM)*.
- [6] Boschetti, Giorgio & Mariscotti, Andrea. (2012). "The parameters of motion mechanical equations as a source of uncertainty for traction systems simulation." 20th IMEKO World Congress 2012. 2. 897-902.
- [7] Sadr, S., Khaburi, D. A., and Rivera, M. (2016). "A new hardware device to simulate the movement of electric train wheel on rail." *2016 7th Power Electronics and Drive Systems Technologies Conference (PEDSTC)*.
- [8] Wang, J., and Rakha, H. A. (2017). "Electric train energy consumption modeling." *Applied Energy*, 193, 346–355.
- [9] Zhang, X., Wang, L., Dunford, W., Chen, J., and Liu, Z. (2018). "Integrated Full-Frequency Impedance Modeling and Stability Analysis of the Train-Network Power Supply System for High-Speed Railways." *Energies*, 11(7), 1714.
- [10] Ratniyomchai, T., Tricoli, P., & Hillmansen, S. (2014). *Recent developments and applications of energy storage devices in electrified railways*. *IET Electrical Systems in Transportation*, 4(1), 9–20.doi:10.1049/iet-est.2013.0031
- [11] Kosowatz, J. (2016). "Using Trains to Send Power to the Grid." *ASME.org*, <https://www.asme.org/engineering-topics/articles/energy/using-trains-send-power-grid>
- [12] Khodaparastan, Mahdiyeh & Mohamed, Ahmed. (2018). "Modeling and Simulation of Regenerative Braking Energy in DC Electric Rail Systems." 1-6. 10.1109/ITEC.2018.8450133.
- [13] Pilo, E. (2010). Power Supply Energy Management and Catenary Problems, WIT Press.13-23, <http://www.solutionaryrail.org>
- [14] US Department of Transportation. "National MFN Map." <https://www.transportation.gov/freight/MFN>. Accessed March 12, 2019.

- [15] United States Geological Survey. “Current and Historical Topographic Maps of the US.” <https://www.usgs.gov/core-science-systems/ngp/tnm-delivery/maps>. Accessed March 12, 2019.
- [16] “High Speed Rail Dataset”. Data Basin. <https://databasin.org/maps/new#datasets=7a9f1867f2e24a1e97ab10419a73b25a>. Accessed March 12, 2019.
- [17] “Convert GPS File to Plain Text or GPX.” GPS Visualizer. <http://www.gpsvisualizer.com/convert?output>. Accessed March 12, 2019.
- [18] Iwnicki, S. (2006). Handbook of Railway Vehicle Dynamics. Boca Raton: CRC Press.
- [19] Standard Steel. “Wheels, Dimensions and Tolerances” <http://www.standardsteel.com/wheels.php> Accessed April 27, 2019
- [20] Quora. “How Much Does a Freight Train Weigh?” <https://www.quora.com/How-much-does-a-freight-train-weigh> Accessed April 27, 2019
- [21] Engineering Toolbox. “Rolling Resistance Coefficients” <http://www.standardsteel.com/wheels.php> Accessed April 27th, 2019
- [22] The Physics Factbook. “Power of a Train” <https://hypertextbook.com/facts/2001/RadmilaIlyayeva.shtml> Accessed April 27, 2019
- [23] Sun, C., Guanetti, J., Borrelli, F., and Moura, S. (2018). “Robust Eco-Driving Control of Autonomous Vehicles Connected to Traffic Lights”
- [24] Pacific Gas and Electric Company (2007). “Energy Efficiency Assessment of Bay Area Rapid Transit Cars” Report No. BASE-PGE-05-09



UC Berkeley

CEE 295 : Energy Systems and Control

Project Report : Flight Path Optimization

Students :

Gaofeng Su, Pierre Melikov, Siyuan Feng, Deep Dayaramani, Jaewoong Lee

10th of May, 2019

1 Abstract

This project is based on a crucial issue in the aviation world : how to optimize the trajectory and controls given to the aircraft in order to optimize flight time and fuel consumption. This project aims to provide elements of a response to this problem and to define, under certain simplifying assumptions, an optimal response. using Model Predictive Control (MPC). Thus, the first step will be to define the dynamic model of the aircraft in accordance with the controllable inputs and wind disturbances. Finally, we will identify a precise objective in terms of optimization and implement an optimization program to solve it.

2 Introduction

2.1 Motivation Background

In today's world, we use multiple different ways of transportation to get from one place to the other. These methods of transportation generate a lot of emissions of CO_2 which lead to global warming and rising sea levels. Because of this, the transportation industry has seen a major shift in the technology used just to lower these emissions. But this hasn't been the same case across all different modes of transportation, especially in the airplane industry. Airplanes contribute to 12.5% of the global CO_2 pollution and 80% of the emissions are from flights of over 1500km. This is one of the main reasons this project focuses on flight path optimization. Flight paths contribute to the amount of fuel used, distance travelled which taking into account the wind speed, air pressure and distance between the two locations. They can provide a lot of advantages such as :

- By optimizing the flight path to consume the least fuel while finding the shortest path possible, we can decrease emissions by a large factor. This reduces the overall environmental impact of airplanes.
- A decrease in amount of fuel used saves the airlines a lot of money while benefiting the passengers by decreasing the cost of their overall journey.
- They can minimize the impact of the weather on the aircraft thus reducing the amount of maintenance required and saving the airlines more money.
- Better management of Air Traffic due to predetermined routes.

There are a lot of challenges that come with creating flight plans which include :

- Taking into account the weather over the whole journey. Weather here refers to pressure of air and wind speed at all points during the journey.
- The weight and amount of fuel being carried. This can also be related to the weight and size of the airplane as larger amounts of fuel are usually carried by larger planes

There are many ways to design and determine flight paths. The method that we chose is Model Predictive Control. A major reason for this was that there wasn't a lot of

research about using Model Predictive Control (MPC) to design flight paths. One reason might be that MPC is mostly used in real-time optimum control problem (e.g. path following problem). Hence, it is interesting to discuss the possibility of using MPC for a nonlinear, large scale optimization problem, which we have further explored through this project.

Another reason we chose this topic was that our team had members taking CE263, Scalable Spatial Analysis where they were learning about data processing (e.g. data decoding, data arrangement, visualization), especially applied to the wind database we included in our model.

2.2 Relevant Literature

A MODEL PREDICTIVE CONTROL APPROACH TO AIRCRAFT MOTION CONTROL
(Luca Deori, 2015–XXVIII November)

This goal of this paper is to develop a Model Predictive Control model that can be used in aircraft reference path following problem. This paper and our project are both try to establish a MPC optimization problem subject to the aircraft dynamics, aircraft performance constraints, wind disturbance. This can provide us some idea how to establish a appropriate aircraft model and relevant constraints.

However, there are some key difference between the paper and our project. This paper use MPC for a path following problem with a 3D aircraft model. The solution involve feedback control strategy and MPC was used to solve a moving finite horizon optimization problem at each time step/feedback loop until it reach the end of the reference path. In this case, the flight path was well-defined before using MPC.

Our project is to use MPC to find the optimum flight path (travel time focus or fuel cost focus), where the finite time horizon cover the whole flight trip, which make the scale of the problem extremely large.

2.3 Objective of this Study

Implement Model Predictive Control (MPC) to optimized the flight path with minimum travel time focus and minimum fuel cost focus from Chicago O'Hare International Airport (ORD) to San Francisco International Airport (SFO) taking forecast wind data into account.

3 Technical Description

The implementation and resolution of the problem is based on four parts. First of all, the dynamic model of the aircraft is configured and developed, according to the objectives set and the degree of precision required. In a second step, we do the same for the wind, and integrate this modeling with the aircraft one. Once the dynamic model is established, it must then be discretized. Indeed, given the complexity and non-linearities of the function describing the dynamic behavior of the system, an analytical solution

to our problem is difficult to find. And so we choose to solve the problem numerically. Finally, the final optimization problem will be implemented in the last phase.

3.1 Aircraft model

A aircraft point mass model (PMM) will be introduced in the path planning optimization problem. The original model is in 3-dimension, however, to simplify the problem, the original model was reduced into a 2-dimension model.

$$\begin{aligned}\dot{x} &= v \cos(\theta) + w_x \\ \dot{y} &= v \sin(\theta) + w_y \\ \dot{v} &= \frac{T - D}{m} = \frac{2T - C_d \rho A v^2}{2m} \\ \dot{m} &= -\eta T \\ \dot{\theta} &= \frac{g}{v} \tan(\phi)\end{aligned}\tag{1}$$

x : west-east distance [m]

y : north-south distance [m]

w_x : wind disturbance on x direction [m/s]

w_y : wind disturbance on y direction [m/s]

v : aircraft ground speed [m/s]

m : aircraft mass [kg]

θ : heading angle [rad]

C_d : drag coefficient [-]

ρ : air density [kg/m^3]

A : aircraft wing area [m^2]

η : thrust specific fuel consumption coefficient [$\text{kg}/\text{N}\cdot\text{s}$]

T : aircraft thrust force [N]

ϕ : bank angle [rad]

In this model, $\dot{\theta}$ involves the aircraft speed, which appears in the denominator. This implies that v cannot be 0. For a path following problem, this is acceptable. However, in a path planning problem, nonzero speed restriction implies that the aircraft can only reach the destination at the end of the given time horizon (i.e. the aircraft cannot reach the destination with a minimum travel time and stop there). Also, for nonlinear programming, the nonlinear solver is very sensitive to the form of the optimization problem and the constraints parameters. Hence, it is better to avoid putting speed in the denominator.

Therefor, the expression of $\dot{\theta}$ was reformulated and replaced by a new input variable φ [rad/s], which represent the turning rate. The final aircraft point mass model that will be used in MPC path planing :

$$\begin{aligned}
\dot{x} &= v \cos(\theta) + w_x \\
\dot{y} &= v \sin(\theta) + w_y \\
\dot{v} &= \frac{2T - C_d \rho A v^2}{2m} \\
\dot{m} &= -\eta T \\
\dot{\theta} &= \varphi
\end{aligned} \tag{2}$$

3.2 Wind model

Historical Track dataset includes 1336 flights between ORD and SFO from July,2013 to August,2013. Departure time, geographical coordinates of a time during the flight, ACID can be read from the dataset. Those flights were completed by two types of aircraft : Airbus 319 and Airbus 320.

After analysing all the 1336 paths, five ARTCC Zones (Air Route Traffic Control Centers), ZAU, ZDV, ZLC, ZMP and ZOA, are commonly crossed by flight departure from ORD to SFO. The boundaries of these five zones would be used to crop wind data. Figure displayed below shows an example of past tracks completed by VRD211.

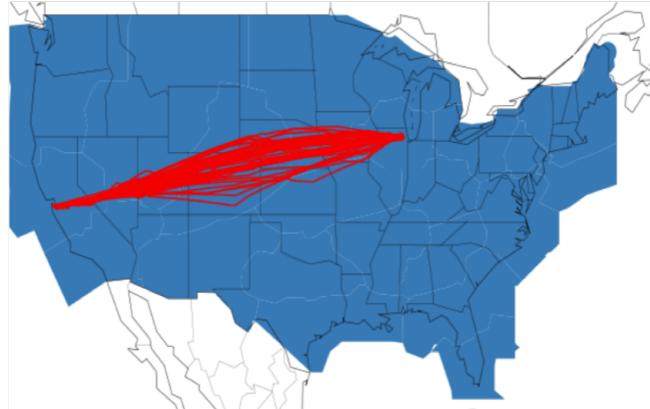


FIGURE 1 – Actual Flight Tracks from ORD to SFO

In our MPC model, forecasted wind data are used. The data are obtained from National Center for Environmental Prediction (NCEP). NCEP generates predicted wind data by Rapid Refresh (RAP) method, the continental-scale hourly-updated National Oceanic Atmospheric Administration (NOAA) assimilation/modeling system. RAP is comprised of a numerical forecast model and an analysis/assimilation system.

The data are given with locations and wind components in 2D-coordinates. The location data contain x and y coordinates in longitude and latitude, and the wind data contain wind speed in x and y direction at the corresponding location.

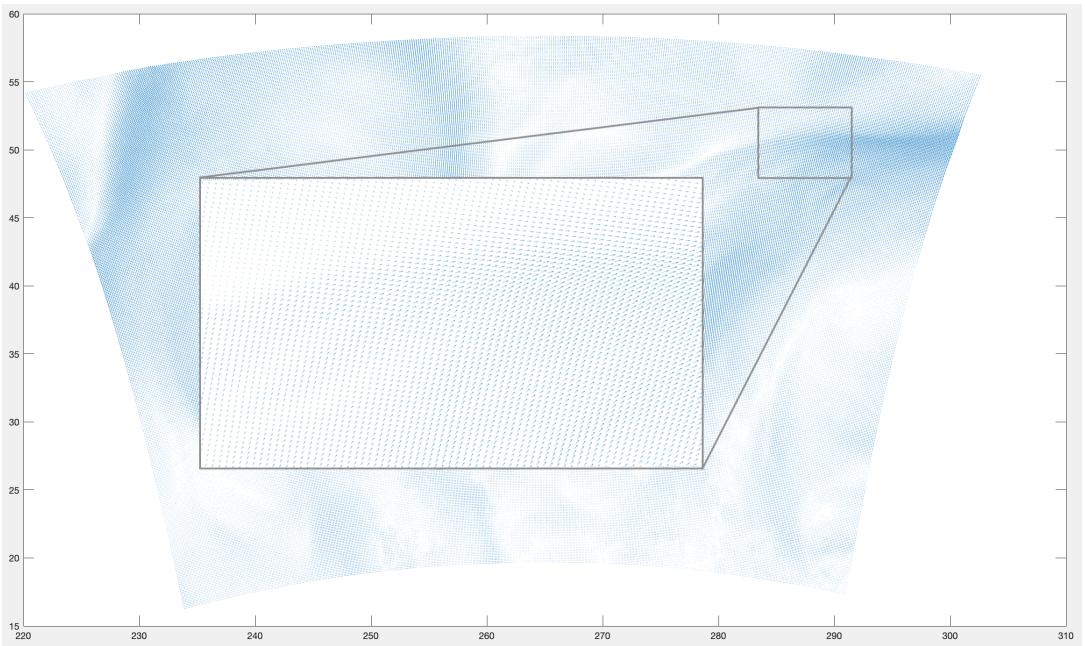


FIGURE 2 – Vector field generated with the raw wind data collected on 07/02/13 at 00 :00.

Wind components are added into our dynamics to create a more realistic MPC model.

$$\begin{aligned}\dot{x}_k &= v_k \cos(\theta_k) + w_{xk} \\ \dot{y}_k &= v_k \sin(\theta_k) + w_{yk}\end{aligned}\tag{3}$$

However, the raw wind data obtained from NCEP cannot be directly combined with our dynamics since the raw wind data components are not directly compatible with the velocity of the aircraft. To convert the raw wind data into the compatible form with the velocity dynamics of the aircraft, we developed a mathematical expression.

We first pick out the five regions defined by Federal Aviation Administration that all the real aircraft path goes through. The wind force are considered in two perpendicular directions u and v . Then we take the average of the wind force for all the 5 time slots that is accessible, and obtained the average wind force spatial distribution in the two directions u and v . We also transform the 2D coordinate systems from WGS84 to WGS 84/Pseudo-Mercator with python package pyproj, and then apply another linear transformation to set the origin of the coordinates system to the departure location. The unit is set to kilometer.

However, the discrete wind data will be hard to handled as input into the aircraft dynamics. Thus, we construct two polynomial functions to fit the two wind force spatial distribution. Here, we use x and y to represent coordinates along longitude and latitude directions. The trick is how to determine the degree and items of the polynomial functions.

From the plots above, we could find the wind force in u direction is highly correlated in x and y, while the correlation is weak for the v direction. Based on this finding, we try different polynomial functions with degrees ranging from 3 to 9 to see their performance in fitting the real wind force distribution. A least square method is applied to specify the parameter of the polynomial functions and the best performances are found when the degree is set to 5. The results are shown below. Figure 2 to Figure 4 show decent fitting performance of the two designed polynomial functions.

$$w_{xk} = a_1 y_k^4 + a_2 y_k^3 + a_3 y_k^2 + a_4 y_k + a_5 + a_6 x_k^4 + a_7 x_k^3 + a_8 x_k^2 + a_9 x_k + a_{10} + a_{11} y_k^3 x_k + a_{12} y_k^2 x_k^2 + a_{13} y_k x_k^3$$

$$\begin{pmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \\ a_5 \\ a_6 \\ a_7 \\ a_8 \\ a_9 \\ a_{10} \\ a_{11} \\ a_{12} \\ a_{13} \end{pmatrix} \approx \begin{pmatrix} 5.404 \cdot 10^{-12} \\ -7.525 \cdot 10^{-9} \\ -1.010 \cdot 10^{-5} \\ 1.8023 \cdot 10^{-3} \\ 3.054 \cdot 10^{-1} \\ 1.071 \cdot 10^{-12} \\ 8.131 \cdot 10^{-9} \\ 1.957 \cdot 10^{-5} \\ 1.360 \cdot 10^{-2} \\ 3.054 \cdot 10^{-1} \\ -4.493 \cdot 10^{-13} \\ 1.372 \cdot 10^{-12} \\ -1.971 \cdot 10^{-12} \end{pmatrix}$$

$$w_{yk} = b_1 y_k^4 + b_2 y_k^3 + b_3 y_k^2 + b_4 y_k + b_5 + b_6 x_k^4 + b_7 x_k^3 + b_8 x_k^2 + b_9 x_k + b_{10}$$

$$\begin{pmatrix} b_1 \\ b_2 \\ b_3 \\ b_4 \\ b_5 \\ b_6 \\ b_7 \\ b_8 \\ b_9 \\ b_{10} \end{pmatrix} \approx \begin{pmatrix} 6.505 \cdot 10^{-12} \\ -2.358 \cdot 10^{-10} \\ 2.009 \cdot 10^{-6} \\ 8.207 \cdot 10^{-6} \\ 6.216 \\ -2.184 \cdot 10^{-12} \\ -1.574 \cdot 10^{-08} \\ -1.790 \cdot 10^{-5} \\ 3.587 \cdot 10^{-2} \\ 6.216 \end{pmatrix}$$

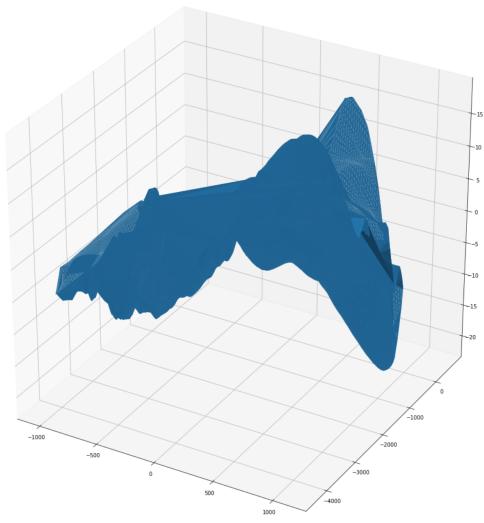


FIGURE 3 – Actual wind force distribution in u direction

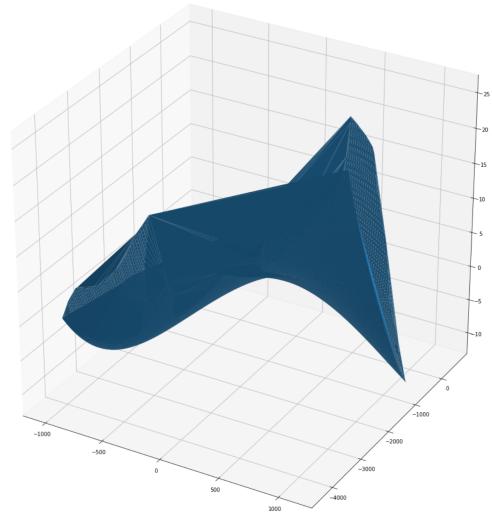


FIGURE 4 – Approximate function in u direction

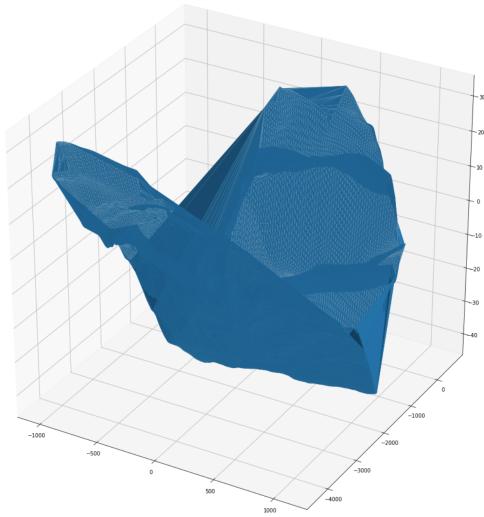


FIGURE 5 – Actual wind force distribution in v direction

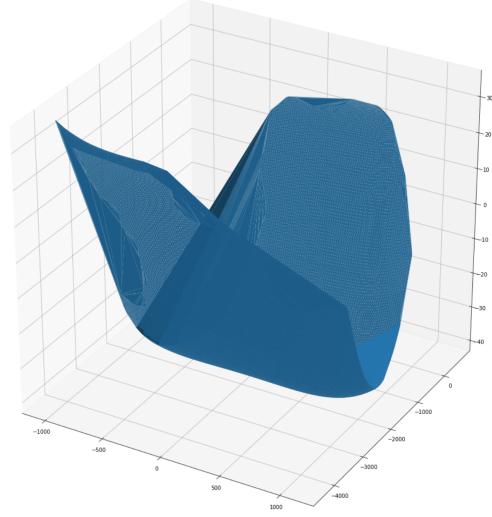


FIGURE 6 – Approximate function in v direction

3.3 Discretization

The continuous dynamic model established earlier is therefore discretized. The time derivative of a variable can be approximated by the relationship $x_{k+1} = x_k + \Delta T \cdot \dot{x}_k$. Thus, according to the model above, we finally have the following discretized dynamic relationship : $x_{k+1} = x_k + \Delta T \cdot f(x_k, u_k, w_k)$

3.4 Non-linear Programming

$$\begin{aligned} & \underset{X, U}{\text{minimize}} \quad \sum_{k=0}^{N-1} (X_k - X_f)^\top Q (X_k - X_f) + U_k^\top R U_k \\ & \text{subject to} \quad X_{lb} \leq X_i \leq X_{ub}, \quad i = 0, \dots, N \\ & \quad U_{lb} \leq U_i \leq U_{ub}, \quad i = 0, \dots, N-1 \\ & \quad X_{i+1} = X_i + \Delta T \cdot f(X_i, U_i, W_i), \quad i = 0, \dots, N-1 \\ & \quad X_0 = \mathcal{X}_0 \\ & \quad X_N = \mathcal{X}_f \\ & \quad X_f = \mathcal{X}_f \end{aligned}$$

with :

$$f(X_k, U_k, W_k) = \begin{pmatrix} \dot{x}_k \\ \dot{y}_k \\ \dot{v}_k \\ \dot{m}_k \\ \dot{\theta}_k \end{pmatrix} = \begin{pmatrix} v_k \cdot \cos(\theta_k) + w_{xk} \\ v_k \cdot \sin(\theta_k) + w_{yk} \\ \frac{2T_k - C_d \rho A v_k^2}{2m_k} \\ -\eta T_k \\ \varphi_k \end{pmatrix}$$

and

$$U_k = \begin{pmatrix} T_k \\ \varphi_k \end{pmatrix}$$

4 Results

The MPC was programmed in MATLAB. The nonlinear solver are very sensitive to the scale of the problem (i.e. numbers of variables). Hence, the total time horizon, time step size, numbers of time step need to be carefully selected.

T = total time horizon [h]

N = total numbers of time step [-]

dT = time step size [s]

A nonlinear solver IPOPT, which is suitable for large-scale nonlinear optimization, was used. Firstly, we focus on optimizing the flight path without considering wind data

and obtain the results as below. To do the optimization, we tried several time by decreasing the whole trip time from 4 hours to 3.6 hours. We wasn't able to make the aircraft arrive at SFO in 3.5 hours. Our optimal travel time is 3.6 hours in total.

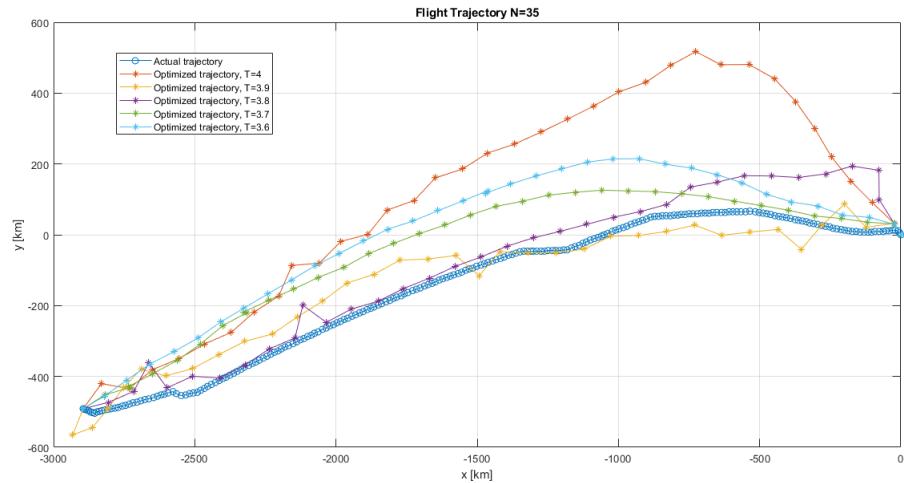


FIGURE 7 – Optimized travel time flight trajectories (no wind) with N=35

Then, based on the polynomial function we fitted for the actual wind data, we apply the function to the state matrix. And the optimal trajectory with wind data is shown as below. If we decrease the total travel time to 3.7 hours, the airplane cannot reach the destination SFO. This time, our optimal flight time is 3.8 hours.

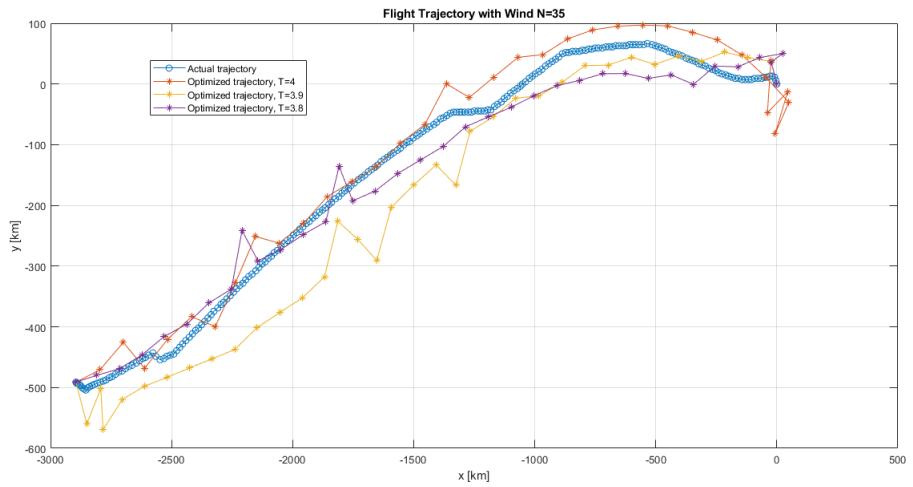


FIGURE 8 – Optimized travel time flight trajectories (with wind) with N=35

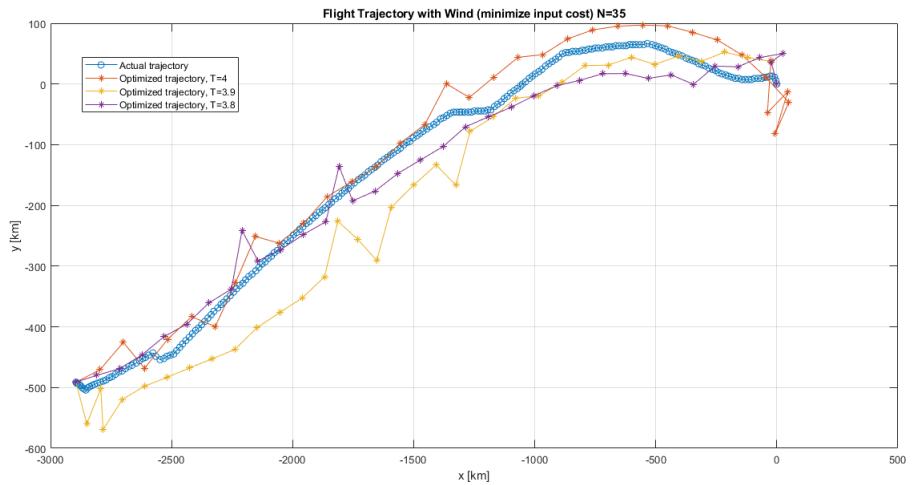


FIGURE 9 – Optimized fuel cost for the flight trajectories (with wind) with N=35

Compare the results without wind data and with wind data above, the optimal trajectories with the input of wind data are much closer to the actual flight path, which shows an acceptable result and the fitted wind function performs very well.

5 Discussion

Our project is aimed to minimize the total travel time and fuel cost to obtain the optimal flight path from Chicago O'Hare International Airport (ORD) to San Francisco International Airport (SFO) taking forecast wind data into account. And compare it to the actual flight truck from the existing sample data.

Since the optimization is a nonlinear, non-convex problem, the main challenge is to find a suitable way to solve this optimization problem. The other challenge is that in a nonlinear, non-convex optimization problem, any solution is a local optimum solution, and there is no guarantee that it is a global optimum solution.

Although there exist some solvers that can solve nonlinear optimization problem, we found out that these solvers are very sensitive to the constraints (especially to the nonlinear aircraft dynamics), the scale of the problem (i.e. numbers of variables).

For example, for a given total time horizon $T=5$ [hour], time step = 60 [s], which lead to total numbers of step = 300. The solver (IPOPT) could not find a solution. However, when reduced the total numbers of step to 40 given $T=4$ [hour] (i.e. time step = 360 [s]), the solver can find a solution.

Also, with different set up (i.e. different total time horizon, different numbers of step), the solution can change drastically. However, we can see that with certain set up, the optimized path is very similar to the actual path. For a travel time focus optimization, the path is reasonable, since not considering fuel cost, the path associate with the minimum travel time should be a straight line on the surface, which is an arc on a x-y plane.

While taking wind into consideration, the path is even much closer to the actual flight, which indicates that the aircraft model is reasonable and the aircraft performance set up is very close to reality.

There are also issues in the result. As mentioned before, the solver is very sensitive to the problem setup. For this optimization problem, the IPOPT solver will only give a unique solution, meaning that the solution is independent to the cost matrix (i.e. minimum travel time path and minimum fuel cost path are the same).

The reason might due to nonlinear programming and the defect of the nonlinear solver. Since the behavior of the nonlinear problem is very complicated and hard to handle. And how to handle the nonlinear problem and make a robust MPC model for this aircraft model should be considered in the future.

6 Summary

It is possible to use MPC method to do a flight path optimization with a nonlinear model. However, due to non-linearity of the problem and the defect of the nonlinear solver, the total number of variables needs to be carefully selected in order to obtain a reasonable result. Hence, the large scale nonlinear problem was reduced to a smaller scale by increase the time step to avoid the solver issue. And this can lead to a low resolution result, which is a drawback of the MPC method.

However, comparing the result and the actual flight trajectories and the, the solution is reasonable, which indicate that MPC method still have the potential to solve the flight path optimization problem. To obtain a more robust result, further improvement needs to be done in handling the non-linearity of the optimization problem.

7 References

- [1] Simone Garatti ; Maria Prandini, Luca Deori. "A MODEL PREDICTIVE CONTROL APPROACH TO AIRCRAFT MOTION CONTROL".
- [2] Majumder, Shibarchi, and Mani Shankar Prasad. "Flight Path Optimization Based on Obstacles and Weather Updates." 2016 3rd International Conference on Signal Processing and Integrated Networks (SPIN), 2016. doi :10.1109/spin.2016.7566732.
- [3] Khardi, Salah, and Lina Abdallah. "Optimization Approaches of Aircraft Flight Path Reducing Noise : Comparison of Modeling Methods." Applied Acoustics 73, no. 4 (2012) : 291-301. doi :10.1016/j.apacoust.2011.06.012.
- [4] Yu, Bin, Zhen Guo, Sobhan Asian, Huaizhu Wang, and Gang Chen. "Flight Delay Prediction for Commercial Air Transport : A Deep Learning Approach." Transportation Research Part E : Logistics and Transportation Review125 (2019) : 203-21. doi :10.1016/j.tre.2019.03.013.

8 Appendix

```
Cd= 0.04; %Drag force coefficient[-]
rho= 0.38; %[kg/m3] air density
eta= 0.01667*10^-3; %[kg/(N*s)]
S= 122.6; %[m2]...Aircraft wing S

%ORD=[272.093, 41.9742]; %start coord
%SFO=[237.617, 37.6211]; %final coord
ORD=[0, 0]; %start coord
SFO=[-2895204.87, -491111.25]; %final coord
```

Wind approximated function coefficient

```
a1=5.404039761756626e-12;
a2=-7.525095226657606e-09;
a3=-1.0097962636800737e-05;
a4=0.0018023200165759225;
a5=0.30541919780328985;
a6=1.0706181973188272e-12;
a7=8.130657646668963e-09;
a8=1.9566392595939457e-05;
a9=0.013598711210671936;
a10=0.3054111118706383;
a11=-4.4925592769414514e-13;
a12=1.3721355294688644e-12;
a13=-1.9705270758495876e-12;

b1=6.505580005194991e-12;
b2=-2.3582120860024345e-10;
b3=-2.0098332293991807e-06;
b4=-8.207601052061964e-06;
b5=6.216049027788961;
b6=-2.184365129705414e-12;
b7=-1.574583855440115e-08;
b8=-1.7909052661551478e-05;
b9=0.0358673394470828;
b10=6.216049027919514;
```

Simulation Parameters

```
N=35;
dT=4*3600/N; %[s] time step
```

Dynamic model/Linearized model

```
%Aircraft dynamics
%x=[x;y;v;m;theta]           u=[T;yaw];

dyn=@(x,u) [x(1)+dT*(x(3)*cos(x(5))+a1*(x(2)^4/1000^4)+...
            a2*(x(2)^3/1000^3)+a3*(x(2)^2/1000^2)+a4*(x(2)/1000)+a5+...
            a6*(x(1)^4/1000^4)+a7*(x(1)^3/1000^3)+a8*(x(1)^2/1000^2)+...
            a9*(x(1)/1000)+a10+a11*(x(1)^3/1000^3)*(x(2)/1000)+...
            a12*(x(1)^2/1000^2)*(x(2)^2/1000^2)+a13*(x(1)/1000)*(x(2)^3/1000^3));
            x(2)+dT*(x(3)*sin(x(5))+b1*(x(2)^4/1000^4)+...
            b2*(x(2)^3/1000^3)+b3*(x(2)^2/1000^2)+b4*(x(2)/1000)+...
            b5+b6*(x(1)^4/1000^4)+b7*(x(1)^3/1000^3)+...
            b8*(x(1)^2/1000^2)+b9*(x(1)/1000)+b10);
            x(3)+dT*(2*u(1)-Cd*rho*S*(x(3)^2))/(2*x(4));
            x(4)-dT*eta*u(1);
            x(5)+dT*u(2)];
```

Cost Matrics

```
Q=diag([10^3,10^3,0,0,0]);    %state cost
R=diag([0,0]);                 %input cost
```

Initial/Final States

```
%x=[x; y; v; m; theta]           u=[T; yaw];
x0=[ORD'; 100; 78000; deg2rad(127)];      %initial states
xf=[SFO'; 0; 0; 0];   %final states
```

State/Input Constraints

```
xlb=[-inf; -inf; 0; 37200; -inf];
xub=[inf; inf; 250; 78000; inf];
ulb=[0; -2.5*pi/180];
uub=[120000*2; 2.5*pi/180];
```

Path Planning

```
tic
[xopt,uopt,flag]=CFTOC2(dyn,Q,R,x0,xf,xlb,xub,ulb,uub,dT,N);
toc

function [xopt,uopt,flag] = CFTOC2(dyn,Q,R,x0,xf,xlb,xub,ulb,uub,dT,N)
disp('initialize...')

%states:[x ;y ;v ;m ;theta]
```

```

%input:[T ;yaw]
x = sdpvar(5,N+1);
u = sdpvar(2,N);

%cost
cost=[];
%initial states constraint:
constraint = [x(:,1)==x0,x(1:3,end)==xf(1:3)]; %initial/final state

for k=1:N
    %states dynamics
    constraint = constraint + [x(:,k+1)==dyn(x(:,k),u(:,k))];

    %states constraints
    constraint = constraint + [xlb <= x(:,k)<= xub];
    constraint = constraint + [-0.6*dT <= (x(3,k+1)-x(3,k))<= 0.6*dT];

    %input constraints
    constraint = constraint + [ulb <= u(:,k)<= uub];
    %cost
    cost=cost+(x(:,k)-xf)'*Q*(x(:,k)-xf)+u(:,k)'*R*u(:,k);
end

%solve optimization problem
disp('optimizing...')
%choose solver
options=sdpsettings('solver','ipopt','verbose',0,'showprogress',1);
%options.ipopt.check_derivatives_for_naninf='yes';
result=optimize(constraint,cost,options);

%check feasibility
flag=result.problem;
disp(yalmiperror(flag));
if flag==0
    %solution found
    xopt=double(x);    uopt=double(u);
else
    %no feasible solution
    xopt=[];    uopt=[];
end
end

```

CE 295 Final Report

Newport Beach Tsunami Evacuation Optimization

May 10, 2019

Li-Wei Chen, Jeremy Kho, Zining Wang, Dihan Yang

Abstract

Coastal cities are often prone to flooding due to hurricanes, tsunamis caused by earthquakes, or other natural disasters. In particular for California, there is a high risk of tsunamis caused by Pacific Rim earthquakes, a simulated 9.1 magnitude earthquake in Alaska could cause a five to ten foot storm surge as far south as San Diego in four hours [1]. During these events, city managers have limited time to implement evacuation routes for the displaced population, which can be highly variable due to beach tourism. Our project's focus is the Balboa Peninsula at Newport Beach, an area with a high tourist population, and limited egress. First, we analyzed signal probe data to estimate the population in the area to include day tourism and their likely routes. Next, we applied a traffic model to determine the number of lanes and mix of pedestrian and vehicle traffic to safely evacuate the entire peninsula's population via multimodal methods in under four hours. We were able to recommend lane usage patterns to safely evacuate both a July 4th peak tourism scenario and a summer base case scenario.

1.0 Introduction

1.1 Motivation and Background

Flooding poses a significant risk to coastal cities through the occurrence of tsunamis, heavy rain, or other natural disasters. Specifically, in Los Angeles County, low lying populations in Newport Beach, Marina Del Rey, and Long Beach are vulnerable to tsunamis. Historical examples of tsunami damage include the 1964 Alaskan Earthquake which caused 10 deaths in California, or the more recent harbor damage from the 2010 Chilean Earthquake and 2011 Fukushima Earthquake [1]. From these distances the resulting tsunamis would generate smaller storm surges in the 3 to 10 feet range, but due to the high degree of urbanization near the coast, major flooding would occur affecting over 750,000 people within a few hours [2,3]. Many communities in the Los Angeles Basin have tsunami evacuation plans [4], however, it is unclear if the selected evacuation routes have the throughput capacity to support the evacuating population.

Balboa Peninsula in Newport Beach offers several unique challenges. First, as shown in Figure 1, Balboa Peninsula in Newport Beach only has two major roads leading out of the area. A five foot high storm surge inundate most of the populated areas in the Balboa Peninsula, the majority of which are on islands created by dredging and infill. The peninsula is also highly tourism dependant, with peak populations on the beach in summer months creating additional evacuation strain. It is unclear if the two major transportation routes, as marked by the two red arrows on the top left corner of figure 1, will have enough carrying capacity as their design parameters are not focused on evacuation scale throughput [5]. One other wrinkle in the estimation of evacuees is their mode of evacuations. It may be faster for pedestrians, and people able to abandon their vehicles to attempt to evacuate on foot instead of by vehicle.



Figure 1. Newport Beach Tsunami Inundation Zone with Evacuation Routes Marked

Unfortunately, city managers have traditionally not been able to access discrete data that can provide them with reliable estimates of population densities and what type of transportation the population is using. The Newport Beach census data only provides information on residents and does not have an accurate estimate of the population on the beaches. However, our team's approach will be to estimate transient population and mode of transport based on the influx of GPS probe points for the period of 2 to 6 July 2018. A visual representation of the probes' density is shown below in Figure 2, which approximates the shape of Balboa Peninsula well.

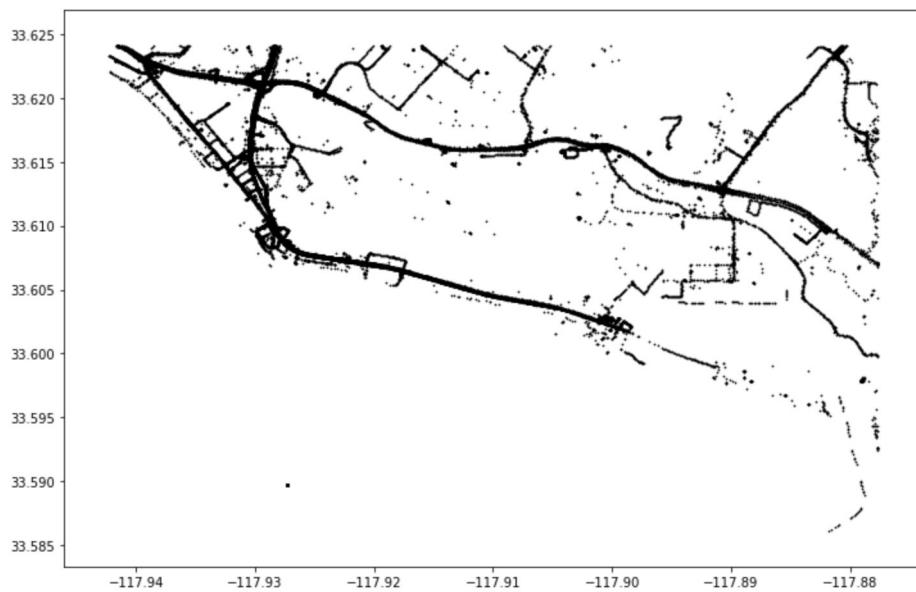


Figure 2. Newport Beach Area Probe Data Provided by CE 263N, 42,785 Data Points

1.2 Relevant Literature

Our team was primarily motivated by the United States Geological Survey Science Application for Risk Reduction published in 2013. The study had broad conclusions for coast communities in California, and particularly highlighted Santa Monica, Marina Del Rey, and Newport beach as communities that could be highly affected with displaced population in the event of a tsunami [2].

One of the limitations indicated in the study was their inability to measure the entire population affected for evacuation planning. In particular, while the researchers were able to use U.S. Census data to baseline population estimates in Balboa peninsula, they were unable to find a reliable way to estimate beachgoers. Currently, tourists in Newport Beach are estimated by volunteer lifeguards and estimates vary widely up and down the coast, and for the USGS study, they estimated 10,000 transients. However, historical police records in Newport Beach indicate that during major holidays almost 100,000 visit Balboa Peninsula [6].

While multiple cities on the California coast have published their tsunami evacuation plans, it is not immediately clear how many of them had rigorously tested their plan based on the physical constraints such as road capacity and bottlenecks such as bridges [5]. Of particular interest was another study done by the US Geological Survey specifically for Alameda, California. In this evacuation scenario, those researchers primarily focused on pedestrian evacuation to safely remove the displaced population from the tsunami hazard [7]. Vehicles were omitted from their simulations due to the limited egress routes off of Bay Farm Island. However, our team felt that focusing on pedestrians would be a significant limitation as it provides limited options for elderly or disabled evacuees unable to make the approximately 4 mile walk off of Balboa Island. Therefore we felt an opportunity to expand on the previous study would be to include pedestrian and vehicle modelling along, to determine an optimal lane usage for the two main routes off of Balboa Peninsula which would allow the entirety of the population to evacuate.

1.3 Focus of this Study

We will use GPS probe data to estimate population density in the Newport Beach area for the July 4th holiday week. Based on the population distribution, we will develop an optimal lane usage framework for Balboa Avenue and Newport Avenue to maximize flow to enable all residents and visitors to Balboa Peninsula to evacuate safely in under four hours while minimizing vehicle losses.

2.0 Technical Description

2.1 Optimization Model

Our model will rely on several assumptions prior to implementation. First, the evacuation speed of pedestrians or vehicles will be constant (time invariant) for the scope of the project. Additionally, we assume that all evacuees will follow Newport Beach's evacuation plans and only follow Balboa and Newport Boulevard off of Balboa Peninsula as described in Newport Beach's evacuation plan [8]. While some pedestrians could use two very limited side streets at the edge of the evacuation error we considered those effects minimal.

We also assumed all roads in Balboa Peninsula will be opened to only outbound flow, and no traffic would be allowed in the opposite direction of evacuation. While there are multiple dedicated turning lanes, we restricted the lane usage to only four lanes per road. Finally, the population and traffic density

in the area beyond the tsunami inundation zone is considered to approach zero, or in other words, there would be no congestion at the edge of our model.

Our model's focus areas will be West Balboa Avenue and Newport Avenue only after the split from Balboa Avenue and will not analyze other sections of the road or other choke points on the island. The goal of the model is to allow for maximum evacuee flow to safely evacuate the whole population on island in under four hours while maximizing vehicle traffic. Our team's motivation to use vehicle traffic as our cost function was twofold, first from a policy standpoint it would be easier to convince residents to evacuate via their cars vice walking, and second, the abandoned vehicles could serve as a property damage indicator.

The models objective function is given in Equation 1, to maximize the vehicles evacuated that travel on Balboa Avenue (A) and Newport Avenue (B). This model is a convex equation with a clear tradeoff between lanes dedicated for pedestrian and vehicle usage per lane, as well as the population assignment to each overall evacuation route.

$$V_{evac} = \sum_{r \in \{A,B\}} N_{lanes,r} \times Q_{veh,r} \times T_{evac,r} \quad (1)$$

Equation 2 is an expansion of Equation 1 to outline specific relationship for the two roads, and the number of lanes l_A and l_B that are dedicated to pedestrian traffic on Balboa and Newport Avenue respectively.

$$V_{evac} = (4 - l_a) \times Q_{veh,A} \times T_{veh,A} + (4 - l_b) \times Q_{veh} \times T_{veh,B} \quad (2)$$

The objective functions are subject to the following constraints and variable definitions. Each constraint is a separate equation per road, however, for convenience only one equation is shown covering both road cases:

First the pedestrians evacuated via Balboa and Newport Avenue are a function of the flow, dedicated lanes, and evacuation time.

$$P_{ped,A,B} = (l_{A,B}) \times Q_{ped,A,B} \times T_{ped,A,B} \quad (3)$$

Next, the vehicles evacuated are also modeled in the same fashion with the number of dedicated vehicle lanes of out the along with an estimated passenger count which was set at five.

$$P_{veh,A,B} = (4 - l_{A,B}) \times Q_{veh,A,B} \times T_{veh,A,B} \times Passengers \quad (4)$$

The total population evacuated by road is the sum of the two evacuation modes.

$$P_{out,A,B} = P_{ped,A,B} + P_{veh,A,B} \quad (5)$$

Next, the population to be evacuated is defined by the different zones on Balboa Island and their likely evacuation routes based on geographical constraints shown in Figure 3.



Figure 3. Newport Beach Area Segmentation

Population evacuating via Balboa Avenue (A) is function of Zone 1 and a percentage of population, given by k , from Zone 4.

$$P_{in,A} = P_1 + k \times P_4 \quad (6)$$

Population evacuating via Newport Avenue (B) is the sum of population in Zones 2 and 3, along with the population in Zone 4 not evacuating via Balboa Avenue.

$$P_{in,B} = P_2 + P_3 + (1 - k) \times P_4 \quad (7)$$

Finally, the population in each zone is related to the evacuation capacity, specifically combining equations 5, 6, and 7 so that all evacuees assigned to a certain road will match the population evacuated.

$$P_{in,A,B} = P_{out,A,B} \quad (8)$$

Additionally physical constraints are added, namely the evacuated vehicles must be less than or equal to the total number of vehicles on island estimated from the 2010 census

$$V_{evac} \leq 16900 \quad (9)$$

The time for all evacuation modes must be less than 4 hours, and more than 0.5 hours and 2 hours for vehicles and pedestrians respectively. The lower constraint represents the estimated average time of vehicle or pedestrian travel from the Eastern end of Balboa peninsula off island.

$$0.5 \leq T_{veh,A,B} \leq 4 \quad (10)$$

$$2 \leq T_{ped,A,B} \leq 4 \quad (11)$$

We then bound the Zone 4 population percentage range, and then discretized the number of lanes for each evacuation mode.

$$0 \leq k \leq 1 \quad (12)$$

$$l_a, l_b \in \{0, 1, 2, 3, 4\} \quad (13)$$

A summary of the variables and parameters are shown in the following table, along with the data source and method for determining the values.

<u>Optimization Variables</u>	<u>Data Source</u>	<u>Method</u>
I_r	Pedestrian Lanes, Optimized	
k	Population of Zone 4 traveling on Balboa Avenue, Optimized, Baseline by LA Probe Data using trajectory splitting	
Model Parameters (data driven)		
$Q_{veh,r}$	Vehicle flow. Estimated from Macroscopic Fundamental Diagram from <u>slowest hourly average vehicle speeds</u> from LA Probe Data	
$Q_{ped,r}$	Pedestrian flow. Estimated from Cheng 2012 simulating subway station evacuation	
P_i	Population. Estimated from 2010 Census expanded by LA Probe Data	
Constraint Parameters (scenario driven)		
$T_{veh,r}$	Vehicle evacuation time. Limited to 4 hour maximum based on USGS Scenario	
$T_{ped,r}$	Pedestrian evacuation time. Limited to 4 hour maximum based on USGS Scenario	
V_{avail}	Vehicles on island. Estimated based on 2010 Census vehicles per household (1.69/hh)	

Table 1. Optimization and Parameter Variables, Origin and Description

After finishing our model for evacuation, the team then analyzed the LA probe dataset to obtain the data needed as input parameters.

2.2 Flagging/Data Cleaning

Our team then preprocessed the LA probe data using a variety of techniques and heuristics taught in CE 263N. Specifically, erroneous speeds (in excess of 160 kph) were corrected, gaps in data were either discarded or linearly interpolated, data points outside of our model's physical limits in Newport Beach were removed, and duplicated data was deleted [9].

2.3 Map Matching

To facilitate data analysis, we first split the target area into four sections as previously described in Section 2.1 “Optimization model,” and for convenience show it again here.



Figure 3. Newport Beach Area Segmentation

The areas chosen are based on proximity to the three major roads and geographical barriers such as bodies of water. We then matched the probe data to West Balboa Avenue (red) and Newport Avenue (green) leading out of the Balboa Peninsula as well as the rest of Balboa Avenue running through the entirety of the peninsula as shown in blue in figure 4.

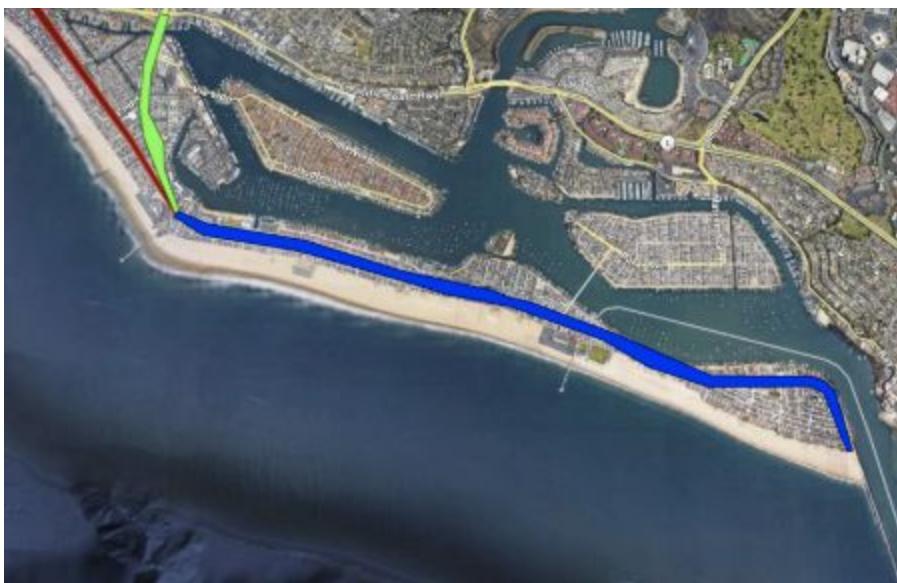


Figure 4. Probe data matching to target roads

As we were not able to find shapefiles for the roads in the area, we manually created shapefiles around the target areas in Google Earth. We then used techniques used in Assignment 0 of CE 263N to

determine if a certain probe data point falls within one of our shapefiles. The python library “Shapely” is used to convert the coordinates of each probe into a Shapely Point object. Each of these point objects are then matched to the shapefile that it belongs to and labelled accordingly for easier identification. A snippet of the labeled dataframe is shown in table 2.

SPEED	PROBE_DATA_PROVIDER	LOCAL_TIME	coord	section	street
16.0	FLEET51	2018-07-01 17:00:59- 07:00	POINT (-117.9194113 33.60690579999999)	4	w_balboa
0.0	FLEET51	2018-07-01 17:02:29- 07:00	POINT (-117.9201456 33.607024)	4	w_balboa
18.0	FLEET51	2018-07-01 17:02:45- 07:00	POINT (-117.9203801 33.6070871)	4	w_balboa
3.0	CONSUMER21	2018-07-01 17:03:33- 07:00	POINT (-117.92847 33.60822)	4	N/A
4.0	CONSUMER21	2018-07-01 17:03:38- 07:00	POINT (-117.9284 33.60815)	4	N/A

Table 2. Snippet of dataframe containing labelled probe data points

The dataframe is labelled by the section and street the point belongs to. A street label of “N/A” signifies that the point does not belong to any of the three targeted roads, and was then excluded from our speed and density analysis.

2.4 Trajectory Segmentation

To further structure our data, we looked into splitting the data collected per person per day into multiple trips. Instead of one trip that took an entire day to finish, we would have multiple trips that the person took that day. To do so, we split the data into separate chunks if the time between the points was longer than 2 minutes. The 2 minute interval was chosen at random but is reasonable for the most scenarios where drivers would stop and then head out for a new destination or if the source of the probe data stopped transmission. Trajectory segmentation allowed us to have a more realistic representation of the traffic patterns and to have more accurate values for average speeds.

Our main focus for trajectory splitting is on which exit road someone leaving the peninsula is more likely to take, W Balboa Blvd or Newport Blvd. Based on the paths identified per probe identification value, we are able to determine the origin and destination of each trip that each probe id took. We then focused on trips that originated from section 4 shown in Figure 3 and ended in one of either W Balboa Blvd or Newport. Through this analysis, we determined that when leaving the peninsula, 64% of travelers would normally choose Newport Avenue over W Balboa Avenue. A visual representation of the increased traffic flow is shown below in figure 5.

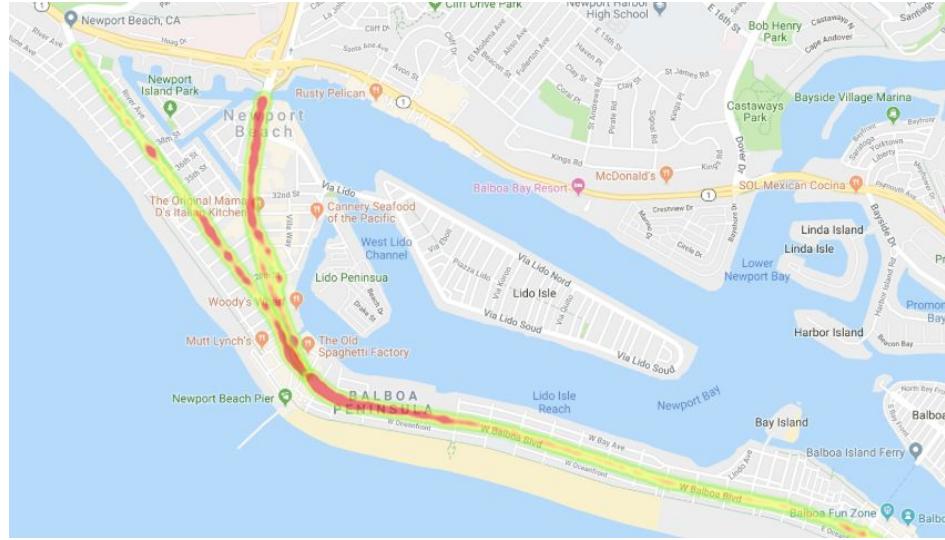


Figure 5. Heatmap of probe densities showing more usage for Newport Avenue

2.5 Average Speed

The average speed of vehicles based on the probe data is needed in order to determine the amount of congestion experienced by the two target roads under normal congestions. We first segmented the data based on the prelabeled roads. We then separated the probe data based on heading. For W Balboa Blvd, we assume that any probe data with a heading value of more than -90 degrees and less than 45 degrees is moving off of the peninsula and any probe data with a value between 145 and -150 degrees is coming onto the peninsula. With Newport Blvd, we assume that any probe data between -140 and 0 is driving off the island and data points with a heading between 90 and -160 degrees is coming onto the peninsula. These values are derived off the average orientation of the roads. We then calculated the average speed of each probe in the targeted road over an hour. Any probes with a mean speed greater than 15 km/hr is assumed to be a vehicle and any that is less than is assumed to be a pedestrian either walking or biking.

Through this analysis, we were able to determine the number of probes and average speeds of probes as sorted by hour, road, trajectory, and vehicle or pedestrian. A snippet of the resulting DataFrame can be seen in table 3.

Hour	Newport_North_Speed_Veh	Newport_South_Speed_Veh	Newport_North_Probe_Veh	Newport_South_Probe_Veh
0	0.000000	0.000000	0	0
1	59.000000	0.000000	1	0
2	30.640351	28.653333	3	3
3	41.842840	41.842840	1	1
4	37.374720	49.000000	2	1

Table 3. Snippet of average speed dataframe

2.6 Calculation of flow rate

2.6.1 Vehicle Flow Rate

Now that the team had determined the worst case congested flow rates for both Newport Beach and Balboa Avenues, we next focused on determining the vehicle flow rate. Initially the team was planning to count probes in order to estimate vehicle densities, but the GPS penetration rate of the probes was too low to make an accurate count. The team instead focused on applying research relating vehicle speed to flow rate.

Vehicle Flow Rate (cars per second) is defined by the following equation [10]:

$$Q = k \cdot V \quad (14)$$

where k is the density and V is the speed of the mode of transportation. Based on data measurements in the urban core of San Francisco, a macroscopic fundamental diagram can be developed to relate each of the terms in the equation to the other [10]. This relationship can be visualized through the fundamental diagrams shown in figure 6 below.

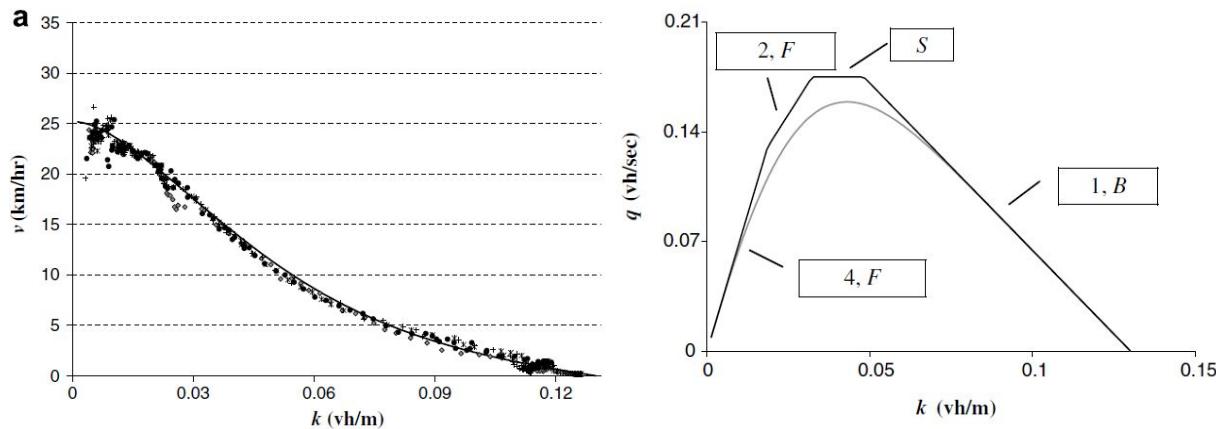


Figure 6. Relationship of vehicle density (k) to vehicle speed (V) and vehicle flow, San Francisco

Specifically in the 2008 study, San Francisco's urban core was analyzed, and after reviewing the number of intersections and lanes, the team believed this model would be a good approximation for Newport Beach's urban core on Balboa Peninsula as well.

Based on our average speed results we then extrapolated our estimates for vehicle flow with results shown below.

Road	Average Speed (kph)	Time of Day	Direction	Estimated Flow (veh/s)
Newport	16	1:00PM	North	0.034
Newport	15.3	12:00PM	South	0.037
Balboa	15	4:00PM	North	0.037
Balboa	15	4:00PM	South	0.037

Table 4. Congested Speed and Flow Estimation Results

2.6.2 Pedestrian Flow Rate

Initially the team planned to measure pedestrian velocity and density using the probes identified as pedestrians based on their behavior, however, the team realized those probes would not likely be exhibiting congested behavior similar to an evacuation. Using a series of subway pedestrian evacuation studies in China, the team selected one study that measured and modeled pedestrians in a 3.5 meter hallway most similar to a road lane. The measurements showed that the peak congested speed for a 3.5 meter wide passageway is 0.502 meters per second at a density of 3.132 people per meters squared [11]. This resulted in a flow of 1.57 people per second and is modeled by the following polynomial equation, where V is speed and k is pedestrian density.

$$V(k) = -0.00056k^4 + 0.0009k^3 + 0.0008k^2 - 0.4242k + 1.8267 \quad (15)$$

2.7 Population estimation

In order to solve the defined optimization problem, we need to determine the population to be evacuated. While census data is available to quantify the resident population, it does not take into account the transient population, of which Balboa Peninsula has plenty. As mentioned in the review of related literature, police roughly estimate a total population of 100,000 during holidays due to the surge in tourist activity. On the other hand, we have access to unlabeled GPS probe data from various sources and devices throughout the Los Angeles Metro Area. The key limitation here, however, is that we do not know the penetration rate of the data, which is protected by intellectual property rights.

In essence, the methodology to estimate population is fairly simple. The trajectories of GPS devices (distinguished in the data through unique IDs) are determined to be either residents or non-residents. Once this is achieved, we assume that the proportion of residents in the GPS data would be a near approximation of the actual proportion of residents to the population. Since we have census data, we can obtain an estimate for the total population.

2.7.1 Identifying Resident GPS Data

The key then is to identify which of the trajectories in the GPS data belong to residents. Based on a priori expectations, we can make the simple assumption that trajectories which terminate in Balboa Peninsula at night (from dusk to dawn) are more likely than not to be residents. However, it is important to note that the neighborhood is particularly affluent and residents going home in the afternoon is highly plausible. These would not be included in the initial setup since the stream of GPS data ceases once the navigation

device or application has been turned off. Hence, we expand the window to be from noon to dawn of the following day. If a trajectory enters the area after noon and does not have an outbound trip until 6:00 AM, we assume that they are residents.

With this framework, one might think to stretch this window from 6:00AM of each day to 6:00AM of the following day, as long as no outbound trip was made. While subjective, this restriction was placed to reduce the possibility of false positives, as homebound trips during that time are less likely and some transient visitors may possibly use navigation systems inbound but not outbound. Regardless, setting the interval start time to noon provides a more conservative estimate. False positives overrepresent the residents and reduce the estimate for the total population. Because we are dealing with an evacuation scenario, it would be better to overestimate the population, making false negatives the preferred error.

Utilizing data on 6 consecutive nights, from the 1st of July to the 6th of July, there were 32.67 residents on average per night represented in the GPS data. It's important to note that not all the residents are necessarily the same for each night, since navigation systems are not always used and people can have intermittent behaviors. However, the average representation of the residents is what is needed to derive population estimates. For the following section on population expansion, this number was simplified to 33.

2.7.2 Population expansion

We based our estimation on the 2010 US Census Data for Newport Beach [1]. As shown in figure 1, we were able to estimate the residential population distribution to correspond with our four evacuation regions P1, P2, P3, P4, respectively, as 100, 143, 1957, 8163, with 10363 overall residents on Balboa Peninsula. Given there were 33 probes identified in Section 2.7.1 as likely residents, we determined a ratio of 1:314, to be used as an expansion factor to represent probes to actual residents in the region.

The overall number of probes on July 4th was 285. Applying the expansion factor on this day, we estimate the total maximum number of people on Balboa Peninsula was 89,490, which is very close to historical estimates in the region of July 4th visitors and residents [6]. We then took the estimated population and separate it by zone to match the census data proportions. The estimated population on July 4th by zone was used as the population parameters for our program and is shown in Table 5.

Zone	Average probe number residents and visitors	Census residents by zone		July 4th probe number residents and visitors	July 4th population estimation by zone	
1	33 residents 110 visitors	Total Census Population 10363	100	65 residents 220 visitors	Total Estimated Population 89490	865
2			143			1235
3			1957			16900
4			8163			70490

Table 5. Resident estimation via census and probe data

3.0 Discussion

Our simulations were run using the CVX solver add on to MATLAB 2018A on a 2014 MacBook using macOS 10.14 Mojave installed.

3.1 July 4th peak visitors results

For the July 4th scenario evacuating 89,490 total residents and visitors off of Balboa Peninsula, the team found the optimal solution would be to open Balboa Avenue fully to vehicle traffic, and to have only pedestrians travel on West Newport Avenue. A total of 2131 cars would be used to travel off island, and 14% of Section 4's population would need to travel on Balboa Avenue, compared to 36% of trips currently made off island via Balboa Avenue.

These results seemed reasonable as our model had a slightly higher flow rate on Balboa than Newport Avenue. Additionally, the pedestrian evacuees only require three hours to safely evacuate, whereas the vehicles evacuate up to the four hour mark. Vehicle only evacuation solutions were all infeasible.

3.2 Summer base case results

In the case of a Summer base case analysis, we used the population data from the week of July 2nd through July 6th, excluding the July 4th holiday. The optimum solution in this case would be to have Balboa Avenue open to vehicle traffic only, along with one lane for vehicles in Newport Avenue. Three lanes in Newport Avenue would be dedicated to pedestrians.

Since there is less total population to evacuate, more cars are able to leave Balboa Peninsula compared to the July 4th case, with the optimal solution having 2631 cars evacuated and 29% of the population leaving via Balboa Avenue. The timeline for evacuation is the same as the July 4th scenario with four hours needed to evacuate the vehicles and three hours to evacuate the pedestrians. These results are intuitive as the lower total evacuation population of 44,745 people allows for less dense modes of transport to be used. Vehicle only evacuation solutions were all infeasible.

3.3 Future work expansion

For future work expansion our team identified three major areas that would enhance our model.

First, we would examine other points along the route for their flow rates. Instead of just focusing on the exits out of Balboa Peninsula, there may exist local choke points e.g. the Lido Island bridge, which may actually be the main constraint to evacuating one of our population zones. The evacuation model could also be expanded north of the Pacific Coast Highway and account for an increasing evacuee density that would back propagate and affect the flow off of Balboa Peninsula.

Another area of expansion could be determining population density on Balboa Peninsula based on the time of day. While our team was able to estimate a worst case scenario, the distribution of probe data shown in figure 7, indicates that holiday visitors come in two waves and later than the average visitors to Newport Beach.

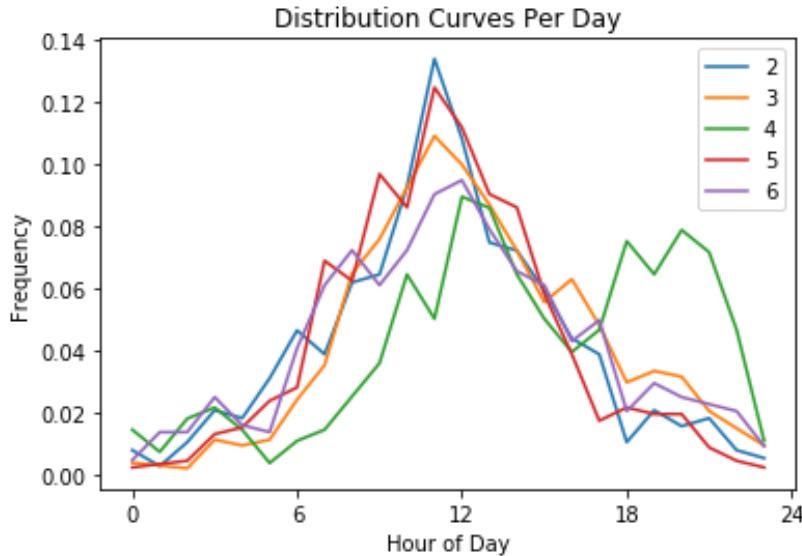


Figure 7. Normalized probe frequency over time of day, July 2nd to July 6th 2018

The varying population distribution could also be applied to the Macroscopic Fundamental Diagram and evacuation scenarios could account not only for differing on island population, but also vehicle flow, based on time of day. Additionally, more time varying components could be added in the analysis. In our optimum evacuation scenarios, pedestrians evacuated completely one hour prior to vehicles so a dynamic programming approach could allow for reoptimization and allocation of those lanes to vehicle traffic to increase vehicles leaving Balboa Peninsula.

A final area to expand our work would be to apply similar tourist population estimation methodologies at different coastal cities in California, such as Marina Del Ray or Malibu, to see if residents and tourists can be distinguished based on their location behavior through time. Additionally, the team's estimation of GPS penetration and probe expansion can be tested against other historical data at other cities.

4.0 Summary

In our project, we used a data driven approach to solve the traditionally difficult problem in estimating tourism population density in order to prepare safe evacuation plans. Using GPS probe data, our team was able to match that data to specific roadways in Newport Beach, as well as identify peak congested average speeds, and behaviors such as route selection. Applying the Macroscopic Fundamental Diagram, our team was then able to use vehicle speed to estimate their flow rate in Balboa Peninsula and applied other research to determine pedestrian flow rate. The team then built an optimization model designed to safely evacuate a peak population estimated on July 4th by allocating dedicated vehicle and pedestrian lanes. Due to the difference in congested flow rates between newport and Balboa Avenue, our optimal solution had all vehicles leaving via Balboa Avenue and pedestrians leaving via Newport Avenue, with the estimated on island July 4th population of 89,940, to all be safely evacuated in four hours. Our model provides a tool for city managers to estimate evacuation flow at certain key points, and can be expanded to analyze different areas, or to account for time delays in arrival and departure from the peninsula.

References:

- [1] R. Xia, R. Lin II, 'California maps will point to tsunami danger zones,' *Los Angeles Times*, March 21, 2014. Available: <https://www.latimes.com/local/lanow/la-me-la-me-in-california-officials-drawing-tsunami-flood-maps-to-aid-future-construction-20140321-story.html> [Accessed: February 20, 2019]
- [2] United States Geological Survey Science Application for Risk Reduction, 'The SAFRR Tsunami Scenario—Improving Resilience for California', 2013.
- [3] United States Geological Survey Science Application for Risk Reduction, 'The SAFRR Tsunami Scenario', 2013.
- [4] Ventura County Sheriff's Office of Emergency Services, 'Ventura County Operational Area Tsunami Evacuation Plan', 2006.
- [5] Transportation Research Board, 'Highway Capacity Manual, 6th Edition,' Washington D.C., 2016.
- [6] T. Martinez, 'Newport Beach's crowded Fourth of July', 2012 [Online]. Available: <https://www.ocregister.com/2012/07/05/newport-beachs-crowded-fourth-of-july/> [Accessed: April 20, 2019]
- [7] J. Peters, N. Wood, R. Wilson, K. Miller, 'Intra-community implications of implementing multiple tsunami-evacuation zones in Alameda, California' in National Hazards, 2016. DOI 10.1007/s11069-016-2469-8
- [8] H. Davis, 'Big Tsunami could flood large swaths of Newport Beach,' Los Angeles Times, March 26, 2018. Available: <https://www.latimes.com/local/lanow/la-me-newport-tsunami-20180326-story.html> [Accessed: February 20, 2019]
- [9] J. MacFarlane, 'Data Preprocessing', CE 263N Lecture, University of California Berkeley, 2019.
- [10] C. Dagazno, N. Geroliminis, 'An analytical approximation of the Macroscopic Fundamental Diagram of Urban Traffic' in Transportation Research, 2008. DOI 10.1016/j.trb.2008.06.008
- [11] F. Li, S. Chen, X. Wang F. Feng, 'Pedestrian Evacuation Modeling and Simulation on Metro Platforms Considering Panic Impacts,' in International Conference on Traffic and Transportation Studies 2014. DOI: 10.1016/j.sbspro.2014.07.209

Appendix

Optimization Code

%% The purpose of this code is to develop optimization scenario for evacuation off Newport Beach. This code is run in a double nested for loop to test all different lane configurations, or you can index within this code itself. Use CVX SDTP solver, available for download at <http://cvxr.com/cvx/download/>

```
function [k,P,T,v] = optimize(l_a,l_b)
    % scenario based
    % l_a: lanes for ped on road A
    % l_b: lanes for ped on road B

    % road A is Balboa Blvd
    % road B is Newport Blvd
    % dummy variables/parameters

    q_a_veh = 0.037; % veh/sec
    q_b_veh = 0.034; % veh/sec

    q_ped = 1600/3600; % ped/sec

    psg = 5; % ren/veh

    % population in 4 shapes
    aug = 4.18 % average
    P1 = 865/2;
    P2 = 1235/2;
    P3 = 16900/2;
    P4 = 70490/2;
    % P1 = 865;
    % P2 = 1235;
    % P3 = 16900;
    % P4 = 70490;

    V_on = 16900; % veh

    Q_a_ped = l_a*q_ped; % ped flow on A
    Q_b_ped = l_b*q_ped; % ped flow on B

    T_evac = 4*3600; %3 hours

    % Optimization
    cvx_begin
        variables k(1) T_a_veh(1) T_a_ped(1) T_b_veh(1) T_b_ped(1) P_a_ped(1) P_a_veh(1) P_a_out(1)
        P_b_ped(1) P_b_veh(1) P_b_out(1)
```

```

% k: percentage of people from Zone 4 to exit via Balboa Island
% ground truth is 0.3555555

maximize((4-l_a)*q_a_veh*T_a_veh + (4-l_b)*q_b_veh*T_b_veh)
subject to

    % road A
    P_a_ped == l_a*Q_a_ped*T_a_ped; % people get out in ped on A
    P_a_veh == psg*(4-l_a)*q_a_veh*T_a_veh; % people get out in veh on A
    P_a_out == P_a_veh+P_a_ped;

    % road B
    P_b_ped == l_b*Q_b_ped*T_b_ped; % people get out in ped on B
    P_b_veh == psg*(4-l_b)*q_b_veh*T_b_veh; % people get out in veh on B
    P_b_out == P_b_veh+P_b_ped;

    % in flow = out flow, every one out!
    P_a_out == P4*k + P1;
    P_b_out == P4*(1-k) + P2 + P3;

    % time within 4 hours
    T_a_veh <= T_evac;
    T_a_ped <= T_evac;
    T_b_veh <= T_evac;
    T_b_ped <= T_evac;
    T_a_veh >= 0;
    T_a_ped >= 0;
    T_b_veh >= 0;
    T_b_ped >= 0;

    % count vehicles
    (4-l_a)*q_a_veh*T_a_veh + (4-l_b)*q_b_veh*T_b_veh <= V_on; % cannot use more cars

    % split factor
    k >= 0;
    k <= 1;
cvx_end

P = [P_a_ped; P_a_veh; P_a_out; P_b_ped; P_b_veh; P_b_out];
T = [T_a_veh; T_a_ped; T_b_veh; T_b_ped];
v = (4-l_a)*q_a_veh*T_a_veh + (4-l_b)*q_b_veh*T_b_veh;
return
end

```

Data Analysis Code (Average Speed, Path Splitting)

```
import numpy as np
import pandas as pd
import geopandas as gpd
import matplotlib
import matplotlib.pyplot as plt
from shapely.geometry import Polygon, Point, LineString
from datetime import datetime

## Read shapefiles into variables

# Read shape files into geopandas
right_half_tri_SF = gpd.read_file("shapefiles/right_half_triangle/right_half_triangle.shp")[['Name', 'geometry']]
left_half_tri_SF = gpd.read_file("shapefiles/left_half_triangle/left_half_triangle.shp")[['Name', 'geometry']]
island_SF = gpd.read_file("shapefiles/island/island.shp")[['Name', 'geometry']]
newport_SF = gpd.read_file("shapefiles/inundation_map/inundation_map.shp")[['Name', 'geometry']]
# Read Street Shape Files
balboa_bld = gpd.read_file("shapefiles/Streets/Balboa/Balboa_Bld-polygon.shp")[['Name', 'geometry']]
w_balboa_bld = gpd.read_file("shapefiles/Streets/W_Balboa/W_Balboa-polygon.shp")[['Name', 'geometry']]
# Read Street Shape Files
balboa_bld = gpd.read_file("shapefiles/Streets/Balboa/Balboa_Bld-polygon.shp")[['Name', 'geometry']]
w_balboa_bld = gpd.read_file("shapefiles/Streets/W_Balboa/W_Balboa-polygon.shp")[['Name', 'geometry']]
newport = gpd.read_file("shapefiles/Streets/Newport/Newport_Bld-polygon.shp")[['Name', 'geometry']]

section_shapefile_list = [right_half_tri_SF, left_half_tri_SF, island_SF, newport_SF]
street_shapefile_list = [balboa_bld, w_balboa_bld, newport]

### Read data for each day

# Assign data for each day
d_1_07012018 = 'Data/Probe_Data/2018_07_01_NewportBeach_basic_probe.csv'
d_2_07022018 = 'Data/Probe_Data/2018_07_02_NewportBeach_basic_probe.csv'
d_3_07032018 = 'Data/Probe_Data/2018_07_03_NewportBeach_basic_probe.csv'
d_4_07042018 = 'Data/Probe_Data/2018_07_04_NewportBeach_basic_probe.csv'
d_5_07052018 = 'Data/Probe_Data/2018_07_05_NewportBeach_basic_probe.csv'
d_6_07062018 = 'Data/Probe_Data/2018_07_06_NewportBeach_basic_probe.csv'
d_7_07072018 = 'Data/Probe_Data/2018_07_07_NewportBeach_basic_probe.csv'

data = pd.read_csv(d_1_07012018,
                   names = ["PROBE_ID", "SAMPLE_DATE", "LAT", "LONG", "HEADING", \
                   "SPEED", "PROBE_DATA_PROVIDER", "X", "Y", "LOCAL_TIME"])
# This now sorts in date order
data.sort_values(by='SAMPLE_DATE', inplace=True, ascending=True)
data.head(3)
```

```

## Analyze Data

class Analyze_Data:

    def __init__(self, data_df, section_shapefile_list, street_shapefile_list):
        # Reading data dataframe into class
        self.data = data_df

        # Read shapefiles into class as lists
        self.section = section_shapefile_list
        self.street = street_shapefile_list

    def plot(Long, Lat):
        # Plot Data
        plt.figure(figsize = (12,8))
        plt.scatter(Long, Lat, s = 0.5, c = 'k')
        plt.show()

    def create_point(self, row):
        # Helper function for turning coordinates into shapely points
        return Point(row['LONG'], row['LAT'])

    def append_points(self):

        self.data['coord'] = self.data.apply(self.create_point, axis=1)
        self.data['section'] = 'N/A'

        return self.data

    def get_section(self):
        # For separating data into sections based on defined shapefiles

        self.append_points()

        section_list = []

        for i in self.data.index.values:
            if self.section[0]['geometry'][0].contains(self.data['coord'][i]):
                section_list.append('right_half_tri')
            elif self.section[1]['geometry'][0].contains(self.data['coord'][i]):
                section_list.append('left_half_tri')
            elif self.section[2]['geometry'][0].contains(self.data['coord'][i]):
                section_list.append('island')
            elif self.section[3]['geometry'][0].contains(self.data['coord'][i]):
                section_list.append('4')
            else:
                section_list.append('N/A')

```

```

        self.data['section'] = section_list

    return self.data

    def get_street(self):
        # For separating data into streets based on defined shapefiles

        self.get_section()
        street_list = []

        for i in self.data.index.values:
            if self.street[0]['geometry'][0].contains(self.data['coord'][i]):
                street_list.append('balboa')
            elif self.street[1]['geometry'][0].contains(self.data['coord'][i]):
                street_list.append('w_balboa')
            elif self.street[2]['geometry'][0].contains(self.data['coord'][i]):
                street_list.append('newport')
            else:
                street_list.append('N/A')

        self.data['street'] = street_list

    return self.data

    def clean_data(self):
        # Remove useless columns and set timezone
        self.get_street()
        self.data_cleaned = self.data[self.data.section != 'N/A'][['PROBE_ID','LAT','LONG',
        'HEADING',
        'SPEED',
        'LOCAL_TIME',
        'PROBE_DATA_PROVIDER',
        'coord',
        'section','street']]

        self.data_cleaned['LOCAL_TIME'] = pd.to_datetime(self.data_cleaned['LOCAL_TIME'])
        .dt.tz_localize('UTC').dt.tz_convert('America/Los_Angeles')
        # Separate to hours
        self.data_cleaned['HOUR'] = self.data_cleaned['LOCAL_TIME'].dt.hour

    return self.data_cleaned

    def clean_data_add_heading(self):
        # Add heading information per probe
        self.clean_data()

        heading_fixed = []

        for i in self.data_cleaned.index.values:
            if self.data_cleaned['HEADING'][i] >180:

```

```

heading_fixed.append(-(360 - self.data_cleaned['HEADING'][i]))
else:
    heading_fixed.append(self.data_cleaned['HEADING'][i])

self.data_cleaned['HEADING_FIXED'] = heading_fixed

return self.data_cleaned

def probe_count(self, section):
    # For counting number of unique probes
    self.clean_data_add_heading()
    uniq_probe = self.data_cleaned.loc[self.data_cleaned['section'] == section].PROBE_ID.unique()

    return len(uniq_probe)

class road_specific_analysis():

    def __init__(self, data_cleaned, street_choice):
        self.data = data_cleaned
        self.street = street_choice

    def road_df(self):
        # Filter and create dataframe based on road choice
        self.data
        self.road = self.data.loc[self.data['street'] == self.street].copy()

    return self.road

    def get_heading(self, df, i, lower_head, upper_head):

        # Helper function for differentiating heading
        head_df = df.loc[(df['HOUR'] == i) & ((df['HEADING_FIXED'] >= lower_head) | (df['HEADING_FIXED'] <= upper_head))][['PROBE_ID', 'SPEED']]

        # This removes all 0 speed instances
        head_df = head_df[head_df['SPEED'] != 0]
        unique_probe_list = list(set(head_df['PROBE_ID']))

    return head_df, unique_probe_list

    def get_heading_info(self, limits, speed_limit, trans_type):

        """
        Inputs:
        df - cleaned up dataframe with hours and streets
        limits - list of heading limits, [lower, higher]
        trans_type - looking for 'vehicle' or 'pedestrian'
        """

```

Outputs:

head_speed - list of 24 north heading speeds averaged over an hour
head_unique_probe - list of 24 counts of unique probe ids in that hour - vehicles
"

```

self.road_df()

head_speed = []
head_unique_probe = []

for i in range(24):
    # Analyze for 24 hours
    head_df, unique_probe_list = self.get_heading(self.road, i, limits[0], limits[1])

    count = 0
    speed = []

    for j in unique_probe_list:

        probe_df = head_df.loc[head_df['PROBE_ID'] == j]
        mean_speed = np.mean(probe_df['SPEED'])

        if trans_type == 'Vehicle':

            if mean_speed >= speed_limit:
                count += 1
                speed.append(mean_speed)

        elif trans_type == 'Pedestrian':

            if mean_speed <= speed_limit:
                count += 1
                speed.append(mean_speed)

    head_unique_probe.append(count)

    if not speed:
        head_speed.append(0)
    else:
        head_speed.append(np.mean(speed))

return head_speed, head_unique_probe

## For getting trajectory

traj = Analyze_Data(data, section_shapefile_list, street_shapefile_list)

```

```

traj_df = traj.clean_data_add_heading()
traj_df.head()

uniq_ID = set(traj_df.PROBE_ID)

# Create empty dataframe
columns = ['PROBE_ID','Route_Num','Start_Section','End_Street','Time_Start','Time_End']
traj_route_df = pd.DataFrame(columns=columns)

for ID in uniq_ID:

    probe_df = traj_df.loc[traj_df.PROBE_ID == ID]

    route = [probe_df.section.iloc[0]]
    route_count = 0
    route_time = [probe_df.LOCAL_TIME.iloc[0]]

    for i in range(len(probe_df)-1):

        if (probe_df.LOCAL_TIME[i+1] - probe_df.LOCAL_TIME[i])<pd.Timedelta(minutes = 1):

            route.append(probe_df.street.iloc[i+1])
            route_time.append(probe_df.LOCAL_TIME.iloc[i+1])

        else:
            route_count += 1
            traj_route_df = traj_route_df.append({'PROBE_ID':ID,'Route_Num':route_count,
            'Start_Section':route[0],'End_Street':route[-1],
            'Time_Start':route_time[0],'Time_End':route_time[-1]}, ignore_index=True)

    traj_rest = traj_route_df.loc[traj_route_df['Start_Section'] == 'rest_of_new_port']

    balboa = traj_rest.loc[traj_rest['End_Street'] == 'balboa']
    newport = traj_rest.loc[traj_rest['End_Street'] == 'newport']

    balboa = balboa.drop_duplicates(subset = ['Time_Start'],keep=False)
    newport = newport.drop_duplicates(subset = ['Time_Start'],keep=False)

    print('Number of Cars going to Balboa Blvd:', len(balboa))
    print('Number of Cars going to Newport Blvd:', len(newport))

# ## Notes
# #### For unique probe count
# #### Section Options:
# - 'right_half_tri'
# - 'left_half_tri'
# - 'island'
# - 'rest_of_new_port'

```

```

#
# ### Street Options
# - 'balboa'
# - 'newport'
#
# ### Blanket sorting for all data

plt.figure(figsize = (12,8))
plt.scatter(data.LONG, data.LAT, s = 0.5, c = 'k')
plt.show()

data.shape

analysis = Analyze_Data(data, section_shapefile_list, street_shapefile_list)
data_cleaned = analysis.clean_data_add_heading()

street_plot = data_cleaned.loc[data_cleaned.street != 'N/A']

plt.figure(figsize = (12,8))
plt.scatter(data_cleaned.LONG, data_cleaned.LAT, s = 0.5, c = 'k')
plt.show()

plt.figure(figsize = (12,8))
plt.scatter(street_plot.LONG, street_plot.LAT, s = 0.5, c = 'k')
plt.show()

data_cleaned[['PROBE_ID','LAT','LONG','HEADING','SPEED','LOCAL_TIME','section','street','HEADING']].head()

# ### Analysis of data based on road and transportation type

# Balboa
bal_north_limits = [-90,45]
bal_south_limits = [145,-150]
# Newport
newport_north_limits = [-140,0]
newport_south_limits = [90,-160]

speed_limit = 15

# For Balboa
analysis_balboa = road_specific_analysis(data_cleaned,'balboa')

# Northbound

```

```

veh_balboa_N_speed, veh_balboa_N_probe = analysis_balboa.
get_heading_info(bal_north_limits, speed_limit, "Vehicle")

ped_balboa_N_speed, ped_balboa_N_probe = analysis_balboa.
get_heading_info(bal_north_limits, speed_limit, "Pedestrian")

# Southbound

veh_balboa_S_speed, veh_balboa_S_probe = analysis_balboa.
get_heading_info(bal_south_limits, speed_limit, "Vehicle")

ped_balboa_S_speed, ped_balboa_S_probe = analysis_balboa.
get_heading_info(bal_south_limits, speed_limit, "Pedestrian")

# For Newport
analysis_newport = road_specific_analysis(data_cleaned,'newport')

# Northbound
veh_newport_N_speed, veh_newport_N_probe = analysis_newport.
get_heading_info(newport_north_limits, speed_limit, "Vehicle")

ped_newport_N_speed, ped_newport_N_probe = analysis_newport.
get_heading_info(newport_north_limits, speed_limit, "Pedestrian")

# Southbound

veh_newport_S_speed, veh_newport_S_probe = analysis_newport.
get_heading_info(newport_south_limits, speed_limit, "Vehicle")

ped_newport_S_speed, ped_newport_S_probe = analysis_newport.
get_heading_info(newport_south_limits, speed_limit, "Pedestrian")

# Create a dataframe of road information per hour
road_info_df = pd.DataFrame({'Hour':range(0,24),
'Newport_North_Speed_Veh':veh_newport_N_speed,
'Newport_South_Speed_Veh':veh_newport_S_speed,
'Newport_North_Probe_Veh':veh_newport_N_probe,
'Newport_South_Probe_Veh':veh_newport_S_probe,
'Newport_North_Speed_Ped':ped_newport_N_speed,
'Newport_South_Speed_Ped':ped_newport_S_speed,

```

```

'Newport_North_Probe_Ped':ped_newport_N_probe,
'Newport_South_Probe_Ped':ped_newport_S_probe,
    'Balboa_North_Speed_Veh':veh_balboa_N_speed,\n
    'Balboa_South_Speed_Veh':veh_balboa_S_speed,\n
    'Balboa_North_Probe_Veh':veh_balboa_N_probe,\n
    'Balboa_South_Probe_Veh':veh_balboa_S_probe,\n
    'Balboa_North_Speed_Ped':ped_balboa_N_speed,\n
    'Balboa_South_Speed_Ped':ped_balboa_S_speed,\n
    'Balboa_North_Probe_Ped':ped_balboa_N_probe,\n
    'Balboa_South_Probe_Ped':ped_balboa_S_probe}\n\n

road_info_df.head()\n\n

def find_slowest(df, column, probe):\n\n
    slow = []\n
    for i in df[column]:\n
        if i>0:\n
            slow.append(i)\n
    slowest_speed = min(slow)\n
    slowest_df = df.loc[df[column]==slowest_speed][['Hour', column, probe]]\n\n
    return slowest_df\n\n

N_N_slow = find_slowest(road_info_df,'Newport_North_Speed_Veh','Newport_North_Probe_Veh')
S_N_slow = find_slowest(road_info_df,'Newport_South_Speed_Veh','Newport_South_Probe_Veh')\n\n

N_B_slow = find_slowest(road_info_df,'Balboa_North_Speed_Veh','Balboa_North_Probe_Veh')
S_B_slow = find_slowest(road_info_df,'Balboa_South_Speed_Veh','Balboa_South_Probe_Veh')\n\n

N_N_slow\n\n

S_N_slow\n\n

N_B_slow\n\n

S_B_slow

```

Comparative Analysis of Data-Driven Modeling for HVAC Systems

Teng Zeng, Aaron Kandel, Damon Zhang, Xiyuan Zhao

May 10, 2019

1 Abstract

In this project, we obtain relevant data from a simulated building HVAC system and use that data to explore several varying modeling approaches for the underlying HVAC system. The building sector accounts for more than 40% of the global energy consumption and 30% of carbon emissions, primarily attributed to a building's heating, ventilation, and air conditioning (HVAC) system. As a result, modeling of dynamics in building HVAC systems comprises a significant area of inquiry in relevant controls and modeling literature. We specifically use an HVAC system as a simple case study to model temperature dynamics using (1) ordinary least squares with and without Lasso regression, and (2) a neural network. After presenting results related to each of our modeling approaches, we also briefly explore some control strategies which we could implement using our varying data-driven models.

2 Introduction

2.1 Motivation and Background

Buildings are responsible for about 30–40% of global energy demand. At the same time, humans spend up to 80–90% of the time in the building. Reducing energy demand through optimal operation is the subject of building control research, while human satisfaction in buildings is studied in the thermal comfort community. Thus, balancing the two is necessary for a sustainable and comfortable building stock. The biggest challenges associated with HVAC systems are maintaining the balance between interior comfort and energy expenditure. To address this balance, accurate and control-oriented dynamical system models are absolutely necessary. This serves as the motivation for our exploration of varying modeling approaches for an HVAC system. Furthermore, our combined expertise as a team spans system dynamics, modeling and control, and various energy systems application (including HVAC systems). Thus, this study can allow us to leverage our existing expertise to delve deeper into exploring a specific energy system using the tools we learning from past research and from CE295.

2.2 Relevant Literature

It is known that building control focuses predominantly on energy savings rather than incorporating results from thermal comfort these decades, especially when it comes to occupant satisfaction. In [1], the current situation of lacking overlap in the thermal comfort and building control research area has been stated, and potential research directions in terms of bridging the two fields by balancing human requirements (comfort, satisfaction) and with energy conservation goals have been identified.

For other ways of taking thermal comfort of occupancy into consideration when design the building control system, in [2], Peng et al. utilize a K-nearest neighbor (KNN) to predict the future occupant behavior to reduce energy consumption, as poor anticipation of occupant behavior has been found to increase building energy consumption by a third. Specifically, based on an occupancy analysis, a learning-based demand-driven control strategy is proposed for sensible cooling. It predicts occupants' next presence and the presence duration of the remainder of a day by learning their behavior in the past and current days, and then the predicted occupancy information is employed indirectly to infer setback temperature setpoints according to rules in this paper. We can learn from the methodology used in this paper undoubtedly, however, it still have some limitations like not highly accurate and lack of real-time thermal comfort feed-backs from occupancy.

A simple way to implement HVAC system control is to use the reactive model-free control (PID). In [3], a self-tuning PID-type fuzzy adaptive control for an expert HVAC system was stated. The modelling, numerical simulation and intelligent control of an expert HVAC (heating, ventilating and air-conditioning) system having two different zones with variable flow-rate were performed by considering the ambient temperature in this study. This is similar from the methods used in our project. In this paper, all models of the variable flow-rate HVAC system were generated by using MATLAB/SIMULINK, and proportional-integral-derivative (PID) parameters were obtained by using Fuzzy sets. For thermal comfort, the temperatures of the two different zones were decreased by 5C with respect to the ambient temperature. The successful results were obtained by applying self-tuning proportional-integral-derivative (PID)-type fuzzy adaptive controller if comparing with the fuzzy PD-type and the classical PID controller. We can definitely learn about how to do PID control of the HVAC system from this study.

A diverse body of existing work exists which seeks to address the problems of energy and comfort optimal building control. In [4], the authors stated that whole building energy model (BEM) is difficult to be used within the classical model-based optimal control framework. The stated reasons were because of the high-dimensionality and expensive computation time. They applied a novel deep reinforcement learning (DRL) based learning framework to circumvent the issue and applied it to a real office building in Pennsylvania. The optimal control policy reaches 15% of heating energy saving. In [5], the authors applied model-free Q-learning that makes optimal control decisions for HVAC system and window systems to minimize both the energy consumption and thermal discomfort. The advantage of a model-free approach is that RL will find an optimal action policy without requiring a model of the building system. They compared between the general heuristic control method and their proposed Q-learning method and concluded that the RL approach showed superior results for maintaining indoor temperature with comfort range while keeping the operation

hours of the HVAC system fewer than the heuristic one.

2.3 Focus of this Study

We use the HVAC system case study to explore different modeling approaches discussed in class. Specifically, we evaluate ordinary least-squares, Lasso regression, and neural networks for modeling HVAC temperature dynamics. We also apply a PID controller to a commercial building system to track a temperature setpoint to maintain thermal comfort.

2.4 Motivating Example

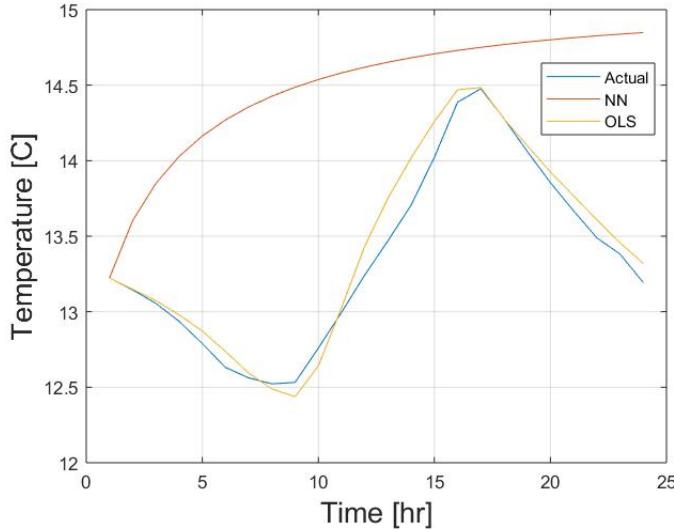


Figure 1: Predicted results from two different building modeling techniques (NN and OLS) against the actual values.

In this example, we have a nonlinear model, namely the feed-forward neural network model and we also have a linear model, the OLS method. The data format that feeds into the OLS model is described in Section 3.2.2 and the data that feeds into the NN model are merely the two state variables, air temperature and relative humidity. Thus, as reflected in the Figure 1, it is immediately apparent that the prediction results from our OLS model strongly out perform those from our NN model (evaluated on one day of data). This is, however, by design. Given data of poor quality, even highly nonlinear modeling architectures like neural networks can fail to accurately describe the underlying dynamics. Thus, the quality and breath of the data is by far the most significant question to address when formulating a data-driven model, regardless of what specific modeling approach one chooses to take.

3 Technical Description

3.1 Building Energy Model (BEM)

To explore different modeling approaches for temperature dynamics in an HVAC system, we need a set of training and validation data of building energy use dynamics. We can use this data to develop models in software like Matlab or Python. Specifically, if we make our outputs of the data-driven model the indoor air temperature and humidity of a certain thermal zone of next time period, then our inputs will be current indoor temperature and humidity of the thermal zone of the building, and also a series of parameters such as the heating and cooling load generated by building energy plant (boiler and chiller) and several internal gains which may influence the temperature dynamics of the building HVAC system. We will discuss in the following subsections about the data analysis methods we used to discover which parameters and disturbances imposed the greatest effect on the overall HVAC temperature dynamics.

In the BEM software (EnergyPlus), by setting inputs related to the BEM in five main parts (Activity, Construction, Openings, Lighting and HVAC), then by running the building energy simulation, we can generate a range of environmental performance data such as: energy consumption, carbon emissions, comfort conditions, daylight illuminance, maximum summertime temperatures and HVAC component sizes. Specifically, we can get data about:

1. heat gains produced through all kinds of building envelope and ventilation;
2. internal gains produced by occupancy, all kinds of devices as well as the solar gains;
3. latent gains which related to the dynamics of relative humidity (RH);
4. environmental related temperatures (such as indoor air temperature, radiant temperature, and operative temperature);
5. comfort matrix (such as Predicted Mean Vote (PMV), discomfort hours and RH);
6. system loads (such as heater and chiller load);
7. fuel types and the amount used by different building systems;
8. carbon emissions;
9. building surface data which can be used to do computational fluid dynamics (CFD) analysis;
10. site weather data such as outside dry-bulb temperature, dew-point temperature, atmospheric pressure and wind and solar related data.

For the building HVAC control purpose, we only need to focus on the parameters that could have relative significant influence on the building temperature dynamics and relative humidity (RH). We specifically selected 11 kinds of data among all those generated data by running annual energy simulations for the building energy model. Specifically, indoor air temperature, relative humidity, zone heating, chiller load, radiant temperature, outside dry-bulb temperature, outside dew-point temperature, general lighting load, computer and equipment load, occupancy load, solar gains through exterior windows. We export those 11 set of data mentioned above using csv files, allowing us to work with it quantitatively using Matlab. We specifically organize the BEM output to fit our data-driven models.

The actual building model we built has three floors (first floor: hall; second floor: gym; third floor: office) and each floor is a single thermal zone. For the consideration of efficiency and accuracy, we started from doing analysis and designing data-driven model using the first floor's hall (one thermal zone). Also, to facilitate our data-driven model design, we chose the data of one typical day in winter (Feb 1st) at the beginning. However, we encountered some issues throughout this first design stage due to an insufficiently rich data. We resolved this issue by redesigning our BEM simulations, and simulating a years worth of HVAC data

for a range of 10 different lower and upper heating and cooling temperature setpoints. This range of simulation conditions gave us better representation of the overall state and input space in our data, allowing us to fit more accurate data-driven models.

3.2 Data-driven Modeling

3.2.1 Ordinary Least Squares (OLS)

We apply ordinary least-squares (OLS) as a simple, baseline model given its simplicity and proven efficacy. Based on our knowledge of the heat transfer phenomena, we choose to use 11 variables from the outputs of the simulation software. The variables are divided into two state variables, two input variables, and seven disturbance variables, as shown in Table 1.

Table 1. Classification of the simulation output

Simulation Output	Classification
Air Temperature (T) Relative Humidity (RH)	State Variable ($\mathbf{x} \in \mathbb{R}^{n=2}$)
Zone Heating Chiller Load	Input Variable ($\mathbf{u} \in \mathbb{R}^{m=2}$)
Radiant Temperature Outside Dry-Bulb Temperature Outside Dew-Point Temperature General Lighting Load Computer + Equipment Load Occupancy Load Solar Gains Exterior Windows	Disturbance Variable ($\mathbf{W} \in \mathbb{R}^{d=7}$)

Based on the governing equation: $Y = \Phi\theta$, we can populate the Y and Φ matrices such that

$$Y = \begin{bmatrix} T(k+1) \\ T(k+2) \\ \vdots \\ \vdots \\ T(N) \end{bmatrix} \quad \Phi = \begin{bmatrix} T(k) & RH(k) & \cdot & \cdot & \cdot \\ T(k+1) & RH(k+1) & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ T(N-1) & RH(N-1) & \cdot & \cdot & \cdot \end{bmatrix} \quad \theta = \begin{bmatrix} a_1 \\ \cdot \\ a_n \\ b_1 \\ \cdot \\ b_m \\ b_{w_1} \\ \cdot \\ b_{w_d} \end{bmatrix}$$

Then we can analytically obtain the set of parameters θ which minimize the sum of square errors between the data and our entirely linear model $\theta^* = (\Phi^T\Phi)^{-1}\Phi^TY$.

3.2.2 The Cross Term Matrix

To account non-linear interactions between the states, disturbances, and inputs, we evaluate a model with squared terms of each variable and the cross products between them such that

$$\Phi = \begin{bmatrix} x_1(k) & \dots & x_n(k) & x_1^2(k) & \dots & x_n^2(k) & x_1(k)x_2(k) & \dots & x_{n-1}(k)x_n(k) \\ \vdots & & \vdots & \vdots & & \vdots & \vdots & & \vdots \\ x_1(N-1) & \dots & x_n(N-1) & x_1^2(N-1) & \dots & x_n^2(N-1) & x_1(N-1)x_2(N-1) & \dots & x_{n-1}(N-1)x_n(N-1) \end{bmatrix}$$

This yields a model which is linear in the parameters, but nonlinear in the states, disturbances, and inputs.

3.2.3 LASSO Regression

We hypothesize the model accuracy will increase due to the added squared terms and the cross products, but we do not know which terms in the Φ matrix possess the greatest importance. To compare the relative impact of each term, we utilize LASSO regression with the governing equation:

$$\min_{\theta} : \|\Phi\theta - Y\|_2^2 + \lambda\|\theta\|_1 \quad (1)$$

The weighted \mathcal{L}_1 penalty on the parameters encourages sparsity in our model. We can solve the LASSO regression problem for a range of weights λ . By comparing which parameters are set to zero as λ increases, we can reveal which parameters and terms in our regressor matrix impose the greatest effect on the overall model accuracy.

3.3 Data-driven Model - Neural Network (NN)

To more effectively account for non-linearities, we implement a deep neural network (NN) to serve as the benchmark for our model accuracy. Figure 2 illustrates a simplified example of a neural network structure. The variable x_i 's are the input data that we provide to the model. During the training stage, variable y is the output data that we provide but during the testing/regression stage, it's then the predicted system output. The circles are called "neurons" and the lines denoted by ω_i 's are the weights. The neural network implements a "back-propagation" method for error propagation back to each neurons and updating the weights over the neurons. As the training error converges to zero or stabilizes to a certain point, we conclude the training process. We've then acquired a model of the HVAC system.

We fit a neural network with 5 hidden layers each with 10 neurons. The input to the model are all state variables, input variables, and disturbance variables. The output of the network is a prediction of T at the next timestep. After we train our model, we simulate its performance and compare to the output of the BEM software.

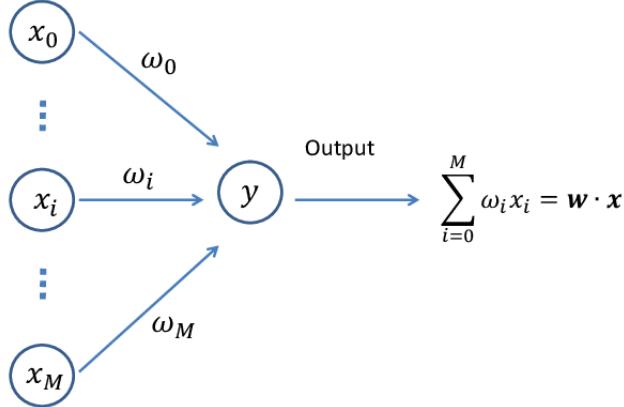


Figure 2: One layer neural network

4 Modeling Results and Analysis

In general, we have to be careful about the modeling techniques we apply. Unfortunately there is no rule-of-thumb guideline for model selections, neither is there a universal optimal model that works for all scenario settings. We will motivate with the following example. [Note: the input data are the same across different models except slight variance in the below section 2.4. One year of data is used to evaluate the effectiveness of the model.]

4.1 Preliminary Models

First, we start with the modeling techniques that we've learned and implemented from class, namely a neural network and OLS, in Figure 3. These are the models with the same input data (all variables). The OLS model includes the nonlinear cross terms in this case. Visually they predict the temperature very well, except at times in between that they aren't able to capture the extreme variance.

The error metric implemented to evaluate the effectiveness of the model is mean square error (MSE), where given values Y_i , $i = 1 \dots n$ and comparing values \hat{Y}_i , $i = 1 \dots n$, MSE is evaluated as

$$\text{MSE} = \frac{1}{n} \sum_{i=1}^n (Y_i - \hat{Y}_i)^2.$$

Therefore, the MSE for NN and OLS are in the same order of magnitude, 0.000807 and 0.000194 respectively.

The OLS model outperforms the NN. This could potentially be caused by over-fitting. As shown in Figure.4a and Figure.4b, the neural network has a tendency to overfit to certain parts of the data. So, while it models that data much more effectively, it fails to generalize to data not seen in its training set.

4.2 LASSO Approach

Since it is quite computation expensive to train a NN model, meanwhile we are also very interested in determining the order of importance for different features (variables), we decide

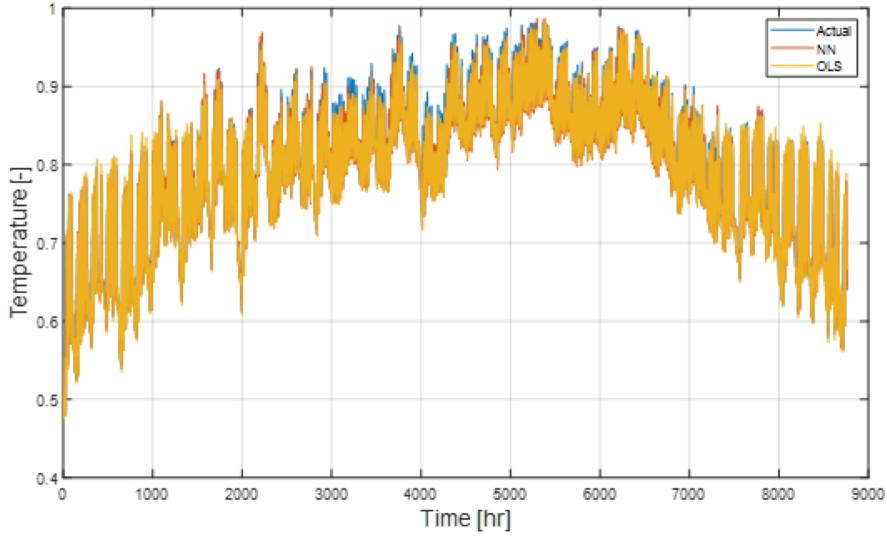


Figure 3: NN and OLS vs. Actual Value

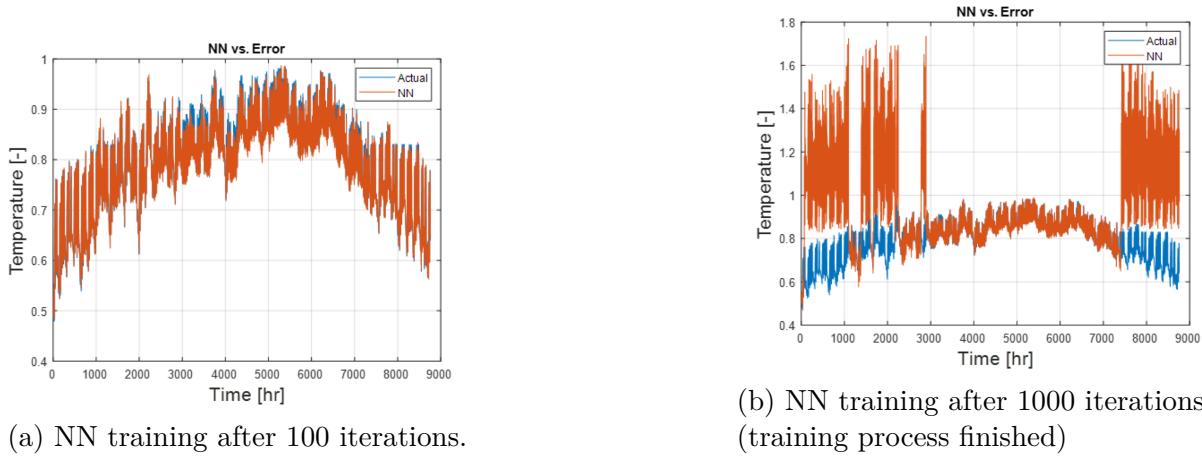


Figure 4: NN Over-fitting

to incorporate LASSO regression method. The methodology is precisely described in Section 3.2.3. As a result, the top three variables we've identified contributing the most to prediction are air temperature, general lighting load, and outside dew-point temperature (Figure 5, numbers in '(.)' are variable indices). Temperature variables are within our expectations but the lighting load is a variable interesting to be discovered the importance.

Further more, we have also investigated the LASSO penalty variable value and error trade off. As the curve in Figure 6 indicates, after λ value reaches a certain point, the model MSE starts to shoot up. Therefore, it's constructive as the curve indicates, we shall only look at λ values that are below such threshold ($\lambda = 0.000016$).

We implement the model with several λ values and align the MSE values with the other preliminary results in Figure 9 ('(.)' after λ indicate indices, the larger the number the larger λ). As the MSE results reflect, when comparing LASSO against NN model, we are trading

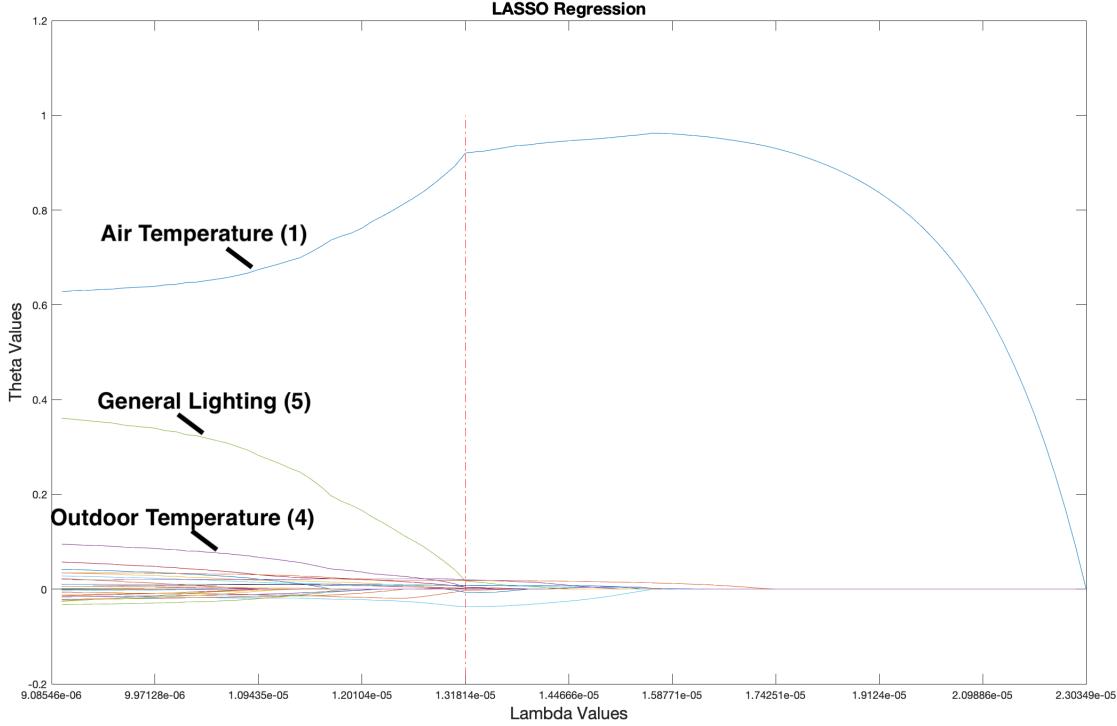


Figure 5: LASSO Regression

accuracy for both complexity and model clarity; whereas LASSO against OLS, it's accuracy versus complexity (MSE of LASSO not on the same order of magnitude to NN and OLS). Interesting to note that in almost all scenarios, OLS actually work the best in terms of MSE. Therefore, when model complexity is not an issue for modeling, OLS is a straightforward and accurate methods to first implement. With special attention to over-fitting (cross-validation method could help), we could expect accurate results from NN as well.

5 HVAC Control

The varied modeling approaches we explore allow us to take several differing approaches to HVAC control.

5.1 Finite Time Optimal Control

We first conduct finite time optimal control using our model and a setpoint of $T_{set} = 20.0$ degree C. Figure 8 shows our optimal control results for a horizon of 12 hours, with a loss function of:

$$J = \sum_{k=1}^N (T_k - T_{set})^2 \quad (2)$$

These results show that our control algorithm takes the temperature to the setpoint as fast as possible, and then consistently applies heat to keep the temperature at the setpoint.

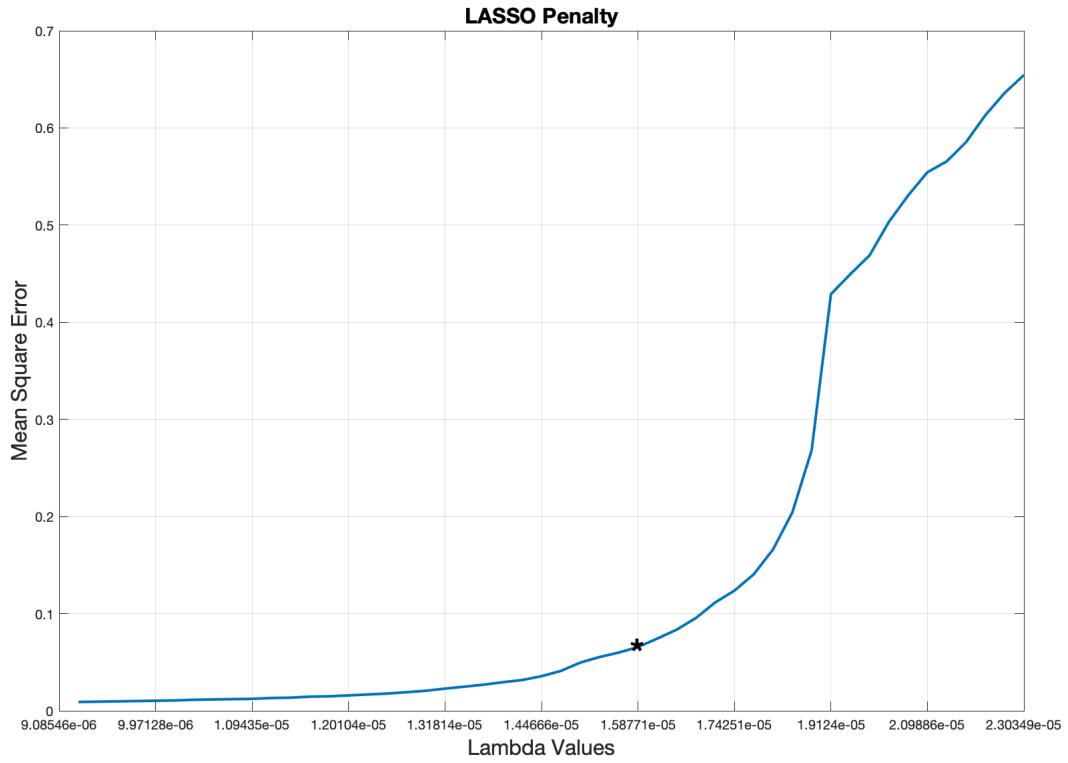


Figure 6: LASSO λ and MSE Trade-off

Models	MSE (Normalized)
Ordinary Least Square	0.000194
Nonlinear Parameters (LASSO, $\lambda = 3.421\text{e-}03$ (40))	0.429
Nonlinear Parameters (LASSO, $\lambda = 1.3492\text{e-}04$ (30*))	0.0656
Nonlinear Parameters (LASSO, $\lambda = 9.9713\text{e-}06$ (2))	0.00941
Deep Neural Network	0.000807

Figure 7: MSE Across Various Models

5.2 PI Control

Our second control analysis is to implement a proportional-integral controller (PI controller) to track a desired temperature setpoint (22 deg C). Figure 9 shows our results from

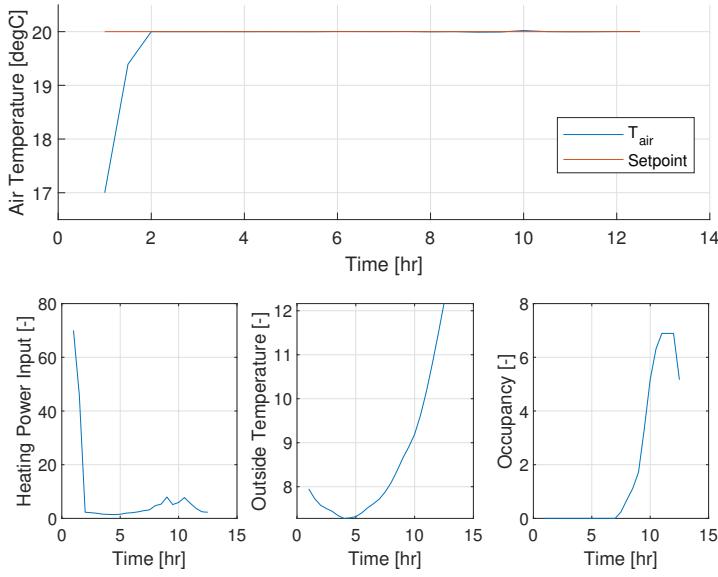


Figure 8: Finite-Time Optimal Control using Black-Box Model

this analysis for 10 days of data using an underlying ordinary least-squares model which is linear in the features and parameters. The proportional gain $k_p = 0.25$ and the integral gain $k_i = 0.5$. These results show that a simple reactive model-free approach is sufficient to

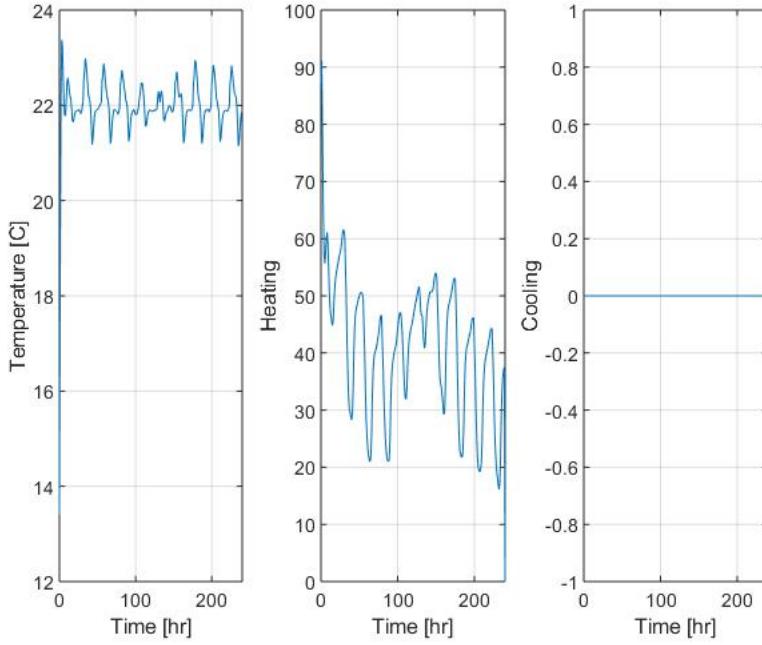


Figure 9: MSE Across Various Models

track a desired temperature setpoint. However, in our case there is some limited oscillation

around the setpoint. We find the large timestep of 1 hour to be one cause for this issue. With further tuning of the PI gains the oscillation decreases, but at the cost of more temperature overshoot.

5.3 Discussion for Control Framework

Overall, our results show that even with the simplest modeling and control frameworks, an HVAC system can systematically track desired state setpoints. Our project began with us implementing a model-free reinforcement learning controller to track thermal comfort for an HVAC system. However, these results indicate that perhaps for setpoint tracking, a reinforcement learning approach is potentially over complicated and simpler model-free techniques like PID control can sufficiently track desired thermal comfort.

We also thought about implementing a linear quadratic regulator to optimally control the system. However, our modeling analysis showed us that the best models we explored were in some cases nonlinear in the states, disturbances, and control inputs. Even with a purely linear model, the requisite inclusion of disturbances means we would have to develop a forecasting scheme in order to implement LQR with any real efficacy. Since we switched projects a week before our scheduled presentation, we ended up not having sufficient time to explore this option.

6 Conclusion

This report presents an exploration of different data-driven modeling approaches using HVAC temperature control as a case study. As we proceeded with this project throughout the semester, we have experienced a fundamental pivot of our project focus. At the beginning of the semester, we thought it would be interesting to apply a reinforcement learning (RL) algorithm for HVAC control. However, as we delved deeper into the problem context and exchanged ideas with researchers (including advisor Prof. Scott Moura) in related fields, we found that for simple control objectives like maximizing thermal comfort with a temperature setpoint, a reactive model-free approach like PID control could be sufficient for the application. We decided to instead focus on a more fundamental question of exploring different methods for modeling temperature dynamics in a building. In particular we evaluate ordinary least squares with linear and nonlinear terms, Lasso regression with our OLS models, and a feed-forward neural network. We have identified the top three most important features of building modeling, namely air temperature, lighting load, and outside dew-point temperature. If model complexity is not the major concern, our experimentation with neural networks provides the most accurate but sometimes inconsistent results. If computation is a concern, we find that a simple ordinary least-squares model which is linear in the features and parameters possesses a surprisingly high degree of accuracy, and is much more control-oriented.

Our brief exploration of control strategies has lead us to confirm our initial suspicion that a model-free reinforcement learning approach is a complicated solution to a simple problem. We implement a PI controller and show that with a simple OLS model, we can effectively track a temperature setpoint to maintain sufficient thermal comfort.

7 Acknowledgement

We would like to express sincere gratitude to Professor Scott Moura and GSI Dong Zhang for the amazing semester lectures and constructive feedback on our project.

References

- [1] June Young Park and Zoltan Nagy. Comprehensive analysis of the relationship between thermal comfort and building control research - a data-driven literature review. *Renewable and Sustainable Energy Reviews*, 82:2664 – 2679, 2018.
- [2] Yuzhen Peng, Adam Rysanek, Zoltán Nagy, and Arno Schlüter. Occupancy learning-based demand-driven cooling control for office spaces. *Building and Environment*, 122:145 – 160, 2017.
- [3] Servet Soyguder, Mehmet Karakose, and Hasan Alli. Design and simulation of self-tuning pid-type fuzzy adaptive control for an expert hvac system. *Expert Systems with Applications*, 36(3, Part 1):4566 – 4573, 2009.
- [4] Zhiang Zhang, Adrian Chong, Yuqi Pan, Chenlu Zhang, Siliang Lu, and Khee Poh Lam. A deep reinforcement learning approach to using whole building energy model for hvac optimal control. In *2018 Building Performance Analysis Conference and SimBuild*, 2018.
- [5] Yujiao Chen, Leslie K Norford, Holly W Samuelson, and Ali Malkawi. Optimal control of hvac and window systems for natural ventilation through reinforcement learning. *Energy and Buildings*, 169:195–205, 2018.



Final Report

CEE 295, Dept. of Civil Engineering, University of California, Berkeley

May 10, 2019

Optimizing Storage and Usage of Distributed Energy from Individual Households Through Machine Learning

Elias Bougrine¹, Arnaud Dyen², Marc-Antoine Loo¹, Matthieu Meeus², Pierre-Louis Missler¹

¹ Civil and Environmental Engineering, UC Berkeley

² Mechanical Engineering, UC Berkeley

I. Abstract

Our project focuses on the optimization of storage and usage of distributed energy from individual households through Machine Learning in order minimize the costs of electricity. In fact, the price of electricity is volatile, for instance from 9pm to 7 am the electricity is cheaper than from 10 am to 6pm. Therefore, we believe that it is critical for us to optimize the electricity storage and usage in order to decrease the monthly energy bill. It is obvious that, due to its higher price, using less electricity generated from electrical grid and more distributed energy can help you save some money. However, the distributed energy generated is not often enough to fulfill our energy need, and thus we always need some energy from electrical grids. Hence, knowing the cost of energy in real time, our goal is to optimize its usage and storage to minimize the cost.

II. Introduction

II.1. Motivation & Background:

Price reduction has always been a motivating factor for various products and companies across all industries. Specifically, in the market for electricity, optimizing prices could significantly reduce costs for users. Electricity prices change every single minute across the day simply due to the laws of supply and demand, much like a stock in the financial market. Just like there are high-frequency trading softwares to decide whether to buy or sell a stock at precise milliseconds, we want to build a software that will decide for a household whether it is better to store the electricity produced by natural resources (in this case solar panels) and buy electricity from the grid, or use the electricity made by the solar panel since the price from the grid is too high.

This application would not only reduce cost for all its users, but it could also potentially reduce the load variations across the grid. This system would give a live portrayal of costs for each household and match the demand instantaneously with the change in price. By optimizing the combination of price and demand at each instant, our program could potentially reduce the large variation in energy prices we see towards 5pm-6pm. This product could in theory stabilize the price by reducing demand at high prices by using the stored energy of the solar panel and allow the household to buy energy at low price times like 2am-3am. By stabilizing price, this could also reduce load variations in the grid, which in turn would reduce maintenance and damages to the system.

As the paper will outline later, this project will require a lot of live data, including the price fluctuations of electricity, hourly energy demands of households and weather fluctuations in order to know the energy provided by the solar panel to make the right decision. This will bring a lot of challenges to our project, but with the talent of our team, this is feasible. Our team is composed of two Mechanical Engineers with a passion for energy technology. For renewable energy to reach its full potential, it should be entirely optimized from an economic perspective. The rest of the group is composed of three civil engineers who look at the bigger picture of

optimizing the electrical grid. Some share a particular interest in finance, whose volatility and data analysis are comparable to this project. Additionally, all of us are aware of our increasingly data-driven world and wish to expand our knowledge in data science and machine learning. This project allows us to dive deep into these methods and apply it to our field of interest.

II.2. Relevant Literature:

Thanks to the new technologic solutions, the idea of reducing the cost of electricity using solar panels and computational models is recurrent in the literature. The focus on optimizing energy flows and demand scheduling with real-time updates of electricity prices, solar radiation data, and uncontrollable demands of the paper [1] gave us lots of information to find our features and some tools to work on our own models. The type of buildings or place where you implement your study is also very important. We use the study of the Oakland EcoBlock to find what kind of technical solutions are we going to use. In this study [2], the authors optimize the use/storage of energy in an Oakland residential block. There were some limitations but overall it proves to be a good base for modeling energy demand and prices for our project. So, on the strength of these papers [2] and [3], we will push the model of price in demand further in order to build a robust prediction model in order to do live decision making for our system.

According to the paper [5], they consider a photovoltaic system, a battery to store energy and a prediction of the solar radiation to improve the insolation prediction of eLum, a company working on energy systems control using Markov chain, in order to better control their storage system. We learned how important the solar radiation prediction is in order to control energy storage. Therefore, we will refer to this article when our model will be built and ready to use with prediction data.

The article [4] describes the optimization of energy use a particular residential 'smart home' in the bay. The purpose is to minimize the total energy cost, with as 4 parameters: energy demand, energy from PV cell, battery, energy from grid. They compute values for solar generation and energy demand from datasets of CAISO and PG&E for specific household with appropriate assumptions. This is all implemented in a simple linear optimization. For further research they suggest including more factors in the optimization model, such as charging an EV and including a detailed working of the battery. We use this information for our optimization model and we decide to choose one battery for our entire study. We focused on the Powerwall for Tesla and according to the technical documents [6], we develop all the model with its characteristics.

II.3. Focus of this Study

Looking to minimize the cost of electricity for users. This will be achieved by a machine learning optimization model that will decide for the user if it is better to use the renewable energy made by the house or buy from the grid in order to minimize cost.

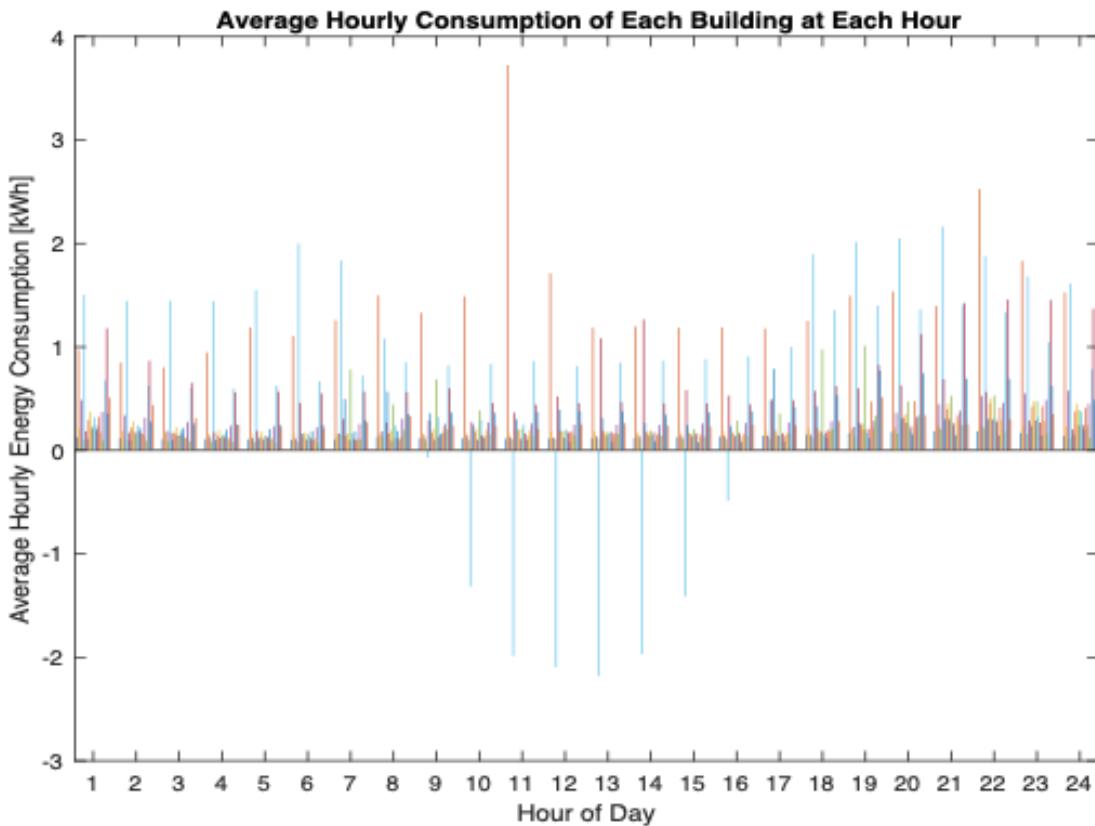
III. Technical Description

III.1. Data Selection and pre-processing

A. Identification of test area

Firstly, it is important to select an area of focus for this project that is both representative to a bigger scale as well as an easy way to get started for the model. Ideas include to start with a specific household or campus building in Berkeley. We worked on a specific household consumption dataset and we also asked for the data consumption of Shires hall at Berkeley and Berkeley High School. However, there is no limitation in the dataset that can be used, since our model can be easily adapted to any dataset. After several tests with data from individual house in Berkeley, we had a lot of problems with small energy values, and most often values of 0\$. This led us to use the data from our class and especially data on energy consumption of the buildings from the HW4.

The following plot represents the data we will use as the Average Hourly Energy Consumption versus the Hour of Day for each of the twenty-three buildings of the dataset.



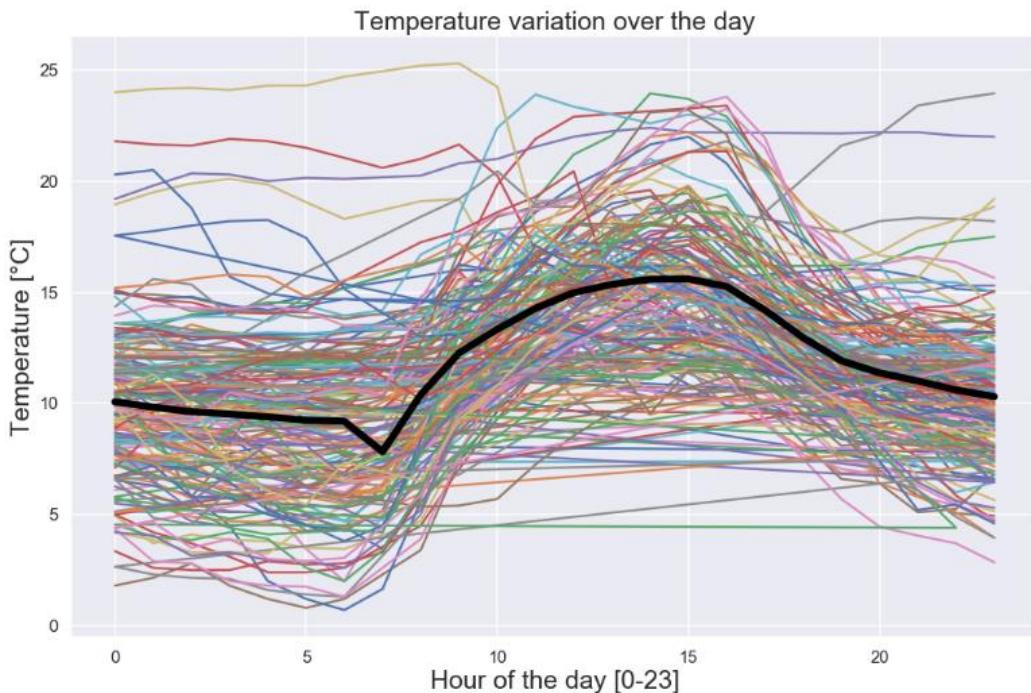
B. Identification of parameters and collection of data for the test area

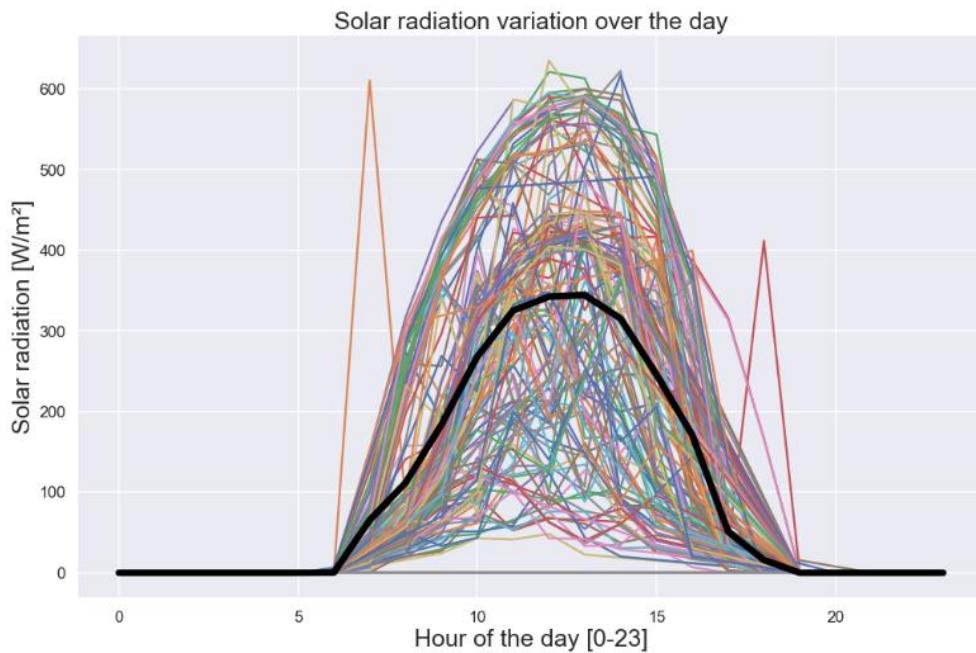
The accuracy of a machine learning model totally depends on its input data. Our first step was to search for all the parameters relevant for our purposes. We intuitively chose to use the following parameters to build our model:

- irradiance of the sun (to determine how much distributed energy can be provided by a solar panel). In fact, we have a program that converts irradiance of the sun into electrical power depending on the latitude and longitude of the solar panel along with the time and day of year.
- weather temperature
- hour of the day

The time period used to build the model will be composed of all the data cited above for every hour and day for one entire year.

We plotted two interesting figures:





From these figures, we can see that the Solar Radiation and the Temperature increase during the middle of the day. We intuitively notice that the Solar Radiation is null before 6am and after 7pm. The Temperature follows the trend of the Solar Radiation.

These two graphs are really interesting in order to have a first estimation of the trend of the price of the electricity during a day. As a matter of fact, we can expect that the price of electricity will be high when the solar radiation (and the temperature) is low. This is supply and demand.

We found the history data for the desired parameters. We cleaned and merged the data in order to fit it in the following table:

	Date	Time	Real Price (in \$/kWh)	Temperature	Solar Radiation
Group					
421176	1/18/2018	00:00:00	0.281196	12.50	0.000
421177	1/18/2018	01:00:00	0.306636	12.40	0.000
421178	1/18/2018	02:00:00	0.277777	12.40	0.000
421179	1/18/2018	03:00:00	0.291616	12.40	0.000
421180	1/18/2018	04:00:00	0.354417	12.25	0.000
421181	1/18/2018	05:00:00	0.338536	12.20	0.000
421182	1/18/2018	06:00:00	0.366386	12.30	0.000
421183	1/18/2018	07:00:00	0.328994	12.00	0.385
421184	1/18/2018	08:00:00	0.145468	12.10	23.865
421185	1/18/2018	09:00:00	0.228765	12.60	125.915
421186	1/18/2018	10:00:00	0.237516	13.60	136.980
421187	1/18/2018	11:00:00	0.205502	15.20	154.370
421188	1/18/2018	12:00:00	0.204402	15.60	84.795
421189	1/18/2018	13:00:00	0.206081	14.30	161.345
421190	1/18/2018	14:00:00	0.214326	14.05	84.210

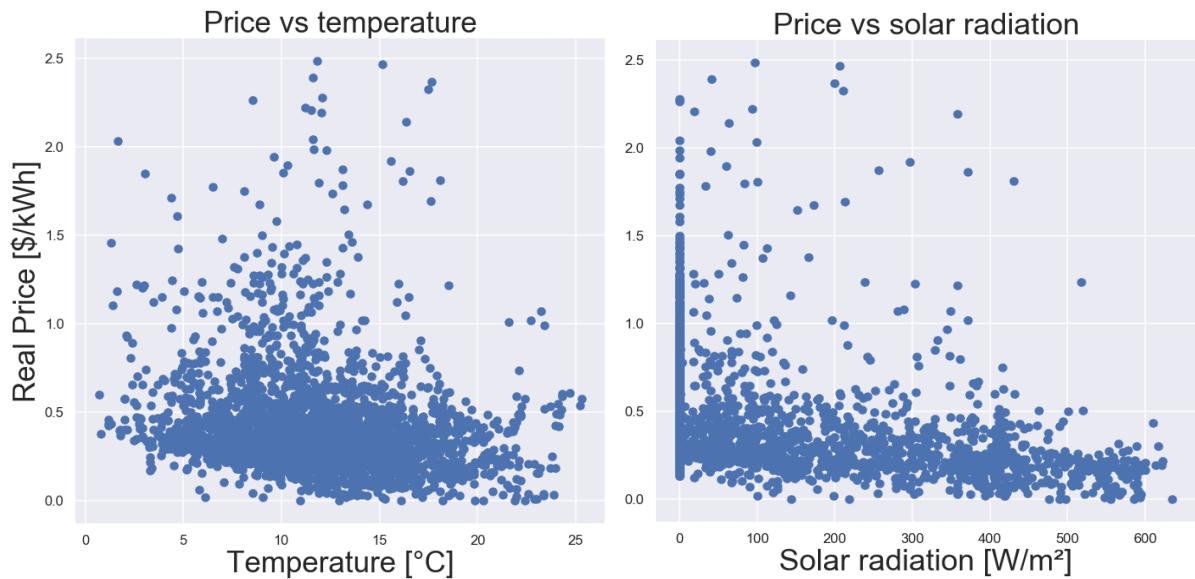
This table shows the real price, the temperature and the solar radiation for each hour of each day of the selected period.

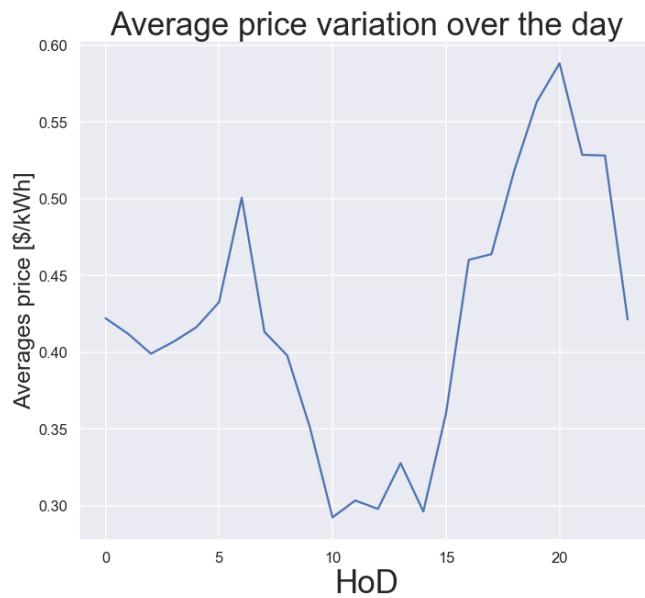
III.2. Model of the cost of energy

This paragraph will discuss how the energy price versus time can be predicted based on multiple features. First, some plots of the energy price data will illustrate if the chosen features indeed show significant correlation. Second, several models will be built and based on the comparison of their accuracy, the best model will be selected for further analysis.

A. Price versus feature correlation

Before even start building a prediction model, it is useful to graphically evaluate the relationship between the value to be predicted -the price [\$/kWh]- versus each feature: temperature [$^{\circ}\text{C}$], solar radiation [W/m^2], hour of day [HoD] and the average price variation. The following plots illustrate the dependence:

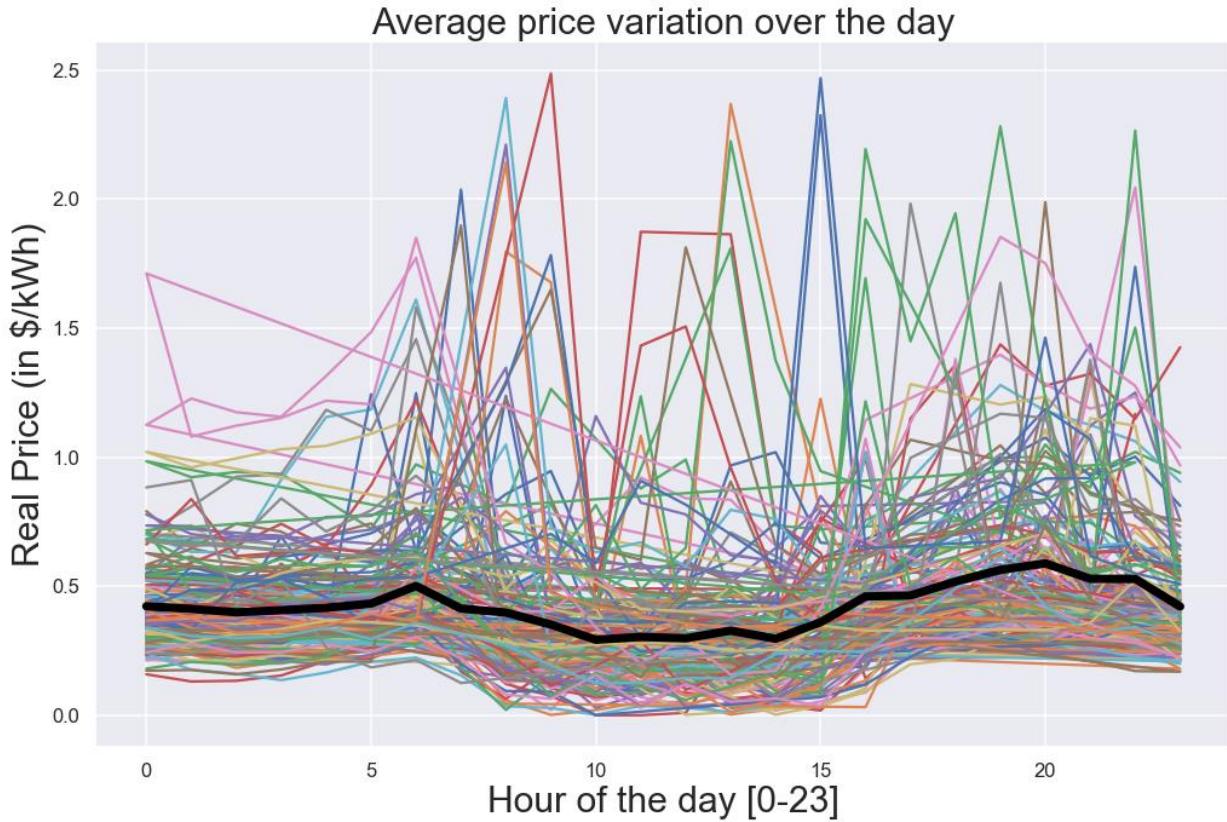




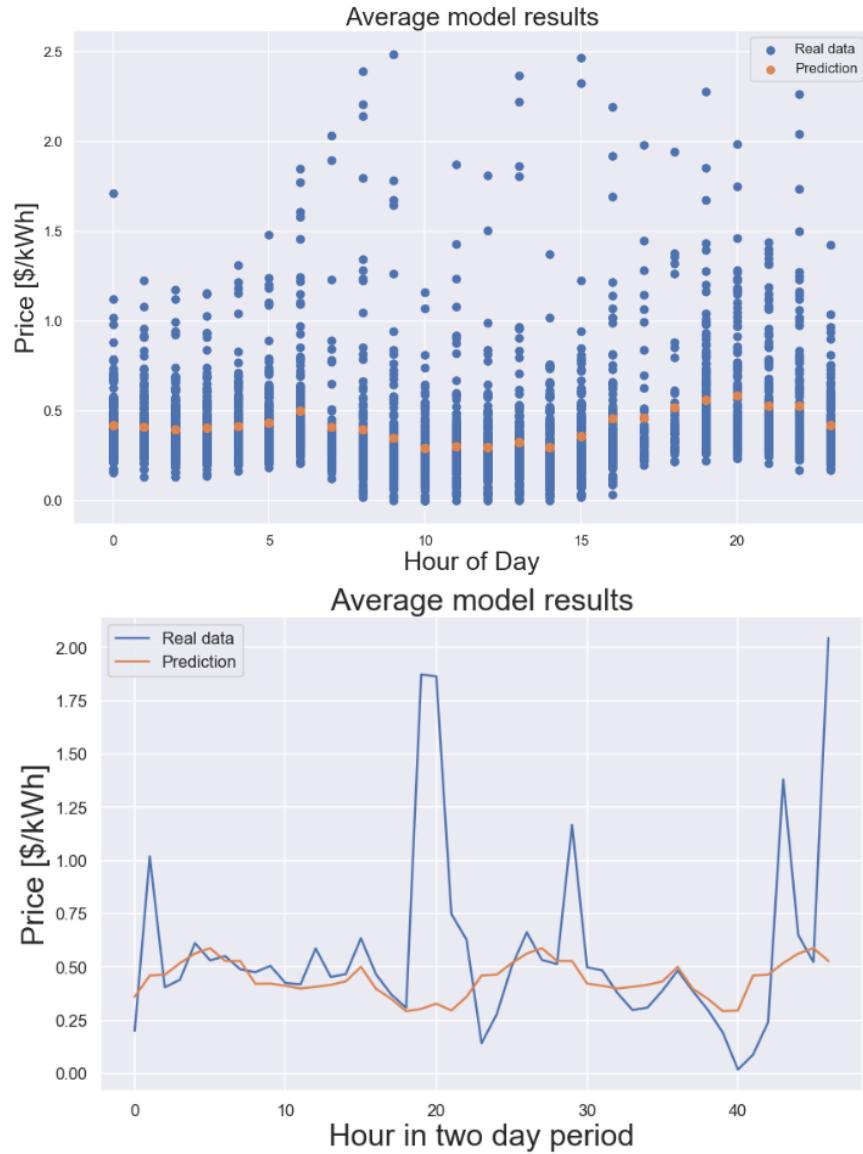
It is clear that the price undergoes significant changes over time during the day, with peaks early in the morning and in the evening. This makes sense since the demand is expected to be the highest at these times. This oscillation is clear in all days of the training data, so it will be useful to take the average price over all training data at a specific HoD and use that as a feature. Furthermore, we see that the price shows a slightly decreasing trend with increasing temperature and solar radiation. This sounds intuitive, since for warm, sunny days and hours, the electricity demand for heating and lighting will be lower, so the prices as well. The implementation of the model will show us whether it is possible to estimate the deviation of the price from the average with these features.

B. Average model

A simple average model was chosen as a first prediction model. The following figure illustrates the price variation in function of the hour of day, aggregated for all the days in the training data. For each hour, the average is computed, and this results in the thick black line in the figure.



In this model, the thick black line is chosen to be a prediction for every day of the test set. This results in the two graphs below. In both cases, the blue points represent the real data, while the orange represents the average prediction. The first graph contains data points of the entire test set, while the second contains a random sample of two consecutive days.



From these graphs, we see that the average model is able to capture major trends and predicts quite well overall but fails to follow sometimes the severe peaks that can be noticed in the real data. As a measure of accuracy of the prediction, the RMSE of the test set can be computed as follows:

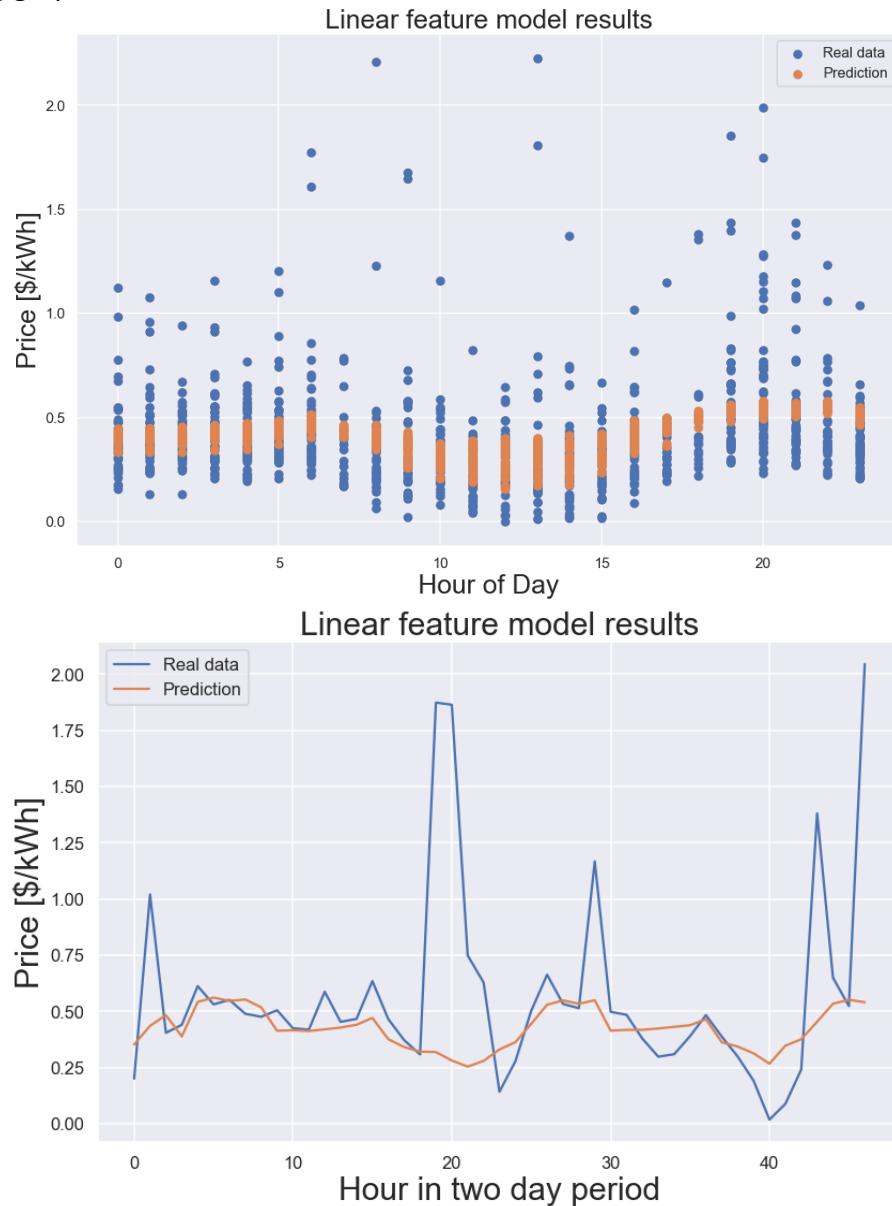
$$RMSE = \sqrt{\frac{\sum_{\text{prices in test set}} (\text{actual price} - \text{predicted price})^2}{\# \text{ of prices}}}$$

For the simple average model, the RMSE of the test set equals 0.27.

C. Linear regression models

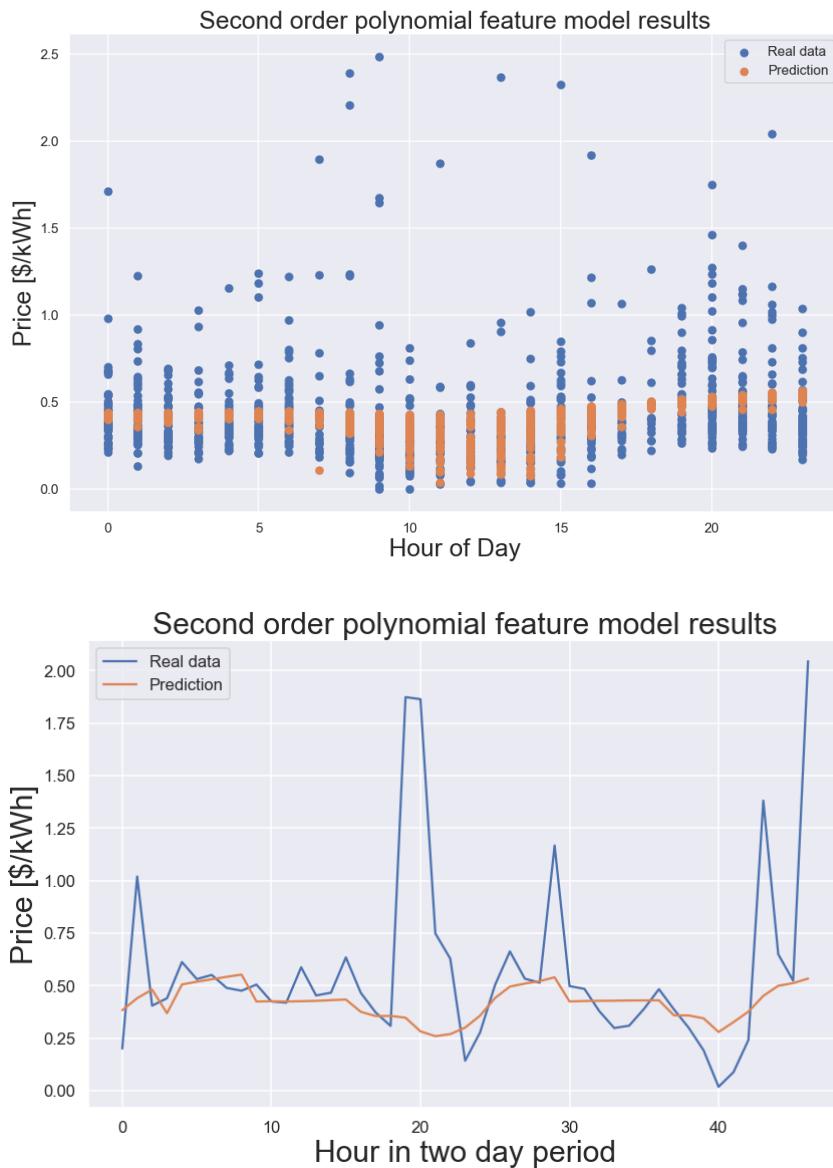
Now we can attempt to improve the average model by adding more features and using linear regression. For all of the following models, linear regression with l1-regularization or Lasso is used. The regularization parameter is determined by cross-validation on the training set. This method was chosen because it is known to drive coefficients of irrelevant features to zero, which allows us to try multiple features and non-linear combination of features.

A first model is linear in the following features: the average value of the price at a specific hour (as determined previously), the solar radiation, the temperature and the hour of day. After applying ‘LassoCV’ from the Python library, the predictions are computed and are illustrated in the following graph.



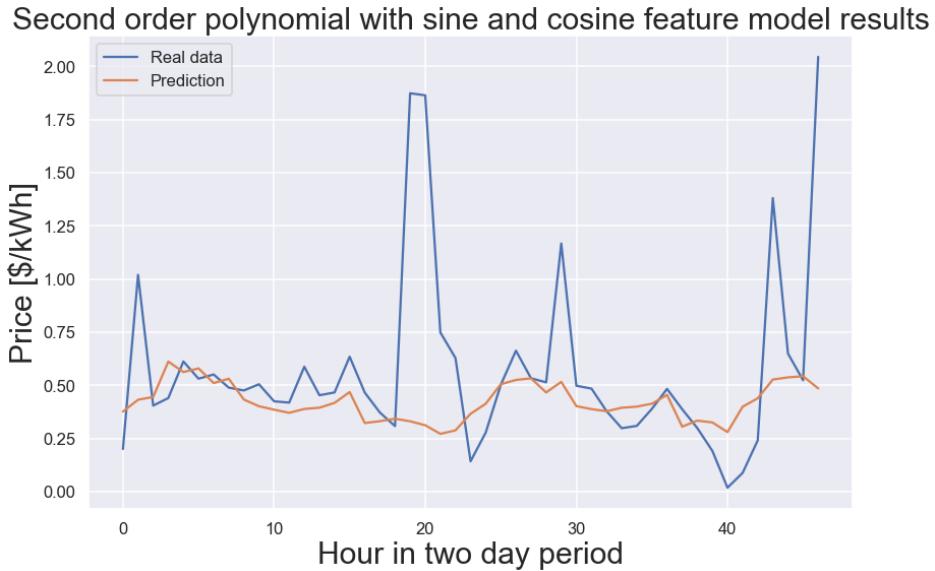
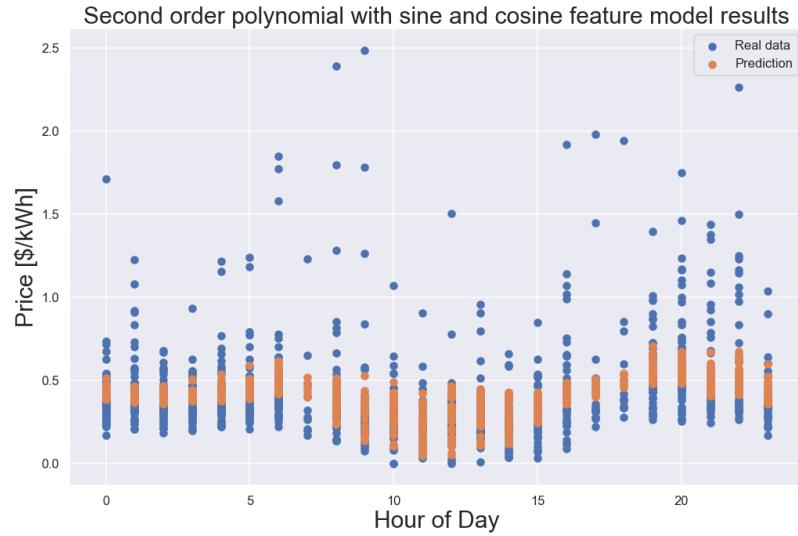
Note that the optimal value for the regularization parameter in this case was equal to zero, which means that there was no regularization, which is why none of the coefficients is zero, although some small. Not surprisingly, the largest coefficient is the one corresponding to the average price. This is clearly reflected in the graphs above, since the results are very similar to the previous average model. However, the RMSE is lower and equal to 0.245.

Subsequently, a second-order polynomial in the feature model was built. This means that, next to the features in the previous model, the square of all these features and their cross-multiples are included as additional features. This results in the following graphs:



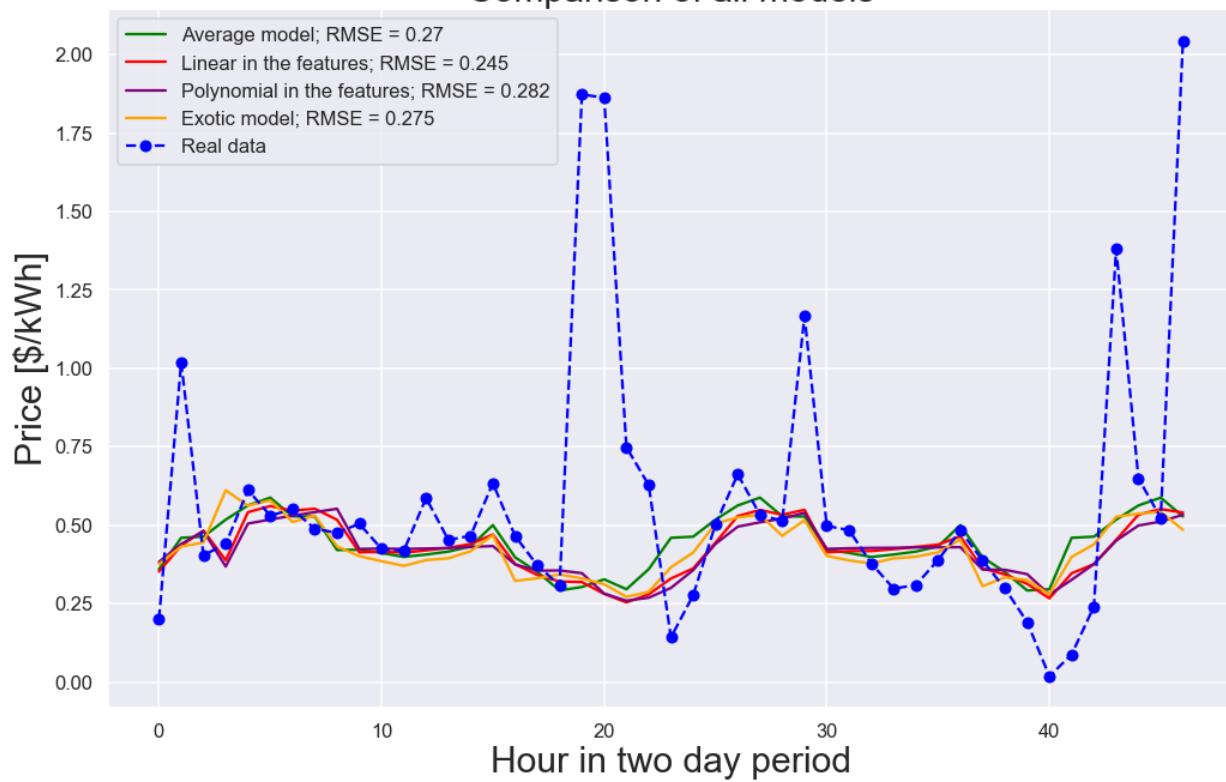
Again, the model is very similar to the first average model. However, the RMSE in this case is higher and equal to 0.282. We can interpret this as such that including polynomials resulted in 'overfitting' of the training data, which resulted in too much variance on the prediction.

Lastly, a more ‘exotic’ model has been built as well. In addition to the features in the previous model, now the sine, cosine and their squares are used as four more features. Because of the periodicity of the price data, this seemed a viable option. The results are illustrated below:



These predictions fluctuate more over time, but do not succeed in decreasing the RMSE below previous models, namely 0.275. The results of all four models are summarized in the plot below:

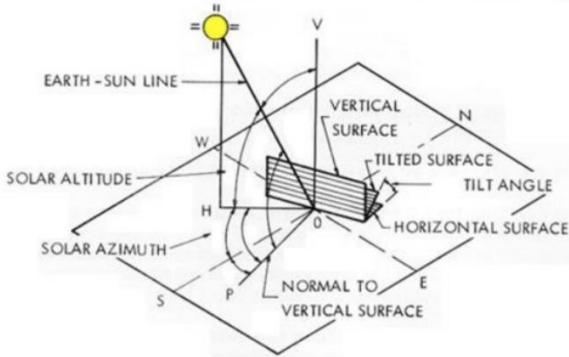
Comparison of all models



All models show very similar results, lying closely to the simple average model. However, the RMSE of the model that was linear in the four features is the lowest. That is why this model has been selected for further use in the project.

III.3. Optimization function

After gathering the price and demand input parameters, the last ingredient we need in order to produce our decision-making algorithm is the prediction of solar energy produced in the upcoming 24 hours. As it was hard to locate recent solar irradiation data, we decided to compute our own.



The calculations are based off the diagram pictured on the right. Essentially the location of the solar panel with respect to the sun along with the time of day and day of year determine the amount of solar irradiation arrives onto the solar panel. The calculations are gone through in the following equations:

$$\alpha = 15 * (t - 12)$$

$$\delta = 23.44 * \sin\left(\frac{360}{365.25}(d - 80)\right)$$

$$\cos(\chi) = \sin(\lambda) * \sin(\delta) + \cos(\lambda) * \cos(\delta) * \cos(\alpha)$$

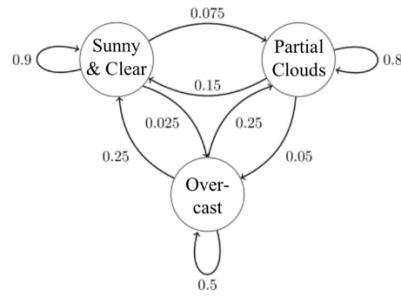
$$\tan(\xi) = \frac{\sin(\alpha)}{\sin(\lambda) * \cos(\alpha) - \cos(\lambda) * \tan(\delta)}$$

$$I_{DN} = A * e^{\frac{B}{\sin(90 - \chi)}} \text{ with } A = 1310 \frac{W}{m^2} \text{ and } B = 0.18$$

$$I_D = I_{DN} * (\cos(\chi) * \cos(\varepsilon) + \sin(\varepsilon) * \sin(\chi) * \cos(\xi - \zeta))$$

Here t is equal to the solar time in [h], λ the latitude [$^\circ$], ζ the surface azimuth angle [$^\circ$], ε the surface inclination from the horizontal plane [$^\circ$], α the hour angle [$^\circ$], d the day of the year, χ the solar zenith angle [$^\circ$], ξ the solar azimuth angle [$^\circ$], I_{DN} the intensity of direct normal radiation [W/m^2], A and B specified constants accounting for the effects of the atmosphere and I_D [W/m^2] the incident direct solar flux.

Once the solar irradiance is computed for each hour of the day, a simple energy efficiency factor will be used to convert incoming solar flux into electrical energy produced by the solar panel. For our analysis, we implemented an efficiency of 15% and used solar panels with an area of five squared meters.



Finally, in order to take into account the annoyance of clouds, a Markov chain will be implemented in order to acquire additional factors that will reduce the incoming solar flux. The diagram shows the probability of one state going to the other and from this we determined what state the system will be in at each hour. Once the state is determined the following coefficients will multiply the incident radiation flux:

- Sunny and Clear: 1
- Partial Clouds: 0.7
- Overcast: 0.25

Now with all three ingredients, the optimization program can be started. We will first start defining a few variables. The index ‘i’ denotes the hour of the day.

$P_G(i)$:	price of grid at i	Input from Price Prediction Model
$D(i)$:	demand at i	Input from Building Data
$P_S(i)$:	solar power generated at i	Input Calculated from Solar Production Model
$c_G(i)$:	consumption from the grid at i	Optimization variable
$c_S(i)$:	consumption of solar energy at i	Optimization variable

The next important feature to understand are the constraints imposed on the optimization problem. The constraints shape the results obtained and clearly guide the solution and the greater the problem is accurately constrained, the more realistic the solution is.

$$\begin{aligned}
& \mathbf{c_G}_i + \mathbf{c_S}_i = \mathbf{D}_i && \text{for } i = 1, 2, \dots, 24 \\
& \mathbf{c_S}_i \leq \mathbf{sum}(\mathbf{S}_{P_i}) && \text{for } i = 1 \\
& \mathbf{c_S}_i \leq \eta * \mathbf{sum}(\mathbf{S}_{P_{1:i}}) + (1 - \eta) * \mathbf{S}_{P_i} - \mathbf{sum}(\mathbf{c_S}_{1:i-1}) && \text{for } i = 2, 3, \dots, 24 \\
& \mathbf{sum}(\mathbf{S}_{P_{1:i}}) - \mathbf{sum}(\mathbf{c_S}_{1:i-1}) \leq 14 && \text{for } i = 1, 2, \dots, 24 \\
& \mathbf{c_S}_i \geq 0 && \text{for } i = 1, 2, \dots, 24 \\
& \mathbf{c_G}_i \geq 0 && \text{for } i = 1, 2, \dots, 24
\end{aligned}$$

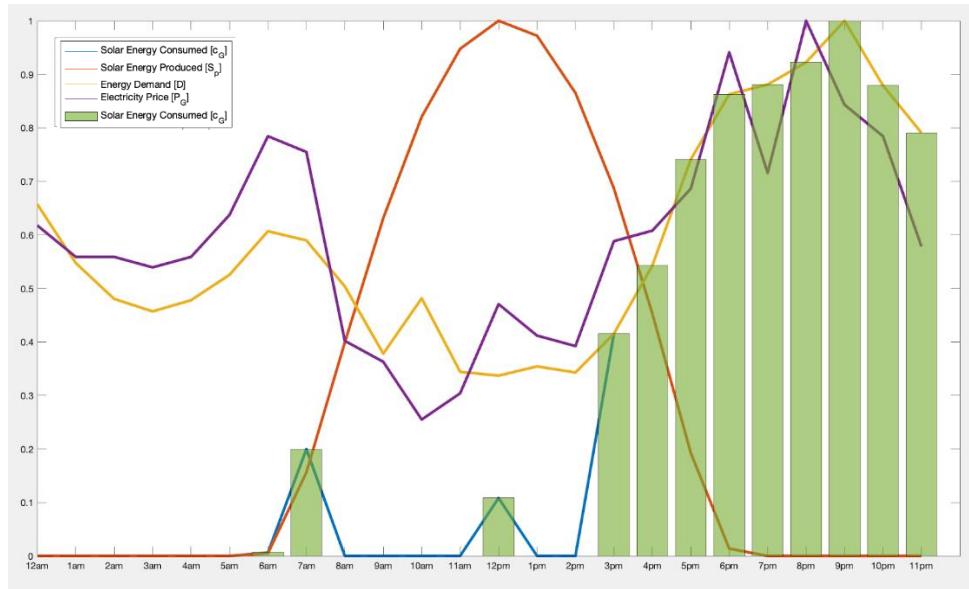
The first constraint requires all demand to be met for each hour of the day. Tenants need to be able to acquire the energy they require, which is why at each hour of the day, the sum of the grid energy and solar energy consumed must meet the demand. The next two constraints take into account the efficiency of the energy as it is stored in the battery. For this specific project we decided to state that the solar energy used at the current hour can be used at 100% efficiency whereas any energy used from energy stored for more than one hour will have an efficiency factor, etc. From literature, we decided to implement a battery storage efficiency of 80%. The fourth constraint imposes the idea of maximum battery capacity. Without this constraint, the battery could infinitely store energy. Being commonly used in the west coast, we decided to use a Tesla Powerwall as the battery storage for our project. This battery has a maximal capacity of 14 KWh, hence explaining the value of 14 in the fourth constraints. Finally, the last two constraints, simply state that both the grid and solar consumption at each hour have to be positive.

With all these constraints placed, the last step is to pose the objective function. For this application, the objective is to reduce the total price, which is simply the cross product of the price vector obtained from the prediction model and the grid consumption vector, which is one of the optimization variables. The equation is summarized below.

$$J = \sum_{i=1}^{24} \mathbf{P_G}_i * \mathbf{c_G}_i$$

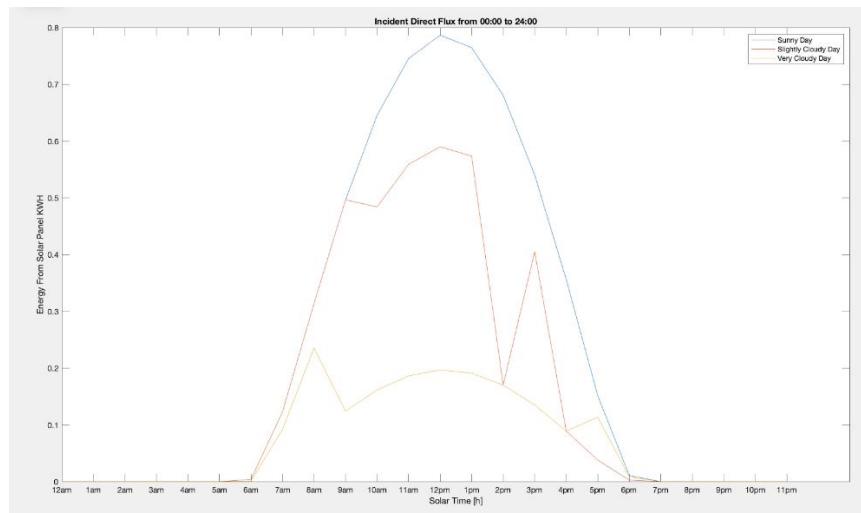
III.4. Results

With all the constraints and data gathered, the following graph represents a typical solution of normalized energy uses observed from the optimization program.

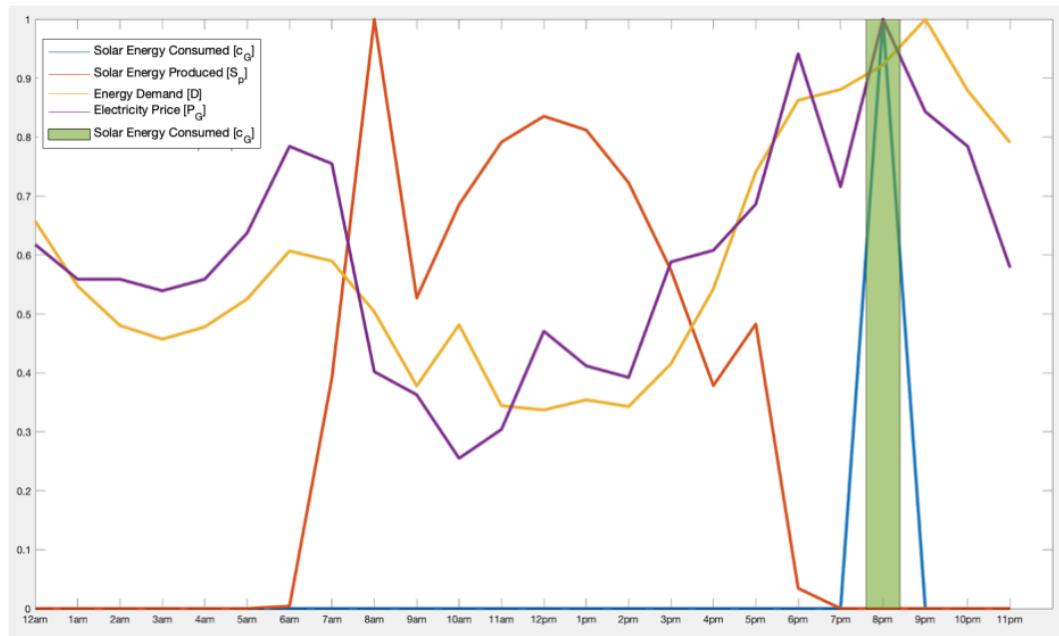
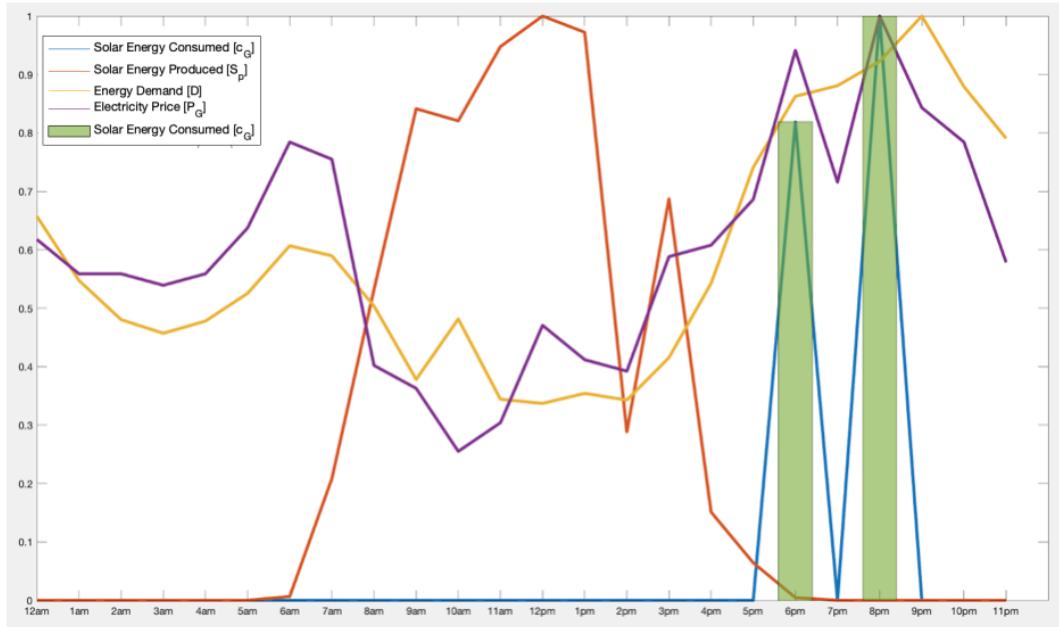


In order to evaluate the performance of our decision-making algorithm, we will compare the prices between our algorithm with the scenario where the solar energy is entirely used to meet the demand at every hour.

The graph above depicts the situation for a sunny day throughout the 24 hours and we have a saving of \$2.04. It is a 13.6% reduction in price compared to non-optimized. Now if we compare this result with days that are slightly cloudy and very cloudy the savings per day decrease tremendously.



The figure above depicts the energy produced by the solar panels on a typical sunny, slightly cloudy and very cloudy day.



The two graphs above represent the decision-making algorithm for both partly cloudy and very cloudy days respectively and the results are tabulated below.

	Savings (\$)	Percent Difference
Sunny	2.04	-
Partially Cloudy	1.42	30 % less
Very Cloudy	0.61	70 % less

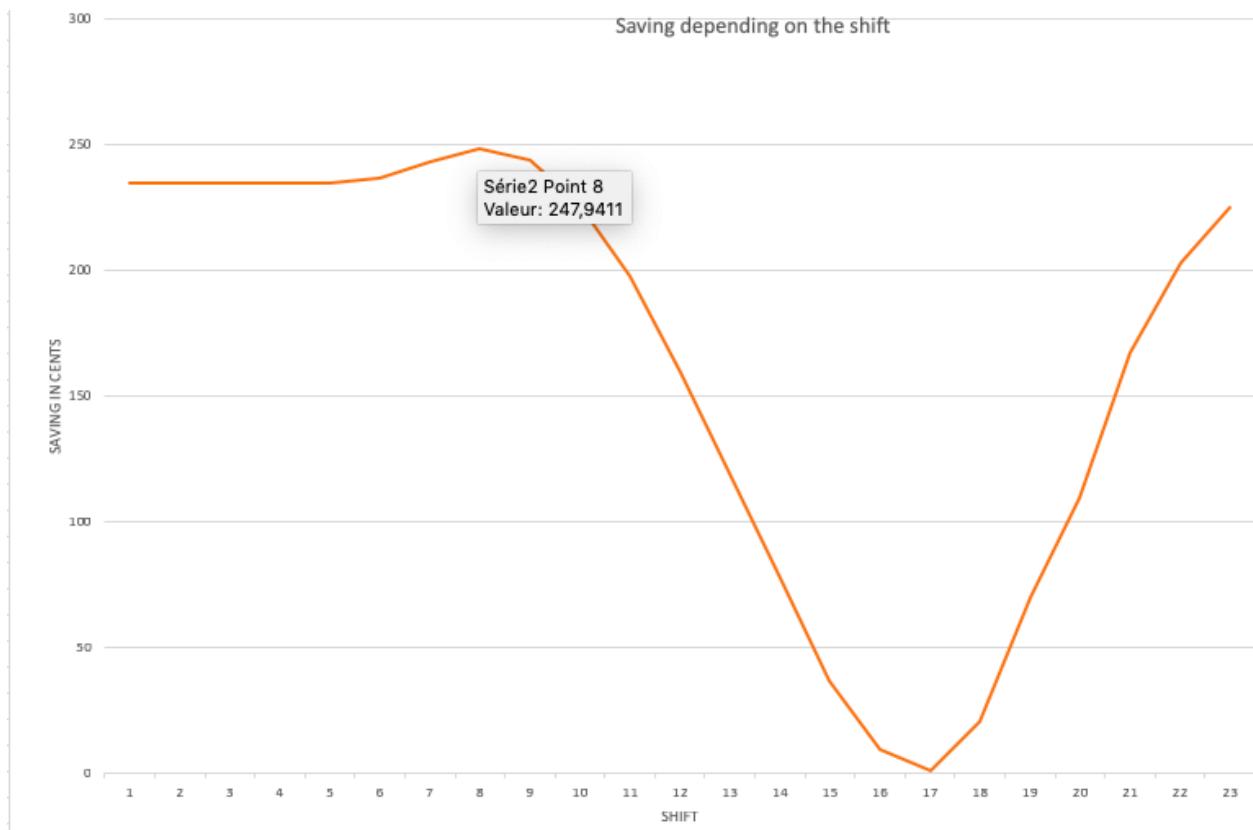
Since our program is optimizing the delivered energy storage only for the next 24 hours, the time at which the program is run is of the utmost importance. All the previous results were found for a starting time of the optimization program of 12:00:00 am. In fact, by starting at 12:00 am, there is no solar energy produced until the sunrise (approximately 8am), and therefore with our current program there is no optimization possible from 12:00am to 8:00 am since there is no solar energy to store. Hence, the optimal starting time of the program has to be found to maximize the saving.

As our time step is 1 hour, we decided to shift the starting time for every hour and compute the saving for every shift. Below is a table of the result obtained:

shift in hour	Saving in cents/day
1	234,4811
2	234,4811
3	234,4811
4	234,4811
5	234,4811
6	236,2354
7	243,067
8	247,9411
9	243,9476
10	226,2204
11	197,2606
12	159,3946
13	119,0154
14	77,3037
15	36,4201
16	9,2645
17	0,7723
18	20,1359
19	69,9081
20	109,238
21	166,7936
22	202,7944
23	224,6363

In this table, a shift of 1 hour consists in running the program at 1 am instead of 12 am, a shift of 2 hours consists in running the program at 2 am instead of 12 am, and so on.

By plotting the savings versus the time shift, we obtained the following graph:



From this plot, we deduced that the time shift that maximizes the saving is 8 hours.

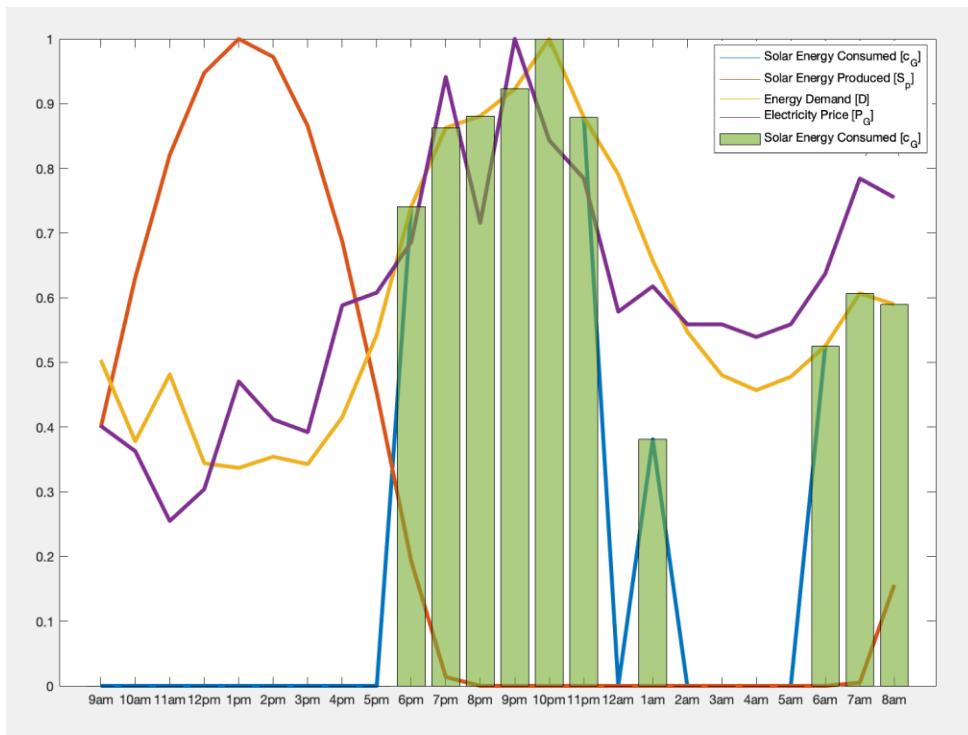
The limit with this result is that the optimal shift found depends on the solar power generated (and therefore of the weather), and the price. This means that we should obtain a different optimal shift for each day since the weather and the price change over the days. Of course, it would be impossible to change our start time every day since our program has been implemented to run for the next 24 hours. However, after having computed the optimal shift for different weather, the optimal shifts ranged from 8 hours to 10 hours. We therefore chose a shift of 9 hours for our program.

Below is a table showing the difference of saving per day by shifting the starting time of the program:

Initial Saving (running the model at 00:00am)	2.04 \$
Saving with a shift of 9 (running the model at 09:00am)	2.48 \$

Therefore, with a shift of 9 hours, we can save 21.6% more than our initial program starting at 12:00am.

Below is the plot obtained for a shift of 9 hours:



The price of a PowerWall from Tesla is around \$7500. We can conclude that the return on investment (ROI) is equal to 8 years for only the batteries. This result seems a lot for a household so cheaper alternatives are needed to make our implementation economically viable.

IV. Discussion and Future Work

The results of the model and the optimization are very promising in terms of scalability. We took the data for a building and we think that the process we use can be multiplied to fit with numerous households and buildings. The conclusion about the cost saving has to be seen as subject to be in a situation without other costs. When we use the Powerwall from Tesla, we didn't include the cost of this battery in the final saving. Actually, it is the same for other investment that people should make to be able to use solar energy. We were focused on a particular situation where the need is to reduce only the price of electricity for the household.

We strongly believe that forecasting the price of electricity could benefit households as it will be the future for energy consumption. The optimization model was improved by using Markov Chain for the stochastic prediction of weather and showed how sensitive savings are to weather. We should probably spend time in discussing archetype of weather situation. We just talked in this project about "Sunny", "Partially Cloudy" and "Very Cloudy" but in the real world a lot of different aspects and features could be used to refine the optimization model. Especially when we are looking for energy demand, it is fluctuating according to the different seasons. So, we could at least create four different sub model to increase the accuracy of the optimization.

Building a device to actively open or close circuits between grid energy and solar energy was also one of our objectives early on. This device would take advantage of our decision-making model and act in real-time to the software developed. This could be a technical solution to reduce efficiently the cost of electricity for people.

To pursue our vision, the main restriction were complete datasets. Actually, to improve the pricing model, we need more relevant features. We already used solar radiation, temperature, average price and hour of day, but we definitely need more. One of the other goals could be to build the decision-making software with real time data. That would be something very profitable and the proof that our process could work anytime in any situation. The decision-making software can also easily be adapted to longer periods instead of only one day because the change could fit with completely abnormal weather and could be the finish line of the entire project with all new features.

V. Summary

In this study, an optimized process to minimize the cost of electricity for a household has been created with a forecasting model of the price of electricity and the use of different features as temperature, solar radiation, the average price of energy and the hour of the day. The result showed that if we just focus on the cost of electricity and the use of the Powerwall battery from Tesla without including the investments behind these systems, our decision-making process over the day saves 13.6% of non-optimized costs and when we shift the hour, the savings increase to 16.5% for the cost of electricity for one building. By connecting the forecast of the cost of electricity and the optimization of the use of solar energy, we succeed to develop a process able to match the demand and the consumption in order to reduce the final bill.

VI. References

- [1] Zhou, Suyanget al. "Real-time Energy Control Approach for Smart Home Energy Management System." *ElectricPower Components and Systems* 42.3-4 (2014): 315–326.
- [2] N. Aytjanova, V. Cheng, P. Garneau-Halliday, E. Porter, J. Richards. "Oakland EcoBlock: Optimal Topology Configuration". UC Berkeley CE 295. 2017
- [3] L. Weaver, K. Ito, S. Mitchell, D. Macdonalds. "Go go green it". UC Berkeley CE295. 2017
- [4] J. Semigran, E. Tsim. "Bay Area Smart Home Cost Optimization with Photovoltaic Electricity". UC Berkeley CE 295. 2017
- [5] A. Bach, J. Kwan, C. Piatt. "Optimal Energy Management on the French Island of Guyane". UC Berkeley CE 295. 2017
- [6] Technical documents of Powerwall from Tesla website. <https://www.tesla.com/powerwall>

Battery State of Charge Estimation

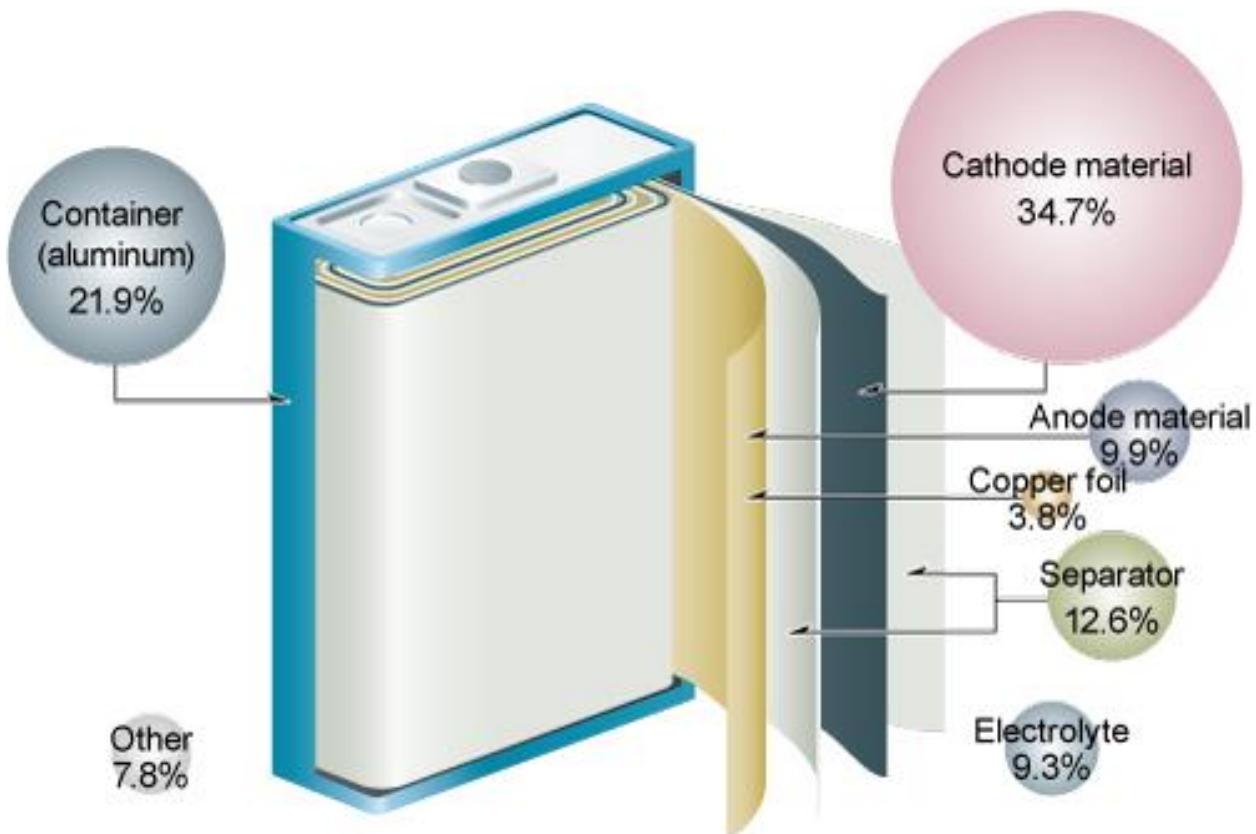
Team members: Jiwei Chen

Zamin Ali Salman

Gowtham Sivaramakrishnan

Aslan Tabyldiyev

William Xi



I. Abstract

Li-ion batteries applications require an electronic system that estimates the charge level in the battery to prevent sudden power loss. The value of battery state-of-charge (SOC) represents the remaining capacity of a battery and can be assumed to be a function of voltage, current, and temperature of the system. This paper presents a method for estimation of the SOC using the Extended Kalman filter (EKF) and neural networks (NN). Given, the non-linear nature of the output Voltage function, its dependence on temperature and the measurement noises present in the equipment, we use EKF to predict Voltage and State of Charge by merely relying on Current as input. We assume that our EKF model is blind to the effects of temperature and measurement noise and we check how well our EKF model performs. The neural network models used temperature, voltage and current as inputs to predict the state of charge. A linear neural network and a non-linear neural network model is developed. In addition, using a circuit model with known resistances and capacitances, we can develop an estimator to test the accuracy of our predictions on real-world data under different operating conditions such as fast charging and low charge level applications.

II. Introduction

a. Motivation and Background

Finite fossil fuels, political, and socio-economic conditions drive the adoption of CO₂-emission free power generation, clean energy, and storage systems. Moreover, such circumstances also proceed with the need to replace traditional internal combustion engine vehicles with electric vehicles. Therefore, a battery with high energy storage and long energy storage lifetime is necessary to address these issues. Also, this battery is needed for the increasing demand for consumer electronics such as mobile phones, laptops, and tablets. In particular, lithium-ion batteries have been widely implemented in various electronic products including electric vehicles due to its low weight, high energy density, long energy storage lifetime, and not a very high cost [1].

Although the electrical performance of Li-ion batteries, such as energy density, power density and C-rate, has improved significantly, non-uniformity of the capacity and the retention ratio associated with ageing are still ongoing major issues regarding cell series [2]. A battery management system (BMS) is a regulator that protects the overall system and provides optimal performance management of the system. Apart from the online monitoring of the terminal voltage, a BMS should observe and give an accurate estimation of SOC which can prevent each cell from over-charging or over-discharging and can potentially extend the battery's lifetime.

b. Relevant Literature

State estimation of the battery has been discussed among researchers for decades. He et al. [3] suggests five different types of equivalent models such as the Rint, RC, Thevenin, PNGV and dual polarisation (DP) model for lithium-ion batteries and obtained model parameter values using

a genetic algorithm, which is used to find the optimal time constant of the model. These five models are evaluated from the dynamic performance and the state charge of estimation (SOC). Based on this paper, the dual polarization has the best dynamic performance and provides the most accurate estimation of the SOC.

Feng et al. [4] discuss an online parameter estimator and the state of charge (SOC) of lithium iron phosphate batteries in an electric vehicle, where SOC estimator is based on the dynamic battery model with predeterminate parameters. An equivalent circuit model using adaptive joint extended Kalman filter algorithm is applied to reduce the errors in estimating battery SOC resulted from the model parameters variances with their different temperatures. Also, open-circuit voltage (OCV) is used to determine SOC in a proposed algorithm.

Ismail et al. [5] discuss a temperature model for a battery based off of 3 components of heat transfer: heat generation from internal resistance, from entropy changes during discharge, and convection to the ambient environment. A lumped capacitance model is assumed for the cell. The validity of the lumped capacitance model is checked by calculating the Biot number which describes the ratio of heat transfer at the surface and the inside of the battery. A biot number under 0.1 is generally accepted.

Charkhgrad and Farrokhi [6] describe using an artificial neural network (ANN) to learn the model describing the state of charge as a function of terminal voltage and current. The neural network is trained offline. The combination of the neural network and the extended Kalman filter (EKF) is used to estimate the state of charge online.

c. Focus of this Study

In this project, our primary focus will be the estimation of SOC of Li-ion batteries using machine learning techniques (neural networks) and Extended Kalman filters. We aim to compare the results obtained by these two models for charging and discharging situations.

III. Technical Description

Methodology

Equivalent circuit models (ECM), that consist of the basic circuit elements such as resistors, capacitors, and inductors, are commonly used to simulate the behaviour of a battery. This model plays an essential role in researching effective management system. Estimation accuracy of the battery is based on the precise selection of the ECM for State of Charge (SOC) estimation. The several aspects should be taken into account for choosing ECM. First of all, the model should imitate the dynamic behaviour of the battery. Secondly, it must not be too complicated to establish state-space equations. The sophisticated ECM can add computation cost. Based on different research papers [3], the second-order equivalent circuit battery model in *Figure 1* is chosen. This model consists of an open-circuit voltage (OCV), a resistor R_0 and two parallel resistor-capacitor networks connected in series.

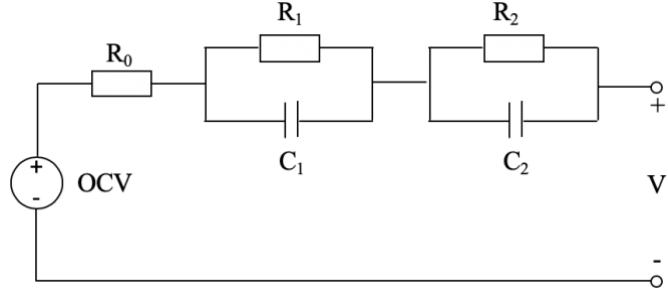


Figure 1: Circuit Model

The energy sources are contained within the two capacitors and the voltage source. Thus our three state variables are the battery state of charge (z) and the voltages across the first and second capacitors from left to right respectively (V_{C1} and V_{C2}). Thus, our state-space model is third order. The controllable input is current, and there is no uncontrollable input. The measured output is the voltage V . The dynamical state equations are developed using Kirchhoff's voltage and current laws and an integrator equation for the state of charge. The output equation for V is a non-linear function of the state of charge due to open circuit voltage $OCV(z)$. The parameters in this EKF system are the resistances R_1 , and R_2 and the capacitances C_1 and C_2 , the battery charge capacity Q and the settings in the polynomial OCV function.

The non-linearities in this model consist of the OCV function and the temperature dependence of the resistance R_0 . The OCV function is approximated as a polynomial of the state of charge z . We also consider using a neural network to learn $OCV(z)$ later on in this project. Lastly, the resistance R_0 is assumed to be a linear function of temperature. The cell temperature itself is described by a first order ODE described in [5]. We consider this equation to be independent of the state space model as it is not a function of the state variables. We determine the cell temperature by solving the ODE as a function of time to obtain a resistance time series. This time series resistance is input into the output voltage equation.

To compare the results obtained by EKF and neural network for various situations like speed charging and discharging, we first run a simulation that uses the above equivalent circuit model to get voltage and state of charge as output. This simulation uses equations from the literature review and assumes the internal resistance of battery as a function of temperature. As the battery charges or discharges, current flows through our equivalent circuit model and heat is generated which raises the temperature of the system and the corresponding internal resistance of our battery. We use this simulation as our actual real-world data.

Next, we design an Extended Kalman Filter that is blind to the effects of temperature on resistance and treats all resistances including the internal resistance of battery as constant parameters. We aim to tune these Eigenvalues of the EKF to make it converge as quickly and efficiently as possible. We don't wish to simply minimize the time of convergence since it will enhance the gain factor of error and be more susceptible to noise.

A neural network is formulated to predict the state of charge blind to the equivalent circuit model. The inputs (or features) into the neural network model are the current, voltage and temperature at any time t . The output is a single continuous prediction of SOC between 0 and 1. Two neural network models are developed. Firstly, a linear model is designed to see if linear

regression is sufficient to achieve high accuracy on this problem. Secondly, a non-linear model is developed using sigmoid activation functions. The optimizer Adam is used for both models. The neural network architecture, hyperparameters and activation functions are selected to try and maximize the prediction accuracy. The neural network is developed using the sequential model in Tensorflow.

Models

Assumptions

1. The resistance is linearly dependent on temperature.

$$R = R_0(1 + \alpha(T - T_{ref}))$$

2. The open circuit voltage is a polynomial function of the SOC.

$$OCV(z) = p_0 + p_1z + p_2z^2 + p_3z^3$$

Parameters	p_0	p_1	p_2	p_3
Value	3.4707	1.6112	-2.6287	1.7175

3. The temperature mode of the resistor takes into account the change in temperature due to the heat generated by resistors, the heat generated due to change in entropy and the convective heat loss to the surroundings [5].

$$mc \frac{dT}{dt} = I^2 R_0(1 + \alpha * T) + T \Delta S \frac{I}{F} + Ah(T - T_{ref}))$$

4. The convective heat transfer coefficient does not change under temperature.

State-space Model and Output Function

From the Kirchoff's Law, assuming the linear dependence of resistance on temperature we get the following equations:

$$V = OCV(z) + V_{C1} + V_{C2} + IR_0(1 + \alpha(T - T_{ref}))$$

$$\dot{V}_{C1} = \frac{I}{C_1} - \frac{V_{C1}}{R_1(1 + \alpha(T - T_{ref}))C_1}$$

$$\dot{V}_{C2} = \frac{I}{C_2} - \frac{V_{C2}}{R_2(1 + \alpha(T - T_{ref}))C_2}$$

$$\dot{z} = \frac{I}{Q}$$

The model is converted into a fourth order state space form with state variables V_{C1} , V_{C2} , z and T .

State-space model is as following:

$$\begin{bmatrix} \dot{V}_1 \\ \dot{V}_2 \\ \dot{z} \\ \dot{T} \end{bmatrix} = \begin{bmatrix} -\frac{V_1}{C_1 R_1 (1 + \alpha(T - T_{amb}))} + \frac{I}{C_1} \\ -\frac{V_2}{C_2 R_2 (1 + \alpha(T - T_{amb}))} + \frac{I}{C_2} \\ \frac{I}{Q} \\ \frac{I^2 R}{mc} (1 + \alpha(T - T_{ref})) + \frac{TSI}{mcF} + \frac{ah(T_{amb} - T)}{mc} \end{bmatrix}$$

After linearizing we get:

$$\begin{bmatrix} \dot{V}_1 \\ \dot{V}_2 \\ \dot{z} \\ \dot{T} \end{bmatrix} = \begin{bmatrix} -\frac{1}{C_1 R_1} & 0 & 0 & 0 \\ 0 & -\frac{1}{C_2 R_2} & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{-ah}{mc} \end{bmatrix} \begin{bmatrix} V_1 \\ V_2 \\ z \\ T \end{bmatrix} + \begin{bmatrix} \frac{1}{C_1} & 0 \\ \frac{1}{C_1} & 0 \\ \frac{1}{Q} & 0 \\ \frac{T_{eq} S}{mcF} & \frac{ah}{mc} \end{bmatrix} \begin{bmatrix} I \\ T_{amb} \end{bmatrix}$$

Parameter values:

The parameters of the second-order model are taken from [7]:

Parameters	R ₀	R ₁	R ₂	C ₁	C ₂
Value	0.02 Ohm	0.02 Ohm	0.003 Ohm	1600 F	17,500 F

The parameters for the temperature dependence equations are [5]:

Parameters	α	Q	m	C	ΔS	F	A	h
Value	0.005 (1/ $^{\circ}$ C)	7920 C	1.5 kg	1350 J/kg.K	-30 J/molK	96,500 C/mol	0.21228 m ²	18.9 W/m ² K

The input parameters are the current and ambient temperature. The ambient temperature is assumed to be a white Gaussian noise of standard deviation of 1 as it is difficult to predict the ambient temperature correctly. The current depends on how the user operates the car. For simplicity, the current is assumed to have 5 charging and discharging cycle. Since the car should be charged faster than the time is taken for it to discharge the current during charging is assumed to be 2.5 A and the current during discharging is 0.5 A. The time for each charging and discharging

cycle was decided based on the time required to take the state of charge from 0.9 to 0.1 and 0.1 to 0.9. All of the input results are shown in *Figure 2*.

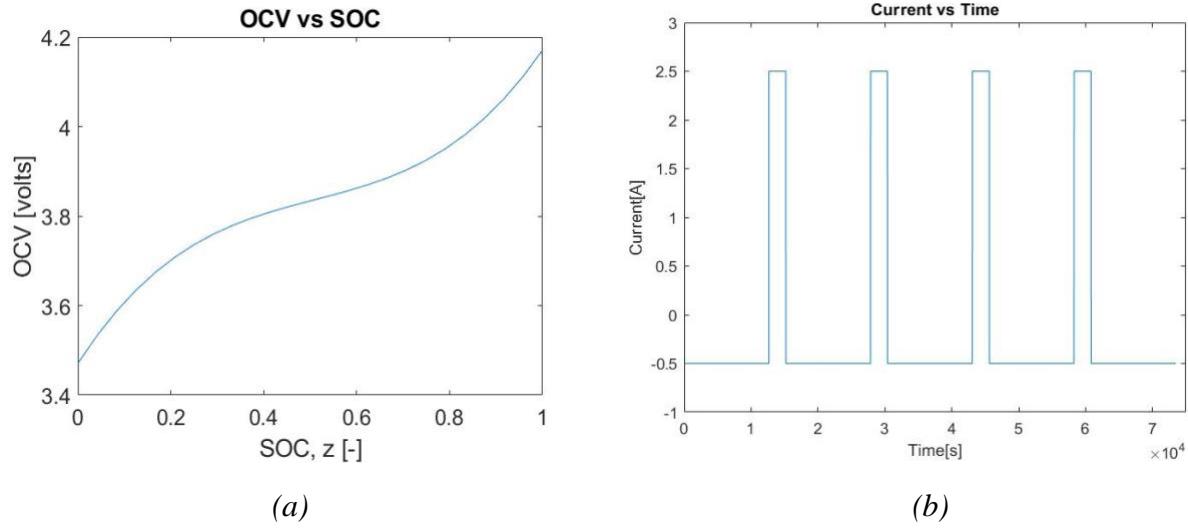


Figure 2: (a) OCV graph as a function of battery SOC; (b) The time requirements for charging and discharging the battery within its maximum and minimum OCV values.

Algorithms

Extended Kalman Filter

The EKF equations are as follows:

$$\dot{\hat{x}} = f(\hat{x}, u) + L(t)[y_m - h(\hat{x}, u)], \quad \hat{x}(0) = \hat{x}_0$$

$$\hat{y}(t) = h(\hat{x}, u)$$

$$L(t) = \Sigma(t)H^TN^{-1}$$

where $\Sigma(t)$ is the solution to the Riccati Differential equation and x corresponds to states of V_{c1} , V_{c2} and z:

$$\dot{\Sigma}(t) = \Sigma(t)F(t)^T + F(t)\Sigma(t) + W - \Sigma(t)H(t)^TN^{-1}H(t)\Sigma(t), \quad \Sigma(0) = \Sigma_0$$

$$F = \begin{bmatrix} -\frac{1}{R_1C_1} & 0 & 0 \\ 0 & -\frac{1}{R_2C_2} & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

$$H = [1 \quad 1 \quad p_1 + 2p_2(z^{eq}) + 3p_3(z^{eq})^2]$$

$$W = \begin{bmatrix} 10^{-3} & 0 & 0 \\ 0 & 10^{-3} & 0 \\ 0 & 0 & 10^{-3} \end{bmatrix}$$

Assuming the values of measurement noise covariance and Σ_0 as follows

$$N = 0.1$$

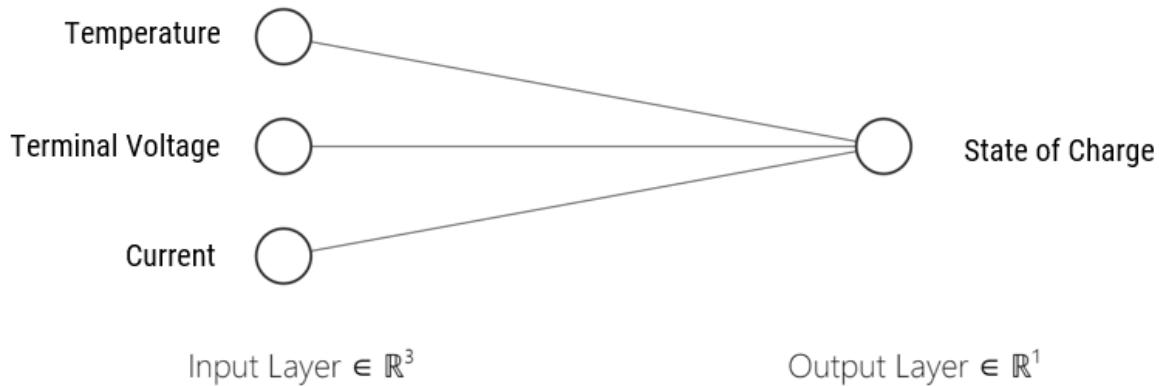
$$\Sigma_0 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

As mentioned earlier our objective is to simply not reduce the convergence time but to do in a way that minimizes the effects of noise while balancing the convergence time. There is no single right answer to this but from the various values we tried for W , we believe that the most optimum value for it is as follows:

$$W = \begin{bmatrix} 10^{-3} & 0 & 0 \\ 0 & 10^{-3} & 0 \\ 0 & 0 & 10^{-3} \end{bmatrix}$$

Neural Network

Neural Network Linear Architecture:



The architecture is presented in the table below:

Layers	Input	Output
Number of Neurons	3	1
Activation Function	N/A	Linear

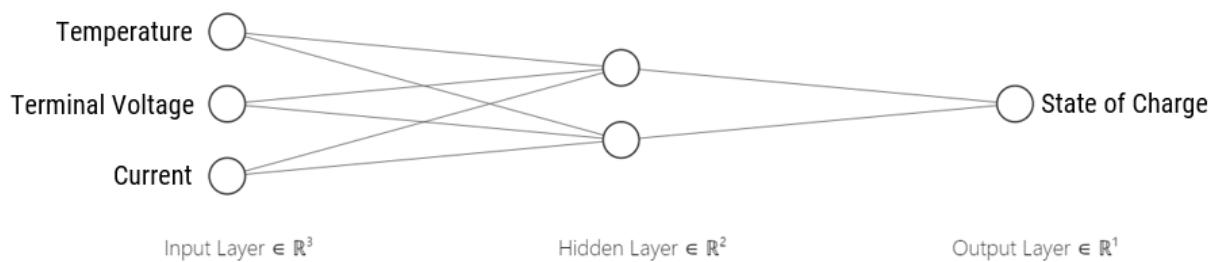
The hyperparameters and optimizer choices are presented in the table below:

Epochs	Batch Size	Optimizer	Learning Rate	Loss Function	Metrics
300	200	Adam	0.01	MSE	MAPE/MAE

The linear architecture was chosen to simulate a linear regression problem. The layers are densely connected. An Adam optimizer is used instead of gradient descent. The activation function is linear which implies that the gradient is constant and not dependent on the training data. The epochs, batch size and learning rate are selected by manually tuning the parameters using multiple for loops.

Data for both the linear and non-linear model are split into a training set, and a test set randomly. 80% of the values are stored in the training set while 20% are stored in the test set. The training set is used to update the neural network weights while the test set is used to evaluate the neural network.

Neural Network Non-linear Architecture:



The architecture is presented in the table below:

Layers	Input	Hidden Layer	Output
Number of Neurons	3	2	1
Activation Function	N/A	Sigmoid	Relu

The hyperparameters and optimizer choices are presented in the table below:

Epochs	Batch Size	Optimizer	Learning Rate	Loss Function	Metrics
300	200	Adam	0.01	MSE	MAPE/MAE

A general rule of thumb in designing an initial neural network is to select one hidden layer with several neurons between the input and output layer. Thus, a hidden layer of 2 neurons is selected. The final activation functions selected are sigmoid for the hidden layer and rectified linear unit for the output layer. The layers are densely connected. The sigmoid and relu functions are given by the following equations respectively:

$$Y = \frac{1}{1 + e^{-x}}$$

$$Y = \max(0, x)$$

The activation functions were selected based on performance. *Sigmoid*, in general, reported higher accuracies than *tanh*. Relu activation functions reported a higher accuracy than sigmoid for the output layer. However, Relu did not converge if it was chosen for the hidden layer. The reason is due to the commonly known problem of dying Relu. This problem occurs when the gradient is negative, and the output is also negative or zero in the case of Relu. When all the outputs are zero, the neural network is unable to learn as the gradient of the Relu function at 0 is also zero. When the gradient is 0, the system is assumed to be at a minimum, and the neural network weights aren't updated.

For the output layer, at learning rates of 0.1 or higher the dying relu problem occurs. Thus, the learning rate is lowered from 0.1 to 0.01 for the non-linear model.

Other hyper-parameters such as the number of epochs and the batch size were optimized using by manually tuning using for loops. The loss function and metrics are detailed by the equations below. A summary of the Adam optimization algorithm is presented with details available in the reference paper.

Lastly, a min max scaler is used to preprocess the data fed into the non-linear neural network. The scaler is chosen to try and improve learning and normalizing gradients across all the features.

$$X = \frac{X - X_{\min}}{X_{\max}} * (X_{\max} - X_{\min}) + X_{\min}$$

Mean Squared Error (MSE):

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_{predicted} - y_{actual})^2$$

Mean Absolute Error (MAE):

$$MAE = \frac{1}{n} \sum_{i=1}^n |y_{predicted} - y_{actual}|$$

Mean Absolute Percentage Error (MAPE):

$$MAPE = \frac{1}{n} \sum_{i=1}^n \left| \frac{y_{predicted} - y_{actual}}{y_{actual}} \right| \times 100\%$$

Adam Optimizer Equations [Reference]:

Given gradient g_t where t represents the current iteration the following set of equations are used to update the neural network weights:

$$m_t = \beta_1 m_{t-1} + (1 - \beta_1) g_t$$

$$v_t = \beta_2 v_{t-1} + (1 - \beta_2) g_t^2$$

Where β_1 and β_2 represented weights for a weighted average of the mean estimation m and variance estimation v which are initialized as 0. These two estimates are corrected for bias to yield $m_{t,corr}$ and $v_{t,corr}$. The following equation then calculates the weight update:

$$\theta_{t+1} = \theta_t - \frac{\eta}{\sqrt{v_{t,corr} + \epsilon}} * m_{t,corr}$$

where n is the general learning rate (the hyperparameter selected), θ represents the weights of each iteration and ϵ represents a smoothing factor.

Results

Simulations

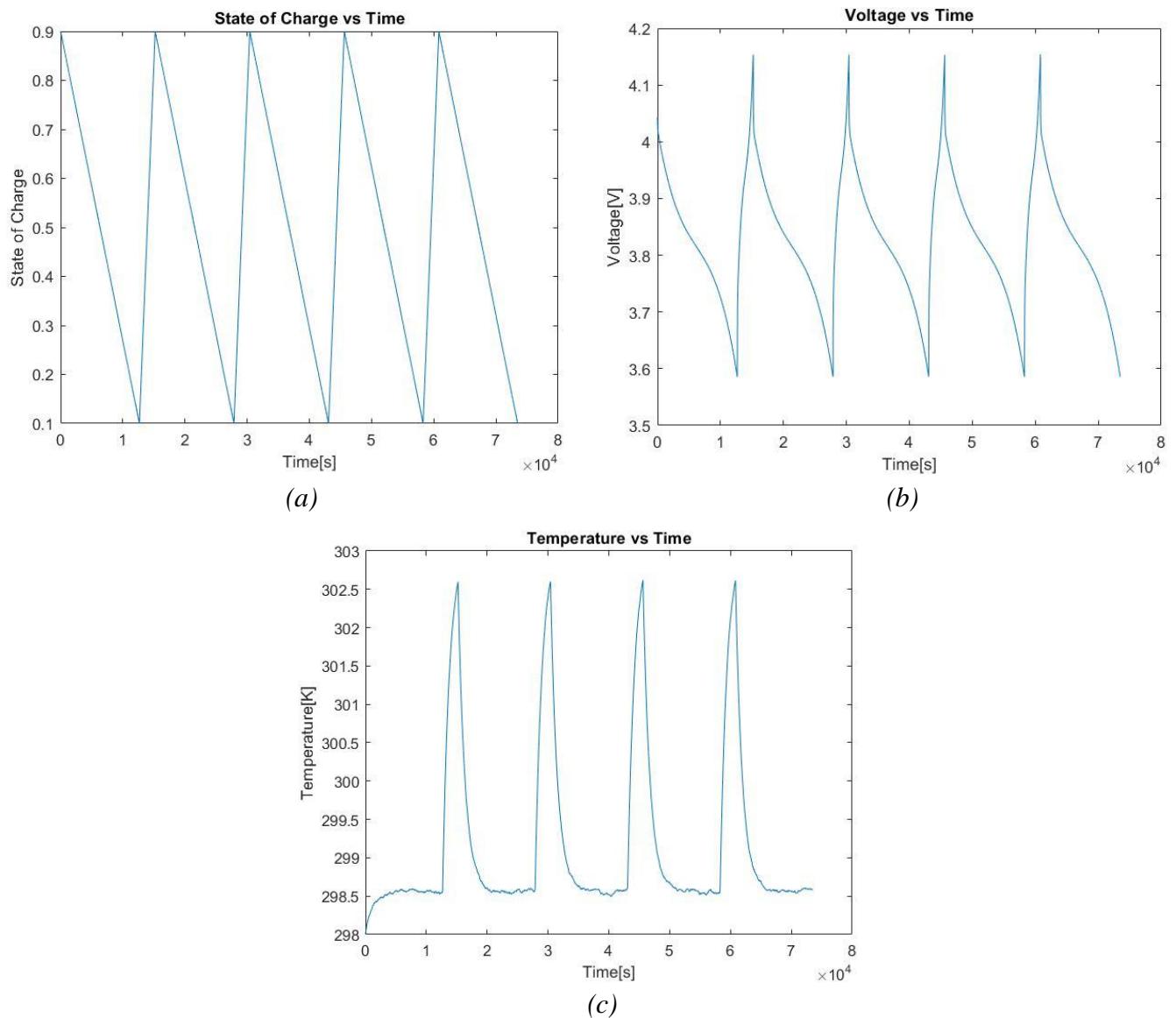


Figure 3:(a) State of Charge vs Time;(b) Voltage vs Time;(c) Temperature vs Time

Figure 3(a) shows that the state of charge changes from 0.9 to 0.1 as we have chosen the charging and discharging times to correspond to these values.

Figure 3(b) shows that the voltage sharply increases during the charging cycles but decreases slowly during discharging cycles. The voltage depends on the current, but there is a time lag between a change in the current and change in voltage due to the capacitors in the circuits. This explains why we see the change in voltage corresponds to an increase or decrease of current, but the change is not perfectly conformal to the change in current.

Similarly, *Figure 3(c)* shows that the temperature sharply increases during charging cycles as we have higher current but decreases during discharging cycles due to low current. The minor fluctuations in temperature are due to the noise in the ambient temperature.

Extended Kalman Filter

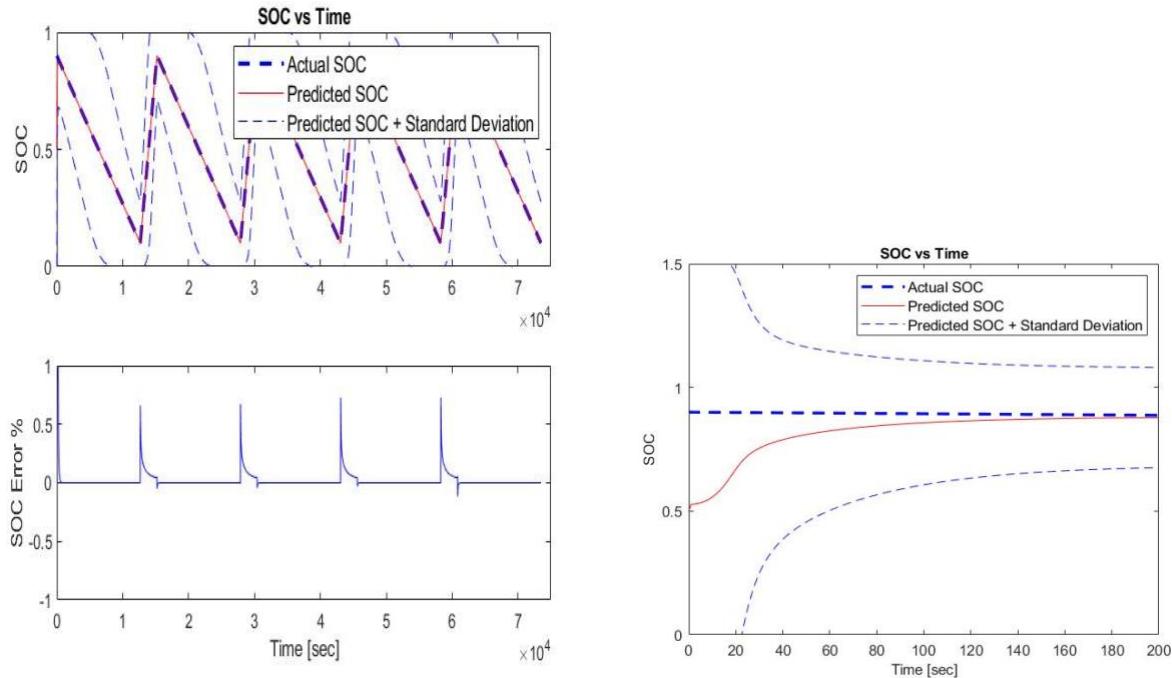


Figure 4:a) State to Charge and SOC Relative Error vs Time b) SOC vs time (zoomed)

The Current is the input function and cyclic in nature with a constant charging current of 2.5 A and a discharge value of -0.5A. As seen from the state space model and EKF equations, SOC derivative is a linear function of current. Hence, if the current is constant over a time period we expect the SOC graph to look linear over that time period. As observed, the state of charge linearly decreases with time when the battery is discharging and increases linearly when the battery is charging. The error in SOC comes from equipment measurement error and temperature effects. As expected the error shoots up to 0.6% every time the charging cycle changes. Finally, if we zoom our SOC vs Time graph we can see that it converges in about 480 seconds (extending the X-axis).

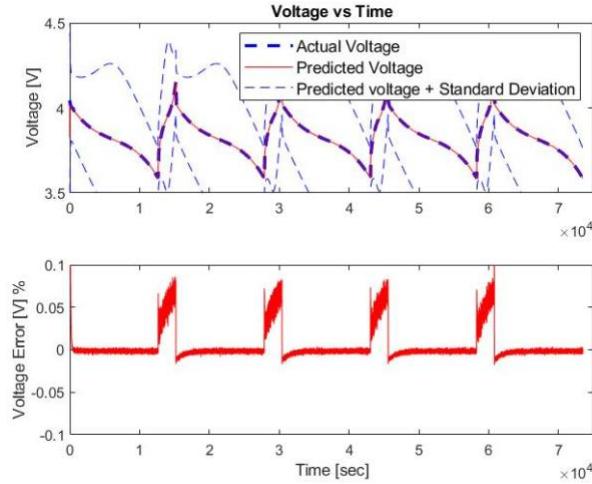


Figure 5: Voltage and Voltage Error vs time

Figure 5 shows the relationship between the time and voltage Error. By voltage error, we mean the difference between the voltage calculated by accounting for the change in internal resistance due to the heating up of the battery after prolonged use and the voltage calculated without accounting for it.

Hence, the Extended Kalman Filter is able to reduce the error and converge both voltage and SOC over time. The EKF manages to converge the predicted value of SOC to actual SOC from Simulation in around 8 minutes. Every time the charging situation changes, EKF notices a slight increase in SOC error to about 0.5% which is very reasonable given the effects of temperature and measurement noise.

Neural Network

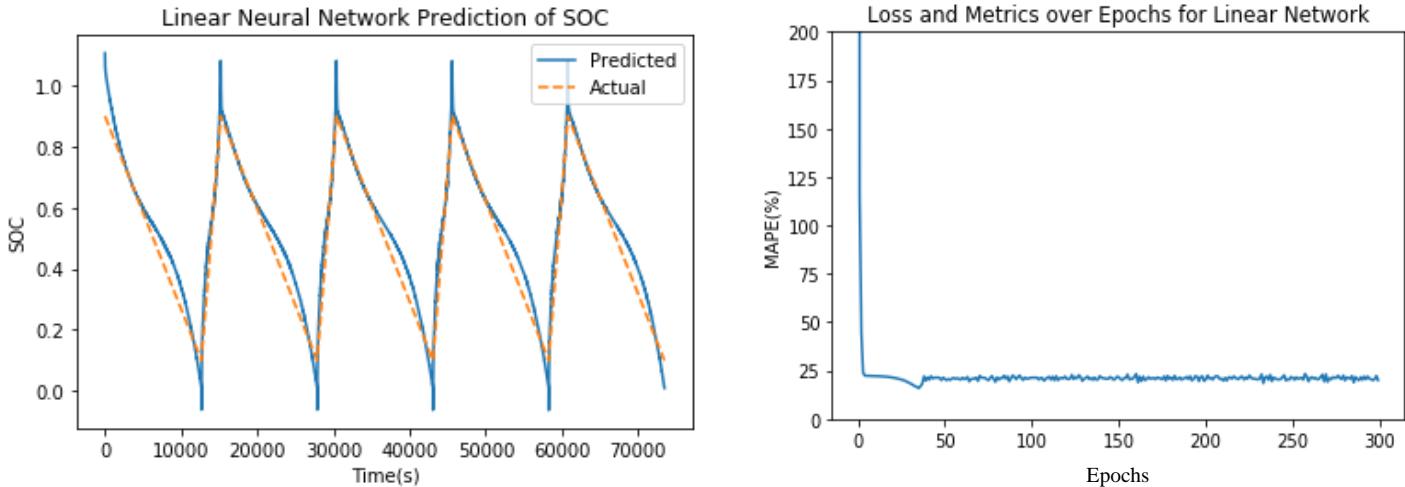


Figure 6:(a) Linear model prediction of SOC; (b) Mean Absolute Error for SOC as a function of Epochs

Figure 6(a) shows that the linear neural network can relatively predict SOC using just voltage, current and temperature as the inputs. As mentioned, SOC is expected to linear as the charge and discharge currents are constant. However, the linear neural network prediction of SOC is non-linear. This is due to the nature of a linear neural network model. Since the gradient of linear function is independent of the input, the neural network just maps the inputs to the output by a linear combination. Thus the non-linearity in the voltage and temperature causes the prediction to be non-linear. Thus linear neural networks are limited in accurately predicting SOC. *Figure 6(b)* shows that the mean absolute percentage error hovers around 23-25%. This can be fatal for an electric vehicle as an empty battery could potentially be read as 25% full.

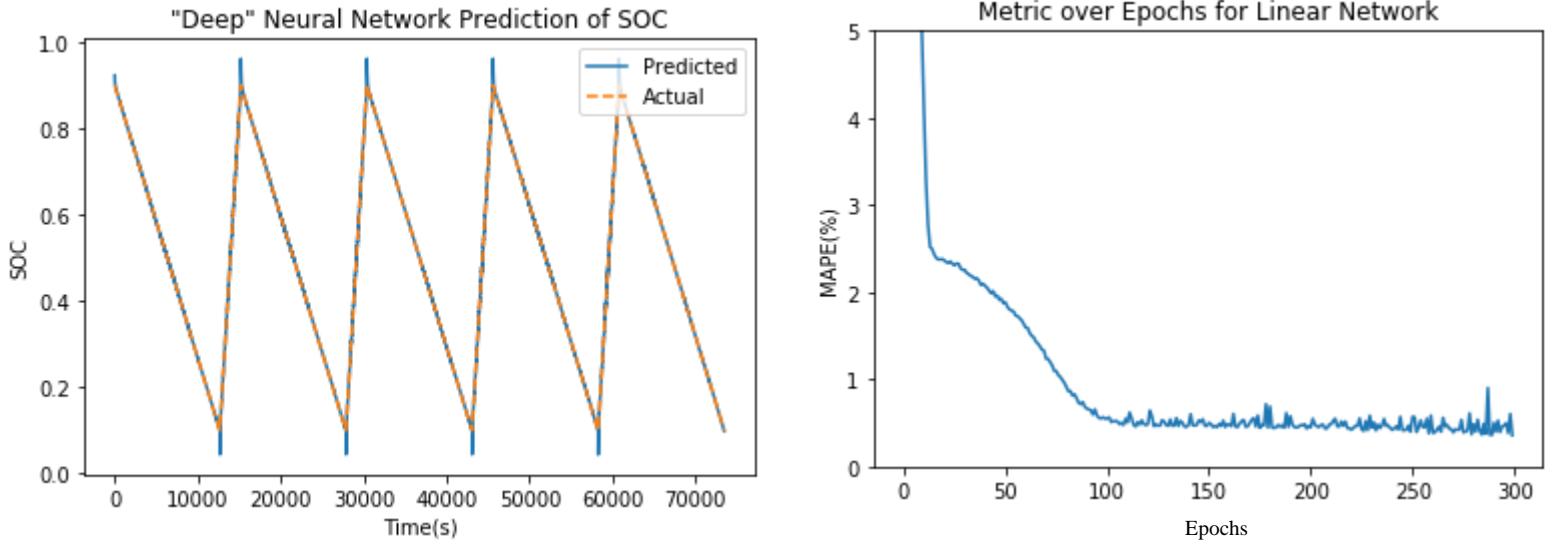


Figure 7:(a) Non-linear model prediction of SOC; (b) Mean absolute error vs Epochs

Figure 7(a) shows the prediction using the non-linear neural network model. The inputs are temperature, voltage and current and the output is SOC. The prediction accuracy is much higher than the linear model, converging at around 0.5% as shown in *Figure 7(b)*. This is expected as a non-linear model can map nonlinear data into a linear SOC better than a linear model. However, MAPE does not seem to be an accurate metric to evaluate the performance of neural networks in predicting SOC. *Figure 7(a)* shows that while the neural network is generally accurate, it is highly inaccurate near the upper and lower SOC limits of 0.9 and 0.1. The most important SOC predictions are when the battery is nearly fully charged or nearly depleted. Thus, the MAPE of 0.5% is deceiving. In the future, a custom metric function that penalizes the neural network for inaccurate predictions near these limits is necessary to improve neural network performance.

IV. Discussion

In this report, we have obtained a dataset by running a simulation based on literature models that relate current as a function of temperature. We have then built an EKF model that is blind to temperature and tries to reduce the error due to temperature effects and measurement noises. We observed that given suitable matrix input, EKF was able to converge the predicted value and actual value of State of Charge in less than 8 minutes. EKF performed well given the fact that even when the charging cycles changed EKF error did not go above 0.5%. Overall the mean absolute % error

was 0.0406% for the EKF. Next, we built an offline neural network that relied on Voltage, Current and Temperature as an input. The network was able to converge to a mean absolute percentage error of 0.3% using 72000 data points (representing 100 hours of data). The only problem with Neural Network, in this case, is that it does not work very well for high and low values of SOC due to the sudden change in current values. This might be a problem since it is important to predict SOC especially when we are changing from a charging cycle to a discharging cycle or vice versa.

From the results displayed above we can see that EKF model performs better than Neural Network in terms of percentage error and the time it takes to converge. However, EKF requires one to have a deep understanding about how the battery works, an Equivalent Circuit Model, physical equations and parameter values, none of which are required in Neural Network. Also, the value of these parameters might change over time and in some rare case even the governing equations. Neural Network is well suited to adapt to these changes unlike EKF that uses a fixed value for these parameters. However, EKF has its advantages too. Unlike Neural Network model, EKF gives more information with lower number of inputs, it also provides us with a confidence interval for the value it predicts and obviously converges better and faster. It's well established and better understood than Neural Networks. Hence the choice of which one is better, ultimately depends on the user requirements. If he has the time to understand the physics, do experiments to find parameter values, wants a confidence interval, requires lower error, convergence time and number of inputs EKF should be his first choice. In all other cases, Neural Network is better.

As seen above, we do not have any real world data and all our models are tested against simulation data, the future CE295 teams could look at real-world data and see which model performs consistently better given various other errors that might be present in the model like errors due to changing convection coefficient of heat with temperature, changing values of heat capacity with temperature, decrease of state of charge due to leakage when no current is flowing, etc. Given these other errors present in real-world data, it would be interesting to see which model performs better. Once we are able to answer this question, it would benefit the Battery Management Systems help increase energy efficiency, reduce costs and drive the adoption of electric vehicles reducing carbon emissions. Building on this work, the future CE295 students with their control and system tools are uniquely positioned to answer this question and advance sustainability in energy systems.

V. Conclusion

In conclusion, we saw that the EKF has a mean absolute error of 0.0406% without the inclusion of temperature whereas the Neural Network has a mean absolute error of 0.3% even with the addition of temperature. Therefore, the extended Kalman Filter has better accuracy to predict the SOC model regardless of the temperature as an impact to the equivalent circuit model. The Neural Network is not suitable to predict the battery SOC when the model has already existed. Therefore, when we know the exact model to predict the SOC, the extended Kalman Filter is the best choice. Other advantages of the neural network include the capability to adapt to changing physical models and the ability to integrate and work with other systems quickly. For example, if the battery deteriorates over time, the physical model and some constants will change. A well-tuned neural network would be able to update its weights to capture these changes. A neural network can also be simultaneously applied to different systems without knowing the models, although likely with varying amounts of success.

VI. References

- [1] Nishi, Y. (2001). Lithium ion secondary batteries; past 10 years and the future. *Journal of Power Sources*, 100(1-2), pp.101-106.
- [2] Huang, S., Tseng, K., Liang, J., Chang, C. and Pecht, M. (2017). An Online SOC and SOH Estimation Model for Lithium-Ion Batteries. *Energies*, 10(4), p.512.
- [3] He, H., Xiong, R. and Fan, J. (2011). Evaluation of Lithium-Ion Battery Equivalent Circuit Models for State of Charge Estimation by an Experimental Approach. *Energies*, 4(4), pp.582-598.
- [4] Feng, F., Lu, R., Wei, G. and Zhu, C. (2015). Online Estimation of Model Parameters and State of Charge of LiFePO₄ Batteries Using a Novel Open-Circuit Voltage at Various Ambient Temperatures. *Energies*, 8(4), pp.2950-2976.
- [5] Ismail, N., Toha, S., Azubir, N., Md Ishak, N., Hassan, M. and KSM Ibrahim, B. (2013). Simplified Heat Generation Model for Lithium-ion battery used in Electric Vehicle. *IOP Conference Series: Materials Science and Engineering*, 53, p.012014.
- [6] M. Charkhgard and M. Farrokhi, "State-of-Charge Estimation for Lithium-Ion Batteries Using Neural Networks and EKF," in *IEEE Transactions on Industrial Electronics*, vol. 57, no. 12, pp. 4178-4187, Dec. 2010.
- [7] Xia, B., Wang, H., Wang, M., Sun, W., Xu, Z. and Lai, Y., 2015. A new method for state of charge estimation of lithium-ion battery based on strong tracking cubature Kalman filter. *Energies*, 8(12), pp.13458-13472.