EVS29 Symposium
Montréal, Québec, Canada, June 19 - 22, 2016

Data Management for Geographically and Temporally Rich Plug-in Hybrid Vehicle "Big Data"

Spencer Vore[1], Mark Kosowski[2], Zachary Wilkins[1], Thomas H. Bradley[1]

[1] *Department of Mechanical Engineering, Colorado State University, Fort Collins, CO,*
Thomas.bradley@colostate.edu, (970) 491-3539 (office)

[2] *Electric Power Research Institute, Palo Alto, CA,*
mkosowski@epri.com, (248) 421-7124 (office)

Summary

The Electric Power Research Institute (EPRI) and its project partners have developed some of the highest resolution and most complete Light- and Medium-Duty Plug-in Hybrid Electric Vehicle Truck operational data for Odyne and Via trucks. This data was collected through a CDMA / GMS transmitter plugged into the CAN communication bus of the fleet. This paper discusses the process of transforming these raw datasets into a scientific database of driving and charging events using data quality management, filtering, processing and decision support tool development. The result is a dataset with demonstrable utility for vehicle design, policy analysis, and operator feedback.

Keywords: PHEV (Plug in Hybrid Electric Vehicle), Medium-Duty, Bucket Truck, Data Acquisition, Policy

1 Introduction

The Electric Power Research Institute (EPRI) and the US utility industry are interested in understanding the means by which grid electricity will enter the transportation energy sector. The quantity, timing, and statistical distribution of electricity consumption have near-term and long-term effects on utility planning for loads, assets, profitability, and sector growth [7, 12].

In the near-term, the function of the various types of OEM (original equipment manufacturer) electrified vehicles that are for sale is the most effective indicator of how consumers will use electrified vehicles [1, 7, 14]. To gather data on the function of these vehicles and the behavior of their users, EPRI and The US Department of Energy have developed a program to gather and store GPS (geographical positioning system) derived location data along with detailed vehicle operation data from a sample set of light- and medium-duty PHEV trucks being utilized as utility bucket trucks and general support trucks [14]. Because of the very large scope of this effort, there exists a need to synthesize these large datasets into databases and toolsets that can communicate the results of these studies to researchers and stakeholders.

With the development of distributed data collection technologies, many other researchers have developed techniques to collect, store and synthesize operation data from vehicle fleets [1, 8, 9, 10, 11, 12, 13, 14]. Characterizing the operation of PHEVs is of particular interest to transportation system researchers because of the well-documented dependency of vehicle fuel consumption on individuals' driving and charging habits [1, 2, 3, 6, 8, 10, 11, 12]. In many previous studies [1, 10, 11, 12], datasets have relied on the data collected from private individuals, and have therefore encountered privacy and traceability concerns. In this study, we have collected vehicle operation data from commercial light- and medium-duty vehicles. In general, these vehicles represent a unique study subset of the US vehicle fleet that have not been studied in detail before. In addition, because these are fleet-owned vehicles, the collection of correlated GPS and vehicle operational data does not present as many privacy concerns.

On the other hand, the increased scope of this data collection effort has led to a variety of technical and “big data” management challenges that have been addressed. In discussing the “reality” of these data collection projects, their limitations, and the technical means used to generate results from them, this paper seeks to improve and contribute to the state of the art in the field of large-scale big data transportation system data collection projects.

Colorado State University (CSU) has developed an algorithm to identify drive and charge events from the raw vehicle data files with the 1-second data sampling frequency. These results can be viewed directly in an Excel Spreadsheet format with charts as demonstrated in the Decision Support Tools Development section, or used for additional scientific analysis, as demonstrated in the Results section. By summarizing the low-level high frequency data into a list of higher level events, the data size can be greatly reduced into a format that provides great utility for answering policy, vehicle design, and operational questions about the vehicle fleet using significantly less computational power.

2 Methods

2.1 Dataset and Project Overview

In this project, EPRI and CSU sought to summarize the very large raw dataset to produce a smaller subset of data with immediate utility to answer policy, design, and operational questions about PHEV vehicles. This summary data was then used to calculate the actual energy consumption of these light- and medium-duty vehicles, investigate charging frequency, and generate utility factor curves. This was done using vehicle-derived data, including driving distance, battery state of charge, fuel injection rate into the engine, charging station power, and more.

For this analysis, the EPRI Commercial Truck dataset was used [14]. This dataset was brought online in January 2015, and the analysis in this paper covers data collected through July 2015. There are two different fleets of vehicles in this dataset. The first fleet consists of 119 medium-duty Odyne Electric Trucks, and the second fleet consists of 177 light-duty Via Electric Trucks [14].

To collect this data, tracking devices were integrated with the CAN communication bus of these vehicles, and a CDMA / GMS transmitter was used to send information to a central database. Most of the CAN signals were recorded every second while the vehicle was in operation, so the data sampling frequency was very high [14]. High-rate data sampling has many advantages. For example, it provides the capability to investigate short term events and improves the overall accuracy of calculation. However, the high data resolution also presents challenges in terms of managing and processing large amounts of data. In total, about 160GB of CSV data were downloaded from the central database and processed for this analysis work. A MATLAB script running on a single core took multiple days to fully analyze the dataset. Note that not every CAN communication bus signal was utilized for this study, and the data that was downloaded from the full database is just a subset of the total available data from the vehicle fleet.

Due to privacy considerations based on the information contained in the database, this dataset is not publically available and is considered protected information. EPRI is solely responsible for granting and denying access to this dataset.

2.2 Data Management and Quality

The dataset was originally downloaded directly from EPRI’s main database hosted on Amazon Redshift using an R script with embedded SQL. The Amazon Redshift database already had some low-level filtering applied to the data to eliminate known bad sensor data. The SQL embedded in R was used to filter out irrelevant CAN signals at the very beginning of the analysis, so they wouldn’t have to be parsed through later in the resulting CSV files. These R scripts created a single CSV file for each vehicle month, which were then processed in MATLAB.

The first step of the MATLAB processing involved converting the CSV data into a .mat file data format and performing some basic data validation. Resaving the data added time to the overall processing sequence, as the data had to be written to and read from the disk an extra time. However, it was also generally more beneficial to save intermediate data, so that later processing steps could be rerun without reprocessing all of the data through the preliminary steps. Due to the large file size, a feature was built into the code that would only read in a portion of a CSV file to break up those files into multiple .mat files. These .mat files could then be loaded individually into the computer memory for analysis to avoid overloading the available RAM.

Basic data validation was also performed in this preprocessing step. The basic data validation primarily verified the data file format, such as correct number of table columns and proper delimiters. Most of the data was properly formatted, but it was also important to remove even very uncommon errors to prevent later analysis scripts from crashing. This data cleaning made the general analysis process more robust.

2.3 Data Processing and Filtering - Compiling Driving Events List from Raw Data

To analyze the raw data into meaningful results, the data processing and filtering steps were closely integrated. Often, the data was re-filtered after each processing and analysis step. This section walks through the process of how the raw 1-second sampling frequency data was converted into a meaningful list of vehicle driving events. These driving events were also presented along with associated statistics, such as trip distance, time duration, energy usage, and State of Charge (SOC). Then, the next section shows how this algorithm to identify driving events can be modified to identify charging events.

The biggest challenge to identifying these drive events was data quality, as there are some situations where the data for an event is incomplete. Therefore, it is critical for this algorithm to be reasonably robust when applied to imperfect data that contains some errors, while remaining effective enough to still produce realistic results.

A drive event is considered to be a single vehicle trip starting when the vehicle begins moving and ending when the vehicle stops moving. Each drive event is recorded in a single spreadsheet row, along with additional information about the date, the time, the trip duration, the distance traveled, the fuel and electricity consumption, the initial battery state of charge, the final battery state of charge, and information identifying the individual vehicle.

The drive mode for each vehicle was also identified along with each drive event, and these are categorized as Charge-Depleting (CD), Charge-Sustaining (CS), or Blended driving mode. A charge-depleting mode is when the vehicle is driving only on battery power, and this drive mode is powered by energy from the electric grid. A charge-sustaining mode is a traditional Hybrid Electric Vehicle (HEV) mode, in which energy from the gasoline engine is recaptured and stored in the battery and the source of vehicle energy comes from conventional gasoline. A blended drive mode is a variation of charge-depleting mode where the gasoline engine is used to slow the rate of battery depletion, and the vehicle is operating on both grid and gasoline energy. For Odyne, the vehicle only operates in a blended and CS mode, so there is no CD mode on Odyne. On Via, the vehicle only operates in CD and CS modes, and the Blended mode is used to categorize driving events where the vehicle transitions from CD to CS mode [14]. How the driving modes are defined could also be further refined in future analysis work.

Data processing and filtering began after the data was converted into a MATLAB data format, as described in the Data Management and Quality Section. When the analysis was run, any individual vehicle data file that had less than 5000 logged data points in a month was removed from the analysis. These small data files could have been the result of vehicles that were never driven or vehicles that did not have a functional tracking system installed. Then, the following data points for relevant CAN bus messages were extracted: Vehicle Speed, Vehicle Odometer, Battery State of Charge, Vehicle Charging Station Voltage (when vehicle is being charged), Vehicle Charging Station Current, and Engine Fuel Injection Rate.

Once these messages were extracted, each signal was run through a time stamp filter that removed duplicate time stamps. This step was very important, as occasionally there were multiple different values reported at the same time for a single CAN bus signal. These multiple reported values caused significant problems later in the data analysis script if not removed. The algorithm kept the first reported value and removed the rest.

Next, the data points from the fuel injection rate, battery voltage, battery amperage, and battery SOC were interpolated onto the timestamp values for vehicle speed using linear interpolation. Since the data was collected asynchronously, interpolation was a good method to realign and project all of the data vectors for different signals onto a single time stamp value. Vehicle speed timestamps were used for the projection target because these data points were generally only recorded when the vehicle is driving.

Drive events and modes were then estimated based on the following criteria:

1) Vehicle speed greater than 0.01 mph was considered driving (vs. other events in the data such as charging, etc).

2) Charge-Sustaining (CS) mode occurred when the battery was at less than 5% SOC for Odyne, and was at less than 22% SOC for Via.

3) Charge-Depleting (CD) mode occurred when the vehicle was driving and not in Charge-Sustaining Mode.

See the sample MALTAB code below to illustrate this logic process. Note that these steps were applied to all of the raw data points collected at the 1-second sampling frequency. Later on, once more information was compiled about the driving events, the CD and CS modes were redefined.

```
% Filter for identifying all drive events
Drive_Filter = Drive_Speed > 0.01;

% State of charge limit for CS Mode in Battery % SOC
CS_SOC_Limit = 5; % Odyne
% CS_SOC_Limit = 22; % Via

%% Primary Filters
% A CS Event is when the vehicle SOC is below the CS SOC Limit
CS_Filter = (Drive_Filter & Drive_SOC < CS_SOC_Limit);

% A CD Event is any time the vehicle is driving and is not in CS mode
CD_Filter = (Drive_Filter & ~CS_Filter);
```

Next, to find the start and end locations for each drive event, the script looked for a change in the vehicle trip conditions where the drive conditions transitioned from true to false in the list of all raw data points. This was done by creating a logical vector based on the drive criteria, offsetting the values by one index value to create another offset vector, and then subtracting the vectors. Any non-zero value in the resulting vector indicated a change in driving conditions, and the sign of the value indicated whether the vehicle started or stopped driving. To augment this method, a secondary method was also implemented to identify large gaps between time stamps greater than 300 seconds. Any event with a time stamp gap greater than 300 seconds was split into two different events at the time gap, as it was assumed that the lack of recorded data indicated that the vehicle was not running. Note that the data collection system installed on the vehicles only recorded data points when the vehicle was in operation or charging. If any of the start and end times for an event were equal, that event was removed from the dataset. Next, if any 2 events took place less than 120 seconds apart, they were recombined into a single event, as a stop of less than 2 minutes was considered inconsequential. Finally, any drive events with less than 5 data points were removed to prevent small idiosyncrasies in the data from being recorded as significant events. Filtering out these small blips in second-by-second sampling frequency data was very important, because there were often small anomalies in the data that added noise to the bigger picture. Combined, these methods robustly identifed events when the vehicle was driving.

Once the start and end times of drive events were located, those starting and ending time stamps (or index values) were used to locate other relevant information from the dataset to calculate statistics of interest for any particular driving event. The statistics calculated for each event include distance traveled, fuel used, electric energy used, change in SOC, the start time and date, and the event duration. A trapezoidal integration was also used to find the integrated values of some variables, such as fuel use, distance traveled, and electric energy usage when only the time-rate signal was available. Note that speed was integrated to calculate distance even though an odometer signal was available because some of the Via odometer data was known to be faulty. Ideally, an odometer signal should be used to calculate distance if it is available, but that was not the case for this dataset.

At this point, some of the previously defined charge-depleting trips were redefined to be either blended or charge-sustaining mode, based on additional information from the relevant summary statistics for each event.

For this data analysis, a vehicle charge-depleting trip was redefined as blended mode when:

1) The delta SOC for a previously identified charge-depleting trip was negative (i.e. charge was decreasing),
2) The fuel consumption was positive, and
3) The initial SOC was above the CD / CS transition SOC that was set at 5% for the Odyne analysis and 22% for the Via analysis.

Charge-depleting trips were redefined to be in charge-sustaining mode based on the following criteria:

1) Delta SOC increased over the duration of the event.
2) Fuel consumption was greater than zero and the vehicle was not in blended mode.

Every other condition was considered to be a charge-depleting mode. See sample MATLAB code below to illustrate this logic:

```
Blended_Condition = ((CD_SOC_Delta < 0) & (CD_Fuel > 0) & …
   (CD_SOC_Initial > CS_SOC_Limit));
CS_Condition = (CD_SOC_Delta >=0 | (CD_Fuel > 0 & ~Blended_Condition));

for Iteration = 1:number_of_charge_depleting_events
  % Determine Drive Mode
  if Blended_Condition(Iteration)
    CD_T_Mode{Iteration} = 'Blended';
  elseif CS_Condition(Iteration)
    CD_T_Mode{Iteration} = 'CS';
  else
    CD_T_Mode{Iteration} = 'CD';
  end
end
```

This last step produced the final values used to define the driving event modes in the event summary spreadsheet. The final results were then exported into an Excel spreadsheet format. This completed the data processing and filtering needed to identify drive events. These results can be reimported into MATLAB for additional analysis or viewed as a stand-alone document.

2.4 Data Processing and Filtering - Compiling Charging Events List from Raw Data

The algorithm used to identify charging events is similar to the algorithm used to identify driving events. For this analysis, a charge event is considered to be a time duration when power was being delivered to the battery from the charging station. Alternatively, a charge event could possibly be defined as a time when there was a voltage across the charging station. Using only voltage would account for time when the vehicle was plugged in but already fully charged, as power transfer stops when the battery is full.

Charge events are also identified as Level 1, Level 2, or Level 2+ charges, which depends on the charging station type. A Level 1 charger has a charging level of 120 volts, a Level 2 charger has a charger voltage of 240 volts, and a Level 2+ charger has a charger voltage of 240 volts with a power greater than 3.3kW [5]. The charge level for charging events was determined by finding which of the rated charging station voltage levels the actual charging voltage signal was closest to.

Here are some notable differences in the charging events summary calculation when it's compared to the driving events summary calculation:

1) This calculation used signals for the charging station voltage, charging station current, and vehicle battery SOC.
2) A vehicle was considered to be in a charging event when:
    a. The charging power was positive and greater than 100 watts. Charging station power was calculated by multiplying the charging station current signal by the charging station voltage signal.
    b. The duration of the charging event was greater than 2 minutes.
3) All signals were interpolated onto either the Charging Station Voltage or Charging Station Current signal time stamps. The projection target was chosen to be the signal with the most time stamp values.
4) Any event with a time stamp gap greater than 2400 seconds was split into two different charging events at the time gap. Any two events with a time stamp gap of less than 2400 seconds were combined into a single charging event.
5) Charging events with no change in SOC were removed. Then, the final state of charge was defined as the maximum SOC for the charging events, due to errors caused by approximating end locations of the charging event. This removed some negative delta SOC values. After the final SOC was redefined, any charging event that still had a negative change in SOC was removed.

2.5 Code Validation

Result validation is very important to the data analysis process. Just because a data analysis script reads in data and produces numbers does not guarantee that those results are useful or accurate. Therefore, it is critical to build validation tools into the data analysis script to track the script execution process so intermediate steps can be evaluated. Below is a list of some of the validation tools that were developed for this project.

- A system to track custom error messages was embedded into the script, so that problems with individual files could be traced back to a specific point in the dataset. The data analyst still had to define and program the error code definitions into this framework, based on their discretion.
- In addition, there was an option to easily create custom graphs of intermediate data from the data analysis process for each vehicle.
- A run log of console print screen output was also recorded. It was up to the data analyst to add print screen statements into the code to document the script execution process in a useful manner.

2.6 Decision Support Tool Development

Once the raw data was compiled into an event summary spreadsheet, it was very easy to post-process the data into charts and figures. Additional software could be written to visualize the data, such as a web app, or the data can be processed directly in Microsoft Excel. Below are some sample Excel graphs to demonstrate how the event summary data can be quickly visualized to support fleet management decisions, vehicle design, public policy, and research questions.

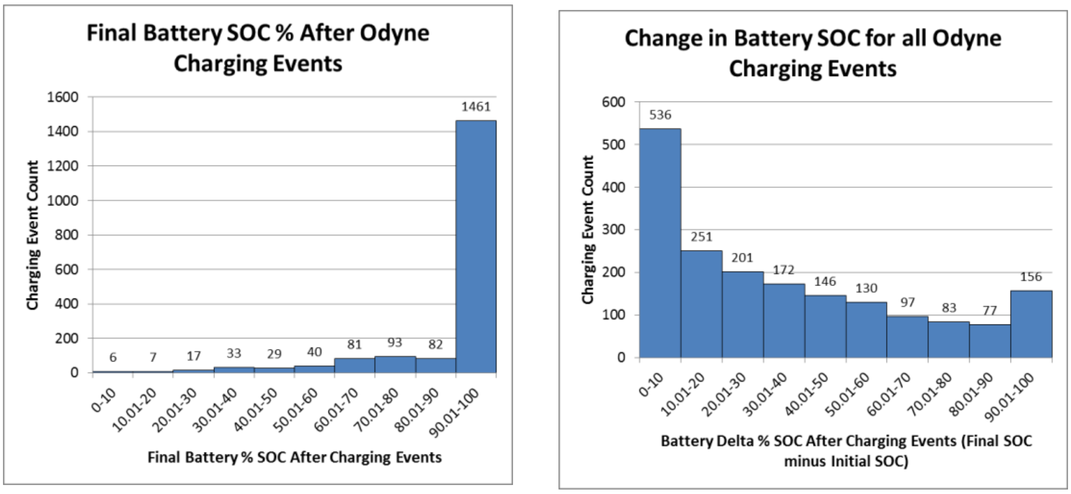

Figure 1: Examples of Data Visualizations in Excel for Decision Support

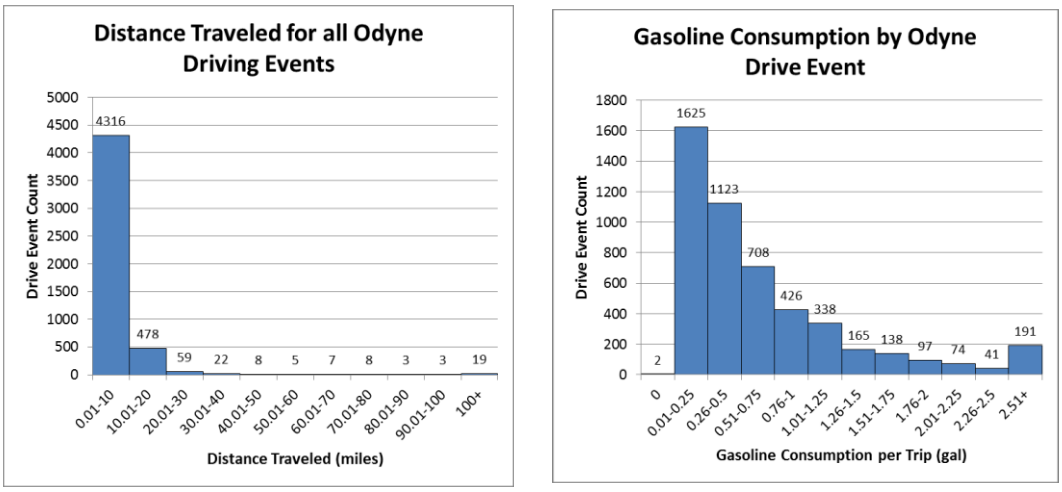

Figure 2: Examples of Data Visualizations in Excel for Decision Support

2.7 Recommendations for future analysis work

For future work, here are some additional suggestions that could possibly be used to improve the data analysis methodology outlined in this paper. These suggestions were not widely implemented in this study, but are instead seen as logical next steps to build on the methods presented in this paper.

As datasets grow larger, it will become more important to parallelize the data analysis algorithm and run it on a computer cluster or multi-core machine. For this study, some very initial preliminary work was done in the area of code parallelization, but it was found that the same results could be produced with less coding work and hardware investment by just letting the machines run for longer. However, if the data size grew by another order of magnitude or if tighter deadline ruled out multi-day run times, parallel computing would have a much larger payoff and should be considered.

More of the available CAN communication bus signals could be integrated into the Utility Factor calculation to develop additional alternate Utility Factor calculations. For example, since both electric charging energy and fuel injection rate are available, the ratio of electric energy to gasoline used could be used as an alternative measure of utility factor.

3 Application of Summarized Dataset and Results

Once the raw dataset is transformed into summarized event data, it's very easy to generate useful numbers and additional analysis to support decisions related to policy, fleet operation, and vehicle design. For example, this study generated utility factor curves from the summarized event data for the fleet of trucks, which supports both policy and vehicle design decisions [2, 7]. Utility Factor (UF) curves for PHEV Medium-Duty Work trucks have also never been published, so the results from Odyne are immediately useful. Then, to better understand how the number of daily charging events could affect utility factory, some additional statistics about the charging events were compiled. The charging event statistics support operator feedback and public policy by indicating that utility factor could be improved if the vehicles could be frequently recharged during the day [2, 6, 7, 11].

3.1 Utility Factor Curve Discussion

This section describes and presents the calculation of Utility Factor (UF) results for both the Odyne and Via Fleets. There are three curves on these graphs. The first curve is the standard utility factor curve in the SAE J2841 specification that was based on NHTS (National Highway Transportation Survey) data [4]. The second curve is the truck UF curve, calculated using the SAE J2841 methodology [4]. The third curve was calculated using the truck data, but with the SOC-based correction factor added to the standard SAE J2841 methodology to account for the possibility of more than one charge per day. It should be noted that the standard SAE J2841 methodology assumes that vehicles are fully recharged only once every day, which may or may not be an accurate assumption [1, 2, 4, 6]. Previous work has explored alternate Utility Factor definitions [2, 6].

The below equations are used to calculate the modified SOC based utility factor curve. These equations are a slight modification to the standard SAE J2841 Fleet Utility Factor (FUF) Equation:

$$UF(R_{CD}) = \frac{\sum_{d_k \in S} \min(d_k, R_{CD} \cdot C_k)}{\sum_{d_k \in S} d_k} \tag{1}$$

$$C_k = \frac{\max(|\Delta SOC_k|, 100 - SOC_{Cutoff})}{100 - SOC_{Cutoff}} \tag{2}$$

In the SOC-based utility factor curve Equation, C_k is a factor that increases the effective R_{CD} in the SAE J2841 Fleet Utility Factor Equation, based on the total daily change in battery State of Charge, ΔSOC_k. To calculate ΔSOC_k, the change in state of charge values for every vehicle trip in a day are added together. Note that the absolute value of ΔSOC_k is taken to ensure that the value is always positive. The term 100 - SOC_{Cutoff} represents the percent SOC change that would occur during one fully charge-depleting trip. The SOC_{Cutoff} value represents the battery state of charge level in which the vehicle transitions from a charge-depleting to a charge-sustaining mode, and in this analysis the value was set to 5% for Odyne and 22% for Via. Note that the SOC_{Cutoff} value depends on the specific model of vehicle and its hybrid control system architecture. The idea behind the C_k factor is that if a vehicle recharges during the day in between trips, its ΔSOC_k will be greater than its 100 - SOC_{Cutoff}, thus increasing the true charge-depleting range of the vehicle for the day. If ΔSOC_k is less than 100 - SOC_{Cutoff}, the utility factor equation for the day will be the same as the SAE J2841 utility factor. It should also be noted that if the vehicles are not charged at least once per day, the SOC-corrected utility factor equation will output the exact same curve as the standard SAE J2841 methodology.

Below are the two graphs that summarize the Utility Factor curves for both the Odyne and Via Fleets:

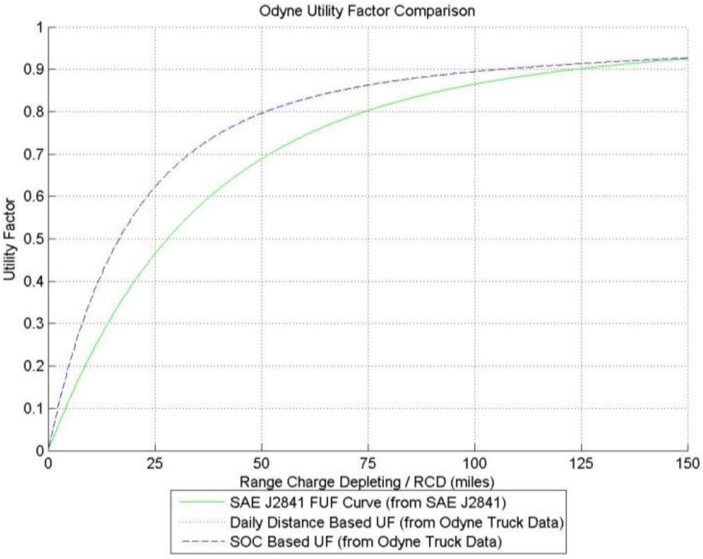

Figure 3. Odyne UF Curve

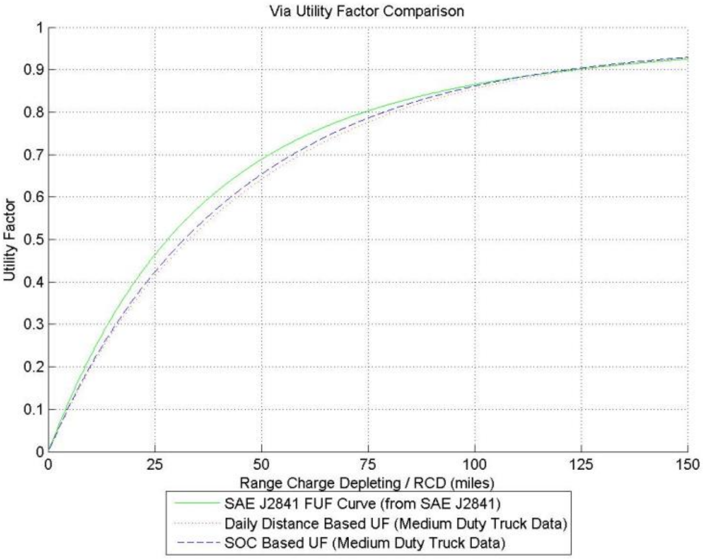

Figure 4. VIA UF Curve

For the medium-duty Odyne trucks, it can be seen that the UF is higher than the standard SAE J2841 UF curve. From a design perspective, this means that these trucks can be designed with a smaller battery than otherwise would be needed if the standard SAE J2841 UF curve was used to meet performance requirements. From a policy perspective, medium-duty trucks may not need as strict of emissions requirements as other commuter vehicles, as they are inherently driven in such a way that increases UF. The SAE J2841 UF curve is just an average of all vehicles in the US, whereas smaller subfleets within the set of all vehicles may have different usage and UF curves [1, 2]. However, it should also be noted that Odyne trucks do not operate in a true all EV (electric vehicle) mode until the battery is depleted, which is an assumption of the SAE J2841 methodology [4, 14]. Instead, the Odyne Trucks use a blended CD vehicle mode where some gasoline power is used to extend the range of the electric battery. So in reality, the actual gasoline displacement of the Odyne trucks will be less than the gasoline displacement estimated by these UF curves. The actual gasoline displacement of these vehicles could likely be improved if their control strategy was reprogrammed to have a true EV mode, or a blended driving mode that used more electric power than the current driving mode.

For the light-duty Via trucks, it can be seen that the UF is much closer to the standard SAE J2841 UF curve than the Odyne trucks. In this situation, it looks like the SAE J2841 UF curve does a good job of approximating the actual real world UF of Via trucks. However, there is still a slight difference, so depending on the accuracy needed, the UF curve presented in this paper may still be required. Unlike Odyne, the Via trucks operate in a true EV mode until the battery is nearly depleted, so the Via UF curves are likely representative of their actual performance in the field.

In the Odyne and Via Truck UF curves, it's also interesting to see that the SOC correction factor did not make a significant difference in the UF curve, so in this situation the standard SAE J2841 assumption of only one charge per day appears to be very reasonable. It's possible that the vehicles are only being charged once per day in line with the SAE J2841 standards, and if the charging pattern of the fleet changed, there would be a significant difference between the two UF methodologies. To better understand the story of why these two curves are so similar, charging event statistics are discussed in the next section to verify this hypothesis.

Overall, the truck UF curves show that light- and medium-duty commercial PHEV's are a great application for PHEV's. The fact that their utility factor curves are equivalent to or higher than the standard SAE J2841 curve means that on average, light- and medium-duty truck PHEV's can displace at least as much if not more gasoline than privately owned commuter PHEV's. Displacing gasoline with electric power can create many positive economic and environmental benefits [7].

3.2 Charging Summary Statistics Discussion

Some additional charging event summary statistics were generated to better understand the Truck UF curves in the previous section. Below is a table of these numbers for both Odyne and Via:

Table 1: Charging Event Summary Statistics for Odyne and Via

	Odyne	**Via**
Total Number of Vehicle Driving Days	1279	1679
Total Number of Charging Events	1850	1076
Average Number of Charging Events per Vehicle Driving Day	1.45	0.64
Mean percent change in battery SOC for charging events	35.0	54.4
Median percent change in battery SOC for charging events	26.5	62.0
Average percent change in battery SOC for all charging events per vehicle driving day (%)	50.7	34.8
Mean final battery SOC after charging event (%)	91.8	90.9
Median final battery SOC after charging event (%)	100.0	100.0

Most charging events almost completely refill the battery, as can be seen in the demonstration graphs under the Decision Support Tools Development section. If a large number of charging events did not completely refill the battery, the SAE J2841 UF curve assumptions may not be accurate.

Combined, these charging event statistics show that in general the fleet is being recharged along the lines of the SAE J2841 assumption of only one fully replenishing battery charge per day. This explains why the two different utility factor curve methodologies presented in the previous section have such similar results. It's also notable that on average, the Via Fleet is not being fully recharged after every day of driving. This information should alert the Via Fleet operators that they can potentially improve their efficiency by recharging the vehicles more often. This recommendation is also noted in EPRI's Report [14] on page 4-26, which was reached independently of the results in this paper.

The Odyne fleet is charged more than once per vehicle driving day, but analysis showed that a large number of charges only replenishing the battery by a very small amount (see graphs in Decision Support Tools Development section). Therefore, it's reasonable to conclude that the Odyne vehicles are being plugged in on days when they are not driving and are already mostly charged.

From an operational perspective, there could be a lot of room for utility factor improvement for both the Odyne and Via fleets if multiple charges could be facilitated throughout the day [2, 6, 7, 11, 14].

All of the charging statistics presented in this section (3.2) were calculated from the summarized event data using about 80 lines of MATLAB code (including whitespace and comments) which ran on a single core in a matter of seconds. The same analysis could have also been performed in Microsoft Excel. For comparison, the MATLAB code written to create the event summary from the raw second-by-second driving CSV data was thousands of lines of code, and took multiple days to run on a single core. Maintaining summarized event data can greatly simplify, streamline, and expedite the data analysis process when new questions of research, design, operation, or policy are posed.

4 Conclusions

The event summary methodologies presented in this paper may be beneficial to future vehicle tracking projects that require analysis of high resolution data from on-board tracking sensors. The ability to summarize a large database of second-by-second driving data points into a list of longer term events is a particularly difficult task, especially when the data collection devices are not 100% reliable. However, when compiled correctly, this event summary list is also very useful. To demonstrate the utility of these event summary statistics, utility factor curves, charging summary statistics, and demonstration histograms were generated.

This work is also novel because it is the first time that UF curves have been published for a fleet of Medium-Duty PHEV Trucks. The presented UF charts for Odyne provide additional evidence that different policies may be needed to govern different vehicle classes, and that a single overarching UF curve for every situation may not be appropriate [2].

The truck UF curves also show that light- and medium-duty trucks are an effective application for PHEV technology, as light and medium-duty PHEV’s displace a large amount of gasoline with electric power. These PHEV’s can displace at least as much gasoline as privately owned commuter PHEV’s, if not more.

References

[1] Salisbury, S., Smart, J., and Bradley, T.H. "Actual Versus Estimated Utility Factor of a Large Set of Privately Owned Chevrolet Volts," SAE International Journal of Alternative Powertrains, May 2014, 2014-01-1803.

[2] Bradley, T.H., and Quinn, C.W., "Analysis of Plug-in Hybrid Electric Vehicle Utility Factors," Journal of Power Sources 195 (2010) 5399--5408.

[3] M. Duoba, R.W. Carlson, J. Wu, Electric Vehicle Symposium, vol. 23, 2008

[4] Society of Automotive Engineers, Utility Factor Definitions for Plug-In Hybrid Electric Vehicles Using Travel Survey Data, SAE J2841, revised Sept 2010

[5] Society of Automotive Engineers, SAE Electric Vehicle and Plug in Hybrid Electric Vehicle Conductive Charge Coupler, SAE J1772, revised 2016-02

[6] Bradley, T. and Davis B., “Alternative Plug in Hybrid Electric Vehicle Utility Factors,” SAE Technical Paper 2011-01-0864, 2011, doi: 10.4271/2011-01-0864

[7] Bradley, T.H. and Frank, A.A., “Design, demonstrations and sustainability impact assessments for plug-in hybrid electric vehicles”, Renewable and Sustainable Energy Reviews, Volume 13, Issue 1, January 2009, pp. 115-128.

[8] Wood, E., Burton, E., Duran, A., and Gonder, J., "Contribution of Road Grade to the Energy Use of Modern Automobiles Across Large Datasets of Real-World Drive Cycles," SAE Technical Paper 2014-01-1789, 2014, doi:10.4271/2014-01-1789.

[9] Al-Khedher, M.A., “Hybrid GPS-GSM Localization of Automobile Tracking System”, International Journal of Computer Science & Information Technology (IJCSIT) Vol 3, No 6, Dec 2011.

[10] Gonder, J., Markel, T., Thornton, M., Simpson, A., “Using Global Positioning System Travel Data to Assess Real-World Energy Use of Plug-In Hybrid Electric Vehicles”, Transportation Research Record: Journal of the Transportation Research Board; Journal Volume: 2017; Journal Issue: 2007. DOI: 10.3141/2017-04

[11] Smart, J., “Electric Vehicle Charging Infrastructure Usage Observed in Large-scale Charging Infrastructure Demonstrations – ARB”, Plug-in Electric Vehicle Infrastructure Information Gathering Meeting; May 27, 2014.

[12] Schey, S., Scoffield, D., and Smart, J., “A First Look at the Impact of Electric Vehicle Charging on the Electric Grid in The EV Project”, Electric Vehicle Symposium 26, May 2012.

[13] Ko, J., R. Guensler, and M. Hunter (in press). “Variability in Traffic Flow Quality Experienced by Drivers: Evidence from Instrumented Vehicles”; Transportation Research Record; National Academy of Sciences; Washington, DC.

[14] “Plug-In Hybrid Medium Duty Truck Demonstration and Evaluation.” EPRI, Palo Alto, CA: 2015. Product ID: 3002006566.

Authors

Spencer Vore is a graduate research and teaching assistant at Colorado State University, and is working towards his MS degree in Mechanical Engineering. He earned his BS degree in Mechanical Engineering from the Georgia Institute of Technology, and has worked in government and the automotive industry with positions at Freudenberg-NOK Sealing Technologies and the National Renewable Energy Laboratory.

Mark Kosowski is a Technical Executive at the Electric Power Research Institute where he works to manage portions of EPRI’s Electric Transportation research and development portfolio. Prior to his work with EPRI, Mark spent 19 years of his 38 years at GM developing Electric vehicles and Plug-in Hybrid vehicles with General Motors including the Volt, EV1 and other hybrid programs.

Zachary Wilkins is an undergraduate research assistant at Colorado State University, and is working towards his BS degree in Mechanical Engineering. He has also worked in government with the National Renewable Energy Laboratory.

Thomas H. Bradley is Associate Professor of Mechanical Engineering at Colorado State University, and Associate Director of CSU’s Systems Engineering Program. He received his B.S. and M.S. degrees from the University of California at Davis, and his Ph.D. from The Georgia Institute of Technology. His research is in the field of systems engineering applied to energy problems with particular emphases in environmental assessment, transportation systems, and policy making.