International Journal of Control and Automation
Vol.7, No.9 (2014), pp.399-410
http://dx.doi.org/10.14257/ijca.2014.7.9.34

State-of-charge Stream Processing and Modeling for Electric Vehicle-Based Trips

Junghoon Lee, Dongwook Lee and Gyung-Leen Park*

Dept. of Computer Science and Statistics,
Jeju National University
Jeju-Do, Republic of Korea
{jhlee, seacove, glpark}@jejunu.ac.kr

Abstract

This paper first describes how to process SoC (State of Charge) streams made up of temporal and spatial stamps, altitude, SoC reading, velocity, and the like. Collected by a probe vehicle on the same road in Jeju city, Republic of Korea, multiple times, each stream has its own initial SoC and speed. The driving distance is calculated by adding up the line segment connecting two consecutive sensor points. For each stream, we plot 1) the absolute SoC change to figure out the SoC dynamics, 2) the consumed SoC change to find the similarity between different streams, and 3) the normalized curves to build a common SoC change model. Next, the artificial neural network is exploited to trace a common pattern out of them, aiming at providing an essential basis for navigation applications supporting electric vehicles. The analysis result reveals that the model can find reasonable representative values out of different streams for each subsection, even with a simple model having just one input variable denoting the distance from the start point and one output variable denoting the SoC reading. Our model keeps the trace error less than 5 % for the duration in which the vehicle speed is the same for each stream.

Keywords: *electric vehicle, state-of-charge stream, artificial neural network, common pattern, normalization*

1. Introduction

The modern power system, called the smart grid, is essentially a cyber-physical system which collects a great deal of sampling data. It intelligently recognizes the current situation from the massive sensor readings, finds an appropriate reaction, and finally runs relevant actuators [1]. Datasets generated in the smart grid are necessarily time series, in which each record corresponds to a time-stamped snapshot of always working devices [2]. Here, if the monitoring target can move, the series becomes a spatio-temporal dataset, in which each record has both time and spatial stamps in addition to the underlying sensor status. In the smart grid system, how to handle those data is a very important problem for diverse control actions to achieve energy efficiency [3]. For example, smart meters periodically sample energy consumption and report to the utility company in real-time, while the utility forecasts the future demand and makes a plan to avoid demand-supply mismatch.

---

* Prof. Gyung-Leen Park is the corresponding author.
This research was financially supported by the Ministry of Trade, Industry and Energy (MOTIE), Korea Institute for Advancement of Technology (KIAT) through the Inter-ER Cooperation Projects.

ISSN: 2005-4297 IJCA
Copyright ⓒ 2014 SERSC

Electric vehicles (EVs) are one of the key elements in the smart grid, especially in the electrified transportation system. An EV consumes electricity stored in its battery while driving. Their charging consumes not a little energy and more EVs will be charged under the control of the grid [4]. Limited battery shortens their driving range, and a fully charged EV can drive up to just 100 *km* in practice due to terrain effects and weather conditions. Hence, a driver, usually a long-distance driver, sometimes wonders if his or her EV can reach its destination with current battery remaining, interchangeably, SoC (State of Charge) [5]. However, how much energy will be needed is very hard to accurately estimate, as it is affected by so many factors such as distance to the destination, road shape, velocity, starting SoC, driving habit, and so on [6]. Particularly, different road features, such as elevation changes and curvatures, necessitate the road-by-road consumption model.

One of the simplest but most efficient ways for battery SoC tracing is to measure the amount of consumption for a specific road several times and then conduct a statistical analysis. The tracing target must be a main road taken by many EVs. With the estimation models for main roads, it is possible to calculate the battery consumption between the start and end points of a trip, as the effect of estimation errors in branch roads is not so significant. Our research team has been collecting the SoC records having the period of 1 second via the test drive along a target road to figure out the SoC change pattern and identify critical parameters effectively affecting battery consumption. The first analysis step is to overview the technical results from the acquired SoC stream for building a SoC-integrated navigation application [7]. Battery modeling makes it possible to design a new application for EV management and control. For example, an emergency rescue system monitors the SoC of rescue vehicles to find the best one to dispatch.

Accordingly, this paper first shows the SoC stream data acquired in our research-and-development project, explains how we process the stream data, and creates a battery consumption model. As an extended version of our previous work [8], this paper has details on the data processing framework capable of evolving to the big data analysis and a neural network approach to develop a battery consumption model for a given road segment. Basically, the raw data fields will be shown for the possibility of adding more environment variables and give the impression on the actual SoC stream is like.

This paper is organized as follows: After issuing the problem in Section 1, Section 2 introduces related work for EV battery modeling systems. Section 3 shows the main features of the collected SoC stream and how to process the raw data. Section 4 builds a common battery consumption model for the multiple streams based on artificial neural networks. Finally, Section 5 summarizes and concludes this paper with a brief introduction of future work.

2. Related work

[6] presents how to predict the power requirement for an EV trip based on a history of vehicle's power consumption, speed, and acceleration. Here, road condition, such as slope and speed limit, is provided by a pre-downloaded digital map. The proposed model categorizes the consumption parameters into 3 groups, namely, *stable*, *dynamic but easy to predict*, and *dynamic and difficult to predict*. The third category, on which this work is mainly focused, includes acceleration and speed, and they depend even on individual's driving behavior and mood. As for acceleration, a driver's behavior is captured in terms of reaction strength and reaction time. Here, the traffic around the vehicle and the traffic regulations also have a significant impact on the acceleration. The power prediction is made available to the battery management system to prevent the damage of battery cells and help efficiently allocate battery cells in real time to meet an EV's power demand [9].

 Copyright ⓒ 2014 SERSC

After addressing the problems of existing SoC estimation schemes including coulomb counting, open-circuit voltage measurement, and other off-line parameter identification methods, [10] proposes a series of models for the battery dynamics considering a simple resistance-capacitance (RC)-equivalent circuit. It also covers an adaptive on-line parameter identification algorithm. Here, the authors deploy a piecewise linearized mapping of the SoC curve and continuously update the relevant parameters. The least squares (LS) identification approach tries to minimize the LS difference between the estimated and actual output values. The parameters to identify over the moving time window are initial and current SoCs, initial and current resistances, and capacitance. For each piecewise time interval, an observer is designed to estimate SoC, finding relevant coefficients with LS error curve fitting techniques. Finally, the battery parameters are extracted from the coefficients and fed to the observer recursively.

[11] presents applications of derived from real-time, adaptive behavior modeling for Li-ion batteries commonly used in EVs. Based on the combination of a Kalman estimator and on-line parameter adaptation, the model gradually removes the estimation error and thus makes the estimation converge to a stable output value, even from erroneous initial conditions. The Kalman estimator is very useful for fitting those states affected by broadband noise, following a discrete time equivalent model. The empirical method can handle not only self-discharging during the non-use intervals of battery cells but also battery aging, namely, natural worn-out. With this capability it is possible to integrate fault diagnosis and fault tolerant control for health monitoring of EVs [12]. As an example, fault detection focuses on the electromechanical system composed of a DC motor, a gear, and the wheel in interaction with the ground. In addition, the overall control strategy embraces a lateral motion controller, ensuring the desired trajectory.

With an accurate estimation mechanism of energy consumption, the routing algorithm can find an energy-efficient path both by pre-trip planners and on-trip navigation systems as indicated in [13]. The authors classify a driving cycle into 4 phases to find a phase-specific parameters. First, *Crusing time* phase is the time interval where the vehicle moves at almost constant speed, namely, the acceleration fluctuates only within a limited range. In *Idle time* phase, the vehicle speed is 0. *Deceleration/Accelerating* phase accounts for the interval in which the vehicle is reducing/increasing its speed, respectively. Each trip includes those 4 phases and their time shares are relatively different. They are affected by not only static factors such as roadway gradient but also dynamic factors such as traffic and weather conditions. To find out the time share for each phase, a statistical analysis is employed from the raw data of the emission factors given in HBEFA (Handbook Emission Factors of road transport). After modeling the energy consumption of each phase, the final estimation process calculates the total energy by multiplying the weight of each phase.

3. SoC data presentation

3.1. Preliminaries

Our research project is performed on Jeju Island, Republic of Korea. Its perimeter is about 200 *km* long, and this city is actively hosting many EV-related enterprises under the ambitious plan of replacing all vehicles by EVs until 2030. Figure 1 depicts the target road which connects two major town areas in this island. The target road segment is marked as a thick line on the map. It runs about 23 *km* and quite winding, while its elevation changes significantly as it crosses a big mountain. Here, the research team has developed a DCC (Driving Condition Collector) module which automatically generates a series of SoC records while driving [14]. Our probe EV captures the SoC every 1 second, creating an SoC stream

Copyright ⓒ 2014 SERSC

consisting of hundreds of SoC records, each of which consists of time stamp, location stamp, SoC, velocity, altitude, temperature, and air pressure. The road segment has no traffic signal, so the EV rarely stops. Here, SoC values can be obtained by the interaction with the underlying battery management system.

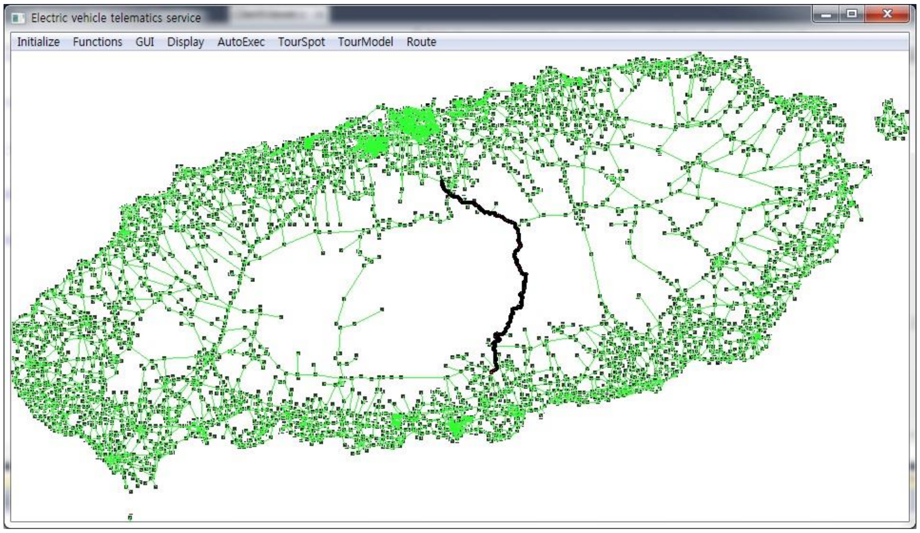


Figure 1. Target Course

Basically, the battery consumption model calculates the SoC consumption, F, in other words, the SoC difference from the start point, when an EV reaches a point a given distance away from the start point. Namely,

$$F(d) = c \tag{1}$$

, where d is the distance from the start point, while c is the consumed battery. To build an estimation model, a lot of $\{C_i, D_i\}$ records must be available. Hence, it is necessary to extract such pairs from the collected SoC stream data. This paper also follows the same method as our previous work [15] and its idea is briefly explained for self-containment. Figure 2 illustrates how to convert a SoC records represented by (x_i, y_i, s_i) for each valid record sequence i. Here, (x_0, y_0, s_0) is the location and SoC at the start point. We take the Euclidean distance between two records for simple calculations, as the sensoring period is just 1 second. Next, output values, namely, the amount of battery consumption from (x_0, y_0), is s_i - s_0 for each valid record i. An SoC stream is converted to a series of SoC records as follows:

$$F(\Sigma D_j) = s_i - s_0, \quad \text{for} \quad 0 \le i < n \tag{2}$$

, where n is the number of valid records. In addition, D_i is calculated as shown in Eq. (3).

$$D_i = \text{sqrt}\,((x_i - x_{i-1})^2 + (y_i - y_{i-1})^2) \tag{3}$$

 Copyright ⓒ 2014 SERSC

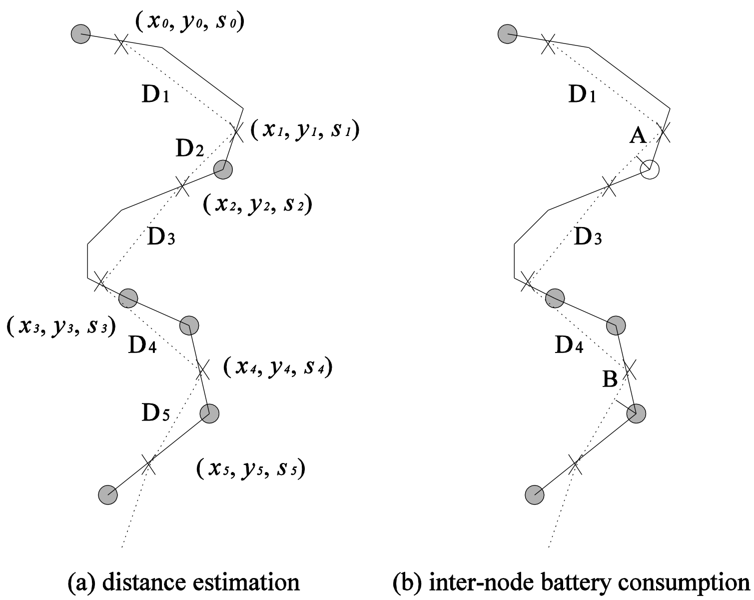


Figure 2. Distance Approximation

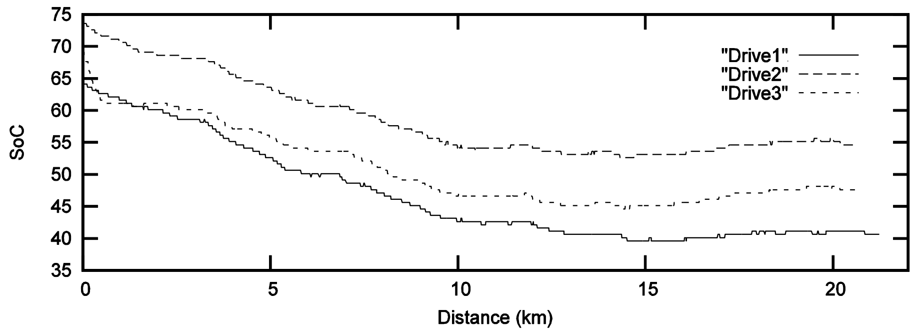
In Figure 2, the actual road shape is plotted by solid lines while the virtual trajectory of the probe vehicle by dotted lines. Each dotted line segment corresponds to the Euclidean distance between two consecutive sensoring points. In addition, Figure 2(b) explains how to obtain the SoC consumption between two nodes, or intersections. For two nodes *A* and *B*, we can find the closest line segments for each one. Then, after calculating the orthogonal points on those two segments, we can estimate the offsets from (x_0, y_0), say, $F(D_A)$ and $F(D_B)$, respectively. Here, $F(D_A)$ is the distance from the start point to D_A, and *A* is not the record sequence. Finally, the battery consumption between *A* and *B* is $F(D_B)-F(D_A)$. For more details, please refer to [15].

We make an EV drive the target course from the same start point to the same end point 5 times. However, 2 streams have many error terms, so they are excluded from the analysis. If the vehicle reaches a stable speed, which is usually the speed limit of the road segment, the vehicle speed hardly changes. However, the driving pattern differs in how fast an EV reaches the stable speed. This factor can be inferred by the initial speed, and it has not a little impact on the battery consumption over the road. In each trial, the coordinate of the start point and the initial speed are shown in Table 1. Start points are not far away from each other, and the difference is negligible. The probe EV starts after the monitoring process begins, hence, the speed of the first several records is 0. The first observed nonzero speed is also shown in Table 1.

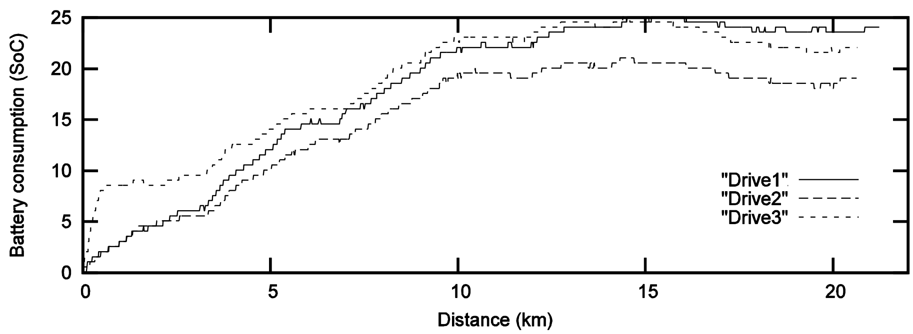
Table 1. Start Points of Drives

Drive	*Longitude*	*Latitude*	*Initial speed*
Drive 1	126.5504	33.5549	5 *kmh*
Drive 2	126.5510	33.4566	3 *kmh*
Drive 3	126.5508	33.4566	10 *kmh*

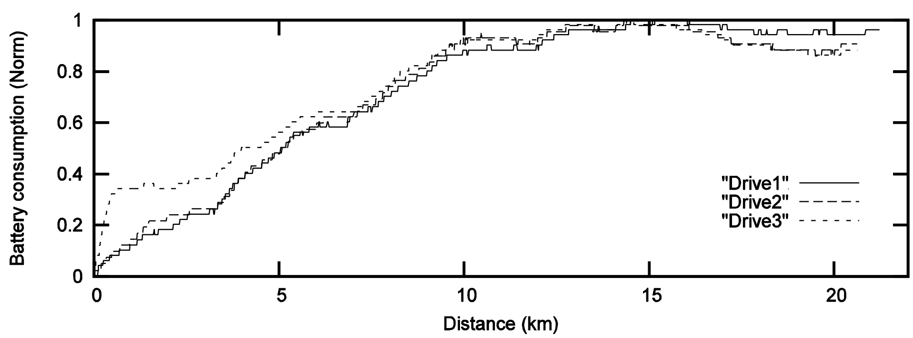
Now, Figure 3(a) plots the SoC change for each stream. Each curve tracks the absolute SoC values and we can find out that the total consumption is different according to the initial SoC level due to intrinsic battery dynamics. SoC is given by the ratio to the full capacity, and the y-axis in Figure 3 shows the percentage values. Here, the full capacity can be influenced by temperature, which makes the battery volume a little bit different. The first trip starts at 64.5 % and consumes 24 %, the second starts at 73.5 % and consumes 19 %, and the third starts at 69.5 % and consumes 22 % in total. According to the figure, if other conditions are equal, we can know that the higher the initial battery, the smaller the overall consumption. It is necessary to further investigate this effect and decide the weight from the initial battery.



(a) Absolute reading



(b) Consumed battery

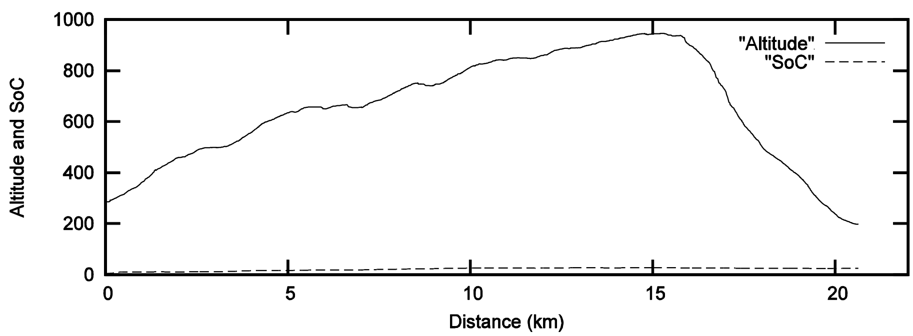


(c) Normalized consumption

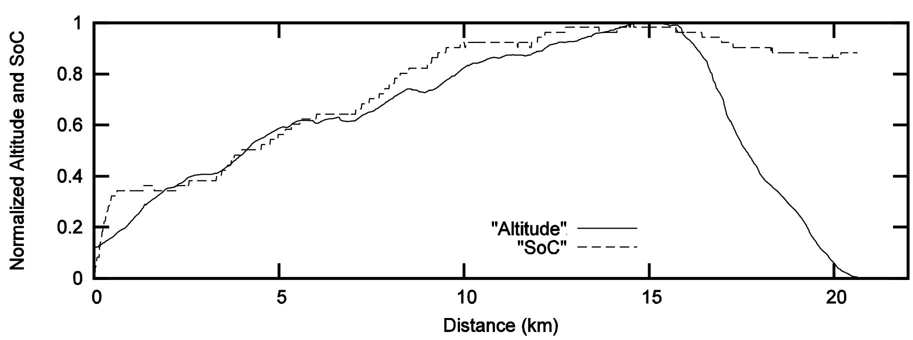
Figure 3. SoC stream Plotting

 Copyright ⓒ 2014 SERSC

Figure 3(b) shows the amount of battery consumption from the start point for each trip. The consumed battery is calculated by subtracting the initial value from the current SoC. Figure 3(b) indicates that there is a similarity for 3 trips. Particularly, the slope changes almost at the same point in the x-axis for 3 curves, indicating that our Euclidean distance estimation is quite acceptable and the complex network distance estimation is not necessary. Beyond 15 *km*, the SoC hardly drops as the EV drives along downslope roads, in which a regenerative braking system even fills the battery. However, *Drive* 3 consumes more electricity at the start, possibly due to the high initial speed. By the comparison of absolute values, we can find just a rough consumption pattern. Hence, Figure 3(c) normalizes the consumption by taking the maximum value as 1.0 for each trip. Three curves get much closer except the interval from 0 *km* to 5 *km* in which *Drive* 3 spends more electricity. With this result, we can develop a common battery consumption model by statistical analysis.



(a) Absolute reading



(b) Normalized plotting

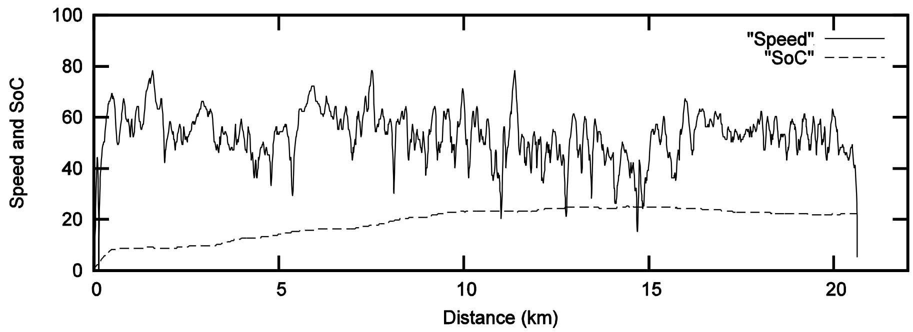
Figure 4. Altitude and SoC

Next, Figure 4 shows the correlation of SoC changes and altitude. An SoC record includes an altitude field. The target road goes up high almost to 1,000 *m* from the sea level. Figure 3(a) just plots the absolute values of altitude and SoC. Even though this figure plots the elevation along the road, the correlation with SoC cannot be known due to scale difference. Figure 3(b) shows the normalized values for two streams. Each stream is made to range from 0 to 1.0 according to its own minimum and maximum values. This figure tells that the elevation gradually increases until the distance of 15 *km*, and drops sharply beyond that point. The two streams seem to have a close correlation until 15 *km* point, as two streams have the almost same slope. On the contrary, for the downslope section, SoC does not drop as sharply

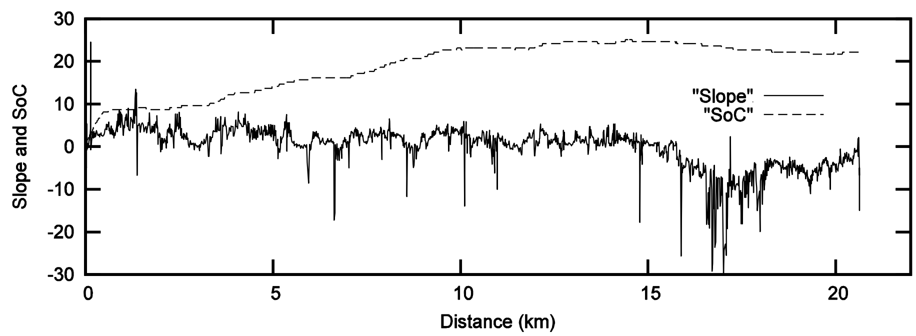
Copyright ⓒ 2014 SERSC

as the altitude, indicating that it is necessary to build different correlation models for upslope and downslope sections.

Micellaneously, Figure 5(a) plots the change of speed along the road captured in *Drive* 3. The vehicle speed field has no zero value except the initial stage. According to the road curvature, the speed changes from 18 to 80 *kmh*. There can be the interference from other vehicles as some part of the road has just a single lane. Anyway, during the upslope section, the speed fluctuates severely, while in the downslope area, the speed oscillates just within a 10 *km* range. As the vehicle doesn't stop on the road, it can be assumed that the speed measurement will be quite accurate. In addition, Figure 5(b) plots the change of the slope field along the road. It is obtained from our driving condition collector which includes a gyroscope. Inherently, this sensor value is not so reliable, as the current vehicle tilt cannot be consistent with the road slope, especially while driving on the uneven surface. Hence, from Figure 5(b), it is necessary to find the fitting curve first before investing the effect of slope changes to SoC. These factors are thought to have an influence on SoC, but additional preprocessing steps are required in the real-life stream analysis.



(a) Speed effect



(b) Slope effect

Figure 5. Other Effects

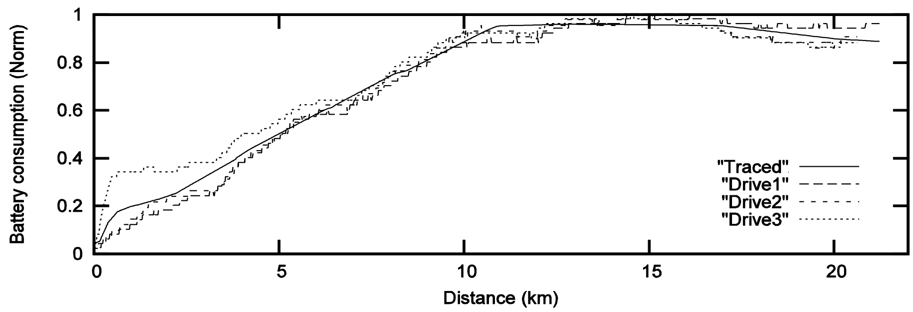
4. Modeling Results

For the 3 streams the measuring SoC changes on the same road, the ANN (Artificial Neural Network) is exploited to build a common battery consumption model. Basically, the ANN can model the complex nonlinear behavior of target objects based on the principle of *learn by example*, without a mathematical or statistical formulation [16]. It is widely used for

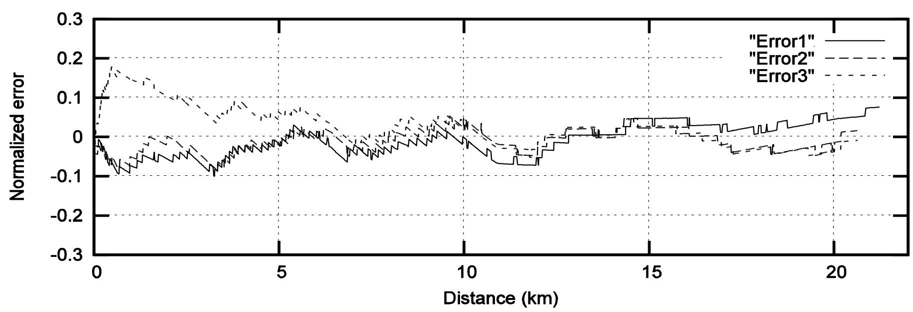
 Copyright ⓒ 2014 SERSC

the approximation problem of time series, by taking a specific time period as input patterns and the next value as the subsequent output [17]. In addition, our experiment makes use of the FANN (Fast ANN) free open source neural network library, which provides a rich set of convenient API functions, particularly making it easy to handle training data set [18]. We need to generate learning patterns from the SoC streams according to the text file format defined in FANN. Then, a C program calls the FANN function to build an ANN and also to get the traced value using the series of provided functions.

Our tracer implementation takes one input variable for the distance from the start point and one output variable for each SoC reading. Hence, the ANN model includes 1 input node, 1 output node, and 10 hidden nodes. The number of hidden nodes is selected empirically. The result is shown in Figure 6(a), in which the solid curve represents the traced result. This figure shows that the simple ANN model finds a reasonable representative value out of different streams. Here, even though *Drive* 3 has a little bit different behavior from others, the tracing error can be masked out with more streams. In addition, Figure 6(b) shows the size of error, that is, the difference between the traced and actual values for 3 streams. The error size is less than 0.1, except the initial stage of *Drive* 3. It is true that the consideration of other variables such as altitude changes and weather conditions will be helpful to the improvement of the model accuracy, the distance-based tracing will catch the overall stream flow.



(a) Traced result



(b) Error characteristics

Figure 6. SoC consumption Estimation

5. Conclusions

EVs are definitely the key component in the electrified transport system, and their deployment will bring energy efficiency and accelerate the developlemt of diverse applications. The role of the information technology is crucial for intelligent management and operation of EV applications. Here, SoC consumption modeling is their fundamental prerequisite. Hence, in this paper, we have processed raw SoC streams as a preliminary step of developing refined battery consumption models for main roads of a target city. Each such record includes a time stamp, location stamp, SoC, velocity, altitude, temperature, and air pressure. It opens the possibility of developing a sophisticated consumption model embracing diverse variables. 3 drives on the same road undergo different SoC changes in absolute values. However, the normalization process makes them much similar, finding out the feasibility of building a common consumption model from repeated trips.

Then, to develop a common battery consumption model out of different SoC streams, which will be further accumulated in our data processing framework, the proposed model takes an artificial neural network, specifically, FANN library. The modeling result and its error analysis have discovered that it can find a reasonable representative value out of different streams for each subsection, even with a simple model having just one input variable for the distance from the start point and one output variable for the SoC reading. Furthermore, the developed model can apply to different initial SoC values by giving an appropriate weight to the SoC change from the start point.

As future work, we are planning to conduct a further analysis using the Hadoop big data processing framework to cope with a larger volume of stream data [19]. This is because our EV systems create a great deal of data, not just SoC streams but also smart meter reading, charger status monitoring, EV taxi trajectory and the like. Moreover, such data will be stored in distributed computers.

Acknowledgment

This paper is a revised and expanded version of a paper entitled “State-of-charge stream processing for electric vehicles” presented at International Conference on Information Technology and Computer Science, Saipan, U.S.A., July 2014.

References

[1] J. Kim, H. Kim, K. Lakshmanan, R. Rajkumar, "Parallel scheduling for cyber-physical systems: Analysis and case study on a self-driving car," International Conference on Cyber-Physical Systems, pp. 31-40, 2013.
[2] Y. Simmhan, M. Noor, "Scalable prediction of energy consumption using incremental time series clustering," IEEE International Conference on Big Data, pp. 29-36, 2013.
[3] X. Fang, D. Yang, G. Xue, "Evolving smart grid information management cloud-ward: A cloud optimization perspective," IEEE Transactions on Smart Grid, Vol. 4, pp. 111-119, 2013.
[4] A. Bandyopadhyay, L. Wang, V. Devabhaktuni, R. Green, "Aggregator analysis for efficient day-time charging of plug-in hybrid electric vehicles," IEEE Power and Energy Society General Meeting, 2011.
[5] Z. Chen, S. Qiu, M. Mansur, Y. Murphey, "Battery state of charge estimation based on a combined model of extended Kalman filter and neural networks," International Joint Conference on Neural Networks, pp. 2156-2162, 2011.
[6] E. Kim, J. Lee, K. Shin, "Real-time prediction of battery power requirements for electric vehicles," International Conference on Cyber-Physical Systems, pp. 11-20, 2013.
[7] M. Rahman, Q. Dua, E. Al-Shaer, "Energy efficient navigation management for hybrid electric vehicles on highways," International Conference on Cyber-Physical Systems, pp. 21-30, 2013.
[8] J. Lee, D. Lee, G. Park, "State-of-charge stream processing for electric vehicles," Advanced Technology Letters, Vol. 51, pp. 242-245, July 2014.
[9] J. Kim, K. Shin, "Scheduling of battery charge, discharge, and rest," 30th IEEE Real-time Systems Symposium, pp. 13-22, 2009.
[10] H. Rahimi-Eichi, F. Baronti, M. Chow, "Online adaptive parameter identification and state-of-charge coestimation for lithium-polymer battery cells," IEEE Transactions on Industrial Electronics, Vol. 61, No. 4, 2014.

 Copyright ⓒ 2014 SERSC

[11] C. Gould, J. Wang, D. Stone, M. Foster, "EV/HEV Li-ion battery modelling and state-of-function determination," International Symposium on Power Electronics, Electrical Drives, Automation and Motion, pp. 353-358, 2012.
[12] R. Loureiro, S. Benmoussa, Y. Touati, R. Merzouki, B. Bouamama, "Integration of fault diagnosis and fault-tolerant control for health monitoring of a class of MIMO intelligent autonomous vehicles," IEEE Transactions on Vehicular Technology, Vol. 63, No. 1, 2014.
[13] K. Kraschl-Hirschmann, M. Fellendorf, "Estimating energy consumption for routing algorithms," Intelligent Vehicles Symposium, pp. 258-263, 2012.
[14] J. Lee, G. Park, B. Lee, J. Han, J. Kang, B. Kim, J. Kim, "Design and development of a driving condition collector for electric vehicles," Lecture Notes in Electrical Engineering, Vol. 280, pp. 1-6, 2013.
[15] J. Lee, M. Kang, G. Park, "Battery consumption modeling for electric vehicles based on artificial neural networks," Lecture Notes in Computer Science, Vol. 8582, pp. 733-742, 2014.
[16] Y. Tan, W. Liu, Q. Qui, "Adaptive power management using reinforcement learning," IEEE/ACM International Conference on Computer-Aided Design, pp. 461-467, 2009.
[17] J. Lee, G. Park, Y. Kim, E. Kim, I. Lee, "Wind speed modeling based on artificial neural networks for Jeju area," International Journal of Control and Automation, Vol. 5, No. 2, pp. 73-80, 2012.
[18] S. Nissen. Neural network made simple. Software 2.0, 2005.
[19] F. Bach, H. Cakmak, H. Maass, U. Kuehnapfel, "Power grid time series data analysis with Pig on a Hadoop cluster compared to multi core systems," 21st Euromicro International Conference on Parallel, Distributed and Network-Based Processing, pp. 208-212, 2013.

Copyright ⓒ 2014 SERSC

 Copyright ⓒ 2014 SERSC