

GreenGPS: A Participatory Sensing Fuel-Efficient Maps Application

Raghu K. Ganti, Nam Pham, Hossein Ahmadi, Saurabh Nangia, and Tarek F. Abdelzaher
Department of Computer Science, University of Illinois, Urbana-Champaign
rganti2, nampham2, hahmadi2, nangia1, zaher@illinois.edu

ABSTRACT

This paper develops a navigation service, called *GreenGPS*, that uses participatory sensing data to map fuel consumption on city streets, allowing drivers to find the most fuel-efficient routes for their vehicles between arbitrary end-points. The service exploits measurements of vehicular fuel consumption sensors, available via the OBD-II interface standardized in all vehicles sold in the US since 1996. The interface gives access to most gauges and engine instrumentation. The most fuel-efficient route does not always coincide with the shortest or fastest routes, and may be a function of vehicle type. Our experimental study shows that a participatory sensing system can influence routing decisions of individual users and also answers two questions related to the viability of the new service. First, can it survive conditions of sparse deployment? Second, how much fuel can it save? A challenge in participatory sensing is to generalize from sparse sampling of high-dimensional spaces to produce compact descriptions of complex phenomena. We illustrate this by developing models that can predict fuel consumption of a set of sixteen different cars on the streets of the city of Urbana-Champaign. We provide experimental results from data collection suggesting that a 1% average prediction error is attainable and that an average 10% savings in fuel can be achieved by choosing the right route.

Categories and Subject Descriptors

J.0 [Computer Applications]: General; K.4 [Computing Milieux]: Computers and Society

General Terms

Experimentation, Measurement

Keywords

Participatory sensing, green navigation, green GPS, model clustering

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

MobiSys'10, June 15–18, 2010, San Francisco, California, USA.
Copyright 2010 ACM 978-1-60558-985-5/10/06 ...\$10.00.

1. INTRODUCTION

In this paper, we develop a novel GPS-based navigation service, called *GreenGPS*, that gives drivers the most fuel-efficient route for their vehicle as opposed to the shortest or fastest route. GreenGPS relies on data collected by individuals from their vehicles and a generalization framework that predicts the fuel consumption of an arbitrary car on an arbitrary street. The service is an example of an emerging category of sensing applications, called *participatory sensing* [2, 9, 12, 23, 24], that rely on voluntary data collection and sharing within a community for common purposes such as mapping of physical phenomena or computing community statistics of interest.

GreenGPS is possible thanks to the *On-Board Diagnostic* (OBD-II) interface, standardized in all vehicles that have been sold in the United States after 1996. The OBD-II is a diagnostic system that monitors the health of the automobile using sensors that measure approximately 100 different engine parameters. Examples of monitored measurements include fuel consumption, engine RPM, coolant temperature, vehicle speed, and engine idle time. A comprehensive list of measured parameters can be obtained from standard specifications as well as manufacturers of OBD-II scanners [4]. Several commercial OBD-II scanner tools are available [3, 4, 5, 6], that can read and record these sensor values. Apart from such scanners, remote diagnostic systems such as GM's OnStar, BMW's ConnectedDrive, and Lexus Link are capable of monitoring the car's engine parameters from a remote location (e.g. home of driver of the car).

GreenGPS utilizes a vehicle's OBD-II system and a typical scanner tool in conjunction with a participatory data collection framework to enable collection and upload of fuel consumption data. In contrast to traditional mapping and navigation tools, such as Google maps [19] and MapQuest [26], which provide either the fastest or the shortest route between two points, GreenGPS collects the necessary information to compute and answer queries on the *most fuel-efficient route*. The most fuel-efficient route between two points may be different from the shortest and fastest routes. For example, a fastest route that uses a freeway may consume more fuel than the most fuel-efficient route because fuel consumption increases non-linearly with speed or because it is longer. Similarly, the shortest route that traverses busy city streets may be suboptimal because of downtown traffic.

The motivation for GreenGPS does not need elaboration. GreenGPS users might be driven by benefits such as savings on fuel or reducing CO₂ emissions and the carbon footprint. With the increase in the use of Bluetooth devices (e.g., cell-

phones) and in-vehicle Wi-Fi, GreenGPS can be easily supported by inexpensive OBD-II-to-Bluetooth or OBD-II-to-WiFi adaptors that can upload OBD-II measurements opportunistically, for example, to applications running on the driver’s cell phone [30]. It can also be supported by scanning tools that read and store OBD-II measurements on storage media such as SD cards. At the time of writing, OBD-II Bluetooth adaptors, such as the ELM327 Bluetooth OBD-II Wireless Transceiver Dongle, are available for approximately \$50, together with software that interfaces them to phones and handhelds.

GreenGPS supports two types of users; members and non-members. Members are those who own OBD-II adaptors or scanning tools and contribute their data to the GreenGPS repository from the OBD-II sensors described above. They have GreenGPS accounts and benefit from more accurate estimates of route fuel-efficiency, customized to the performance of their individual vehicles.

Non-members can use GreenGPS to query for fuel-efficient routes as well. Since GreenGPS does not have measurements from their specific vehicles, it answers queries based on the average estimated performance for their vehicle’s make, model, and year (or some subset thereof, as available).

The paper makes two general contributions. First, we demonstrate how to use participatory sensing to develop a fuel-saving navigation service that relies on voluntary data collection by individuals to influence their routing decisions. Second, we provide a brief experimental evaluation of the system, where users are shown to save 6% on average over the shortest route and 13% over the fastest.

A related contribution is to demonstrate how sparse samples of high-dimensional spaces can be generalized to develop models of complex nonlinear phenomena, where one size (i.e., model) does not fit all. We develop prediction models that enable us to extrapolate from fuel-efficiency data of some vehicles on some streets to the fuel consumption of arbitrary vehicles on arbitrary streets. While, in this case, the utility of such extrapolation may be short-term (soon all cars will be able to measure their own fuel-efficiency), the basic mechanisms and principles behind it can be used for a variety of other participatory sensing applications that share the need for generalizing from sparse data.

GreenGPS utilizes prediction models, developed in this paper, to abstract vehicles and routes by a set of parameters such that fuel efficiency can be computed simply by plugging in the parameters of the right car and route. Using Dijkstra’s algorithm, the minimum-fuel route can then be computed. An experimental study is performed over the course of three months using sixteen different cars with different drivers (and a total of over 1000 miles driven) to determine the accuracy of prediction models. It is shown that a prediction accuracy of 1% is attainable.

The rest of this paper is divided into nine sections. Section 2 presents a feasibility study that investigates the amount of fuel savings that can be achieved by using GreenGPS and by following fuel-efficient routes. The details of GreenGPS system are described in Section 3. Models for estimating fuel consumption are presented in Section 4. Implementation details and evaluation results are presented in Section 5 and Section 6, respectively. Section 7 discusses the results and lessons learned. Section 8 reviews related work. Finally, we conclude with directions for future work in Section 9.

2. A FEASIBILITY STUDY

In this Section, we present a feasibility study that provides the reader with a proof of concept estimate of fuel savings that can be achieved by driving on the most fuel efficient routes.

We compute fuel consumption between landmarks in Urbana-Champaign for three different cars (and drivers) and compare these values across multiple routes between the same pairs of landmarks. The landmarks chosen were frequently visited destinations such as the work place of the authors, a major shopping center, and a football stadium. Three landmarks were initially chosen. The shortest and fastest routes were obtained using MapQuest [26]¹. In Figure 1, we plot the fuel consumption for the shortest route, the fastest route, and the route that consumes the least fuel (as computed from our models) for the aforementioned landmarks.

We observe, from Figure 1, that in the first experiment, the fastest route is also the most fuel-efficient route. In the second experiment, the shortest route consumes the least amount of fuel. In the third experiment, the most fuel-efficient route is different from both the shortest and the fastest routes. We conclude from the above observations that simply choosing the shortest or the fastest route will not always be fuel-optimal.

For example, if the user always chooses the fastest route, their extra fuel consumption compared to taking the optimal route is 0%, 24%, and 10% for the three landmarks, respectively (an average of about 11% overhead). Similarly, if the user always chooses the shortest route, their average extra fuel consumption is about 11.5%. Hence, following the fuel-optimal route can translate (at the current national average gas price, which at the time of writing this paper was USD 2.86 [1]) into savings of at least 30 cents per gallon, which is not bad for “cash back”.

The above results are only a proof of concept. They simply show that there may exist situations where using a fuel-optimal route can save money. A more extensive study of route models and savings is presented in the evaluation section.

To estimate the amount of savings that can be achieved on a global scale, we provide back of the envelope calculations based on data from the Environmental Protection Agency (EPA) [13]. An estimated 200 million light vehicles (passenger cars and light trucks) are on the road in the US. Each of them is driven, on an average, 12000 miles in a year. The average mile-per-gallon (mpg) rating for light vehicles is 20.3 mpg. Even if 5% of these vehicles adopted GreenGPS and 10% fuel savings were achieved on only a quarter of the routes traveled by each of these vehicles, the amount of overall fuel savings is nearly 177 million gallons of fuel $((12000/20.3) * 0.3 * (0.05 * 200M) * 0.1)$. This translates into nearly half a billion dollars in savings at the pump (based on the current national average pump prices for a gallon of gasoline). The authors consider the above prospective savings acceptable. The rest of the paper presents details of the GreenGPS service and a more extensive evaluation.

¹Google maps provides only the shortest route. MapQuest provides both fastest and shortest routes. Hence, we use MapQuest to get route information.

3. THE GREENGPS SYSTEM

The service provided by GreenGPS is similar to a regular map application, such as Google maps [19] or MapQuest [26]. Google maps and MapQuest provide the shortest or fastest routes between two points, whereas GreenGPS computes the most fuel-efficient route. A snapshot of the Web-based GreenGPS’s user interface is shown in Figure 2 along with the most fuel efficient route between two points for a user with a Toyota Celica, 2001. In the following subsections, we will discuss the GreenGPS concept, then present the participatory sensing framework that we utilize for data collection and data sharing and the specifics of the hardware used for the purpose of data collection.

3.1 The GreenGPS Concept

Individuals who want to compute the most fuel-efficient route between two points enter the source and destination address via the interface provided by GreenGPS. Members of GreenGPS (i.e., those individuals who contributed participatory data) can register their vehicles that were used for data collection. Hence, GreenGPS can compute the route specifically for the registered vehicle. Other users may enter their vehicle’s make, model, and year of manufacture. Since different vehicles have different fuel consumption characteristics, these car details are used to compute the most fuel-efficient route for the given vehicle brand. The advantage for the users who contribute data is that the system provides better estimates of the most fuel-efficient routes to these individuals, thus allowing them to have higher savings.

Currently, it is impractical to assume that GreenGPS members will measure all city streets and cover all vehicle types. Instead, measurements of GreenGPS members are used to calibrate generalized fuel-efficiency *prediction models*. These models, discussed in Section 4, show that the fuel consumption on an arbitrary street can be predicted accurately from set of *static* street parameters (e.g., the number of traffic lights and the number of stop signs) and a set of *dynamic* street parameters (such as the average speed on the street or the average congestion level), plus of course the vehicle parameters (e.g., weight and frontal area). It is the mathematical model describing the relation between these general parameters and fuel-efficiency that gets estimated from participant data. Hence, the larger and more diverse is the set of participants, the better the generalized model.

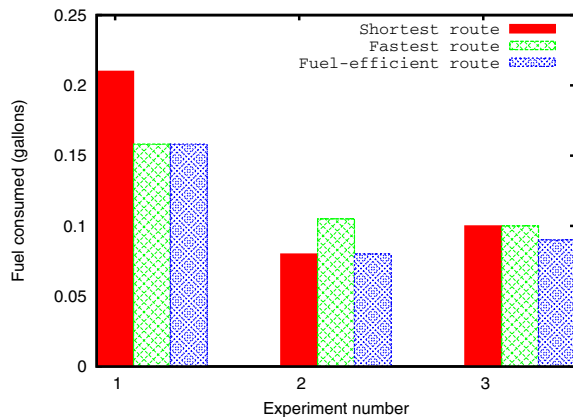


Figure 1: Figure showing fuel consumption for multiple routes between multiple selected landmarks for different cars and drivers

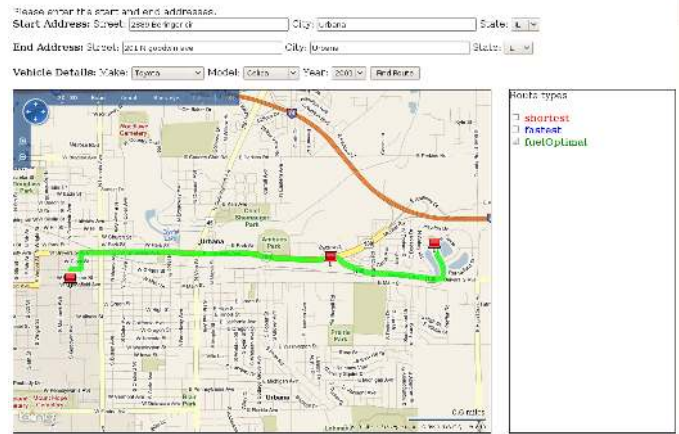


Figure 2: Figure showing the user interface of GreenGPS with the most fuel efficient route between two points on the map for a Toyota Celica, 2001.

For most streets, static street parameters can be readily obtained from traffic databases. For example, the number of traffic lights and the number of stop signs on streets can be obtained from the red light database [20]. Dynamically changing parameters such as the congestion levels or average speed are more tricky to obtain. In larger cities, real-time traffic monitoring services can supply these parameters [35]. Many GPS device vendors, such as TomTom, also collect and provide congestion information. Finally, participatory sensing applications, such as Traffic Analyzer [17] and CarTel [24], have been described in prior literature that have the potential to provide congestion and speed data.

In this paper, speed information is obtained from the collected data using the hardware described in the next section. The speed data is aggregated for different city blocks, based on the GPS data. Thus, given a street name (or the latitude/longitude of a location), GreenGPS provides the average speed information for the corresponding block.

3.2 A Participatory Sensing Framework

We utilize a participatory sensing framework developed in our prior work, called PoolView [17], to implement GreenGPS. PoolView facilitates developing data collection applications. It provides a client-side interface for data upload and delivers all data to a central server called the *aggregation server*, that is application-specific. We implemented GreenGPS by writing our aggregation server for PoolView. An individual who wants to share their OBD-II sensor data can thus download the client side software of PoolView, and use it to upload their data to the GreenGPS aggregation server. The aggregation server uses these data to calibrate models that relate street and vehicle parameters to fuel-efficiency and offers the GreenGPS query interface for fuel-efficient routes.

Individuals who wish to contribute OBD-II data to GreenGPS install, in their vehicle, a commercial OBD-II scanner along with a GPS unit. In our deployment, we use one such off-the-shelf device for data collection purposes, called DashDyno [4], shown in Figure 3. The DashDyno’s OBD-II scanner is interfaced to a Garmin eTrex Legend GPS [18] to get location data. The DashDyno records trip data (including

Garmin’s GPS location) on an SD card that the user later uploads it to the GreenGPS server.

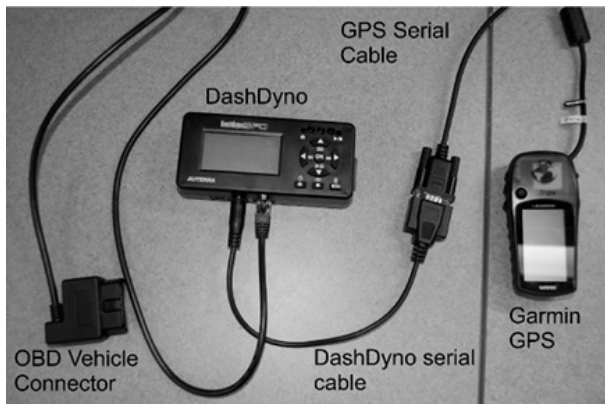


Figure 3: Hardware used for data collection

A total of 16 parameters are obtained from the car and the GPS, the most important of them being instantaneous vehicle speed, total fuel consumption, rate of fuel consumption, latitude, longitude, and time.

4. GENERALIZING FROM SPARSE DATA

In this section, we demonstrate the foundations of one of the key mechanisms in participatory sensing applications that are tolerant to conditions of sparse deployment; namely, the generalization from sparse multidimensional data. Such generalization is complicated by the fact that, in high-dimensional data sets, one size does not fit all. Hence, for example, developing a single regression model to represent all data is highly suboptimal. In the case of GreenGPS, the lack of widespread availability of OBD-II scanner tools suggests that the data contributed by users of our participatory sensing application will be a sparse sampling of routes and cars. Hence, we aim to use data collected by a smaller population to build models capable of predicting the fuel consumption characteristics of a larger population. Admittedly, the conditions of sparse deployment are typically temporary, making the above contribution short-lived in nature. Nevertheless, it solves a key problem at a critical phase of most newly deployed systems, which makes it important. Before we explain the details of the generalization mechanism, we will provide a brief description of our data collection for the purpose of developing models.

4.1 Data Collection

The vision for GreenGPS, when fully deployed, is to collect data from everyday users, which can be employed to update and refine predictions of fuel consumption when such users (or others with similar vehicles) embark on new itineraries. Having said so, for the purposes of this paper, we conducted a limited proof-of-concept study involving sixteen users (with different cars) over the course of three months. A total of over 1000 miles were driven by our users to construct the initial models. Figure 4 shows a partial map of the paths on which data were collected. The details of the car make, model, year, and the number of miles of data collected for each car are summarized in Table 1.

In the aforementioned experiments, each user was asked to drive among a specific set of landmarks in the city. We split



Figure 4: Coverage map for the paths on which data were collected

| Car make | Car model | Car year | Miles driven |
|-----------|------------|----------|--------------|
| Ford | Taurus | 2001 | 135 |
| Toyota | Solara | 2001 | 45 |
| BMW | 325i | 2006 | 70 |
| Toyota | Prius | 2004 | 140 |
| Ford | Taurus | 2001 | 136 |
| Ford | Focus | 2009 | 95 |
| Toyota | Corolla | 2009 | 45 |
| Honda | Accord | 2003 | 102 |
| Ford | Contour | 1999 | 22 |
| Honda | Accord | 2001 | 18 |
| Pontiac | Grand Prix | 1997 | 25 |
| Honda | Civic | 2002 | 11 |
| Chevrolet | Prizm | 1998 | 16 |
| Ford | Taurus | 2001 | 10 |
| Mazda | 626 | 2001 | 9 |
| Toyota | Celica | 2001 | 120 |
| Hyundai | Santa Fe | 2008 | 22 |

Table 1: Table summarizing the cars used and the amount of data collected

each drive into smaller *segments* to capture the variation in the fuel consumption on individual streets. These segments were the “samples” used to capture the variables affecting fuel consumption and develop initial prediction models.

4.2 Derivation of Model Structure

The first part of data generalization is to derive a model structure. The structure describes how various parameters are related, but does not evaluate the various constants and proportionality coefficients. In this case, we derive the structure of fuel prediction models from physical analysis. The analysis is straightforward but is included for completeness.

To motivate the need for modeling, we plot the distribution of miles per gallon (mpg) for all the data collected in Figure 5. We observe from this figure that the distribution is nearly uniform with the mpg values varying between 5 and 50. The standard deviation of the mpg distribution is 9.12 mpg, which is pretty high. Hence, an appropriate model is needed to estimate the fuel consumption on various segments.

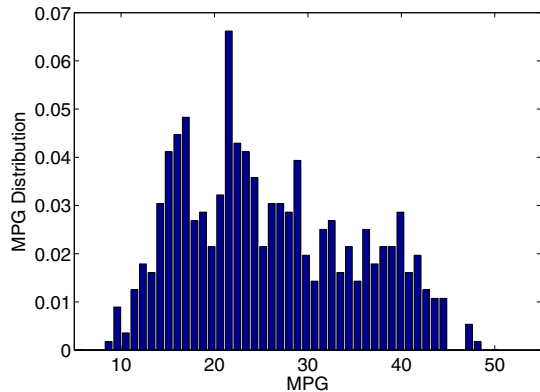


Figure 5: Figure showing the real mpg distribution for all the sixteen users

The inputs to our prediction model include segment parameters and car parameters. We do not consider driver factors in the model because the sample size of drivers was small in our dataset. We will explore the effect of driver factors on fuel consumption in our future work. Note that, we are interested in predicting long-term fuel consumption only. While actual savings of a user on individual commutes to work may vary, the user might be more concerned with their net long-term savings. Hence, it is important to capture only the statistical averages of fuel consumption. As long as the errors have near zero mean, the savings are accurate in the long term. As a given user drives more segments, a value of interest is the total end-to-end prediction error that results (which improves over time as the individual positive and negative segment errors cancel out). We call that end-to-end error the *cumulative error*. It is useful to normalize that error to the total distance driven. We call the result *cumulative percentage error*. It represents how far we are off in our estimate of total fuel consumption.

We derive the model structure for fuel consumption from the basic principles of physics. Many such models exist in prior literature [7, 15, 36], which simplifies the task. We divide the parameters that affect fuel consumption into (i) *static segment parameters*, namely, numbers of stop signs (ST), numbers of traffic lights (TL), distance traveled (Δd) and slope (θ), (ii) *dynamic segment parameters*, namely, average speed (\bar{v}), and *car specific parameters*, namely, weight of the car (m) and car frontal area (A).

The approximated fuel consumption model as a function of the above parameters was derived and can be found in the Appendix. It is given as follows (where gpm is the inverse of mpg and the unit of measure is gallons per mile):

$$gpm = k_1 m \bar{v}^2 \frac{(ST + \nu TL)}{\Delta d} + k_2 m \frac{\bar{v}^2}{\Delta d} + k_3 m \cos(\theta) + k_4 A \bar{v}^2 + k_5 m \sin(\theta) \quad (1)$$

We plot the distributions for various parameters (for individual segments) in Equation 1 for the data that we collected in Figure 6. In the next section, we show that the coefficients of our model, k_1, k_2, k_3, k_4 and k_5 differ among different vehicles making it harder to generalize from vehicles we have data for to those we do not.

4.3 Model Evaluation: One Size Fits All?

Regression analysis is a standard technique for estimating coefficients of models with known structure. In this section,

we demonstrate that a single regression model is a bad fit for our data. Said differently, while a regression model that accurately predicts fuel consumption can be found for each car from data of that one car, the model found from the collective data pool of all cars is not a good predictor for any single vehicle. Hence, in a sparse data set (where data is not available for all cars); it is not trivial to generalize. We illustrate that challenge by first evaluating the performance of car models obtained from their own data (which is good), then comparing it to the trivial generalization approach: one that finds a single model based on all car data then uses it to predict fuel consumption of other cars. A solution to the challenge is presented in the next section.

One should add that while the generalization challenge is common to many participatory sensing applications, our evaluation is not intended to be a definitive study on vehicular fuel consumption. For example, we evaluate fuel consumption in Urbana-Champaign only, which is quite flat. Hence, $\theta = 0$ is a good approximation. (We therefore set the last term, $k_5 m \sin(\theta)$, of our physical model to zero, so k_5 is no longer needed.) Furthermore, the city is rarely congested. Moreover, the range of cars used in the study is rather skewed towards sedans, and hence not representative of the diversity of cars on the streets. Fortunately, even this rather homogeneous data set is sufficient to show that generalization is hard.

First, we determine the length of the segment empirically. We vary the segment length from 0.5 miles to 2 miles in increments of 0.5 miles and evaluate the accuracy of our model in each of these cases. We observed that the accuracy of the model is best when the segment length is 1 mile. Hence, we fix the segment length to be 1 mile in the rest of our experiments. We evaluate the accuracy of models derived from vehicle data using a cross validation approach. We choose a random data point (i.e., a given *segment* of a street driven by some car) to predict fuel consumption for. We then use other points to train a model. We distinguish models based on other segments of the same car from models based on data from other cars in predicting the fuel consumption of the one segment. The 4th and 5th columns of Table 2 summarize the resulting errors, respectively, for a fraction of the used cars.

| Car make | Car model | Car year | Individual cumulative error % (magnitude) | General cumulative error % (magnitude) |
|----------|-----------|----------|---|--|
| Hyundai | Santa Fe | 2008 | 2.89 | 23.63 |
| Honda | Accord | 2003 | 0.89 | 15.3 |
| Ford | Contour | 1999 | 0.83 | 91.4 |
| Ford | Focus | 2009 | 0.12 | 27.35 |
| Ford | Taurus | 2001 | 0.75 | 24.85 |
| Toyota | Corolla | 2009 | 0.61 | 89.97 |
| Ford | Taurus | 2001 | 0.56 | 6.9 |

Table 2: Table summarizing the cumulative percentage errors for the individual car models and the generalized case when all the data (except one car) is used to obtain the model

We also plot the error distribution for individual segments (for one car) in Figure 7. We observe that this distribution is

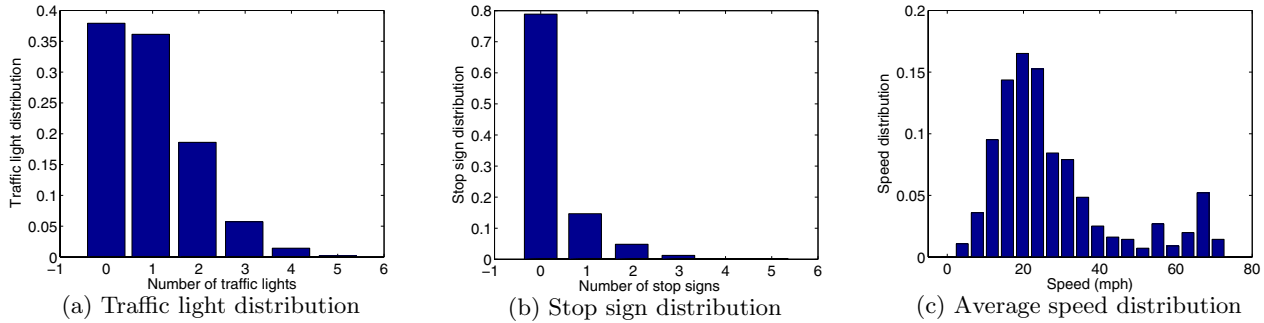


Figure 6: Figures showing the distributions of number of traffic lights, stop signs, and average speed

near normal and the mean is near zero (0.26%). We observe a similar distribution for other cars too.

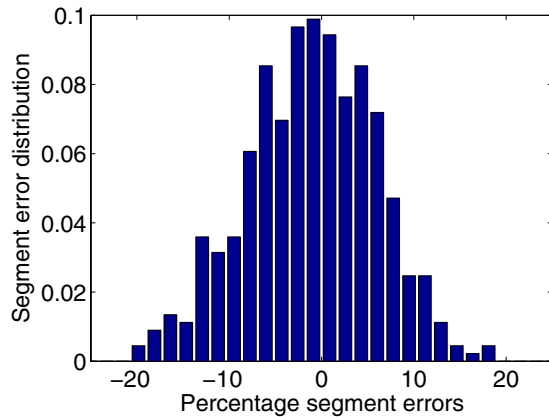


Figure 7: Figure showing the segment error distribution for one car

We also observe from the Table 2 that the cumulative percentage error for individual car models are quite good. Most of them are below 2%. On the other hand, when we predict one car’s consumption using data from other cars, the errors are quite high. This suggests the existence of non-trivial bias in error that does not cancel out by aggregation. In the next section, we propose a way to mitigate this problem based on grouping cars into clusters, such that prediction can be done based on other *similar* cars by some metric of similarity.

4.4 Model Clustering

The above suggests a need for better generalization over vehicle data. Different car types behave differently. Even though the model is parameterized by factors such as car weight and frontal area, they are not enough to account for differences among cars. This is a common problem in high-dimensional data sets collected in participatory sensing applications. The question becomes, if we cannot generalize over the whole set, can we generalize over a subset of dimensions?

A solution is borrowed from the general literature on data cubes [21]. Data cubes are structures for Online Analytical Processing (OLAP) that are widely used for multidimensional data analysis. They group data using multiple attributes and extract similarities within each group. For example, previous work showed how to efficiently construct

regression models for various subsets of data [10]. The data cube framework can thus help compute the optimal generalization hierarchy in that it can help generalize data based on those dimensions that results in the minimum modeling error.

We consider three major attributes (data dimensions) of a given car: *make*, *year*, and *class*. Based on these three attributes, data can be grouped in eight ways. At one extreme, all cars may be grouped together, thus producing a single regression model (which we have shown is not acceptable). At the other extreme, cars can be partitioned into clusters based on their (make, year, class) tuple. A separate model is derived for each cluster. Therefore, a 2001 compact Ford is modeled differently from a 2001 mid-size Ford, a 2002 compact Ford or a 2001 compact Toyota.

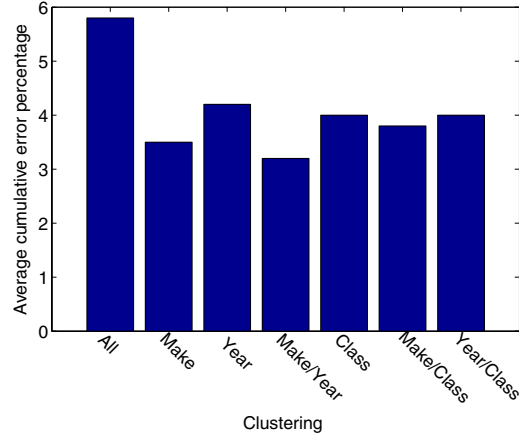


Figure 8: Cumulative error percentage of the models obtained from various clustering approaches

Between these two extremes, to find out which clustering scheme gives the best accuracy, we obtain the cumulative percentage error for each scheme. The results, summarized in Figure 8, show that different generalizations have different quality. These generalizations are somewhat better than using all car data lumped together. While our data set is too small to make general conclusions (from only 16 cars), as more data are collected in a deployed participatory sensing application (e.g., say deployment reaches 100s of cars), progressively better generalizations can be attained.

To use results of Figure 8, one would build models for each pair of make and year (the lowest error clustering scheme). If a car is encountered for which we do not have data on

its (make, year) cluster, we go one level up and use (make) clusters or (year) clusters as generalizations for the (make, year) cluster. If there are no models corresponding to either make or year of a given car, we have no recourse but to go one level up and use the model computed from all data. Figure 9 depicts the generalization process among various model clusters.

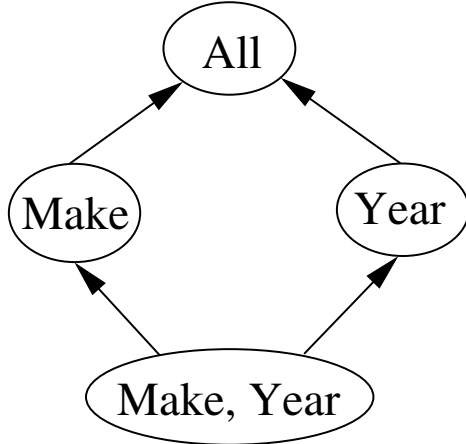


Figure 9: Model generalization from fine grained clusters

| Car make | Car model | Car year | Cumulative error % |
|----------|-----------|----------|--------------------|
| Hyundai | Santa Fe | 2008 | 0.73 |
| Honda | Accord | 2003 | 1.01 |
| Ford | Contour | 1999 | 1.42 |
| Ford | Focus | 2009 | 2.7 |
| Ford | Taurus | 2001 | 3.38 |
| Toyota | Corolla | 2009 | 1.28 |

Table 3: Table showing the cumulative error percentage for each individual car when model clustering is used

We evaluate the performance of our model clustering technique by measuring how accurately an individual car can be modeled using the data from cars with similar make or year. Specifically, we construct the model cluster while removing data of a certain car type. We use the model cluster to estimate the fuel consumption for a given car. The resulting cumulative error percentage is presented in Table 3.

To put the above results in perspective, the reader is reminded that the nature of the landscape in Urbana-Champaign limits our study in that we do not have data on hilly terrain. The study would have been more interesting if conducted on uneven grounds, where changes in incline modulate fuel consumption. We expect that future data collected will be used to evolve our current model by considering the terrain (θ in Equation 1) parameter. Further, new data collected will be used to update the model. Another limitation of our modeling approach arises from the class of cars for which data has been collected. We observe from Table 1 that the majority of the cars are sedans (with the exception of one SUV). We observe that the generalization tree (Figure 9) does not use the *class* of the car. This generalization tree is specific to

the dataset collected. The point of this section is to illustrate an approach to improve prediction in the temporary but important conditions of sparse deployment. Ultimately, when all cars have their own OBD-II readers supplying data to drivers’ cell-phones, we shall not need the generalization scheme described above.

5. IMPLEMENTING GREENGPS

The GreenGPS server combines several open source software services to provide the fuel-efficient route computation service. The various modules that are part of the GreenGPS implementation are depicted in Figure 10. GreenGPS maintains the map of a given area as an *OpenStreetMap (OSM)* [29]. OSM is the equivalent of Wikipedia for maps, where data are collected from various free sources (such as the US TIGER database [37], Landsat 7 [27], and user contributed GPS data) and an editable street map of the given area is created in an XML format. The OSM map is essentially a directed graph, which is composed of three basic object types, *nodes*, *ways*, and *relations*. A node has fixed coordinates and expresses points of interest (e.g. junction of roads, Marriott hotel). A way is an ordered list of nodes with tags to specify the meaning of the way, e.g. a road, a river, a park. A relation models the relationship between objects, where each member of the relation has a specific role. Relations are used in specifying routes (e.g. bus routes, cycle routes), enforcing traffic (e.g. one way routes).

GreenGPS maintains the street variables affecting fuel consumption as additional parameters in the OSM map. This information is stored as a tag/value pair in the way object, where tag is a street parameter and value is the corresponding value of the parameter. We are currently working on populating the street variables into the OSM database for Urbana-Champaign in an automated manner. Further, the car and driver specific parameters are maintained in a separate database. The model to compute fuel consumption on a given way (for a given car and driver) queries these databases and computes the fuel consumption on the way.

The OBD-II data shared by individuals is used to compute regression models that predict the fuel consumption on specific streets given the car details (e.g. make, model, age) and driver behavior. The regression variables which are street specific are stored in the OSM map database, whereas the car and driver specific variables are stored in a similar database.

5.1 Model Clustering Implementation

GreenGPS implements the model clustering technique described in Section 4.4 using *Data Cubes* [21].

We implement a 3-dimensional (make, class, year) regression cube [10] in C++. Each one mile segment is organized as a row in a database where *five* of its attributes are the values of physical model parameters (see Section 4.2) and are used for regression. Three other attributes (make, class, year) are used for grouping. After computing the regression models for all clusters (i.e. materializing the cube), search for a specific triple of (make, class, year) is done consecutively within the (make, year) cluster, the (make) cluster, and the (year) cluster. The first regression model that matches the query is used for prediction.

5.2 Routing in GreenGPS

Routing is achieved in GreenGPS by customizing the open source routing software, Gosmore [28]. Gosmore is a C++

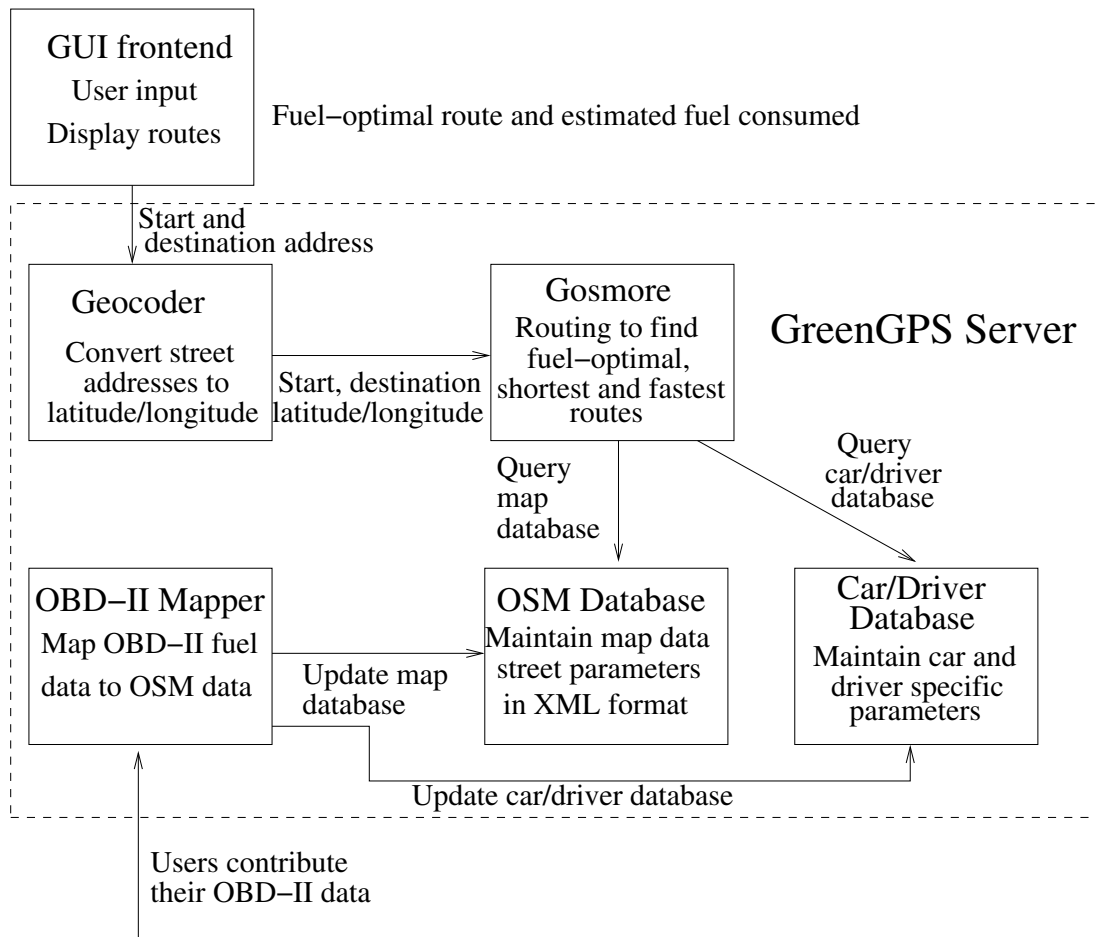


Figure 10: Figure depicting the various modules of GreenGPS

based implementation of a generic routing algorithm that provides shortest and fastest routes between two arbitrary end-points. Gosmore uses OSM XML map data for doing routing. Gosmore’s routing algorithm is a heuristic that by default computes the shortest route. This routing algorithm can be thought of as a weighted Dijkstra’s algorithm on the OSM map, where the nodes of the graph are OSM nodes and the edges of the graph are OSM ways and the weights of the edges are the lengths (distance) of the ways. The fastest route is computed by multiplying the distance by an inverse speed factor (thus giving lower weights to faster ways). Our fuel-optimal routing algorithm multiplies the distance by an inverse mpg (miles per gallon) metric that results in lower weights for fuel-optimal ways.

5.3 Other Implementation Issues

Street address inputs provided by the user are translated into latitude/longitude pairs using the open source geocoding perl module, Geo::Coder::US. This module is used for geocoding US addresses only. Geocoding is the process of finding corresponding latitude/longitude data given a street address, intersection, or zipcode.

The GUI frontend to display the fuel-optimal route (shown in Figure 2) utilizes Microsoft Bing maps. Routes are color coded and rendered as *polylines* on Bing maps. For example, the fuel-optimal route is a “green” color polyline.

When a query is posed to GreenGPS for the fuel-optimal route between the start address and destination address, the addresses are first geocoded into their corresponding latitude and longitude pairs using the geocoder module. The latitude and longitude pairs of the start and destination addresses are then fed to the routing module which computes the fuel-optimal route (along with the shortest and fastest routes) using the OSM XML database and the prediction models of fuel consumption on streets (computed from the OBD-II sensor data contributed by users). The computed routes are then displayed on the Bing maps based GUI frontend.

6. EVALUATION

We evaluate the performance of GreenGPS in two stages. First, we evaluate the performance of our model by using it to predict the end-to-end fuel consumption for long routes. Second, we evaluate the potential fuel savings of an individual using GreenGPS.

6.1 Model Accuracy

We first evaluate the accuracy of our prediction model in estimating fuel consumption on long routes. These routes are continuous sequences of segments that individuals drove. Only six cars are used in this experiment² because the data

²Ford Focus, 2009; Ford Taurus 2001; Toyota Corolla, 2009;

from the rest of the cars did not include multiple paths (and hence we would not be able to do path-based cross validation, where data collected on one path is used to predict fuel consumption on another). We consider the path error as the end-to-end prediction error for the given path (which is the metric used for evaluation in Section 4). For cross validation, we remove the data points associated with a given path and obtain a model for the car, then obtain the error in predicting fuel consumption for this path based on the computed model. We repeat the above for all the paths.

The entire path error distribution corresponding to the above experiment when prediction for each car is used based on data of the same car (on other paths) is shown in Figure 11. We observe that the path error distribution is nearly normal and that the mean of this distribution is near zero (<1%). We conduct a similar experiment to derive the path error distribution that is achieved by employing clustering such that fuel consumption of cars is predicted from that of other cars in the nearest cluster. To experiment with prediction accuracy of clusters, we remove the data points for each car (as if that car was not known to the system) and cluster the rest of data points, as described in Section 4.4, based on make, year, and both. Fuel consumption of the removed car is then predicted using the nearest cluster. Namely, we first check if a cluster exists with the same car make and year. If no such cluster exists, we check for a cluster of the same make or the same year, respectively. Finally, a model based on all car data is used if all the previous steps fail. The prediction error for each path is computed as before and the distribution is presented in Figure 12. Again, a normal distribution of the path errors is observed with near zero mean (<4%).

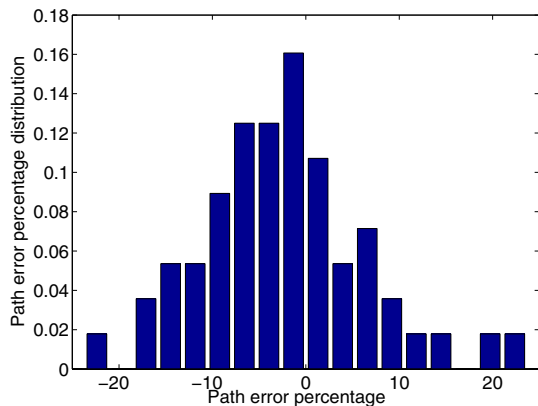


Figure 11: Distribution of path error percentages when training is done using individual cars

In order to understand how path errors vary with path lengths, we bin the paths based on their length and compute the average of the absolute path errors as a function of path length. We repeat this experiment for the case where models are derived for each car individually and the case where models are derived for clusters and the nearest cluster is used. We plot the mean of the absolute path errors for varying path lengths in Figure 13. We observe from Figure 13 that the error decreases with increasing path length for both the individual and cluster based models. As expected, models

Ford Taurus, 2001, Honda Accord, 2001; and Ford Taurus, 2001.

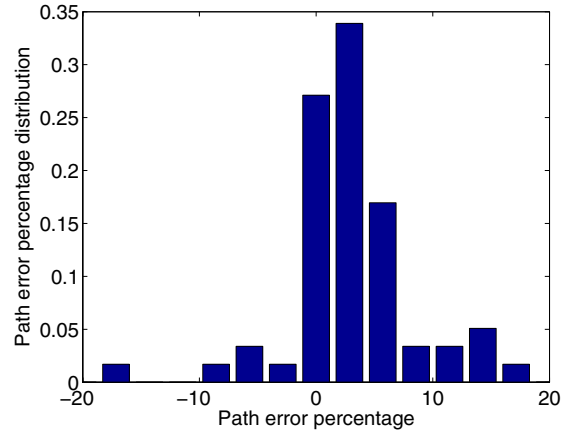


Figure 12: Distribution of path error percentages for the clustering approach

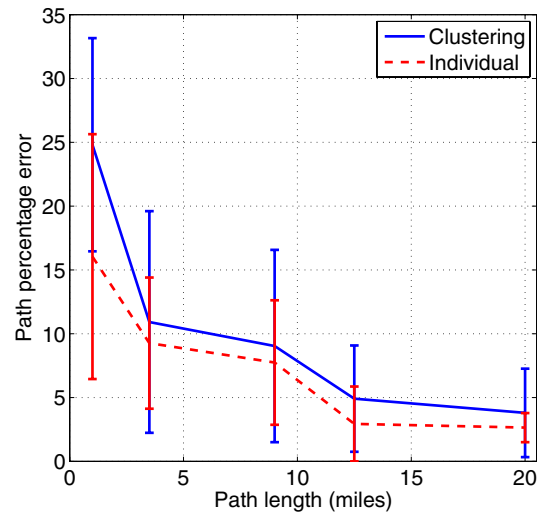


Figure 13: Mean path error when path length is varied for individual car models and cluster based models

based on the owner’s car do better than models based on the nearest cluster, but the cumulative error continues to decrease with distance driven, which is what we want. We have not explored if this holds true when the commutes have large dynamics in speeds, such as in larger cities. The current data set is limited in that it was collected in a fairly quiet town.

From the perspective of building participatory sensing applications, the above suggests the importance of finding models that do not have *biased error*. Since the models often try to predict aggregate or long-term behavior (such as long term exposure to pollutants, annual cost of energy consumption, eventual weight-loss on a given diet, etc), if the error in day-by-day predictions is normally distributed with zero mean, the long-term estimates will remain accurate. Hence, rather than worrying about exact models, GreenGPS attempts to find *unbiased* models, which is easier.

6.2 Fuel Savings

In this section, we evaluate the fuel savings achieved when using the GreenGPS system. As we outlined in the implementation section, we are integrating the street parameters such as the stop signs, traffic lights, and average speed information into the OSM database. To evaluate fuel savings, we chose landmarks in the city of Urbana-Champaign that the authors visit in their daily commutes, such as work, gym, frequently visited restaurants, and shopping complexes. To eliminate subjective choice of routes between the selected landmarks, each of the authors selected a pair of landmarks then looked up both the shortest route and fastest route between these landmarks on MapQuest. The person then drove eight round trips (of approximately 20-40 minutes each) between their selected pair of landmarks; four on the shortest route and four on the fastest route, recording actual fuel consumption for each round trip. The landmarks together with the shortest and fastest routes are shown in Figure 14. We then used the GreenGPS system to predict which of the two compared routes for each pair of landmarks is the better route, which it did correctly in every case.

The fuel consumption data for each roundtrip on the shortest and fastest routes for all the cars in this experiment are shown in Table 4.

We observe from Table 4 that the fuel-optimal route for destinations of the Honda Accord and Ford Taurus was the shortest route, whereas, for the other three destinations it was the fastest route. Hence, picking the shortest or fastest routes consistently is not optimal. To confirm that the differences in fuel consumption between the compared routes are not due to measurement noise, we tested the statistical significance of the difference in means using the two paired *t*-test. The test yielded that the differences are statistically significant with a confidence level of at least 90%. The average savings (by choosing the correct route over the alternative) for each pair of landmarks and car are summarized in Table 4.

Comparing the total fuel consumed on the optimal route to the average of that consumed on the shortest route and fastest route (assuming the driver guesses at random in the absence of GreenGPS), the savings achieved are roughly 6% over the shortest path and 13% over the fastest, which is consistent with data we reported earlier in the feasibility study.³ This is by no means statistically significant, since only a handful of routes were used in the experiments above, but it nevertheless shows promise as a proof of concept.

7. LESSONS LEARNED

This section presents, in its two respective subsections, a brief discussion of our experiences with the GreenGPS service and the limitations of the current study.

7.1 Experiences with GreenGPS

Several lessons were learned from GreenGPS, as an example of participatory sensing applications. First, we observed that data cleaning is an important problem and it is application dependent. We had several occasions when several fields were missing from the data. For example, the GPS sometimes failed to communicate with the DashDyno and the location fields were then empty. A simple scheme was

³The feasibility study used different routes from those reported above

used to filter complete datasets from those that were missing values. Another data-related issue was the presence of noise in the data. For example, in our setup, we observed that (in some car models) whenever the GPS communicated with the DashDyno, the *fuel rate* measurement had a large spike. This was likely due to improper use of sensor IDs, which led to data overwriting. Solutions have to be developed that filter the noise at the source. For example, we developed a simple filter (as a plugin to PoolView) that removes outliers from the data before storing it. An application-specific challenge was observed due to the slight variations in the OBD-II standards among different cars. For example, we observed that the Toyota Prius (by default) outputs the speed and fuel measurements in the metric system, rather than the Imperial system (which happens to be the default for the remaining cars in our dataset). It is harder to propose generic solutions to such problems. They suggest, however, that unlike small embedded systems, participatory sensing applications involve a much larger number of heterogeneous components (e.g., different car types in GreenGPS). As such components interact with each other or with aggregation services, subtle compatibility problems will play an increasing role. Troubleshooting techniques are needed that are good at identifying problems resulting from unexpected or bad interactions among different individually well-behaved components. This is to be contrasted, for example, with debugging tools that attempt to find bugs in individual components.

Next, privacy challenges come to the forefront in participatory sensing systems. A large class of participatory sensing systems monitor location information continuously, which poses significant privacy issues. Simple anonymization of data will not work in such situations, as the GPS traces can lead to privacy breaches (e.g., reveal the home location of the user and thus uncover their identity). Techniques such as the ones proposed in [17] and [31], which rely on data perturbation can be used to preserve privacy, while still allowing accurate modeling. In our current study, individual users simply switch off data collection devices when they feel the need for privacy.

Finally, another lesson learned relates to the initial experimental deployment of participatory sensing systems. A major hurdle in getting participatory sensing systems off the ground is to provide the right incentives to the individuals (who are part of the system) [32]. We believe that the initial deployment, which tends to be sparse, should be carefully designed in order to provide incentives for larger adoption. It should therefore be useful from the very early stages.

7.2 Limitations of Current Study

Apart from the limitations arising from the small size of the data set, discussed earlier, we also make the following observations. As expected, the main factors affecting fuel consumption of a vehicle on a path are the average speed, the speed variability (estimated by averaging the speed squared), and the engine idle time (estimated from the number of stop signs and stop lights on the path). A limitation of the study is that we did not explore the use of real-time traffic conditions for purposes of fuel estimation. Rather, we opted to use statistical averages of speed, speed variability and idle time. It is easy to see how such statistical averages can be computed for different hours of the day and different days of the week given a sufficient amount of historical data, yielding expected fuel consumption (in

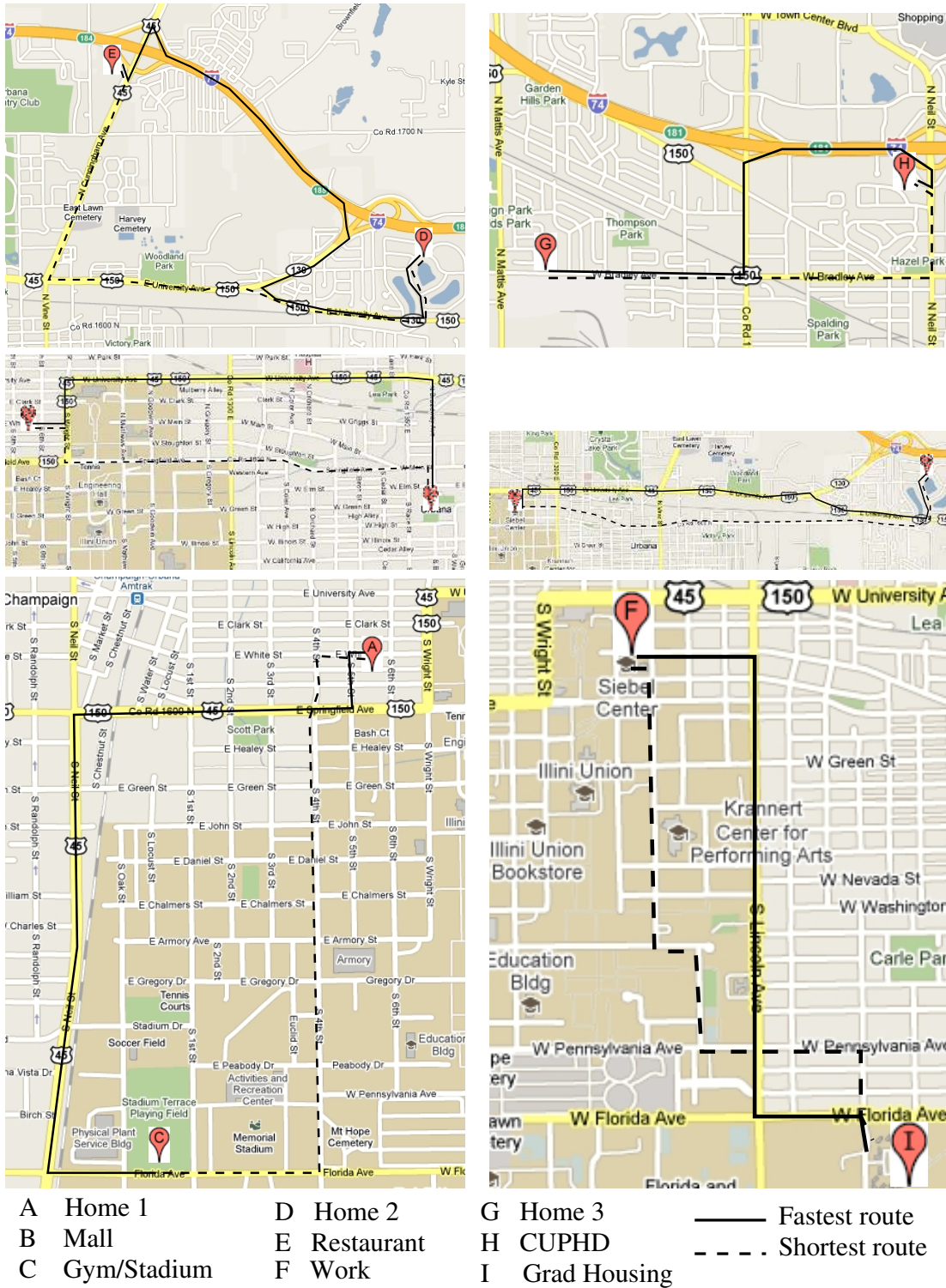


Figure 14: Figure showing the landmarks and corresponding shortest and fastest routes

the statistical sense of expectation). The outcome is that individual trips may differ significantly from the statistical expectation. However, by consistently following routes that have a lower expected fuel consumption, savings will accumulate in the long term. Drivers may think of GreenGPS as

a long-term investment. Short-term results may vary, but long-term expectations should tend to come true.

A limitation of the study, as discussed in Section 4, is that the selection of cars used in our current study (mostly compact and mid-sized sedans) result in a generalization hierarchy that ignores the car *class* (currently incorporates

| Car type | Landmarks | Route type | Fuel consumption (gallons) | | | | GreenGPS prediction | Savings % |
|--------------------|----------------------|------------|----------------------------|------|------|------|---------------------|-----------|
| Honda Accord 2001 | Home 1 to Mall | Shortest | 0.19 | 0.16 | 0.19 | 0.16 | Shortest | 31.4 |
| | | Fastest | 0.22 | 0.23 | 0.25 | 0.22 | | |
| | Home 1 to Gym | Shortest | 0.19 | 0.20 | 0.19 | 0.18 | Shortest | |
| | | Fastest | 0.21 | 0.23 | 0.22 | 0.25 | | |
| Ford Taurus 2001 | Home 2 to Restaurant | Shortest | 0.24 | 0.23 | 0.23 | 0.22 | Shortest | 26 |
| | | Fastest | 0.3 | 0.28 | 0.29 | 0.29 | | |
| Toyota Celica 2001 | Home 2 to Work | Shortest | 0.18 | 0.16 | 0.18 | 0.17 | Fastest | 10.1 |
| | | Fastest | 0.17 | 0.14 | 0.16 | 0.15 | | |
| Nissan Sentra 2009 | Home 3 to CUPHD | Shortest | 0.14 | 0.15 | 0.15 | 0.15 | Fastest | 8.4 |
| | | Fastest | 0.13 | 0.13 | 0.14 | 0.14 | | |
| Honda Civic 2002 | Grad housing to Work | Shortest | 0.33 | 0.32 | 0.33 | 0.3 | Fastest | 18.7 |
| | | Fastest | 0.25 | 0.28 | 0.27 | 0.24 | | |

Table 4: Table showing fuel consumptions for the various roundtrips between different landmarks

only car make and year). Future deployments will consider a broader range of vehicles, such as SUVs, minivans, and light trucks. The data from these deployments will be used to recompute a better generalization hierarchy.

In order to achieve the next level of optimization, a next generation of GreenGPS can take into account the real-time situation. Our experience reveals, not surprisingly, that the degree of congestion plays the largest role in accounting for fuel consumption variations among individual trips of the same vehicle. On lightly-utilized streets, another main factor is the degree to which traffic lights are synchronized. Lack of synchronization accounted for up to a 50% increase in fuel consumption in our measurements.

Another limitation of the current service is that it does not properly account for turns. Turns on the path add fuel consumption, often because delays in the turn lane differ from those in the through lane. In particular, our measurements show that left turns may add a considerable amount of delay to a path. Hence, routing should account for the type of turn as well.

Finally, we expect that fuel savings in larger cities will be higher than those reported in this paper, both due to the larger variability in traffic conditions that could be taken advantage of, and because of the larger connectivity which offers more alternatives in the choice of route. With the above caveats, we believe that the study remains of interest in that it explores problems typical to many participatory sensing applications, such as overcoming conditions of sparse deployment, adjusting to heterogeneity, and living with large day-to-day errors towards estimating cumulative properties. The GreenGPS study could therefore serve as an example what to expect in building similar services, as well as a recipe for some of the solutions.

8. RELATED WORK

We divide this section into two parts, the first part presents related work in participatory sensing and the second examines fuel efficiency related literature.

8.1 Participatory Sensing

Our navigation service is an example of participatory sensing services, that have recently become popular in networked sensing. The concept of participatory sensing was introduced in [9]. In participatory sensing, individuals are tasked with data collection which is then shared for a common pur-

pose. A broad overview of such applications is provided in [2]. Several early applications have been published. Examples include CenWits [23], a participatory sensing network to search and rescue hikers, CarTel [24], a vehicular sensor network for traffic monitoring, BikeNet [12], a bikers sensor network for monitoring popular cyclist routes, and ImageScape [33], cellphone camera networks for sharing diet related images. Our application, GreenGPS, introduces a novel example of this genre that enables individuals to compute fuel efficient routes within a city.

8.2 Fuel Efficiency

A comprehensive study that provides optimal route choices for lowest fuel consumption is presented in [14]. In the paper, fuel consumption measurements are made through the extensive deployment of sensing devices (different from the OBD-II) in experimental cars. These fuel consumption measurements are then used to compute the lowest fuel consumption route. In contrast to the work in [14], our paper uses a sparse deployment to build mathematical models for predicting fuel consumption for other streets and cars. In [8], the influence of driving patterns of a community on the exhaust emissions and fuel consumption were studied. Feedback was provided to the community regarding the driving patterns to cut back on the fuel consumption and exhaust. A driver support tool, FEST, was developed in [11]. FEST uses sensors installed in the car along with a software to determine the driving behavior of the driver and provide real-time feedback to the individual for the purpose of reduction in fuel consumption. An extension to FEST that includes more experiments and further evaluation can be found in [38]. A feedback control algorithm was developed in [34] that determines speed of automobiles on highways with varying terrain to achieve minimal fuel consumption. An extension to the work in [34] was developed in [22]. In [22], suggestions of driving style to minimize fuel consumption were made for varying road and trip types (e.g. constant grade road, hilly road). The problem was formulated using a control theoretic approach.

UbiGreen [16] is a mobile tool that tracks an individual's personal transportation and provides feedback regarding their CO₂ emissions.

In a separate study [25], it was shown that rising obesity has a significant impact on the total fuel consumption in the US. Models were developed that studied the impact of obesity on the amount of fuel consumed in passenger vehicles.

Our work represents the first participatory sensing service that aims at improving fuel consumption. Using data collected from volunteer participants, models are built and continuously updated that enable navigation on the minimum-fuel route.

9. CONCLUSIONS AND FUTURE WORK

In this paper, we developed a navigation service, called GreenGPS, that computes fuel efficient routes. This service relies on OBD-II data collected and shared by a set of users via a participatory sensing framework, called PoolView. Lessons were described that extrapolate from experiences with this service to broad issues with participatory sensing service design in general. This paper shows that significant fuel savings can be achieved by using GreenGPS, which not only reduces the cost of fuel, but also has a positive impact on the environment by reducing CO₂ emissions. An important issue addressed was surviving conditions of sparse deployment. GreenGPS achieves this by using a hierarchy of models developed in this paper to estimate the fuel consumption, and shooting for models that are unbiased, if not accurate. Our future work will address the challenges associated with real-time prediction, as well as experiences from a longer-term deployment. We will also explore the use of data cubes in the context of building generalized hierarchical models.

10. ACKNOWLEDGEMENTS

The authors thank the shepherds, Dr. Maria Ebling and Prof. Brian Noble, and the anonymous reviewers for providing valuable feedback that significantly improved the paper. We would also like to thank all the drivers who volunteered to collect data, without whom this paper would have not been possible. The work described in this paper was funded in part by NSF grant CNS 05-54759, Microsoft Research, and the Siebel Foundation.

11. REFERENCES

- [1] AAA. National average gas prices. <http://www.fuelgaugereport.com/>, April 2010.
- [2] T. Abdelzaher et al. Mobiscopes for human spaces. *IEEE Pervasive Computing*, 6(2):20–29, 2007.
- [3] Actron. Elite autoscanner. http://www.actron.com/product_category.php?id=249.
- [4] Auterra. Dashdyno. <http://www.auterraweb.com/dashdynoseries.html>.
- [5] AutoTap. Autotap reader. <http://www.autotap.com/products.asp>.
- [6] AutoXRay. Ez-scan. http://www.autoxray.com/product_category.php?id=338.
- [7] D. M. Bevely, R. Sheridan, and J. C. Gerdes. Integrating ins sensors with gps velocity measurements for continuous estimation of vehicle sideslip and tire cornering stiffness. In *Proc. of American Control Conference*, pages 25–30, 2001.
- [8] K. Brundell-Freij and E. Ericsson. Influence of street characteristics, driver category and car performance on urban driving patterns. *Transportation Research, Part D*, 10(3):213–229, 2005.
- [9] J. Burke et al. Participatory sensing. Workshop on World-Sensor-Web, co-located with ACM SenSys, 2006.
- [10] Y. Chen et al. Regression cubes with lossless compression and aggregation. *IEEE Transactions on Knowledge and Data Engineering*, 18(12):1585–1599, 2006.
- [11] V. der Voort. Fest - a new driver support tool that reduces fuel consumption and emissions. *IEE Conference Publication*, 483:90–93, 2001.
- [12] S. B. Eisenman et al. The bikenet mobile sensing system for cyclist experience mapping. In *Proc. of SenSys*, November 2007.
- [13] EPA. Emission facts: Greenhouse gas emissions from a typical passenger vehicle. <http://www.epa.gov/OMS/climate/420f05004.htm>.
- [14] E. Ericsson, H. Larsson, and K. Brundell-Freij. Optimizing route choice for lowest fuel consumption - potential effects of a new driver support tool. *Transportation Research, Part C*, 14(6):369–383, 2006.
- [15] J. Farrelly and P. Wellstead. Estimation of vehicle lateral velocity. In *Proc. of IEEE Conference on Control Applications*, pages 552–557, 1996.
- [16] J. E. Froehlich et al. Ubigreen: Investigating a mobile tool for tracking and supporting green transportation habits. In *In Proc. of Conference on Human Factors in Computing*, pages 1043–1052, 2009.
- [17] R. K. Ganti, N. Pham, Y.-E. Tsai, and T. F. Abdelzaher. Poolview: Stream privacy for grassroots participatory sensing. In *Proc. of SenSys '08*, pages 281–294, 2008.
- [18] Garmin eTrex Legend. www8.garmin.com/products/etrexlegend.
- [19] Google. Google maps. <http://maps.google.com>.
- [20] GPS POI. Red light database. <http://www.gps-poi-us.com/>.
- [21] J. Gray et al. Data cube: A relational aggregation operator generalizing group-by, cross-tab and sub-totals. *Data Mining and Knowledge Discovery*, 1(1):29–54, 1997.
- [22] J. N. Hooker. Optimal driving for single-vehicle fuel economy. *Transportation Research, Part A*, 22A(3):183–201, 1988.
- [23] J.-H. Huang, S. Amjad, and S. Mishra. Cenwits: a sensor-based loosely coupled search and rescue system using witnesses. In *Proc. of SenSys*, pages 180–191, 2005.
- [24] B. Hull et al. Cartel: a distributed mobile sensor computing system. In *Proc. of SenSys*, pages 125–138, 2006.
- [25] S. H. Jacobson and L. A. McLay. The economic impact of obesity on automobile fuel consumption. *Engineering Economist*, 51(4):307–323, 2006.
- [26] MapQuest. <http://www.mapquest.com>.
- [27] National Aeronautics and Space Administration (NASA). Landsat data. <http://landsat.gsfc.nasa.gov/data/>.
- [28] Nic Roets. Gosmore. <http://wiki.openstreetmap.org/wiki/Gosmore>.
- [29] OpenStreetMap. Openstreet map. <http://wiki.openstreetmap.org/>.
- [30] Owen Brotherwood. Symbtelm. <http://sourceforge.net/apps/trac/symbtelm/>.
- [31] N. Pham, R. Ganti, Y. Sarwar, S. Nath, and

- T. Abdelzaher. Privacy-preserving reconstruction of multidimensional data maps in vehicular participatory sensing. In *LNCS Proc. of EWSN*, pages 114–130, 2010.
- [32] S. Reddy, D. Estrin, and M. Srivastava. Recruitment framework for participatory sensing data collections. In *To Appear in Proc. of Intl. Conference on Pervasive Computing*, 2010.
- [33] S. Reddy et al. Image browsing, processing, and clustering for participatory sensing: Lessons from a dietsense prototype. In *Proc of EmNets*, pages 13–17, 2007.
- [34] A. B. Schwarzkopf and R. B. Leipnik. Control of highway vehicles for minimum fuel consumption over varying terrain. *Transportation Research*, 11(4):279–286, 1977.
- [35] Traffic. Real-time traffic conditions. <http://www.traffic.com/>.
- [36] H. E. Tseng. Dynamic estimation of road bank angle. *Vehicle System Dynamics*, 36(4-5):307–328, 2001.
- [37] US Census Bureau. Tiger database. <http://www.census.gov/geo/www/tiger/>.
- [38] M. van der Voort, M. S. Dougherty, and M. van Maarseveen. A prototype fuel-efficiency support tool. *Transportation Research, Part C*, 9(4):279–296, 2001.

Appendix: Deriving the physical model for fuel consumption

Assuming that the engine RPM is ωs^{-1} , the torque generated by the engine is $\Gamma(\omega)$, the final drive ratio is G , the k -th gear ratio is g_k , and the radius of the tire is r , then F_{engine} is given by the following equation: $F_{engine} = \frac{\Gamma(\omega)Gg_k}{r}$.

The frictional force $F_{friction}$ is characterized by the gravitational force acting on the car, given by $mg\cos(\theta)$, where m is the mass of the vehicle and g is the gravitational acceleration and the coefficient of friction, c_{rr} . The equation for frictional force is: $F_{friction} = c_{rr}mg\cos(\theta)$.

The gravitational force, F_g , due to the slope is given by the following equation: $F_g = mgsin(\theta)$.

Finally, the force due to air resistance, F_{air} , is given by the following equation: $F_{air} = \frac{1}{2}c_dA\rho v^2$.

In the above equation, c_d is the coefficient of air resistance, A is the frontal area of the car, ρ is the air density, and v is the current speed of the car.

Assuming that the car is on an upslope, the final force acting on the car is given by the following equation: $F_{car} = F_{engine} - F_{friction} - F_{air} - F_g$.

In order to obtain a relation between the fuel consumed and the above forces, we note that the fuel consumed is related to the power generated by the engine at any instance of time t . If f_r is the fuel rate (fuel consumption at a given time instance) and P is the instantaneous power, then $f_r \propto P$. Power is related to the torque function, $\Gamma(\omega)$, and engine RPM, ω as follows: $P = 2\pi\Gamma(\omega)\omega$. Hence, we obtain $f_r = \beta\Gamma(\omega)\omega$.

In the above equation, β is a constant. Further, we also have the relationship $v = r\omega$ from rotational dynamics. From the above equations, we obtain the fuel consumption rate as a function of the forces acting on the car shown be-

low:

$$\begin{aligned} F_{car} &= ma \\ &= \frac{f_r G g_k}{r \omega \beta} - c_{rr} m g \cos(\theta) - \frac{1}{2} c_d A \rho v^2 \\ &\quad - m g \sin(\theta) \\ m a v &= \beta' f_r - c_{rr} m g \cos(\theta) v \\ &\quad - \frac{1}{2} c_d A \rho v^3 - m g v \sin(\theta) \\ f_r &= k_1 m a v + k_2 m v \cos(\theta) \\ &\quad + k_3 A v^3 + k_4 m v \sin(\theta) \end{aligned}$$

Finally, we can obtain the equation for the fuel consumed, f_c by integrating the rate of fuel consumption with respect to time. We obtain the following equation:

$$\begin{aligned} f_c &= \int_{t_1}^{t_2} f_r(t) dt \\ &= \int_{t_1}^{t_2} (k_1 m a v + k_2 m v \cos(\theta) + k_3 A v^3 \\ &\quad + k_4 m v \sin(\theta)) dt \end{aligned}$$

In order to further derive a model that can be used for regression analysis, we will detail the various components that are part of the fuel consumption of a car. As shown in the above equation, a moving car at a constant speed on a straight road which does not encounter any stop lights or traffic will only need to overcome the frictional forces caused by the road, the air, and gravity. These are represented by $\int_{t_1}^{t_2} k_2 m v \cos(\theta)$, $\int_{t_1}^{t_2} k_3 A v^3$, and $\int_{t_1}^{t_2} k_4 m v \sin(\theta)$, respectively. On the other hand, the first component $\int_{t_1}^{t_2} k_1 m a v$ can be broken down further into two components, one is the extra fuel consumed due to encountering stop signs (ST) and traffic lights (TL) and the second one is the extra fuel consumed due to congestion. Hence, the previous equation now becomes the following:

$$\begin{aligned} f_c &= \int_{t_1}^{t_2} (k_{11} m a v (ST + \nu TL) \\ &\quad + k_{12} m a v) + k_2 m v \cos(\theta) \\ &\quad + k_3 A v^3 + k_4 m v \sin(\theta)) dt \end{aligned}$$

If we replace v with \bar{v} , the average speed, assume that θ remains constant, and we know $a = dv/dt$, we can further simplify the above integral to the following:

$$\begin{aligned} f_c &= k_{11} m \bar{v}^2 (ST + \nu TL) + \frac{k_{12} m \bar{v}^2}{2} \\ &\quad + k_2 m \Delta d \cos(\theta) + k_3 A \bar{v}^3 \Delta t \\ &\quad + k_4 m \Delta d \sin(\theta) \end{aligned}$$

In the above equation, Δd is the distance traveled and Δt is the time traveled. Dividing the above equation by Δd gives us the metric *fuel consumed per mile (gpm)*, which is appropriate for our analysis purposes. Hence, our final model will now be:

$$\begin{aligned} gpm &= k_{11} m \bar{v}^2 \frac{(ST + \nu TL)}{\Delta d} + k_{12} m \frac{\bar{v}^2}{2 \Delta d} \\ &\quad + k_2 m \cos(\theta) + k_3 A \bar{v}^2 + k_4 m \sin(\theta) \end{aligned}$$