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

2/28/2011

Presenter: Steve Kopman

Abstract

- GreenGPS uses participatory sensors to determine a fuel-efficient route between two arbitrary end-points.
- Utilizes the OBD-II interface to retrieve data from sensing automobiles
- The most fuel-efficient route is not always the shortest or fastest

GreenGPS Challenges

- Sparse deployment and data
  - "How does the system generalize from sparse sampling of high-dimensional space to produce compact descriptions of complex phenomena"

Major Contributions

- Demonstrate the use of participatory sensing system to develop a fuel-saving navigation system
- Show a 6% on average savings vs. shortest route and 13% over fastest
- Demonstrate how to generalize sparse samples of high-dimensional spaces to develop models of complex non-linear phenomena; one size does not fit all

SYSTEM OVERVIEW

Current Navigation vs. GreenGPS

	Google Maps/MapQuest	GreenGPS
Data Source	Web-based maps	Web-based maps plus user fuel economy data
Priority Route	Fastest or Shortest	Fuel Efficient

- Example
  - A freeway route might be the fastest but fuel economy decreases non-linearly with speed.
  - City streets might be the shortest path however traffic might decrease total fuel efficiency

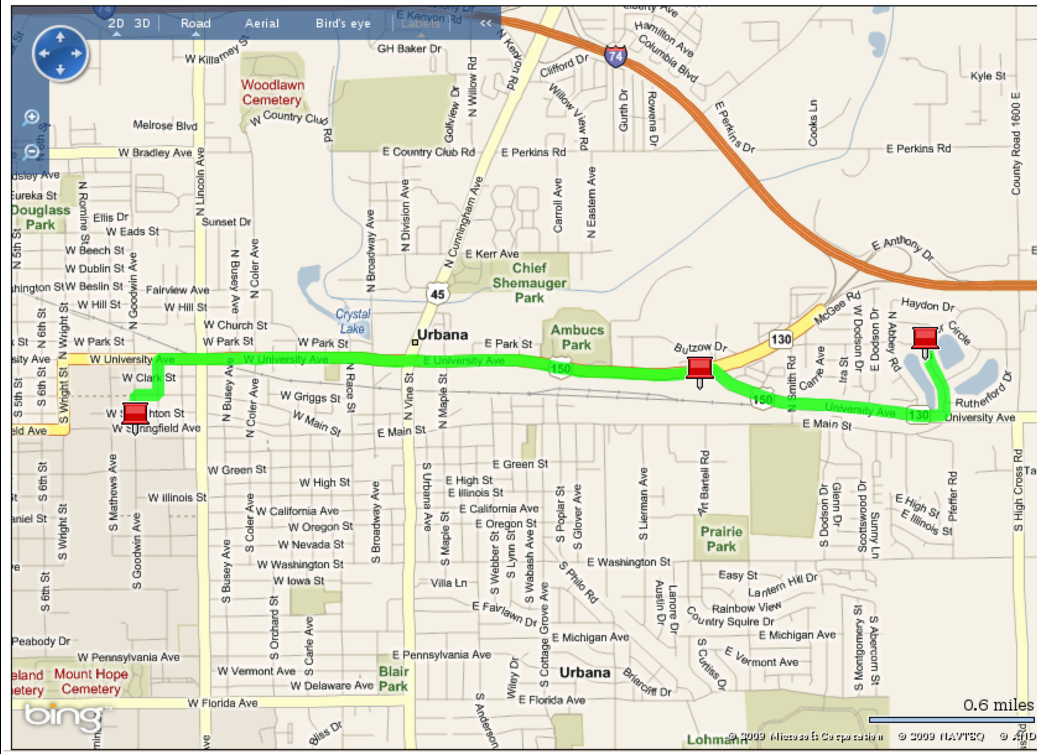
GreenGPS Map Interface

Please enter the start and end addresses.

Start Address: Street: 2889 Beringer cir City: Urbana State: IL

End Address: Street: 201 N goodwin ave City: Urbana State: IL

Vehicle Details: Make: Toyota Model: Celica Year: 2001 [Find Route]



Route types

- ☐ shortest
- ☐ fastest
- ☑ fuelOptimal

Data Collection

- On-Board Diagnostics (OBD)-II System
  - Present on all cars sold since 1996
  - Provides monitoring of engine RPM, coolant temperature, fuel consumption, speed and idle time.
  - A standard OBD-II scanner tool to retrieve data
- Remote monitors can also be used such as On-Star or Lexus Link
  - Not used in this paper

GreenGPS Hardware

- System consists of
  - OBD Vehicle Connector
  - Garmin GPS
  - DashDyno data recorder
- Data is stored on a SD card in the DashDyno
- Can be connected to a mobile phone via a bluetooth to OBD-II adapter

Figure 3: Hardware used for data collection

Users

- GreenGPS consists of two types of users
  - Members – have OBD-II equipment and upload data specific to their vehicle
  - Non-members do not contribute data and therefore receive reduced accuracy for fuel economy

GreenGPS Concept

- User enters a start and end address
  - Not all streets can be measured for all vehicles (impractical)
- Goal is to use a generalized prediction model
  - Static street parameters such as # traffic lights and stop signs are pulled from traffic databases
  - Dynamic parameters such as avg speed, avg congestion are gathered via GPS and traffic monitoring services
  - Vehicle parameters are entered by user (weight, frontal area)

Sensing Framework

- Framework is based on previous work called PoolView
- PoolView is a client-side interface for data uploads
- Data is delivered to a central server
- Software needs to be downloaded by the user to their device (PC or phone)
- The DynoDash records 16 parameters including vehicle speed, total fuel consumption, rate of fuel consumption, GPS

University of Central Florida

EXPERIMENT AND FEASIBILITY

Experiment

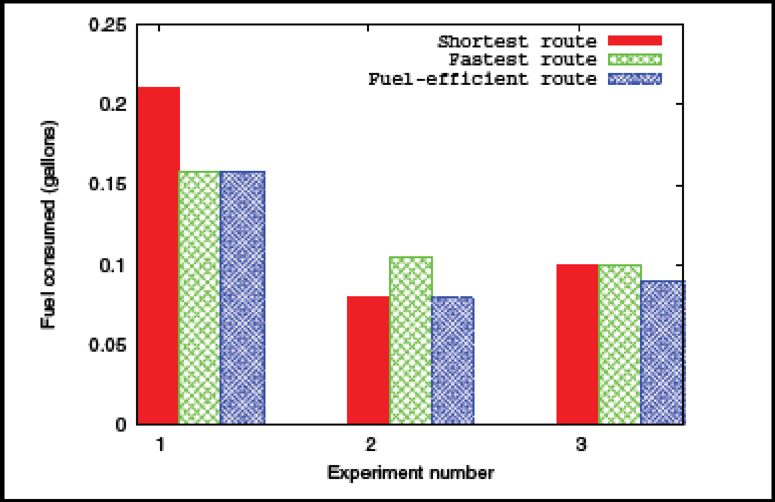
- Use Dijkstra’s algorithm to calculate the minimum fuel route
- Three month study using 16 cars with different drivers
- Over 1000 miles driven between Urbana-Champaign (UC)

Small Scale Feasibility Study

- Three different cards and drivers between UC landmarks
  - Shopping Center, authors work and football stadium
  - Shortest and fastest routes calculated using MapQuest

Small Scale Results

- Average of 11% overhead for always taking the "fastest" route
- Average of 11.5% overhead for "shortest"



- Estimated 200m light vehicles in the U.S., avg of 12,000mi/yr, avg mpg=20.3
- If 5% of vehicles used GreenGPS for a 10% fuel savings on ¼ of all routes, overall fuel savings is 177 million gallons of fuel (~$500m)

MODELING STRUCTURE

Cars and Miles Driven

- Issue – Data is sparse and multi-dimensional
- Limited proof-of-concept was performed using cars and mileage listed in Table 1
- Limited to landmarks in Urbana-Champaign

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

University of Central Florida

Derivation of Model Structure

- Describes how various parameters are related but does not evaluate the various constants and proportionality coefficients
- The first step was to plot the distribution of miles per gallon (mpg) for all data collected (shown on next slide)
- Results – Distribution was 5-50mpg with a standard deviation of 9.12 mpg
  - Very high

Metrics

- Cumulative Error
  - Total end-to-end prediction error
- Cumulative Percentage Error
  - Normalized error to the total distance drive

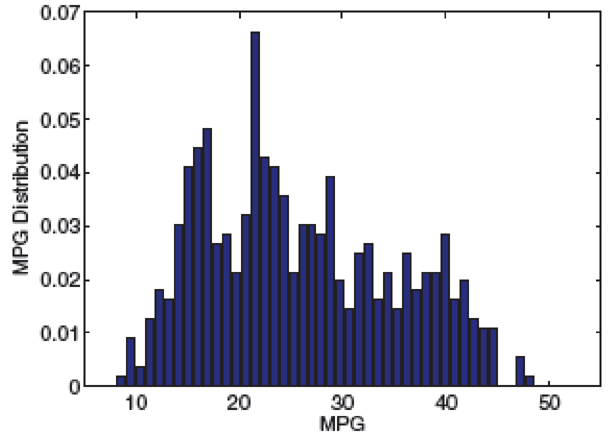


Figure 5: Figure showing the real mpg distribution for all the sixteen users

Fuel Consumption Model

- Parameters
  - Static Segments
    - # of stop signs (ST)
    - # of traffic lights (TL)
    - Distance traveled (Δd)
  - Dynamic Segment
    - Average speed (v)
  - Car Specific
    - Weight of car (m)
    - Car Frontal Area (A)

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

Fuel Consumption Model

- Using this model worked well when using data from one car for that car however using all data, except the car under evaluation, increased the cumulative error

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

Model Clustering

- Clustering can be used to improve generalization by grouping similar items
- Three data dimensions chosen for this experiment
  - Make, Year and Class
  - Provides eight combinations
  - The cumulative error percentage is calculated for each one to find the best perform – Make/Year is best

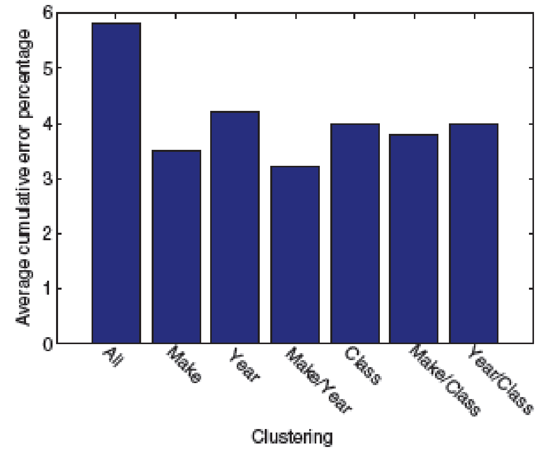


Figure 8: Cumulative error percentage of the models obtained from various clustering approaches

Model Clustering Errors

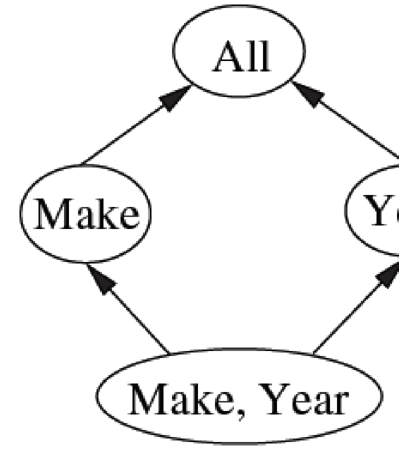


Figure 9: Model generalization from fine grained clusters

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

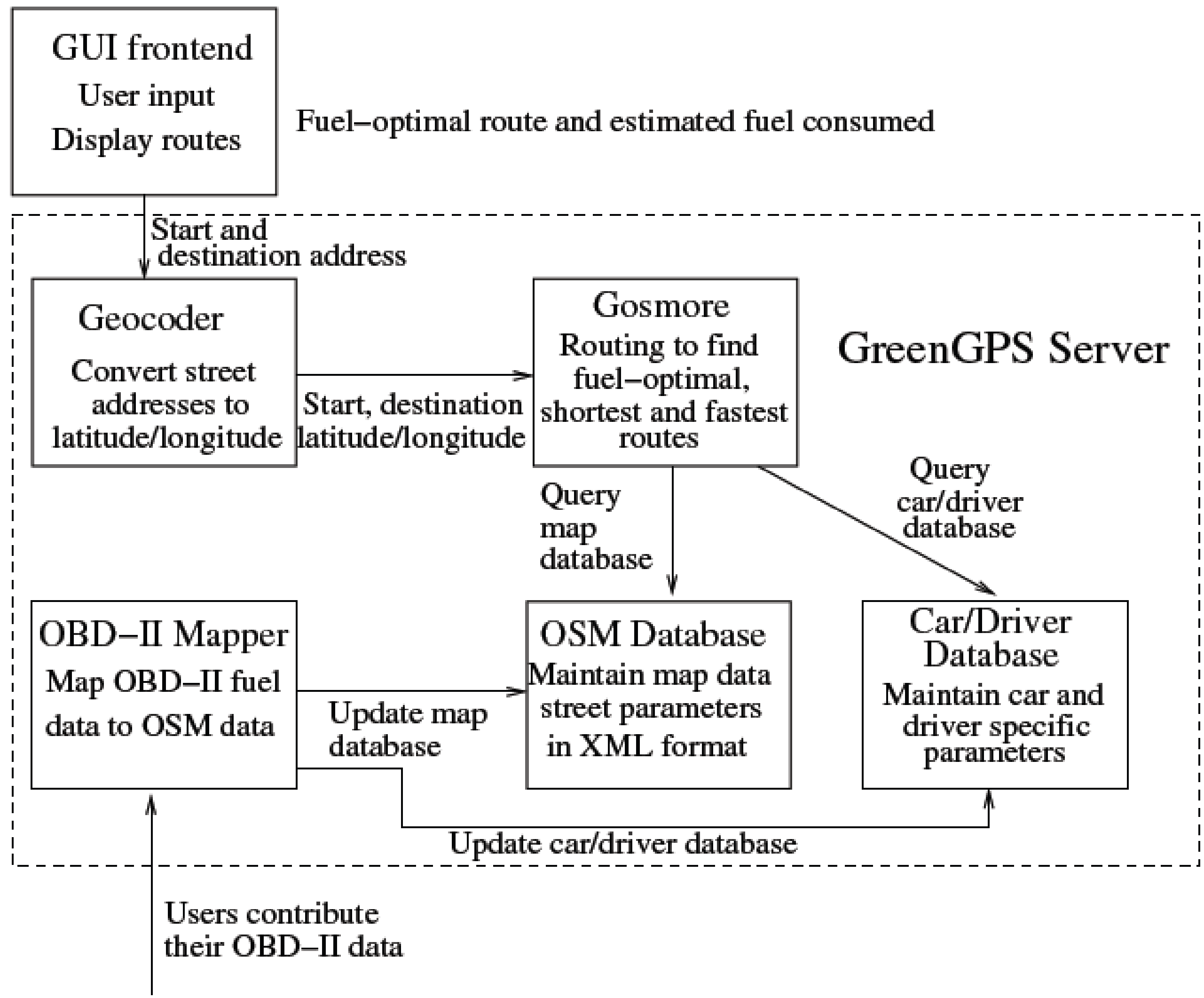
Table 3: Table showing the cumulative error percentage for each individual car when model clustering is used

University of Central Florida

IMPLEMENTING GREENGPS

University of Central Florida

Modules



GUI frontend
User input
Display routes
Fuel–optimal route and estimated fuel consumed
Start and destination address
Geocoder
Convert street addresses to latitude/longitude
Start, destination latitude/longitude
Gosmore
Routing to find fuel–optimal, shortest and fastest routes
GreenGPS Server
Query map database
Query car/driver database
OBD–II Mapper
Map OBD–II fuel data to OSM data
Update map database
OSM Database
Maintain map data street parameters in XML format
Car/Driver Database
Maintain car and driver specific parameters
Update car/driver database
Users contribute their OBD–II data

Server

- Combines open source software services to provide the fuel-efficient route computation
  - OpenStreetMap (OSM)
    - Combines collected data and an editable, XML street map
    - Directed graph consisting of nodes, ways and relations
    - Node – fixed coordinates; expresses a POI
    - Way – Ordered list of nodes with tags to specify the meaning (i.e. road, river, etc.)
    - Relations – models a relationship between objects where each member has a specific route (bus route, etc.)

University of Central Florida

Model Clustering

- Utilizes Data Cubes to implement a 3-dimensional regression cube in C++
  - Make, class, year
- Broken up into 1 mile segments that are rows in a database
  - Each row contains five additional attributes that are the values of the physical model parameters
- The regression model is computed for all clusters and a recursive search for a specific triple (make, class, year) within the (make, year) pair, then (make) cluster and (year) cluster
- The first match found is used for prediction

Routing

- Uses a open-source software called Gosmore
  - Implemented in C++
  - Uses OSM XML map data to find a route
  - Basically a weighted Dijkstra algorithm on the OSM map
    - Nodes are OSM nodes, edges are OSM ways, lengths are the weights
  - Fastest route = Distance/speed factor
  - Fuel optimized route = Distance/mpg
  - Lower weights are better

EVALUATION

Evaluation

- Performed in two stages
  - Stage 1 – Use to predict end-to-end fuel consumption for long routes
  - Stage 2 – Evaluate potential fuel savings of an individual using GreenGPS

Model Accuracy

- Utilized only the six cars that included multiple paths
  - Need to do path-based cross-validation
- Cross-validation
  - Removed the current path and compute a model for the car
  - Compare predicted fuel consumption against the path to obtain the error
    - Repeat for all paths

Model Accuracy

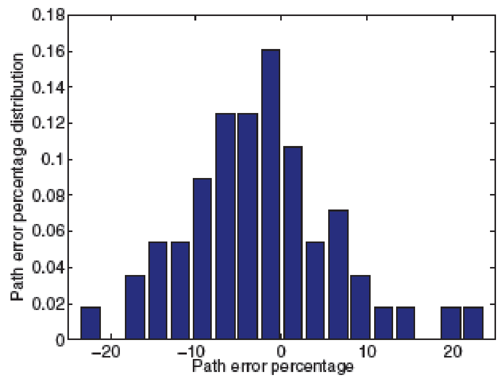


Figure 11: Distribution of path error percentages when training is done using individual cars

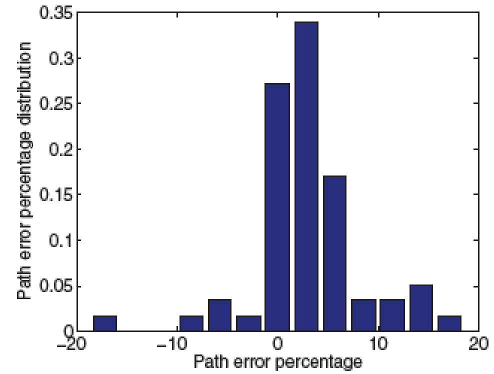


Figure 12: Distribution of path error percentages for the clustering approach

- Figure 11 – Path error percentages calculated by removing the path but calculating prediction from the same car on different paths
- Figure 12 – Removed all data for that car and used clustering the find an appropriate model
- Both versions have normal distributions with a near zero mean (long term savings)
- Additionally, path error is reduced with the length of travel

Fuel Savings

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	19.7
		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

Lessons Learned

- Experience with GreenGPS
  - Data cleaning is important
    - Sometimes GPS would not provide/record data to the DashDyno
    - Some cars used metric, other imperial system
    - Need to filter complete datasets
  - Privacy
    - User activity is traced via GPS
    - User can turn this off but this affects data
  - What incentive should be provided to the user?
    - Need to mitigate sparse data

Future Work/Conclusion

- **Future work**
  - Include real-time traffic data to improve idle/stop time accuracy
  - Need more experience with longer term deployment
  - Explore the use of data cubes in the context of building generalized hierarchical models
- **Conclusion**
  - GreenGPS shows significant savings for fuel and CO2 emissions
  - Prediction models can be made from sparse data using clustering