Pedestrian-Aware Engine Management Strategies for Plug-In Hybrid Electric Vehicles

Yingqi Gu, Mingming Liu, Joe Naoum-Sawaya, Emanuele Crisostomi, Giovanni Russo, and Robert Shorten

***Abstract*— Electric vehicles (EVs) and plug-in hybrid EVs (PHEVs) are increasingly being seen as a means of mitigating the pressing concerns of traffic-related pollution. While hybrid vehicles are usually designed with the objective of minimizing fuel consumption, in this paper we propose a engine management strategies that also consider environmental effects of the vehicles to pedestrians outside of the vehicles. Specifically, we present the optimisation-based engine energy management strategies for PHEVs that attempt to minimize the environmental impact of pedestrians along the route of the vehicle, while taking account of route-dependent uncertainties. We implement the proposed approach in a real PHEV and evaluate the performance in a hardware-in-the-loop platform. A variety of simulation results are given to illustrate the efficacy of our proposed approach.**

***Index Terms*— Plug-in hybrid electric vehicles (PHEVs), engine management, robust optimisation.**

I. Introduction

WITH the increasing concerns over air quality issues arising from traditional internal combustion engine (ICE) vehicles, plug-in vehicles are seen worldwide as an efficient and effective means of transportation that may help in addressing pollution related challenges. Plug-in Hybrid Electric Vehicles (PHEVs), in particular, are becoming increasingly popular with the general public. Traditionally, such vehicles have two modes of operation: a fully electric mode, and a hybrid mode, the latter of which is designed by the manufacturer to maximise fuel efficiency [1]–[5]. Recently, several authors [6]–[8] have suggested exploring the actuation possibilities in such vehicles, namely to automate the on/off switching of the fully electric mode, to address not only fuel efficiency but also pollution issues in urban areas. Specifically, in [6], a number of control strategies were implemented on PHEVs to regulate traffic-related pollution in an urban environment. This was achieved by considering PHEVs as power-split devices and electric energy was used to keep the pollutant level always below a safety pollutant threshold. In particular, the authors had formulated these problems as utility maximisation problems, in terms of different notions of fairness, and had addressed them in a distributed resource-allocation framework using ideas from Internet congestion control. These ideas are further explored for a network of vehicles in [9] and [10] by formulating a constrained optimisation problem. In a similar fashion, for a single vehicle in [11] Guardiola *et al.* introduced an online strategy to adapt the engine calibration to the driving conditions continuously, with the final objective of minimising fuel consumption while fulfilling some emission limits for ICE vehicles. A key contribution of this latter work is accounted for the complexity of the ICE in the optimisation problem. Our objective in this present work is to further investigate these ideas. Specifically, we wish to make use of the fact that there are many data feeds that are available in real time in our cities that give reliable information concerning the density of pedestrians. For example, twitter feeds, mobile phone mobility data, and other historical data, all reveal this information. Given this information, and a budget on available electrical power, we wish to orchestrate switching between fully electric mode, and hybrid mode, so that the impact on pedestrians is kept to a minimum. We argue that such data should be, and perhaps must be, incorporated into engine management strategies, so that the on/off switching of the fully electric mode can be orchestrated to minimise the environmental impact of the vehicle on the population.

The current paper extends our previous work in a number of ways. In particular, the work in the present paper builds on [6]–[8], [12]. In [6], we had proposed using feedback control theory to regulate pollution level in a geo-fenced urban area. This was achieved by orchestrating the switching into fully electric mode of a network of vehicles using distributed stochastic algorithms. In [7] and [8], distributed engine management strategies for a fleet of hybrid vehicles were proposed to optimally manage a budget of energy. A severe limitation of this work was that the routes travelled by each vehicle were assumed to be known a priori. This latter issue is addressed in a different context in [12] in which Markov-decision-based route-prediction engines are proposed and validated.[1] Our present work uses elements of all four papers to create a framework in which a meaningful optimisation can be formulated with a view to protect pedestrians. Thus, the main

Manuscript received December 23, 2016; revised April 6, 2017; accepted April 8, 2017. Date of publication May 23, 2017; date of current version December 26, 2017. The work of Y. Gu, M. Liu, and R. Shorten was supported by Science Foundation Ireland under Grant 11/PI/1177. The work of J. Naoum-Sawaya was supported by NSERC Discovery Grant RGPIN-2017-03962. The Associate Editor for this paper was J. E. Naranjo. *(Corresponding author: Mingming Liu.)*

Y. Gu, M. Liu, and R. Shorten are with the School of Electrical, Electronic and Communications Engineering, University College Dublin, 4 Dublin, Ireland (e-mail: yingqi.gu@ucdconnect.ie; mingming.liu@ucd.ie; robert.shorten@ucd.ie).

J. Naoum-Sawaya is with the Ivey Business School, University of Western Ontario, London, ON N6A 3K7, Canada (e-mail: jnaoum-sawaya@ivey.ca).

E. Crisostomi is with the Department of Energy, Systems, Territory and Constructions Engineering, University of Pisa, 56122 Pisa, Italy (e-mail: emanuele.crisostomi@unipi.it).

G. Russo is with IBM Research, 4 Dublin, Ireland (e-mail: grusso@ie.ibm.com).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TITS.2017.2697044

[1] See https://goo.gl/fpqdt3. for applications of this engine.

contribution of this paper is to propose a new paradigm that manages the way in which a PHEV discharges its limited battery with response to the population density across various routes that may be travelled by a vehicle during a particular journey. This work goes beyond our previous work and other work in the literature by incorporating the following features into the engine management strategies.

A. A route prediction engine is designed which, given historical data of the driver, estimates the probability of the routes likely to be travelled by the driver.

B. Given A, and an energy budget, online engine management optimisation strategies are proposed, that are based on the distribution of the population over all expected travel routes, as characterised by the route prediction engine.

C. Our proposed engine management strategies have been implemented in a real PHEV, and validated using our hardware-in-the-loop (HIL) platform.

The remainder of this paper is organised as follows. Related works are reviewed in Section II. Notation and the problem statement are given in Section III. The system architecture and the optimisation are discussed in Section IV, where we calculate the probability of each road segment of routes to formulate our optimisation problems. In particular, we note here that a similar approach, by using Markov chain model, can also be applied to calculate the probability as required. For completeness, we also present a high level description on the Markov model in our context in the Appendix section. Implementations of the proposed strategies in the Simulation of Urban MObility (SUMO) package and the HIL platform are presented in Section V. Simulation results from SUMO and the HIL platform are discussed in Section VI. The limitations of our method and future extensions are remarked in Section VII. Finally, a brief conclusion is presented in Section VIII.

II. Related Work

Following the work [6], related work was then developed in [7] and [8]. The key idea of such papers was to regulate the energy consumption of EVs on road traffic in a coordinated manner, using forecasts of available future energy, in order to balance demand and supply and facilitate the adoption of demand side management strategies. Specifically, by considering the knowledge of both vehicles and energy available at the next charging period, vehicles could control energy consumption along their routes, in a manner that some desired performances indexes of all vehicles could be optimised. These initial ideas are also explored for a network of vehicles in [9], [10], and [13]. In addition, Guardiola *et al.* [11] have investigated the optimal control strategies to continuously adapt an engine calibration of an ICE vehicle for minimising the fuel consumption while keeping a limited amount of the pollutant emissions over an unknown driving cycle. The optimisation problem was formulated by taking into account the driving behaviours of a given driver and the varying pollutant limits depending on the location of the vehicle and other boundary conditions. The uncertainty of the driving cycle was estimated by using stored historical engine speed and torque demands of the vehicle in a probability manner. Our work is also related to conventional management strategies for hybrid electric vehicles. There are a number of interesting surveys and papers on this mature topic, see [1]–[5]. Most of this work focuses on minimising fuel consumption.

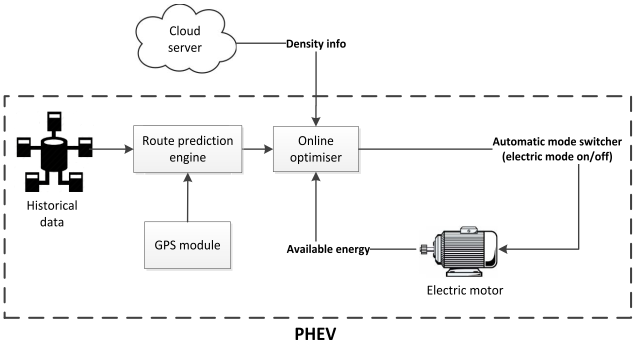


Fig. 1. A schematic diagram of the system architecture.

III. Problem Statement

Our objective in this paper is to develop an engine management strategy that takes into account the density of pedestrians along a particular route. To do this, we use a relatively simple strategy that does not take into account atmospheric dispersion models for pollutants, nor does it take into account weather or topology information from urban environments. Rather, we take the view that when a vehicle is in polluting mode (hybrid), the probability that it is causing harmful damage is proportional to the population density in that given location. Thus, our strategy is to manage the switching of the engine in order to minimise the impact of pollution on pedestrians; namely, to recommend vehicles to drive in EV mode in high density areas.

In what follows we will make use of the probability on each segment of routes to formulate the uncertainty that is central to our optimisation formulation; namely, driver intention. For this purpose, we define a route $\mathcal{R}$, or a journey, as a sequence of road segments r_i from an origin to a destination point, i.e., $\mathcal{R} = \{r_1, r_2, \dots, r_N\}$, where N is the number of road segments of a specific route. We then define a road segment as the part of a road that connects two consecutive junctions or, if this is too long (e.g., with a density of people that greatly varies along such a part of the road), we may assume that one same road segment could not exceed some fixed length (e.g., 500 meters). Other definitions of road segments may be however used as well, if more convenient (e.g., fixed length). Note that according to the previous definition, a road segment could belong to more than one route. Also, the same origin and destination point may be connected through different routes, if different sequences of segments can be taken to get to the (same) destination.

IV. System Architecture

Our proposed system architecture is shown in Fig. 1. The system requires inputs from four functional blocks: 1. historical data of the vehicle (history of routes travelled, energy consumption along these routes) 2. a route prediction engine that tries to anticipate the driver intention; 3. an online optimiser that controls the engine management; 4. and a cloud server that informs the vehicle of likely pedestrians density. Based on this architecture, the pedestrian-aware engine management strategies operate as follows.

A. We assume that a PHEV has an available energy budget for a particular journey. In practice, the energy budget can be estimated from historical energy consumption patterns and the vehicle's current state of charge (SOC).
B. We use the route predictor to obtain the probability distribution over all routes likely to be travelled by the driver based on the historical travelling patterns of the driver and the real-time GPS location of the vehicle (for online optimisation).
C. We require that there exists a central agent (e.g., a cloud server) that has access to real-time population density info in a given area where the vehicles may travel.
D. We use this real-time density info as an external input for the PHEV that will subsequently optimise the sequence of engine switching.
E. The switching to different modes is operated by using a dedicated-device on board that is controlled by the vehicle without input from the driver.
F. The mode switching of the PHEV is determined after every road segment in an online optimisation manner. In particular, at the end of a road segment the route predictor may want to update its predictions and rule out some possible routes; the remaining available energy budget of the vehicle may be recalculated. Also, a new estimate of the time-varying number of pedestrians in the different road segments may become available.

In the following we outline the details of such functional blocks.

A. Route Prediction Engine

The objective of the route prediction engine is to predict the unknown route of the driver, and, more specifically, also to predict the probability of taking every possible road segment. In particular, consider a generic road segment r_i where the vehicle is currently along with, and assume that this segment belongs to a number of routes, say $\mathcal{R}_1, \ldots, \mathcal{R}_k$. Moreover, we denote by N_t the total number of times that route $\mathcal{R}_t$ was taken in the past. Then, a probability, say $p(\mathcal{R}_t)$ is computed for each of these routes as follows:

$$p(\mathcal{R}_t) = \frac{N_t}{\sum_{j=1}^{k} N_j}, \quad \forall t \in \{1, 2, \ldots, k\}. \tag{1}$$

In particular, if a route does not pass by r_i, then the probability that is on the actual route is considered zero. Also, if only one route passes by r_i then its probability is considered 1. Then, the probability p_s^i of taking segment r_s in the future is simply computed as the sum of the probabilities of all routes that contain that road segment. In the notation p_s^i, the superscript i reminds that the probability of a route, and consequently, of a road segment within that route, depends on the specific road segment r_i where the vehicle is travelling at the time when a new prediction of the route is performed.

Remark: In our model, we implicitly assume that a car has the ability to store routes travelled in the past, and the number of times each route was taken. For instance, we could realistically assume that each vehicle can store up to 100 routes. After this number, any new route replaces an older rare route (e.g., one route that was taken only once). Thus, storing route information in the manner described here can be challenging in terms of data storage. A compact method of storing driver route choices is to embed driver's intention in a Markov chain. For simplicity, we do not describe the Markov chain approach in detail here, but some details and useful references are given in the Appendix.

B. Optimisation

In this section we formulate our online energy management optimisation problems. For this purpose, we add the following notation: *(i)* we denote by $\mathcal{S}$ the set of all road segments appearing in all the past routes; *(ii)* we denote by $\overline{e}_s$ the expected energy consumption along the s'th segment of $\mathcal{S}$, when travelling in pure electric mode (again, this can estimated from the historical data); *(iii)* we denote by d_s^i the expected number of people along the s'th road segment of $\mathcal{S}$ (as mentioned this can be computed using twitter feeds, mobile phone data, or can be simply estimated again from some historical data). The superscript i refers to the fact that we are using the information that is available when the vehicle is along road segment r_i; *(iv)* we denote by E_{av}^i an available energy budget when the vehicle is located at the i'th segment. For instance, this could simply be the energy left in the battery when the vehicle is along road segment r_i; finally *(v)* we denote by $x_s^i \in [0,\ 1]$ the set of decision variables that we wish to optimally compute. In particular, x_s^i is the percentage of time that the vehicle is in full electric mode on each segment $r_s \in \mathcal{S}$. The superscript i here reminds again that such an optimal prediction is performed when the vehicle is driving on road segment r_i, and a new prediction will be performed when the vehicle enters a new road segment, when a new (possibly more accurate) prediction of the route will be performed.

Problem 1: Mathematically, the first optimisation problem that we are interested in is:

$$\begin{cases} \max\limits_{x_s^i} \ \sum_{s \in S} p_s^i d_s^i \overline{e}_s x_s^i \\ \text{s.t. } \sum_{s \in S} p_s^i \overline{e}_s x_s^i \leq E_{av}^i \\ \\ \quad 0 \leq x_s^i \leq 1, \end{cases} \tag{2}$$

where $S = \{i \in \mathbb{N} | r_i \in \mathcal{S}\}$ (i.e., the constraint involves all the road segments). Roughly speaking, Problem 1 aims at making the vehicle travel in electric mode more likely when there are more pedestrians. In particular, this is done in a probabilistic manner, due to the uncertainty of the route, by giving more importance to the most likely routes.

Remark: Problem 1 is solved iteratively every time the vehicle enters a new road segment; in fact, when the vehicle enters a new road segment, it is possible to update the prediction of the route, to possibly update the remaining energy in the battery and the number of pedestrians as well.

Problem 2: While Problem 1 refers to an average optimisation (averaged over all possible routes with their corresponding probabilities), it may be also of interest to optimise the variables x_s^i with respect to the most energy-consuming route. In this case, the optimisation problem is solved with respect to the worst-case scenario (i.e., energy will be instantaneously allocated taking into account that the actual route might be the most energy demanding one). In this case, the mathematical problem may be formulated as:

$$\sum_{s\in R} \overline{e}_s x_s^i \le E_{av}^i, \quad \text{for all routes,} \tag{3}$$

where $R = \{i \in \mathbb{N} | r_i \in \mathcal{R}\}$ (i.e., the constraint involves all the road segments belonging to a given route, and must then hold for all routes). This formulation guarantees that the available battery level is never exceeded, irrespective of the driver choices. As we shall see later, such a worst-case scenario usually gives rise to a more conservative energy consumption pattern.

Example: We now provide a simple example to clarify our approach. In particular, we assume that one vehicle starts from the beginning of road segment r_1 that in the historical database appears in three different routes, as below:

- Route 1: $r_1 \to r_2 \to r_3$, this route was taken 100 times;
- Route 2: $r_1 \to r_2 \to r_4$, this route was taken 200 times;
- Route 3: $r_1 \to r_5 \to r_4$, this route was taken 400 times.

Then $\mathcal{S} = \{r_1, r_2, r_3, r_4, r_5\}$. Segments r_3 and r_5 appear only in route 1 and 3 respectively, so their probabilities p_3^1 and p_5^1 correspond to the probabilities of the corresponding routes (i.e., 1/7 and 4/7 respectively). On the other hand, probability p_2^1 is 3/7 (as both routes 1 and 2 lead to road segment r_2) and the probability p_4^1 is 6/7 (as both routes 2 and 3 lead to road segment r_4). In this case, we have that the objective function of the optimisation problem (first equation of (2)) is

$$d_1^1\overline{e}_1x_1^1 + \frac{3}{7}d_2^1\overline{e}_2x_2^1 + \frac{1}{7}d_3^1\overline{e}_3x_3^1 + \frac{6}{7}d_4^1\overline{e}_4x_4^1 + \frac{4}{7}d_5^1\overline{e}_5x_5^1, \tag{4}$$

where the objective is to compute the optimal values of x's, given the knowledge of the number of pedestrians and the expected energy consumption along each road segment. Also, the energy constraint is then

$$\overline{e}_1x_1^1 + \frac{3}{7}\overline{e}_2x_2^1 + \frac{1}{7}\overline{e}_3x_3^1 + \frac{6}{7}\overline{e}_4x_4^1 + \frac{4}{7}\overline{e}_5x_5^1 \le E_{av}^1, \tag{5}$$

if we wish to solve Problem 1, and

$$\begin{cases} \overline{e}_1x_1^1 + \overline{e}_2x_2^1 + \overline{e}_3x_3^1 \le E_{av}^1 \\ \overline{e}_1x_1^1 + \overline{e}_2x_2^1 + \overline{e}_4x_4^1 \le E_{av}^1 \\ \overline{e}_1x_1^1 + \overline{e}_5x_5^1 + \overline{e}_4x_4^1 \le E_{av}^1, \end{cases} \tag{6}$$

if we address the worst-case scenario of Problem 2.

Remark (Networked Vehicles): Here we briefly remark that, our current framework can be easily extended in a variety of ways to incorporate the behaviours of networked vehicles for minimising the group pollutants to pedestrians. For instance, given a fleet of PHEVs having similar travelling route maps in a fixed city area, one may wish to adopt our proposed framework for all vehicles optimally switching their engine modes while considering the impact on traffic flow on all possible routes of all vehicles. In practice, this problem can be easily formulated for each PHEV in the form of (2) which gives rise to our Problem 3.

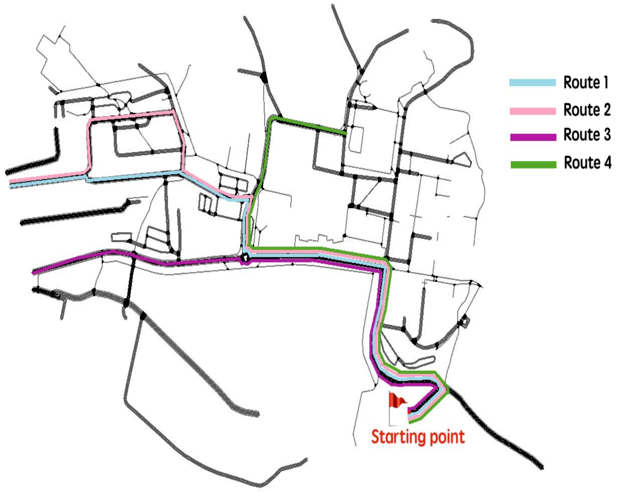


Fig. 2. Four real routes used in the field experiments of our test PHEV.

Problem 3: Formally speaking, the problem of the networked vehicles can be formulated in a distributed manner, namely each PHEV solves its own problem individually. Mathematically, we are interested in solving:

$$\begin{cases} \max\limits_{x_s^i} \ \sum_{s\in S} p_s^i d_s^i \overline{e}_s x_s^i f_s^i \\ \text{s.t.} \ \sum_{s\in S} p_s^i \overline{e}_s x_s^i \le E_{av}^i \\ \quad 0 \le x_s^i \le 1. \end{cases} \tag{7}$$

where f_s^i denotes the traffic flow on road segment s relative to all other road segments when the vehicle starts from the beginning of the i'th segment.

V. SUMO and HIL Simulations

In this section, we mainly introduce our simulation set-up in SUMO and present our algorithm implementations in the hardware-in-the-loop (HIL) platform embedded with a real car. First note that a complete description of the HIL platform is presented in [14]. In this work, we shall test our applications based on this platform with proper modifications and extensions for specific experimental purposes. Here we repeat some contents from [14] for readers' conveniences.

A. Simulation Set-Up in SUMO

In this section, we evaluate the performance of our algorithm in a realistic traffic scenario in the University College Dublin (UCD) campus, where the mobility of our test PHEV (Prius) is simulated in the popular traffic simulator software package SUMO [15]. The road network of the campus is imported from OpenStreetMap [16] and loaded in SUMO for further simulation and analysis. Specifically, we assumed that our test PHEV (Prius) had four route records in the UCD campus from its vehicle database. The road network and the corresponding route info is shown in Fig. 2. Note that we also assumed that all routes of the vehicle shared the same

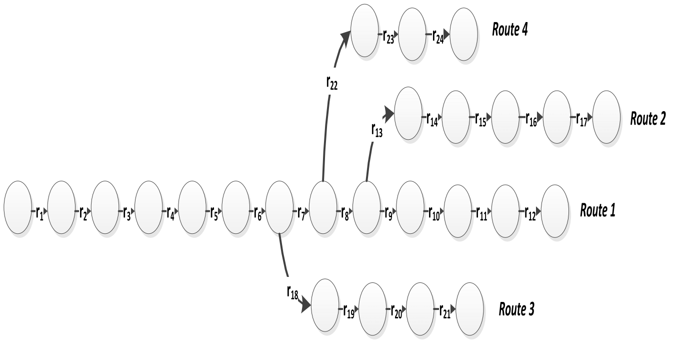


Fig. 3. A probability model including four real test routes in UCD campus. Note: 1. Each circle represents the beginning of one (or more) road segment(s), where the vehicle solves the online optimisation problem; 2. Length of the arrow is for indicating purpose only and it does not represent the real length of a road segment.

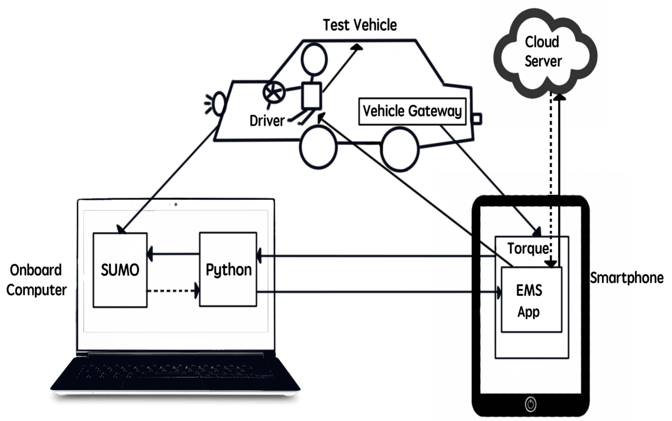


Fig. 4. A schematic digram of the real car implementation. Note that in this figure EMS is short for Energy Management System, and two dash lines represent two different ways that Python can obtain real time density info of pedestrians for online optimisation. In this work we simply adopt the interface from SUMO as an external input.

starting point but ended up with different exits (or junctions), which represent different destinations that the vehicle might drive out of the campus. It is worth noting that all routes have some road overlaps with each other; and this can be seen clearly from its probability model pictured in Fig. 3.

B. Real Car Implementation

In this section, we introduce our real car implementation using the HIL platform. A schematic diagram of our real car implementation is shown in Fig. 4. The system mainly consists of a test vehicle, an onboard computer, a smartphone, and a cloud server.

1) Test Vehicle: The test vehicle we used for our field experiments is a 2015 Toyota Prius VVTi 1.8 5DR CVT Plug-in Hybrid vehicle and is pictured in Fig. 5. One of the advantages of using the Prius as our field-test vehicle is because of its flexible engine management system, which allows us to operate the vehicle in fully electric mode and hybrid mode. It is this degree of freedom that allows us to explore the optimality of switching in response to pedestrians. To facilitate the automation between different mode switch (e.g., from fully electric mode to hybrid mode), we designed a dedicated mechanical interface (like a "Finger") which overrides the manual EV button in the vehicle. In our application, this interface is triggered by receiving Bluetooth control signals from a smartphone. The control signal is transmitted from phone to the "Finger", after every fixed driving distance of the Prius (e.g., 100 meters), based on the optimal algorithm output calculated from the onboard computer running with SUMO.

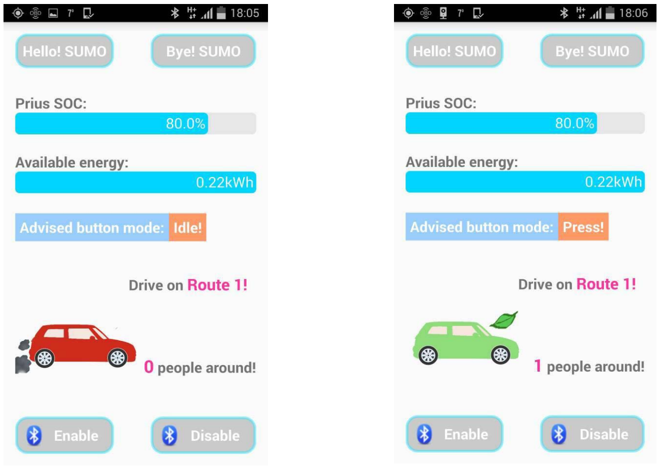
Fig. 5. Field-test vehicle: 2015 Toyota Prius.

Finally, we constructed a special-purpose hardware to permit communication between a smartphone and the controller area network (CAN) bus. The Prius provides a CAN access on the vehicle diagnosis *On Board Diagnosis II (OBDII) interface* and the OBDII interface device that we used is the Kiwi Bluetooth OBD-II Adaptor by PLX Devices.[2] Our hardware module acts as a gateway between this CAN interface and the smartphone. The module is directly connected to the CAN and to the smartphone via Bluetooth.

2) Onboard Computer: In our application, the optimisation algorithm is written in Python and is implemented on an onboard computer running with SUMO. SUMO is a mobility traffic simulator that we used to simulate the mobility of the test vehicle in our scenarios. We used SUMO to load the map imported from OpenStreetMap, and used its handy interface, namely TraCI (short for Traffic Control Interface) [17], to retrive online states info of the real vehicle from the smart phone via Python and control the behaviours of the vehicle in SUMO on the fly. Specifically, we used SUMO to model the behaviours of pedestrians walking in the campus as well, and based on this to provide population density info to Python as an external input. It is worth mentioning that a similar function can be achieved by using a cloud server, and we shall give brief comments for doing this in the Cloud Server section.

Based on all data collected, which includes real time data, such as Prius's GPS location, battery level, speed, distance and pedestrians data, as well as historical travelling data from the driver, Python implements the algorithm and determines the optimal duration of the vehicle that the electric engine mode should be engaged for pedestrian-aware driving.

[2] PLX Devices Inc., 440 Oakmead Parkway, Sunnyvale, CA 94085, USA. Phone: +1 (408) 7457591. Website: http://www.plxdevices.com



18:05

Prius SOC: 80.0%

Available energy: 0.22kWh

Advised button mode: Idle!

Drive on Route 1!

0 people around!

18:06

Prius SOC: 80.0%

Available energy: 0.22kWh

Advised button mode: Press!

Drive on Route 1!

1 people around!

Fig. 6. App GUI design for the field experiments. Figure on the left suggests the PHEV should be operating in hybrid mode and the figure on the right advises the PHEV should be switched to electric mode.

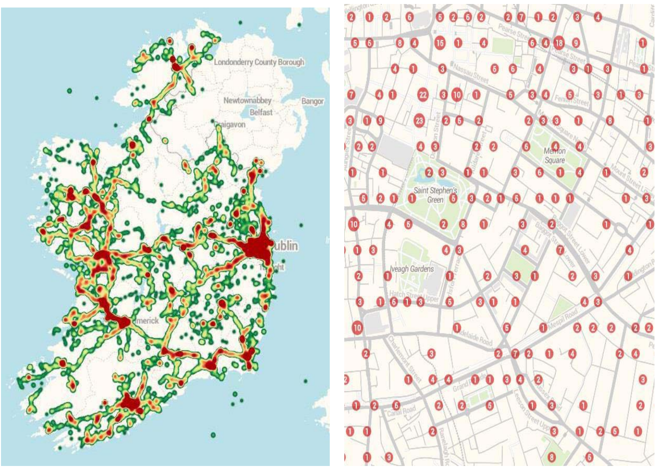


Fig. 7. Heat map of the density distribution of mobile phone users in Ireland (left) and the cluster map of the density in areas near Dublin city centre (right) on a typical working day, where the red colour in the heat map indicates higher density than the green one and the number in cluster map indicates the number of signal measurements received from the cell towers. Note: data is collected via OpenCellID.org API: http://opencellid.org/api on 5th of July, 2016.

3) Smartphone: In our architecture, the smartphone is required to collect data transmitted from the vehicle gateway (and from the cloud server if it is required) and forward such data to the onboard computer for online optimisation. From this viewpoint, the smartphone is only acting as a relay in our implementation system. In this work we developed a specific app on a Samsung Galaxy S IV (model no. GT-I9500) smartphone running the Android KitKat operating system (version 4.4.2) to demonstrate essential vehicle info to the driver. The GUI of the App we designed is shown in Fig. 6.

4) Cloud Server: A cloud server is only required to act as a central agent being capable of sending specific pedestrians info, via realisable communication channels, to smartphone whenever it receives request. In practice, a twitter feed[3] or any applicable third party service supplier (e.g., free mobile station data from OpenCellID)[4] can be directly used, or indirectly used, where exact pedestrians data is not available, on behalf of the cloud server to take actions in this step. As a concrete example, a real Ireland-based density distribution for mobile phone users (only those who installed the OpenCellID app) on a specific day is given in Fig. 7, where the density data across Ireland is filtered from the original data obtained from the OpenCellID database. Finally, we note that it is not required in our design that the cloud server should be capable of collecting all historical travelling data from Prius, and thus this not only reduces heavy computing burdens on the cloud side but also preserves privacy of the driver.

VI. Results and Discussions

In this section we discuss the simulation results obtained by implementing the proposed engine management strategies on a real car. To begin with, we use the proposed probability model in Fig. 3 and we define the maximum length of a road segment as 100 meters. To generate multiple historical energy consumption data of the Prius in full electric mode, we randomly distribute the energy on each road segment between 0 and 0.05 kWh according to the estimation from our real experimental data. As an example, Fig. 8 shows the real driving speed and the state of charge of the Prius in pure electric mode during one of our field experiments on route 2. Further, we assume a probability of 40% for the PHEV to travel on route 1, 30% for route 2, 20% for route 3, and 10% for route 4. Finally, we assume that the density of pedestrians along all routes is time-varying and is available on request.

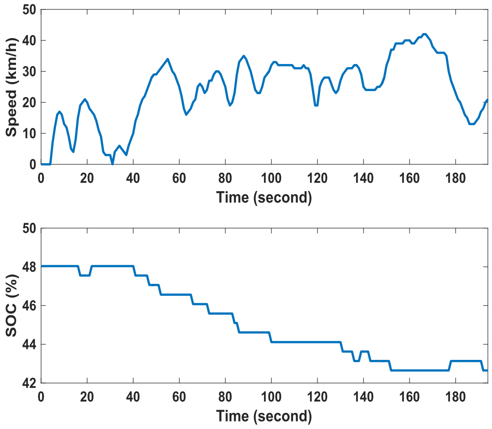


Fig. 8. The speed and SOC of Prius in pure electric mode in one of the field experiments on route 2.

A. Case Study: Problem 1

In this section we illustrate the simulation results obtained by solving optimisation Problem 1. Specifically, we compare different energy management schemes of Prius in three scenarios, while the PHEV was driving on route 1. We assume that the Prius started from the beginning location of segment r_1 in all scenarios and has the same initial energy budget 0.22 kWh which corresponds to 5% of its battery size (4.4 kWh) at the beginning of driving.

In the first scenario, the PHEV solves the optimisation Problem 1 at the beginning of every road segment along

[3] http://twitterfeed.com/

[4] http://www.opencellid.org

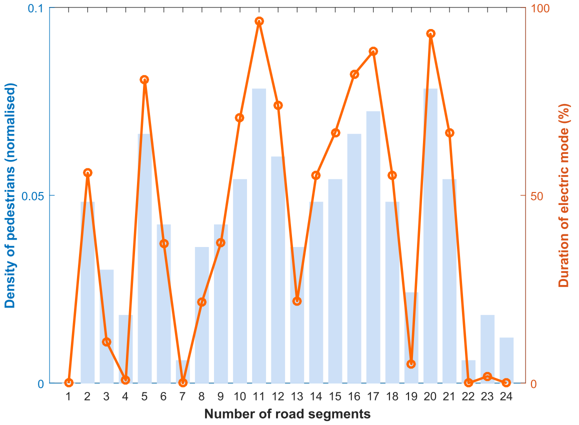


Fig. 9. Comparison of normalised density of pedestrians and the averaged duration of electric mode, with respect to 1000 different initial energy budgets, when the PHEV starts to drive at r_1.

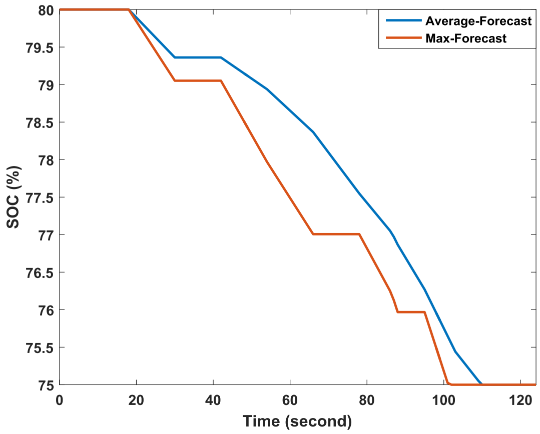


Fig. 10. State of charge of the PHEV in the first two scenarios.

route 1. In particular, the $\overline{e}_s$ is calculated using real data measured from the Prius averaged over multiple experiments.[5] In the second scenario, instead of averaging, $\overline{e}_s$ is taken to be the maximum value observed for a particular road segment over all experimental data. The third scenario, which is included as a benchmark, does not use any optimisation. Instead, the electric mode is engaged in proportion to the maximum observed $\overline{e}_s$ along a given route. For clarity, we abbreviate these scenarios as “Average-Forecast”, “Max-Forecast”, and “None-Opt” in the following.

Fig. 9 shows the relation between the normalised population density and the duration of electric mode in the first scenario when the PHEV starts to drive at r_1. The duration of electric mode at each road segment is averaged by solving optimisation Problem 1 in 1000 times by linearly spacing the available energy E^i_{av} between 0 kWh and 0.22 kWh (i.e., step size equals $0.22 \cdot 10^{-3}$). The results show that our approach preferentially allocates electric energy to those road segments with higher density of pedestrians. A comparison of results between “Average-Forecast” and “Max-Forecast” are shown in Fig. 10 and Fig. 11. Fig. 10 shows that the state of charge (SOC) of the PHEV decreases more sharply in “Max-Forecast” scenario than in the other scenario due to the fact that the largest energy consumption patten of the PHEV was adopted, while it also shows that the proposed method manages the energy consumption of the PHEV in exactly 0.22 kWh (5% of the battery size) along route 1 in both scenarios. As expected, Fig. 11 illustrates that the cumulative value of the objective function in “Average-Forecast” is indeed larger than “Max-Forecast”. Finally, Fig. 12 compares the value of cumulative clean air factor along route 1 in all scenarios, where the clean air factor is defined as $d^i_s x^i_s$ at each road segment s when the PHEV is in i. In there, we observe that a continuous improvement in air quality is achieved by a simple implementation of our proposed strategy, compared to the trivial equal energy allocation scheme (“None-Opt”).

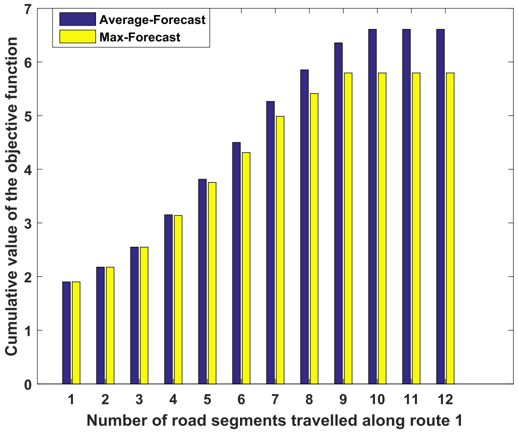


Fig. 11. Comparison of the cumulative value of the objective function in the first two scenarios.

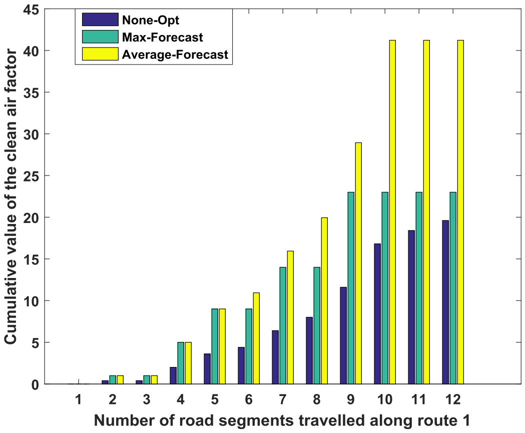


Fig. 12. Comparison of the cumulative value of the clean air factor in all scenarios.

B. Case Study: Problem 2

In this section we illustrate the simulation results obtained by solving optimisation Problem 2. In particular, we compare our results to the first method of solving Problem 1. In this study, we assume that the Prius has 0.1 kWh energy budget (2.5 % of battery size) before driving and we evaluate the overall clean air factors as if the Prius was driving on 4 different routes starting from the beginning location of r_1. Our results are shown in Fig. 13 and Fig. 14. Fig. 13 shows that the

[5] It can also be calculated from a vehicle model and data corresponding to congestion information at different times of the day.

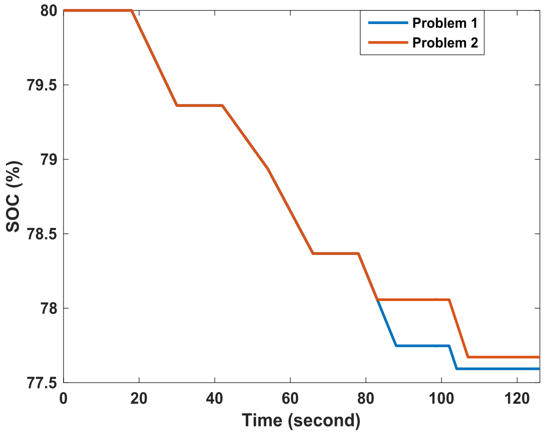


Fig. 13. Comparison of the SOC of the PHEV on Route 3 using expected (Problem 1) and robust (Problem 2) energy constraints.

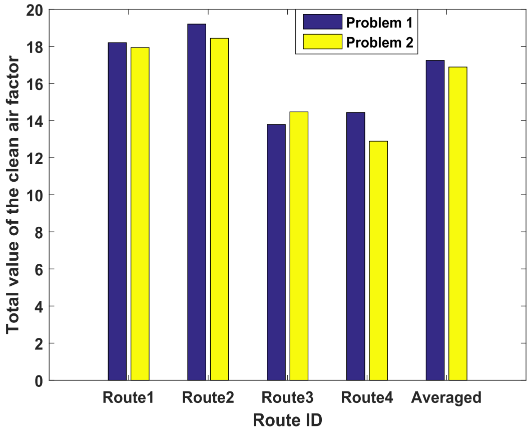


Fig. 14. Comparison of the total clean air factor using expected (Problem 1) and robust (Problem 2) energy constraints on different routes.

energy consumption of the PHEV is indeed more conservative (after 80 seconds) compared to using Problem 1 on Route 3, which is exactly the most energy-inefficient route in our study. Most importantly, from the last two bars in Fig. 14 we can see that, on average, the overall air quality evaluated using the expected energy constraint (i.e., the first constraint in (2)) is better compared to when robust energy constraints are used (i.e., equation (3)). However, as shown on route 3 in Fig. 14, for the route where the PHEV achieves the worst air quality, the robust energy constraints lead to better air quality.

C. Case Study: Problem 3

In this section we investigate the impact of pollutants on pedestrians in different scenarios with networked vehicles. To illustrate the benefits of deploying our extended engine management strategies, namely taking account of both pedestrians data and the traffic flow factor on different roads segments, we consider a symmetric "Y" style traffic network with three road segments pictured in Fig. 15.

In this network, we assume that the vehicles can either start from the beginning of the road segment r_1 or r_2 but both terminate at the end of the road segment r_3, as what is shown in Fig. 15. We also assume that the average energy consumption on all road segments is the same for all vehicles, which equals 0.025 kWh, and every PHEV has a same initial energy budget equal to 0.01 kWh at its own starting point. Most importantly, we assume that the number of pedestrians on each road segment is 50, and the traffic flows on r_1, r_2 and r_3 are assumed to consist of 20, 20 and 40 vehicles at the steady state. In this context, we perform our experimental activities in the following three scenarios:

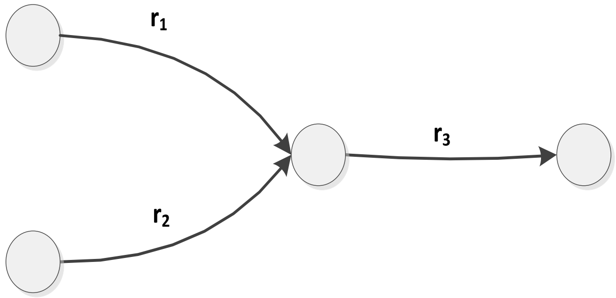


Fig. 15. A schematic diagram of the "Y" style traffic network with three road segments.

1) all vehicles solve Problem 1 without considering the traffic flow impact on all road segments;
2) all vehicles solve Problem 3 considering the traffic flow factor on all road segments;
3) based on 2), a feedback signal is further added to restrict the maximum number of pollutant units on a particular road segment.

In particular, we call the third scenario "limited pollutants" and we assume that the specific road segment in our example is r_3 with a maximum number of 800 pollutant units allowed. The comparison results are shown in Fig. 16, where we easily observe from the first two scenarios that by simply extending our algorithm by including a traffic flow factor, the number of pollutant units on road segment r_3 decreases significantly from 1600 units to 1200 units (25% less), which tremendously reduces the impact on pedestrians on r_3 due to the heavy traffic. Furthermore, by including a feedback signal on r_3 in the "limited pollutants" scenario, the number of pollutant units further decreases to 800 units, which satisfies all constraints and yields an enhanced performance for our proposed engine management strategy.

Finally we briefly note that our current framework can also be easily extended to carry out more restricted environmental enforcements ideas in reality, such as the "green zone" scenario [8], in which no pollutant is allowed in a given zone. In this circumstance, the driver can simply reserve the amount of energy required for driving on those segments, and implement the same proposed algorithm to optimally allocate the remaining available energy. To demonstrate this idea, we consider a "green zone" scenario in the same network as in Fig. 15. The number of pedestrians and vehicles are assumed to be the same as above. However, we assume that each vehicle has a 0.04 kWh energy budget to be consumed, and specifically, in the "green zone" scenario we explicitly require that there should be no pollutants on road segment r_3. After the implementation of our algorithm, we obtain that the number of pollutant units on r_1, r_2, and r_3 equals 400, 400, 0,

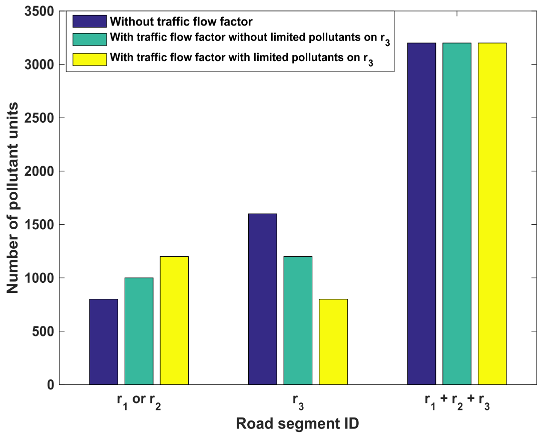


Fig. 16. A comparison diagram for the number of pollutant units with and without considering traffic flow factor.

respectively. With a total amount of 800 pollutant units on all road segments, our strategy is very promising to be deployed in practice.

VII. Limitations and Extensions

The work presented in this paper is an important step towards "pedestrian-aware engine management". However, it is only a first step and neglects some aspects of a complete solution, and these will be the subject of future work. For instance, the proposed system does not model the evolution of pollution and realistic dispersion models, as well as the impact of topology. Finally, we used the probability model to account for uncertainty in driver intention. The other uncertainty in the context of electric vehicles is the energy needs of the drivers, and models to capture this uncertainty may be investigated in future work.

VIII. Conclusion

In this paper, we have presented a novel engine management system for PHEVs. This management system is designed to benefit those most harmed by automotive emissions; namely, pedestrians. This is in contrast to traditional algorithms which benefit the polluters; namely, vehicle owners. Also, the proposed approach paves the way to a dynamic and less conservative definition of city restricted areas. We have implemented our proposed strategies in a real test vehicle. Limited field tests have demonstrated the efficacy of the algorithm. The result presented in [11] can be used as a basis for extending the work presented in this paper to ICE vehicles.

Appendix
Markov Chain Models and Driver Intention

Markov chains provide a compact method of storing large volumes of data (assuming certain assumptions are satisfied). Formally speaking, a Markov chain model is a tuple $\langle S, \mathcal{A}, P \rangle$, with S being a finite set of states $(s_1, \ldots, s_n)$, $\mathcal{A}$ being a finite set of actions, and P being the transition function $P : S \times \mathcal{A} \times S \to \mathbb{R}$. In our context, a junction between road segments in a journey defines a node, or state, in a graph $\mathcal{G}$, and the road segment between two nodes defines a link in $\mathcal{G}$. Measured turning probabilities at each junction are used to construct the entries of P. This transition matrix essentially captures journeys from all possible origins to all possible destinations for this driver and can be used to predict driver intention. In particular, the objective of the route prediction engine is then to use this transition matrix P and the vehicle's current position to get all possible road segments to be travelled by the driver and their associated probability values. In this paper we adopted a simpler and equivalent model to the Markov chain approach, to simplify the discussion of the model. However, we recognise that the Markov chain approach would be more efficient in storing details of historical routes of single vehicles. For more details regarding the use of Markov chains for traffic modelling in vehicular applications, see references [12], [18]–[21].

References

[1] S. Stockar, V. Marano, M. Canova, G. Rizzoni, and L. Guzzella, "Energy-optimal control of plug-in hybrid electric vehicles for real-world driving cycles," *IEEE Trans. Veh. Technol.*, vol. 60, no. 7, pp. 2949–2962, Sep. 2011.

[2] S. Barsali, C. Miulli, and A. Possenti, "A control strategy to minimize fuel consumption of series hybrid electric vehicles," *IEEE Trans. Energy Convers.*, vol. 19, no. 1, pp. 187–195, Mar. 2004.

[3] A. Sciarretta, M. Back, and L. Guzzella, "Optimal control of parallel hybrid electric vehicles," *IEEE Trans. Control Syst. Technol.*, vol. 12, no. 3, pp. 352–363, May 2004.

[4] K. Chau and Y. Wong, "Overview of power management in hybrid electric vehicles," *Energy Convers. Manage.*, vol. 43, no. 15, pp. 1953–1968, 2002.

[5] Z. Amjadi and S. S. Williamson, "Power-electronics-based solutions for plug-in hybrid electric vehicle energy storage and management systems," *IEEE Trans. Ind. Electron.*, vol. 57, no. 2, pp. 608–616, Feb. 2010.

[6] A. Schlote *et al.*, "Cooperative regulation and trading of emissions using plug-in hybrid vehicles," *IEEE Trans. Intell. Transp. Syst.*, vol. 14, no. 4, pp. 1572–1585, Dec. 2013.

[7] Y. Gu, F. Häusler, W. Griggs, E. Crisostomi, and R. Shorten, "Smart procurement of naturally generated energy (SPONGE) for PHEVs," *Int. J. Control*, vol. 89, no. 7, pp. 1467–1480, 2016.

[8] J. Naoum-Sawaya, E. Crisostomi, M. Liu, Y. Gu, and R. Shorten, "Smart procurement of naturally generated energy (SPONGE) for plug-in hybrid electric buses," *IEEE Trans. Autom. Sci. Eng.*, vol. 14, no. 2, pp. 598–607, Apr. 2017.

[9] F. Häusler, M. Faizrahnemoon, E. Crisostomi, A. Schlote, and R. Shorten, "A framework for real-time emissions trading in large scale vehicle fleets," in *Proc. 9th ITS Eur. Congr.*, Dublin, Ireland, 2013, pp. 1–13.

[10] F. Häusler, M. Faizrahnemoon, E. Crisostomi, A. Schlote, I. Radusch, and R. Shorten, "A framework for real-time emissions trading in large-scale vehicle fleets," *IET Intell. Transp. Syst.*, vol. 9, no. 3, pp. 275–284, 2014.

[11] C. Guardiola, B. Pla, P. Bares, and H. Waschl, "Adaptive calibration for reduced fuel consumption and emissions," *Proc. Inst. Mech. Eng. D, J. Automobile Eng.*, vol. 230, no. 14, pp. 2002–2014, 2016.

[12] S. Sinnott, R. H. Ordóñez-Hurtado, G. Russo, and R. Shorten, "On the design of a route parsing engine for connected vehicles with applications to congestion management systems," in *Proc. 19th Int. Conf. Intell. Transp. Syst. (ITSC)*, 2016, pp. 1586–1591.

[13] F. Knorn, M. J. Corless, and R. N. Shorten, "A result on implicit consensus with application to emissions control," in *Proc. 50th IEEE Conf. Decision Control, Eur. Control Conf. (CDC-ECC)*, Dec. 2011, pp. 1299–1304.

[14] W. M. Griggs, R. H. Ordóñez-Hurtado, E. Crisostomi, F. Häusler, K. Massow, and R. N. Shorten, "A large-scale SUMO-based emulation platform," *IEEE Trans. Intell. Transp. Syst.*, vol. 16, no. 6, pp. 3050–3059, Dec. 2015.

[15] *SUMO, Simulation of Urban Mobility*, accessed on Apr. 1, 2015. [Online]. Available: http://www.sumo-sim.org

[16] M. Haklay and P. Weber, "OpenStreetMap: User-generated street maps," *IEEE Pervas. Comput.*, vol. 7, no. 4, pp. 12–18, Oct. 2008.

[17] A. Wegener, M. Piórkowski, M. Raya, H. Hellbrück, S. Fischer, and J.-P. Hubaux, "TraCI: An interface for coupling road traffic and network simulators," in *Proc. 11th Commun. Netw. Simulation Symp.*, Ottawa, ON, Canada, 2008, pp. 155–163.

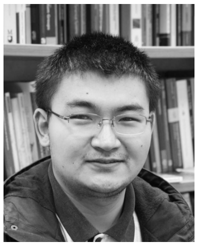
[18] J. Krumm, "A Markov model for driver turn prediction," Soc. Autom. Eng. (SAE), Detroit, MI, USA, Tech. Rep. 2008-01-0195, 2008.

[19] E. Horvitz and J. Krumm, "Some help on the way: Opportunistic routing under uncertainty," in *Proc. ACM Conf. Ubiquitous Comput.*, 2012, pp. 371–380.

[20] J. Krumm, "How people use their vehicles: Statistics from the 2009 national household travel survey," Soc. Autom. Eng. (SAE), Detroit, MI, USA, Tech. Rep. 2012-01-0489, 2012.

[21] J. Krumm, R. Gruen, and D. Delling, "From destination prediction to route prediction," *J. Location Based Services*, vol. 7, no. 2, pp. 98–120, 2013.

Yingqi Gu received the B.E. degree (Hons.) in electronic engineering from Maynooth University in 2013 and the M.Sc. degree in signal processing and communications from the School of Engineering, The University of Edinburgh, in 2014. She is currently working toward the Ph.D. degree with University College Dublin, under the supervision of Prof. R. Shorten. Her research interests are the modeling, simulation, and optimization of intelligent transportation systems.



Mingming Liu received the B.E. degree (Hons.) in electronic engineering from National University of Ireland Maynooth in 2011 and the Ph.D. degree from Hamilton Institute, Maynooth University, in 2015. He is currently a Post-Doctoral Research Fellow with University College Dublin, with Prof. R. Shorten. His research interests are nonlinear system dynamics, distributed control techniques, modeling and optimization in the context of smart grid, and smart transportation systems.

Joe Naoum-Sawaya received the B.E. degree in computer engineering from American University of Beirut, and the M.A.Sc. and Ph.D. degrees in operations research from University of Waterloo. He is an Assistant Professor in management science with Ivey Business School. His research interests include large-scale optimization methods for practical problems arising in the industry.

Emanuele Crisostomi received the B.S. degree in computer science engineering, the M.S. degree in automatic control, and the Ph.D. degree in automatics, robotics, and bioengineering, from University of Pisa, Italy, in 2002, 2005, and 2009, respectively. He is an Assistant Professor in electrical engineering with the Department of Energy, Systems, Territory and Constructions Engineering, University of Pisa. His research interests include the control and optimization of large-scale systems, with applications to smart grids and green mobility networks.

Giovanni Russo received the Ph.D. degree from the University of Naples Federico II in 2010. His work had a focus on stability of nonlinear dynamical systems with applications to networked control. From 2012 to 2015, he was the Lead System Engineer and the Integrator of the Honolulu Rail Transit Project, the first driverless mass transit railway in the Unites States. In 2015, he joined IBM Research, Ireland. His areas of interest include connected and autonomous cars, railways, road traffic control, Internet of Things applications, and the management of shared infrastructures. He is a member of the Board of Editors of IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS I for topics related to Control Theory, Networks and Networked Systems.

Robert Shorten is a Professor of Control Engineering and Decision Science with University College Dublin and holds a position with IBM Research. His research spans a number of areas. He has been active in computer networking, automotive research, collaborative mobility, including smart transportation and electric vehicles, and basic control theory and linear algebra. His main field of theoretical research has been the study of hybrid dynamical systems.