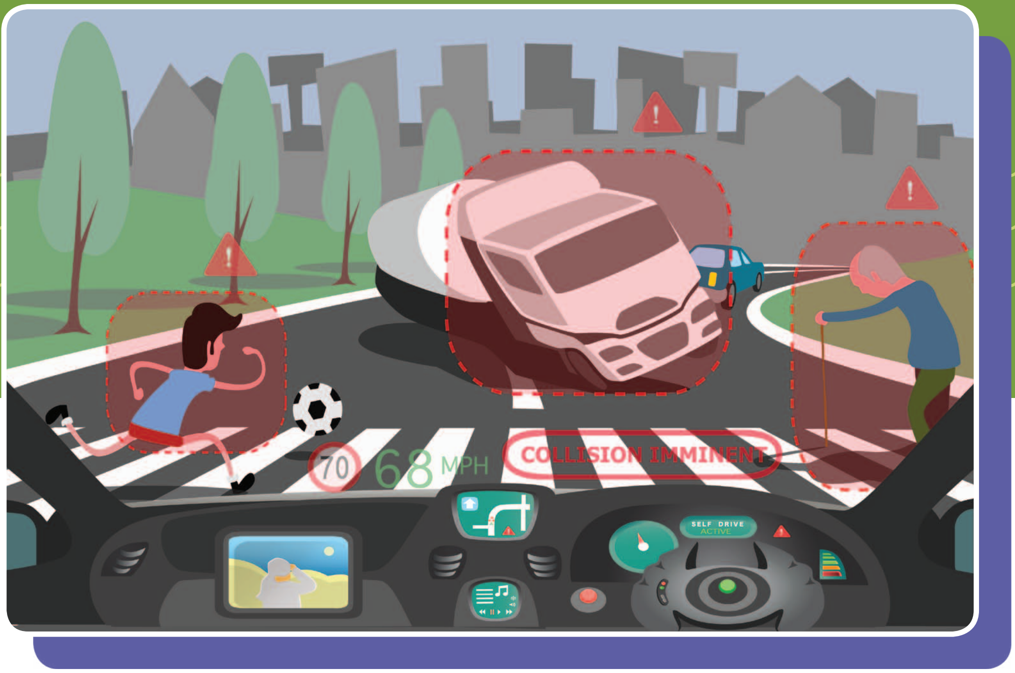
Cloud-Based Pedestrian Road-Safety with Situation-Adaptive Energy-Efficient Communication



©ISTOCKPHOTO.COM/A_CROTTY

Mehrdad Bagheri and Matti Siekkinen
Department of Computer Science - Aalto University - Espoo, Finland, Email: mehrdad.bagheri@aalto.fi, matti.siekkinen@aalto.fi

Jukka K. Nurminen
Department of Computer Science - Aalto University - Espoo, Finland; Systems Modeling and Simulation - VTT Technical Research Centre of Finland Email: jukka.k.nurminen@aalto.fi

Digital Object Identifier 10.1109/MITS.2016.2573338
Date of publication: 22 July 2016

***Abstract*—**Pedestrian detection using wireless communication complements sensor-based pedestrian detection in driverless and conventional cars. This fusion improves road-safety particularly in obstructed visibility and bad weather conditions. This paper seeks developing such wireless-based vehicle-to-pedestrian (V2P) collision avoidance using energy-efficient methods and non-dedicated existing technologies namely smartphones (widespread among pedestrians and drivers), cellular network and cloud. Our road-safety mobile app can be set to driver mode or pedestrian mode. This app frequently sends vehicle and pedestrian geolocation data (beacons) to cloud servers. Cloud performs threat analysis and sends alerts to road users who are in risky situation. However, constant pedestrian-to-cloud (P2C) beaconing can quickly drain smartphone battery and make the system impractical. We employ adaptive multi-mode (AMM) approach built on situation-adaptive beaconing. AMM reduces power consumption using beacon rate

1939-1390/16©2016IEEE

control while it keeps the data freshness required for timely vehicle-to-pedestrian collision prediction. AMM runs on cloud servers and commands the mobile apps to change P2C beaconing frequency according to collision risk level from the surrounding vehicular traffic. City-scale mobility simulation demonstrates energy efficiency of our approach. We evaluate battery lifetime according to geolocational variations over the city map. Results show that road-safety system imposes a small mean overhead on smartphone battery's state-of-charge. Furthermore, our evaluation of computation and network load shows feasibility of running such road-safety systems on conventional cellular networks and cloud providers. We use server-side prototype experiment to estimate minimum cloud resources and cloud service costs needed to handle computation of city-scale geolocation data.

I. Introduction

Autonomous driving and advanced driver assistance systems (ADAS) have generated much interest as the future trend of mobility. In this regard, automated and assisted collision avoidance is a crucial requirement to ensure safety of road users including drivers, cyclists and pedestrians. Special attention should be paid to safety of pedestrians since they are among the most vulnerable road users involved in fatal accidents [1], [2].

Driverless and conventional cars can use advanced sensors to detect obstacles and avoid vehicle-to-pedestrian (V2P) collisions [2]–[4]. Sensors mounted on such vehicles include cameras (i.e. imaging sensors with computer vision), RADARs, LASER scanners, Infrared (IR) or Photonic Mixer Device (PMD) [2], [3]. However, they still have shortcomings such as: Problem in detecting a person obstructed by cars, trees or building corners (non-line-of-sight (NLoS)), and a decreased precision of camera-based image recognition at night or in rain, snow and fog. They also have difficulty in seeing or sensing the other side of a hilly or curvy road. For example, the car is unable to detect the risk in time when a pedestrian steps on the road from behind of a parked vehicle or in an intersection where line-of-sight is blocked by a building. In these scenarios, it is also more difficult for drivers to notice the pedestrian in time and avoid the accident. Although sensor fusion can tackle this to some extent, the NLoS scenarios will not be resolved [2], [3], [5]. Deploying sensors on road-side to cover obstructed spots is an expensive approach [2] and only practical in limited scenarios (e.g., Cooperative ITS Corridor [6]).

In such obstructed visibility and bad weather conditions, we can employ vehicle-to-pedestrian wireless communication to exchange messages between road users, conveying their geolocation, speed and direction retrieved from GPS or other sensors. This approach can complement sensor-based methods, increase reliability of pedestrian detection and collision prediction and improve road-safety. Wireless communication (e.g., WiFi, 3G) is able to tackle NLoS problem by detecting road users even when obstructed by other physical objects. For instance, if a pedestrian is approaching the road in rural areas at night or from behind parked cars, it is possible to anticipate a potential accident and warn both driver and pedestrian. In addition, distinguishing a pedestrian from a car or roadside object (a challenge for sensor-based methods) will become trivial, as a pedestrian is any person who holds a communication device with a designated app. V2P message exchange will also increase driverless car's awareness of its surrounding environment. For V2P applications, communication on the pedestrian side can be established by means of a smartphone.

Wireless communication is also used in vehicular networking for vehicle-to-vehicle (V2V) collision avoidance but little research has addressed integrating pedestrians into vehicular networks to develop pedestrian road-safety applications. The IEEE 802.11p of Dedicated Short Range Communications (DSRC) [7] is specifically standardized for V2V ad-hoc communication [8], [9]. However, at the moment it is not feasible to employ DSRC for V2P communication. It is expensive and impractical to equip pedestrians with DSRC modules. DSRC hardware is not available on commercial smartphones yet and the effort to adapt it for smartphones has been only recently started [10]. Moreover, DSRC is power-hungry and therefore it is a challenge to use it on smartphones [11]. Besides, realizing vehicular networks faces challenges such as costs of producing and deploying DSRC hardware, upgrading road infrastructure (e.g., installing road-side units), and the anticipated long market penetration and user adoption period [7]–[9].

One promising approach to enable earlier adoption and development of wireless-based V2P collision avoidance is to utilize existing communication methods and devices available to all road users. Therefore, a prospective V2P networking solution should not require large-scale installation of new hardware (e.g., should not be dependent on the road-side units). This approach can be a cost-effective temporary alternative for autonomous vehicles that are not equipped with high-end sensors or dedicated communication hardware. In this regard, works such as [5], [11]–[14] use existing infrastructure and devices for pedestrian road-safety. They employ mainstream smartphones together with either WiFi ad-hoc network or cellular-based Internet or a combination of both. Experiments and evaluations [15], [16] show that utilizing WiFi is not practical in all collision prevention scenarios. Unlike IEEE 802.11p, WiFi suffers from limited communication range (100 m) and weak mobility support (e.g., sensitivity to Doppler effect due to its 20 MHZ channel width). On the other hand, although cellular technologies (e.g., 3G and LTE) are not designed for vehicular networks, evaluations [5], [16] and experiments [12], [14] show their potential for this purpose. This is because of their high mobility support, and high bit-rate, communication range and communication capacity. Some vehicles are already equipped with cellular connectivity (on-board SIM card) but otherwise the driver's smartphone can be used as an alternative.

Having above discussion, our pedestrian road-safety system utilizes smartphones (with a designated mobile app) together with cellular technologies as the only communication method. This approach naturally leads to a cloud-based system [17] (centralized, not ad-hoc) as device-to-device (D2D) communication such as LTE Direct is not yet available in commercial cellular networks. Both pedestrian smartphone and vehicle need to periodically send update messages (also called *beacons* or *probes*) to cloud-servers. Beacons mainly contain *geolocation*, *speed* and *direction* of road users. Server-side performs threat analysis based on the received probes. Collision warnings are sent from cloud to road-users when risky situation is detected.

From a financial point of view, we argue that consumers will be motivated enough to use the system in spite of its extra data usage over their mobile subscription (such as 3G, 4G, 5G). Firstly, our simulation results (presented later in this paper) show that bandwidth (BW) overhead of the road-safety service is not very large per user in realistic conditions. Mean estimate is only 71 MB of extra data per month for each user. This is acceptable even for cheap subscriptions considering the added benefit of road-safety. Secondly, [18] analyzes financial aspects of such mobile-based road-safety systems in more detail. The mobile app can still be provided for free while the service offers an important value proposition of increased safety for pedestrians. Another motivating value proposition can be the decreased annual insurance cost for pedestrians and drivers. Finally, cars are likely to get connected to high BW cellular Internet in near future to enable other services (e.g., real-time traffic information, infotainment) that users are more willing to pay for.

Smartphones suffer from limited battery capacity and their short battery lifetime can be a bottleneck in realization of pedestrian road-safety system. To our best knowledge, no previous work has analyzed energy-efficiency of cellular-based road-safety systems on smartphones. Due to the power consumed for wireless communication, periodic beaconing quickly drains smartphone battery even when sending small amounts of data. Innovative battery technologies [19] could potentially solve this problem but still have a lot of uncertainty and need several years before reaching the market. As an example, our earlier work [20] shows that periodic beaconing completely discharges smartphone battery after around 5 hrs while the same smartphone normally would have around 16 hrs of battery lifetime. This paper further investigates this problem by evaluating the overhead of road-safety system on power consumption of smartphone in realistic vehicular traffic scenarios. We propose an approach based on situation-adaptive beaconing to tackle battery lifetime limitation. Adaptive beaconing schemes have been proposed [21], [22] to optimize beaconing according to wireless channel capacity. However, this is different from objectives of our work. In this paper, we study the impact of adaptive rate control on energy-efficiency of communication in the context of vehicular networking safety applications.

This paper makes the following contributions:

- We present a mobile-based pedestrian road-safety system. Our system is centralized and no ad-hoc D2D communication is made between road users. All road users utilize a mobile app to beacon their geolocation information to cloud. Cloud predicts V2P accidents and sends collision warnings to vehicles and pedestrians.
- Section III proposes adaptive multi-mode (AMM) approach to achieve a practical smartphone battery lifetime and thus enable running pedestrian road-safety systems. AMM is an implementation of situation-adaptive beaconing for cloud-based V2P collision avoidance. It switches mobile app's operation mode (between sleep and active) according to pedestrian's own movement and location context. AMM changes mobile app's beaconing frequency depending on surrounding vehicular traffic and more specifically based on collision risk level (a factor of V2P distance, relative V2P speed and direction). The goal is to reduce communication power consumption but maintain acceptable freshness of position data for timely collision prediction.
- Sections IV to VII demonstrate feasibility of running pedestrian road-safety system on mainstream smartphones and conventional cellular networks and cloud providers. Section V performs city-scale simulation using realistic vehicle trace data of Cologne (Köln) in order to evaluate the overhead imposed on smartphone battery lifetime in realistic situations. Simulation results explain how AMM communication and battery lifetime are influenced by pedestrian geolocation within the city and time of the day. We also evaluate advantage of AMM over non-adaptive approach in terms of energy-efficiency. Section VI evaluates the feasibility in terms of computation load and its associated cloud service costs. We present details on implementing a road-safety cloud system and explain how cloud servers can perform timely threat analysis over the received probe data. We implement a server-side collision prediction prototype to measure execution times and delays, and estimate minimum costs to meet the system compute demand. Section VII evaluates the road-safety system in terms of network load overhead.

II. Background and Related Work

Regarding communication in road-safety systems, European Telecommunications Standards Institute (ETSI) categorizes the exchanged messages into two main classes [23]: (a) Cooperative Awareness Message (CAM) that is time-triggered (periodic) and conveys information related to geolocation of vehicles. (b) Decentralized Environmental Notification Message (DENM) that is event-triggered and conveys hazard warnings. Example of a DENM message for pedestrians is traffic light violation warning on their smartphones. ETSI allows maximum latency of 100 ms for delivery of CAM and DENM messages as well as the time interval between CAM messages ranging from 100 ms to 1.0 s depending on the use case [24]. For this purpose, LTE has a fair enough performance with a maximum latency of 100 ms [16]. In addition, practical experiment [5] demonstrates feasibility of using earlier cellular technologies such as UMTS and HSPA.

Regarding computation, threat analysis and collision prevention can be performed in one of the following architectural arrangements [5] and the choice depends on which method provides a better performance and energy-efficiency: (a) all computation performed on smartphone, (b) all computation performed on vehicle on-board unit (OBU), (c) all computation performed on back-end servers. In this paper, we adapt and use the last method employing cloud-based servers.

As mentioned before, there are a few similar works which employ wireless communication and smartphones to develop pedestrian road-safety systems. Authors of [5] and [12] use smartphones for communication on pedestrian side. They investigate different architectural arrangements and employ a hybrid of both cellular connectivity and 802.11 (WiFi-based ad-hoc). In [5], authors address the non-structured mobility behavior of pedestrians by using filters to identify and ignore pedestrians' non-risky movement patterns. In [12], authors develop a V2P prototype and perform simulation. Cellular network (3G) is used to send probe data to a server which performs initial and predictive calculation. In case this calculation concludes a risk, vehicle and pedestrian are notified to start a direct WiFi-based V2P communication.

In [25], authors employ only WiFi as communication medium for V2P collision avoidance and implement a tablet-based prototype for pedestrian side. Authors of [11] implement a WiFi-based protocol and a smartphone app for vehicular communication scenarios. However, as mentioned before in section I, WiFi may not perform well in road-safety scenarios because of its limited communication range and high speed of network nodes. Authors of [10] integrate DSRC into mainstream smartphones without doing any hardware or chip upgrade. Smartphone is used only on pedestrian side while vehicles use dedicated hardware. However, [10] does not thoroughly investigate the feasibility of using the power-hungry DSRC [11] on smartphones. Using only cellular communication, authors of [14] run experiments to analyze the accuracy of GPS data provided by smartphones and to evaluate feasibility of using 3G Internet in terms of latency.

Unlike previous solutions, we evaluate power consumption overhead of the road-safety system and viability of smartphone battery lifetime in varying geolocational conditions in the city. We propose an energy-efficient approach in order to use the battery-limited smartphones as the main communication device. In addition, unlike [5] and [12], we use cellular network for communication in all cases and no switching is performed from cellular to 802.11p (or regular WiFi) when a risky situation is detected by the cloud-based server. Cellular-based D2D communication is not available yet and therefore a cloud-based system is studied in this paper. No broadcast is performed to neighboring vehicles, but instead all beacons are sent directly to cloud servers.

Adaptive beaconing mechanisms are classified in [21] and [26]. Based on these schemes, implementations of efficient adaptive beaconing have been proposed that are summarized and compared in [22]. These implementations adjust beaconing rate (affects data freshness) and/or transmit power (affects transmission range) in order to reach an optimal point of preventing communication channel overload and at the same time keeping acceptable tracking accuracy and freshness of position data depending on the scenario (e.g., for traffic management or road safety). In line with situation-adaptive beaconing schemes proposed in [21], works such as [26]–[28] present models to incorporate surrounding vehicular traffic situation into adaptation of beaconing. Surrounding traffic is assessed either in a macroscopic (e.g., traffic density in an area) or microscopic scope (e.g., proximity of vehicles). As an example, the goal in [26] is to achieve the smallest tracking error and packet loss values but at the same time prevent overloading the shared DSRC channel. It increases beaconing rate when position error passes certain threshold and decreases the rate when channel is being congested. Authors evaluate their method using microscopic simulation.

However, above approaches differ from objectives of our work. We study the impact of adaptive rate control on energy-efficiency of mobile-based communication and smartphone battery lifetime in vehicular safety applications. State-of-the-art beaconing approaches have used DSRC module on-board of vehicle, and therefore power consumption optimization made little sense since the module is attached to vehicle's electricity system. Unlike previous works, this paper seeks reducing power consumption of communication while maintaining required data freshness for timely vehicle-to-pedestrian collision prediction. This paper proposes an implementation of situation-adaptive beaconing [21] for cloud-based pedestrian road safety. We perform beacon rate adaptation (not transmit power adaptation) depending on the microscopic scope of surrounding vehicles' movement. Beacon rate is calculated for each individual pedestrian and not set globally. The trigger for modifying the rate is a change in pedestrian collision risk level. Because our focus is safety, we take into account real-time dynamics namely relative V2P velocity, heading and distance in order to evaluate collision risk. Details are explained in following sections.

III. Adaptive Multi-Mode (AMM) Pedestrian Road-safety System

As outlined before in section I, our road-safety system consists of server-side application running in cloud and client-side road-safety application (mobile app) running on road users' devices. Server-side performs all the data processing and computation (threat analysis and collision prediction) whereas the mobile app only sends periodic update beacons to the cloud (no D2D communication). Beacons are also known as Basic Safety Message (BSM) in DSRC [29] and mainly contain road user's up-to-date *geolocation, speed and heading* obtained from GPS, Galileo or other sensors. This basic information alone is enough for the server-side to anticipate possible accidents and accordingly alert the road users. Collision warnings are sent from cloud to road-users when risky situation is detected. Pedestrians use their smartphone to run the mobile app and connect to the cloud application (e.g. over 3G or LTE Internet) and vehicles utilize either a built-in module (with a dedicated SIM-card) or driver's smartphone.

When road users are involved in a risky situation that may lead to an accident, they should send BSMs with intervals short enough to keep the data freshness and make it possible for cloud to predict collisions in time. According to use cases defined by ETSI for BSMs in the CAM category [23], namely Collision Risk Warning and Intersection Collision Warning, beacons should be generated with a 100 ms interval (frequency of 10 Hz) [24]. However as explained in section I, such arrangement is not practical in terms of power consumption because constant beaconing can quickly drain smartphone battery [20]. In order to address this problem, this paper presents an approach based on situation-adaptive beaconing as outlined above in section II. Our adaptive multi-mode approach (AMM) saves electrical energy by reducing unnecessary beaconing according to surrounding situation. The goal is to reduce communication power consumption but at the same time maintain freshness of position data required for timely collision prediction. This section also evaluates communication delays and timing analysis considering the role of cloud in our system. Table 1 summarizes parameters and values used in this paper.

A. AMM System

In realistic scenarios, pedestrians are often in risk-free or low-risk situations for example when they are stationary (e.g., not walking, spending time at home or office) or when they are walking but far enough from vehicles or roads. In such situations a constant full-rate beaconing is not required. Therefore, AMM adjusts the *mode* and *frequency* of pedestrian-to-cloud (P2C) communication according to pedestrian's own movement, location context and vehicle-to- pedestrian collision risk level. In brief, AMM evaluates collision risk per each individual pedestrian and depending on the microscopic scope of surrounding vehicular traffic. In other words, collision risk is a factor of V2P distance, relative V2P speed and heading. AMM performs beacon rate control considering real-time V2P kinematics relations and the most threatening nearby vehicle. In addition, AMM switches to sleep mode (beaconing turned off) when the user is stationary and far enough from roads or vehicles. More details are explained below.

AMM considers three collision risk levels for pedestrian: risk-free, low-risk and high-risk. A risk-free situation is a scenario with no probability of leading to a collision. It is when the smartphone user is staying indoors (e.g., at home or office) or far enough from streets, and therefore not considered a pedestrian for the time being. Such risk-free situations are detected by an event based context-aware algorithm running client-side on smartphone itself. Such context-aware algorithm can be implemented using methods explained in [30]–[32], and detects whether the person is stationary or moving. It utilizes smartphone's built-in motion sensors (e.g., accelerometer) and consumes negligible amount of battery capacity. A high-risk situation is a scenario that can potentially lead to a collision. It is when pedestrian is mobile (e.g., walking) and at least one vehicle is within proximity of $w_{\max}$ (high-risk range window) while vehicle's movement direction (its speed vector) indicates it is getting closer to the pedestrian. Example of a high-risk situation is when a pedestrian is walking towards the street in order to cross it while a vehicle only 200 m away is approaching towards the pedestrian at 50 km/h. A low-risk situation is when pedestrian is mobile but either no vehicle is closer than $w_{\max}$ or all vehicles within this proximity are moving away from pedestrian. In other words, road-safety system filters out vehicles which although within the high-risk range, either have passed by the pedestrian or are moving in a direction away from the pedestrian.

Client-side road-safety app running on pedestrian's smartphone works in three modes of operation: *sleep mode, low-rate mode* and *full-rate mode*, each corresponding to one of the risk-levels explained above. When the system recognizes that pedestrian is in a risk-free situation, it switches the P2C communication to *sleep mode*. P2C communication is turned off and no beacon message is sent in this mode. As soon as the context-aware algorithm on smartphone detects person's mobility (e.g., person starts walking), it enables P2C communication (enables beaconing). Fig. 1 illustrates the communication and system workflow during pedestrian's mobility. During low-risk situations, pedestrian's smartphone works in the energy-saving *low-rate mode* in which beaconing to cloud is kept at a lower rate (f_{lowrate}).

On the other hand, cloud-based server-side application is running a predictive algorithm that performs threat analysis by pair-checking incoming beacons received from pedestrians and vehicles located in close proximity. When the server-side detects a high-risk situation for any of pedestrians, it sends a notification message (e.g, using push message) which commands the pedestrian smartphone to adapt accordingly and switch to *full-rate mode*, in which P2C beaconing is performed at full 10 Hz rate (f_{fullrate}). Similarly, when

Table 1. Parameters influencing the adaptive multi-mode and road-safety system.

Parameter	Definition	Value
W_{max}	High-risk radius where there is a potential for a V2P or V2V collision.	200 m for V2P, 500 m for V2V (explained in text).
W_{prb}	Total distance traveled by car while collision prediction, reaction to collision alert, and braking is being performed.	A function of v: $1.38 \times v + v^2/13$
W_{extra}	Safe distance margin usually equivalent to ≈ 2 s of safe time margin.	$W_{max} - W_{prb}$
v	Maximum speed of vehicles surrounding the pedestrian.	Depends on traffic conditions and geolocation in city
$f_{fullrate}$	Frequency of beacon messaging in full-rate mode.	10 Hz according to ETSI [24]
$f_{lowrate}$	Frequency of beacon messaging in low-rate mode.	A function of v as illustrated in Fig. 3
$d_{process}$	Distance traveled by car while the system is performing communication and computation required to predict a collision.	$v \times t_{process}$
$d_{reaction}$	Distance travelled by car while the auto-brake system or the driver are reacting to the alert before they activate the brakes.	$v \times t_{reaction}$
d_{brake}	Distance travelled by car after brake activation until vehicle completely stops	$v^2 / (2_{abrake})$
$t_{process}$	Total time taken for collision prediction, comprising pedestrian-to-cloud (P2C) and vehicle-to-cloud (V2C) message transmission and threat analysis performed by cloud.	Equal to $t_{comm} + t_{comm}$
t_{comm}	Time spent sending the beacon message from smartphone to cloud server and the collision alert message from cloud to smartphone.	0.05 s (equal to LTE RTT)
t_{comp}	Time spent by cloud to perform collision prediction on all incoming probe records.	0.1 s to 0.5 s depending on system workload (deadline is 0.5 s).
$t_{reaction}$	Reaction time before car brakes is activated.	0.83 s [36]
a_{brake}	Vehicle deceleration during braking.	6.5 m/s^2
$t_{mobility}$	Duration of person's mobility (e.g., total 2 hrs during a day).	$t_{mobility} = t_{lowrate} + t_{fullrate}$ Differs from person to person depending on daily life.
$t_{lowrate}$	Duration of P2C beaconing in low-rate mode (pedestrian in low-risk situation).	Depends on traffic speed and density, and pedestrian's geolocation (urban area).
$t_{fullrate}$	Duration of P2C beaconing in full-rate mode (pedestrian in high-risk situation).	Depends on traffic speed and density, and pedestrian's geolocation (urban area).
$r_{fullrate}$	Ratio of P2C beaconing in full-rate mode in relation to the whole mobility period of pedestrian.	$r_{fullrate} = t_{fullrate}/t_{mobility}$ Depends on traffic speed and density, and geolocation (urban area).
C_{rem}	Smartphone battery SOC after pedestrian's mobility period.	Calculated according to (6) as explained in the text
T	Smartphone battery lifetime during a day.	Calculated according to (7) as explained in the text
C	Smartphone battery capacity (when fully charged).	7.98 Wh (Samsung Galaxy S3, 2100 mAh, 3.8 V)
P_0	Typical power consumption of smartphone (when road-safety app is off)	0.5 W (Battery lifetime around 16 hrs)
$P_{mobility}$	Additional power consumed by the road-safety app during person's mobility.	Calculated according to (8) as explained in the text
$P_{transmit}$	Power consumption of one beacon transmission.	1.52 W
P_{tail}	Power consumption of tail.	1.216 W
R_{ul}	Uplink network bitrate available for each road user's smartphone.	Max 50 Mbps for LTE (assumed 5 Mbps effective)
S	Size of a beacon message. Collision alert and switch command also same size.	0.8 Kbit
$t_{transmit}$	Beacon message data transfer time.	1.6×10^{-4} s (S/R_{ul})
t_{tail}	LTE tail	0.1 s for low-rate and 9.984×10^{-2} s for full-rate mode

(continued)

Table 1. Parameters influencing the adaptive multi-mode and road-safety system. (*continued*)

Parameter	Definition	Value
i_{lowrate}	Beacon messaging interval in low-rate mode.	$1 / f_{\text{lowrate}}$. A function of v
i_{fullrate}	Beacon messaging interval in full-rate mode.	100 ms
t_{idle}	LTE idle time	$i - t_{\text{transmit}}$, where i is the beaconing interval. $i_{\text{lowrate}} - 1.6 \times 10^{-4}$ if, low-rate mode 9.984×10^{-2} if, full-rate mode
P_{fullrate}	Additional power consumed by road-safety app in full-rate mode	1.216 W
P_{lowrate}	Additional power consumed by road-safety app in low-rate mode	A function of f_{lowrate} : $1.218 \times 10^{-1} \times f_{\text{lowrate}}$
PC_{city}	Number of pair-checks performed by cloud in each collision prediction loop.	According to (11) as explained in the text
r	Total number of regions in the city (city is divided into regions by server-side).	$r = \text{area}_{\text{city}}/\text{area}_{\text{region}}$
$\text{PC}_{\text{region}}$	Number of pair-checks for each city region.	Depends on traffic conditions and road user density.
N	Total number of road users (vehicles, pedestrians) moving in the city.	Depening on the city
n	Number of running parallel processes	Depends on workload and server loop deadline.
N_{ped}	Number of pedestrians in the region	Depends on pedestrian mobility and density.
N_{veh}	Number of vehicles in the region	Depends on traffic conditions and density.
$\text{UL}_{\text{region}}$	Uplink overhead of the road-safety system in each urban region	Calculated according to (12) as explained in the text
UL_{node}	Uplink overhead for each network node (pedestrian's or driver's smartphone).	$f_{\text{beacon}} \times S$
$\text{DL}_{\text{region}}$	Downlink overhead of the road-safety system in each urban region	Calculated according to (14) as explained in the text
DL_{node}	Downlink overhead for each network node	$(f_{\text{aler}}t + f_{\text{switch}}) \times S$
$N_{\text{colliding}}$	Number of detected colliding nodes within the urban region	Depends on traffic conditions
f_{beacon}	Average frequency of P2C messaging for pedestrians in the same region	$(t_{\text{fullrate}} \times t_{\text{fullrate}} + t_{\text{lowrate}} \times f_{\text{lowrate}})/t_{\text{mobility}}$
f_{alert}	Frequency of sending collision alert messages from cloud to colliding road users as long as the collision is not prevented.	10 Hz in worst-case
f_{switch}	Frequency of mode-switch messages sent from cloud to road users in a region.	Depends on traffic conditions

the high-risk situation no longer exists, cloud sends a notification to mobile app so that it switches back to *low-rate mode*.

Server-side executes its collision avoidance loop with a high precision of 10 times per second. It performs threat analysis based on the location, speed and heading of each {vehicle, pedestrian} pair moving within each other's high-risk range. As soon as a high probability of collision is predicted, server informs both vehicle and pedestrian's smartphone with a critical alert (shown in Fig. 1). At this point, either the auto-brake system or the driver activates the brakes and makes the vehicle stop. These collision warnings should be sent soon enough so that according to kinematics there is enough time to react and stop the car. Different types of warnings and critical alerts can be sent depending on the relative V2P distance and speed vectors, for example by increasing warning sound volume as distance gets closer. However, details of kinematics calculation and human-computer interaction are outside scope of this work. This paper

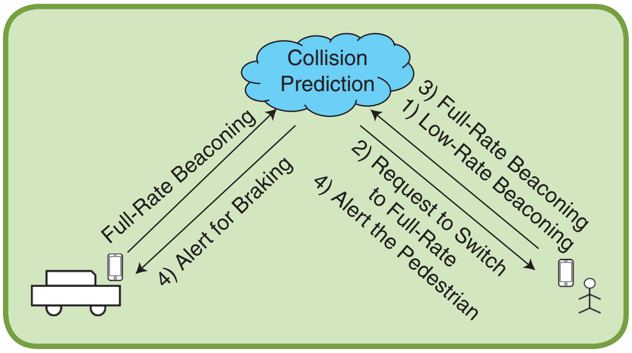


FIG 1 Communication and workflow during pedestrian's mobility: how the road-safety system adapts from low-risk to high-risk situation and how collisions are avoided. Road users are constantly sending their geolocation and speed data to cloud. Vehicle-to-cloud (V2C) beaconing is always performed in full-rate (10 Hz) and pedestrian-to-cloud (P2C) beaconing often in low-rate (e.g., 0.5 Hz) (1). When cloud detects a high-risk situation, it commands (2) pedestrian mobile app to also switch to full-rate beaconing (3) so that the system has maximum position data accuracy required to predict and alert (4) a potential collision.

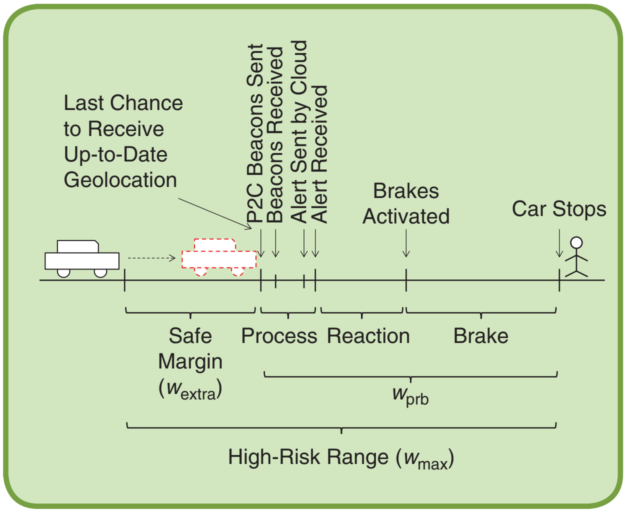


FIG 2 Timing analysis of AMM illustrated for collision avoidance in a worst-case scenario. X axis indicates distance and the arrows point to the last location (last chance) on road where certain action should be taken. Safe Margin (w_{extra}) gives the chance to system to switch pedestrian-to-cloud (P2C) beaconing from low-rate to full-rate. To successfully predict and prevent the accident, last vehicle-to-cloud (V2C) and P2C beacons should be sent at worst when vehicle reaches the dashed red location. Parameters are explained in more detail in section III.B. Also refer to equations (1) to (4).

focuses on how AMM manages messaging frequency and switching between different modes.

In our system, vehicle-to-cloud (V2C) communication is always performed in full-rate mode in order to frequently enough keep the server-side updated about vehicle location information. For this reason, in case V2C communication is made using driver's smartphone (not using built-in cellular module), the client-side mobile app should detect smartphone user's mode of transportation as explained in [32] and [33], so that it can switch from *pedestrian mode* to *driver mode* (always full-rate V2C beaconing) when smartphone user starts driving. We assume that driver's smartphone is plugged to vehicle's electricity system using a car adapter, and therefore does not have battery limitation problem. However, if such arrangement does not exist, for example if driver forgets to plug the smartphone, client-side mobile app reminds the driver about this problem.

B. Setting Beaconing Frequency of Low-rate Mode

This section formulates minimum P2C beaconing frequency for the low-rate mode so that it allows timely collision prevention depending on surrounding vehicular traffic. The lower the frequency of P2C messaging in low-rate mode (f_{lowrate}), the more energy-efficient the mobile app gets. However, cloud-based server-side application needs frequent enough location and heading updates and thus f_{lowrate} should be high enough to allow the road-safety system to detect high-risk situations early enough so that it can switch the smartphones to full-rate mode and avoid a possible collision in time.

Fig. 2 illustrates a collision avoidance scenario. Collision avoidance timing analysis is based on [34]. In addition, the low-rate P2C beaconing and our cloud-based model are taken into account. We should consider message transmission delays, server-side processing time, reaction delay before brakes are engaged and finally the distance travelled by vehicle while braking. Considering these values, in a worst-case scenario, the last V2C and P2C beacon messages should be sent at last when vehicle arrives in location specified by arrow on Fig. 2, so that prediction process is performed by server-side, the collision alert is received by vehicle just in time and vehicle stops right before colliding with the pedestrian. The distance traveled by the vehicle during this whole process is named w_{prb} (*process, reaction and brake window*). Ideally, earlier collision warnings are also sent within the *safe margin window* of w_{extra} (e.g, for bad weather conditions or older drivers), however this section only considers the worst-case scenario in which only one collision alert is sent.

As explained before, vehicle and pedestrian are in a high-risk situation after they get closer than $w_{\max}$ while approaching each other. After vehicle has entered the $w_{\max}$ window, server-side detects this high-risk situation as soon it receives a beacon from pedestrian containing up-to-date geolocation. However, P2C is being performed in low-rate mode before high-risk situation is detected and therefore this beacon might be received with a considerable delay because of the interval between subsequent P2C messages. To successfully predict and prevent the accident, the system should receive this beacon before the vehicle reaches w_{prb}. Based on this discussion, low-rate messaging interval (i_{lowrate}) should be adjusted so that the distance traveled by vehicle during beaconing intervals does not exceed w_{extra}.

$$i_{\text{lowrate}} = w_{\text{extra}}/v \tag{1}$$

Therefore minimum messaging frequency possible for low-rate mode is:

$$f_{lowrate} = v/(w_{\max} - w_{\text{prb}}) \tag{2}$$

Where v is the maximum assumed value for V2P relative speed. Pedestrian's speed is assumed 0 and therefore v denotes the vehicle speed before brakes are activated. $w_{\max}$ is defined in the system as a constant value. w_{prb} is calculated as follows.

$$w_{\text{prb}} = d_{\text{process}} + d_{\text{reaction}} + d_{\text{brake}} \tag{3}$$

$$w_{\text{prb}} = v \times t_{\text{process}} + v \times t_{\text{reaction}} + v^2/(2a_{\text{brake}}) \tag{4}$$

Where d_{brake} is the distance travelled by vehicle after brake activation until the vehicle completely stops. a_{brake} denotes the vehicle deceleration when brakes are activated and its value depends on the car make (e.g., how good the brakes are), road conditions (e.g., type of pavement, wet or dry road). d_{reaction}

is the distance travelled by vehicle while the auto-brake system or the driver are reacting to the alert and before brakes are activated. d_{process} is the distance travelled by vehicle while the system is performing communication and computation required to predict a collision. This collision prediction process takes $t_{\text{process}} = t_{\text{comm}} + t_{\text{comp}}$, where t_{comm} is the time spent transmitting the beacon message from smartphone to cloud server and the collision alert message from cloud to smartphone. Because both BSM and alert message are very small, their data transfer time (t_{transmit}) is split millisecond and thus ignored. Therefore, t_{comm} comprises only the average ping response time from smartphone to cloud server. t_{comp} is the time spent by cloud to run the predictive algorithm and perform threat analysis of all road users in a city.

C. Numerical Evaluation of Low-rate Beaconing Frequency

Value of w_{prb} is evaluated as follows. As explained above, t_{comm} is equal to the average ping response time over LTE and assumed to be 50 ms [35] (DSRC recommends latency of less 100 ms). In addition, our experiment (later explained in section VI) shows that t_{comp} can be as small as 500 ms. As a result, t_{process} equals 0.55 s. t_{reaction} is set to 0.83 s as stated in [36]. According to statistical data on car brake incidents provided by UK Highway Code [37], we estimate an average a_{brake} equal to 6.5 m/s^2, which is indifferent to v, car make and road conditions (details of this estimation in [20]). As a result, w_{prb} is a function of v calculated as $w_{\text{prb}} = 1.38 \times v + v^2/13$.

Similar collision avoidance systems work in a certain limited range which can be considered same as the $w_{\max}$ high-risk radius in our system. For example, the safety brake system in [38] and the pedestrian detection system in [13] have the detection range of 200 meters, and the V2P safety system in [39] has the detection range of around 180 meters. In this paper, we also conclude a similar value for $w_{\max}$ as follows. The worst-case scenario of collision avoidance is to prevent V2P collision when updates have reached the server when V2P distance is w_{prb}. Assuming the system supports vehicle speeds up to $v = 30$ m/s (around 110 km/h), above formula gives $w_{\text{prb}} = 162$ m. In addition, a safe margin of $w_{\text{extra}} = 30$ m (around 2 second) is considered. Therefore, high-risk situation is detected within $w_{\max} = 200$ m radius when vehicle's heading indicates that it is approaching the pedestrian. Full-rate collision avoidance process starts after this point. Put w_{prb} and $w_{\max}$ in (2), Fig. 3 illustrates f_{lowrate} in relation to v. Speed of vehicles is assumed to range from 0 to 144 km/h (0 to 40 m/s) considering usual speed limits of streets and intracity highways. As shown in this figure, the minimum possible beacon frequency quadratically increases with increasing vehicle speed. Low-rate messaging frequency does not exceed 1 beacon per second (1 Hz) as opposed to constant 10 Hz frequency of full-rate mode.

IV. Energy Efficiency and Battery Lifetime

This section discusses energy efficiency of AMM pedestrian road-safety and its practicality in terms of smartphone battery lifetime. Frequent beaconing results in additional power consumption and thus an overhead on typical discharge rate of smartphone battery as well as overhead on total battery lifetime.

As explained in previous section, AMM on client-side (smartphone road-safety app) activates P2C communication only during person's mobility. Therefore, the resulting battery lifetime overhead varies depending on duration of this mobility. Person's mobility during a day may comprise walking (as a pedestrian), cycling, riding public transport, or driving. Duration of person's mobility is denoted by t_{mobility} and its value differs from person to person depending on daily life. However, there is no reliable and accurate trace data about this duration, and how long people walk and are exposed to vehicular traffic is not accurately known to transportation professionals and researchers [40]. As a result, we resort to realistic assumptions of person's mobility period for our evaluations in this work.

Moreover, AMM makes the smartphone switch between full-rate and low-rate communication during person's mobility. Time spent operating in full-rate mode is specified as t_{fullrate} and the time spent in low-rate mode as t_{lowrate}. We have:

$$t_{\text{mobility}} = t_{\text{lowrate}} + t_{\text{fullrate}} \tag{5}$$

Because full-rate and low-rate modes perform beaconing with different frequencies, the power consumption overhead of each mode is different as well. This concludes that battery lifetime also changes depending on variations in full-rate activity ratio ($r_{\text{fullrate}} = t_{\text{fullrate}}/t_{\text{mobility}}$) during person's mobility. Duration of full-rate mode depends on how long pedestrian is exposed to collision risk, and therefore its value varies depending on factors such as density and speed of vehicles surrounding the pedestrian. The denser and faster the traffic gets, t_{fullrate} increases and t_{lowrate} decreases. Traffic density and speed itself depends on geolocation of urban areas where pedestrian passes as well as time of the day. For example, whether the pedestrian is located in the busy city center or in the suburbs might make a difference in full-rate activity

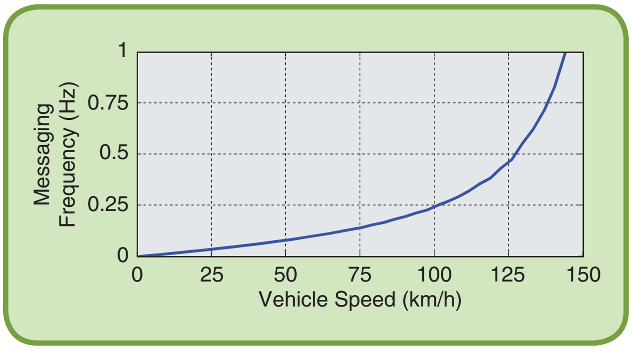


FIG 3 Minimum allowed beacon frequency of low-rate mode (f_{lowrate}) during pedestrian's mobility, in relation to maximum speed of surrounding vehicles (v). This beaconing frequency keeps minimum required freshness of position data for the system to perform timely collision prediction and prevention.

Similar collision avoidance systems work in a certain limited range which can be considered same as the w_{max} high-risk radius in our system.

ratio. Therefore, $r_{fullrate}$ largely depends on variations in geolocational and temporal factors: *geolocation within the city* and *time of the day*.

To sum it up, power consumption overhead caused by road-safety app and total battery lifetime depend on *geolocation* of areas where pedestrian passes, *time of the day*, and *duration of mobility*.

A. Formulating Smartphone Battery Lifetime

For the sake of this evaluation, it is assumed that person's mobility duration ($t_{mobility}$) is continuous and happens only once during a day, and battery is fully charged before pedestrian's mobility starts. With these assumptions, C_{rem} denotes battery state-of-charge (SOC) after mobility period and T is total battery lifetime.

$$C_{rem} = C – t_{mobility} \times (P_0 + P_{mobility}) \tag{6}$$

$$T = \begin{cases} t_{mobility} + C_{rem}/P_0 & \text{if } C_{rem} > 0 \\ C/(P_0 + P_{mobility}) & \text{if } C_{rem} \leq 0 \end{cases} \tag{7}$$

Where C refers to the battery capacity and P_0 refers to the average baseline power consumption of smartphone with typical daily usage. In other words, P_0 indicates average power consumption of applications or functionalities that the user might be running in addition to our road-safety mobile app. Value of P_0 varies per user and depends on different factors such as apps the user runs (video streaming, phone call, browsing, etc.) or how often the device screen is used as well as smartphone model. $P_{mobility}$ refers to the amount of additional power consumed by the road-safety mobile app during person's mobility:

$$P_{mobility} = (t_{fullrate} \times P_{fullrate} + t_{lowrate} \times P_{lowrate})/t_{mobility} \tag{8}$$

Where $P_{fullrate}$ is the amount of additional power smartphone consumes while running in full-rate mode and $P_{lowrate}$ is the additional power consumed in low-rate mode:

$$P_{lowrate} = (P_{tail} \times t_{tail} + P_{transmit} \times t_{transmit}) \times f_{lowrate} \tag{9}$$

$$P_{fullrate} = (P_{tail} \times t_{tail} + P_{transmit} \times t_{transmit}) \times f_{fullrate} \tag{10}$$

In order to send a beacon message, the system firstly needs to turn on radio. The radio consumes P_{tail} power while it is on and it goes off after t_{tail}. In addition, the system needs to transmit the beacon message through cellular network which takes $t_{transmit}$ and consumes $P_{transmit}$. More details about power consumption calculation is presented in [41].

In case of using a non-adaptive communication method, P2C beaconing is performed with a full-rate frequency during the whole mobility period and therefore the power it consumes is simply calculated as $P_{mobility} = P_{fullrate}$.

B. Numerical Evaluation of Battery Lifetime

We assume the pedestrian's smartphone to be Samsung Galaxy S3 with battery capacity of $C = 7.98$ Wh (with 2100 mAh and 3.8 V). t_0 is battery lifetime with typical daily usage (when the road-safety app is switched off) and it is not influenced by pedestrian mobility. As mentioned before, baseline power consumption P_0 varies depending on active applications, user usage habits and smartphone model. P_0 used in this paper is an estimated average value (0.5 W) based on the assumption that with a battery capacity of 2100 mAh, battery charge lasts around $t_0 = 16$ hrs on average for a typical user ($P_0 = C/t_0$).

It is assumed that the system is using optimized LTE (with Discontinuous Reception (DRX)) with 0.1 s tail and $R_{ul} = 5$ Mbps minimum effective uplink bitrate for each road user. The amount of transferred data with each beacon message (S) comprises the message itself in addition to TCP/IP overhead. The message is estimated 60 bytes which includes probe data according to BSM in DSRC [29] in addition to extra bytes specific to our system. TCP/IP overhead is 40 bytes. Therefore, $S = 100$ bytes (0.8 Kbit) and S/R_{ul} gives $t_{transmit} = 1.6 \times 10^{-4}$ S.

As explained in section III.B, $f_{fullrate}$ is constant 10 Hz and $f_{lowrate}$ changes depending on v as illustrated in Fig. 3. t_{tail} is evaluated as follows. For $P_{lowrate}$, because messaging interval ($i_{lowrate}$) is a variable equal or larger than 1 s, idle time ($t_{idle} = i_{lowrate} - t_{transmit}$) is always larger than DRX tail, and therefore t_{tail} is considered equal to DRX tail ($t_{tail} = 0.7$ s). For $P_{fullrate}$ however, because $i_{fullrate}$ is constant 0.1 s, t_{idle} is 9.984×10^{-2} s which is smaller than the DRX tail, and therefore t_{tail} is considered equal to t_{idle} ($t_{tail} = 9.984 \times 10^{-2}$ s). Furthermore, P_{tail} and $P_{transmit}$ have been measured as 1.216 W and 1.52 W by experiments in [41].

Putting above values in (9) and (10), $P_{fullrate}$ is a constant value 1.216 W, and $P_{lowrate}$ is a function of $f_{lowrate}$ calculated as $P_{lowrate} = 1.218 \times 10^{-1} \times f_{lowrate}$. Having above evaluation, (6), (7) and (8) denote that C_{rem} and T change depending on variations in maximum speed of surrounding vehicles (v), full-rate activity ratio ($r_{fullrate}$) and pedestrian's mobility period ($t_{mobility}$).

We firstly evaluate SOC in relation to v, when $r_{fullrate}$ and $t_{mobility}$ are constant. It is assumed that $t_{mobility} = 2$ hrsand full-rate mode is active always half of this mobility time and thus $r_{fullrate} = 0.5$ (regardless of geolocation and vehicular traffic conditions). It is concluded that influence of v on C_{rem} and T is minor, causing only around 2% difference in SOC when speed of vehicles ranges between 10 km/h (for which $C_{\text{rem}} = 72\%$) and 144 km/h (for which $C_{\text{rem}} = 70\%$). In the rest of this paper, we assume v to be 144 km/h, thus according to Fig. 3, low-rate messaging frequency is set at $f_{\text{lowrate}} = 1$ Hz.

We also examine how variations in r_{fullrate} and t_{mobility} influence power consumption and SOC. In (8), P_{fullrate} is always larger than P_{lowrate} because their only difference is in beaconing frequency for which always $f_{\text{fullrate}} > f_{\text{lowrate}}$. As explained before, with denser and faster vehicular traffic, t_{fullrate} increases and t_{lowrate} decreases, which results in larger P_{mobility}. As a result, SOC and battery lifetime linearly decrease with growing traffic. In other words, SOC and battery lifetime have inverse linear relation with r_{fullrate}. Furthermore, it is concluded that SOC has inverse linear relation with t_{mobility} as well. Fig. 4 presents SOC for different t_{mobility} and r_{fullrate} values as well as comparison with non-adaptive approach. As a concrete example, when r_{fullrate} is 50% and pedestrian is mobile for 6 hrs, 12% battery charge remains after this mobility and the battery lifetime is reduced to around 8 hrs (half of original lifetime). However, being in a moving state for such long hours during a day is a very pessimistic assumption. As explained before, there is no accurate trace data for t_{mobility} and its value differs from person to person. Nevertheless, considering that most of daily hours might be spent indoors, our assumption is that a typical smartphone user has maximum 2 hrs of mobility during a day (e.g. 1 hr commuting to work, 0.5 hr shopping walk, and 0.5 hr running exercise).

Based on above discussion, for a scenario in which pedestrian has been mobile for 2 hrs during the day, battery lifetime is 13.5 hrs which is only 15% less than the 16 hrs typical lifetime. It is concluded that, it is *practical* to run AMM road-safety app on a mainstream smartphone as this system is *not power-hungry*. Due to its small overhead on battery SOC, road-safety app can be used during the whole mobility period, and a practical battery charge of 70% still remains after 2 hrs. Next section further evaluates variations in r_{fullrate}, C_{rem} and T based on city-scale simulation.

V. Mobility Simulation Results

This section presents our city-scale simulation results in order to evaluate feasibility of AMM pedestrian road-safety in terms of smartphone battery lifetime in realistic urban situations. We also evaluate advantage of AMM over non-adaptive approach.

Numerical evaluation in previous section assumed r_{fullrate} of 25%, 50% and 75% during pedestrian's mobility regardless of location and time. However as explained before, in real-life P2C communication activity and thus power consumption are highly influenced by pedestrian's geolocation in the city and time of the day. Realistic simulation of city-scale daily vehicle traffic helps examine how often pedestrians are in high-risk or low-risk situation throughout different urban regions, and thus gives a more precise estimation of power-consumption and smartphone battery lifetime.

We use SUMO (Simulation of Urban Mobility) [42], [43] for microscopic vehicle mobility simulation and extend it to emulate our adaptive road-safety system. Simulation is performed for city of Cologne using realistic vehicle traffic data retrieved from the TAPASCologne project [44], [45], providing 2 hrs (06:00 am to 08:00 am) of trace information that includes peak traffic time. TAPASCologne SUMO scenario has imported the city's road networks from OpenStreetMap (OSM) [46] map of Cologne. The whole urban area simulated is around 31 $km < 00D7 >$ 33 km including parts of suburbs. Following sections analyze our simulation results from geolocational and temporal point of view.

A. Geolocational

This section evaluates influence of geolocation in city on P2C communication and battery lifetime. In order to evaluate with suitable geolocational granularity, our simulation divides the city into 100 m X 100 m divisions (100 regions in each square km) making up a total of 102,977 urban regions on the Cologne map. Simulation results are compiled into PDF and CDF (figures 5, 7, 8 and 9) to show the realistic chances of staying in energy-saving low-rate mode and achieving considerable power savings depending on geolocation of users throughout the city. We illustrate the results for parameters r_{fullrate}, SOC and T. Probability distribution and cumulative distribution are calculated according to fraction of urban regions in relation to the illustrated parameter.

1) Communication in Full-rate and Low-rate Mode

First of all, simulation results show how often it is possible for the road-safety system to switch to and stay in low-rate mode when smartphone user is mobile for 2 hrs during peak traffic time in each urban region. In other words, we evaluate what are possible realistic values for r_{fullrate} depending on geolocation throughout the city.

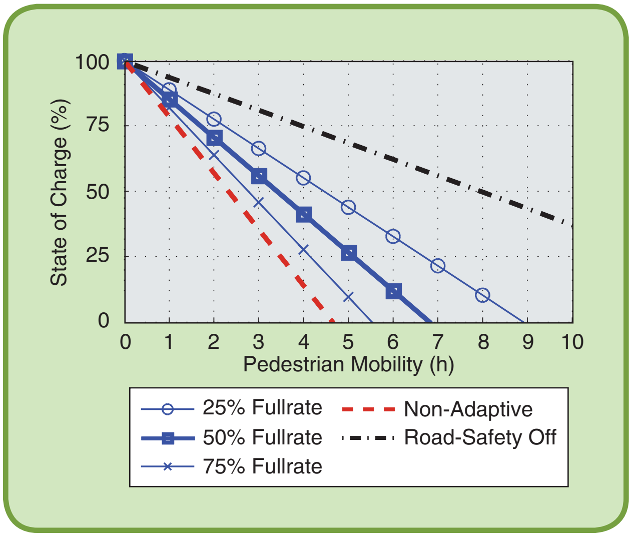


FIG 4 Smartphone state-of-charge (SOC) in relation to pedestrian mobility period (t_{mobility}), having different ratios of fullrate mode activity (r_{fullrate}). With longer mobility periods, changes in r_{fullrate} result in larger variance in SOC.

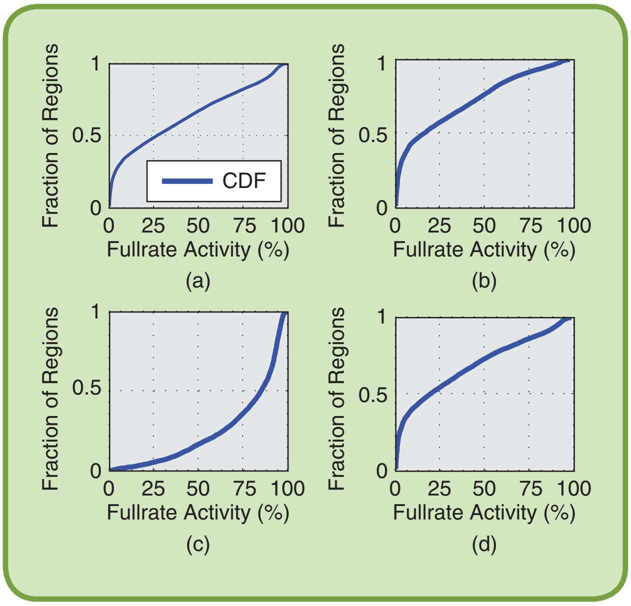


FIG 5 Ratio of full-rate mode pedestrian-to-cloud (P2C) communication (r_{fullrate}) out of total mobility (t_{mobility}), when pedestrian has been mobile for 2hrs in different geolocations (urban regions) throughout a) the whole city, b) western part of the city, c) city center, and d) eastern part of the city. These CDFs show probability of having less frequent full-rate beaconing (lower r_{fullrate} values) throughout the city, indicating higher energy-efficiency and longer battery lifetime.

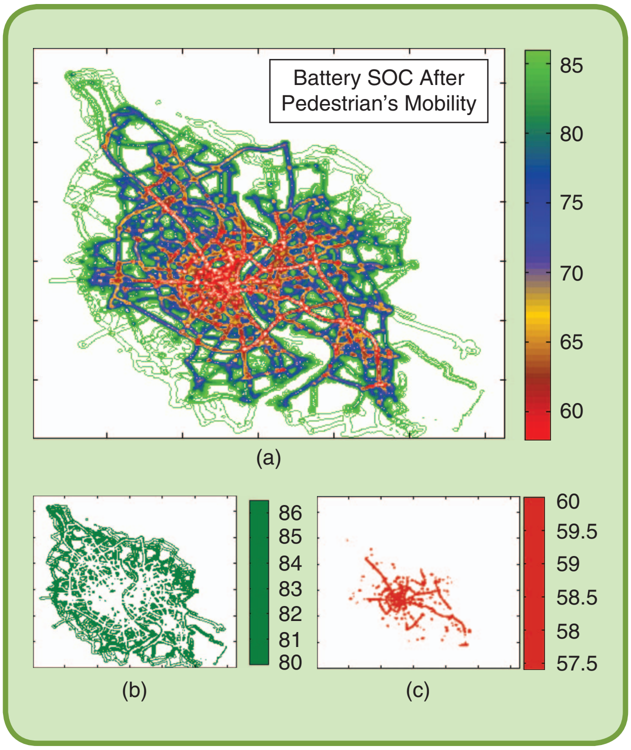


FIG 6 Smartphone state-of-charge (SOC) after pedestrian has been mobile in different geolocations for 2 hrs during peak traffic time, illustrated on Cologne city map, a) SOC throughout the whole city (white areas have no vehicle traffic), b) Showing only low activity areas with SOC larger than 80%, c) Showing only high activity regions with SOC smaller than 60%.

Fig. 5 illustrates ratio of full-rate mode communication (r_{fullrate}) among urban regions. Fig. 5.a shows that around half of regions have a rather efficient r_{fullrate} of less than 25%. Referring back to Fig. 4, $r_{\text{fullrate}} < 25\%$ indicates practical SOC of more than 75% and battery lifetime of more than 14 hrs. As expected, CDFs of Fig. 5 show that highest communication activities are more probable in city center. As an example, in city center, the probability (p) to have $r_{\text{fullrate}} < 25\%$ is very small ($p \approx 0.05$) , implying few chances of reducing power consumption of beaconing. Whereas, outside city center, probability of low-rate activity is much higher ($p \approx 0.58$ in western part and $p \approx 0.55$ in eastern part of the city) implying more chances of reducing power consumption and achieving longer battery lifetimes. Nevertheless, there is still high activity in some areas outside center, anticipated to be areas such as main highways and roads.

2) Energy Efficiency and Battery Lifetime

Secondly, we evaluate battery lifetime of smartphone depending on geolocation (urban regions). The smartphone is assumed to be Samsung Galaxy S3 with battery lasting around 16 hrs with typical daily usage (when the road-safety app is switched off). Battery is assumed to be fully charged before person starts walking in traffic-prone areas.

Fig. 6 illustrates smartphone SOC in different geolocations over the map of Cologne city, after smartphone user has been mobile for 2 hrs while using the road-safety app during peak traffic. As expected, this heat map also shows that urban regions with highest full-rate communication (thus highest power consumption and lowest SOC) are mostly located either in city center or otherwise over the main roads and highways. Fig. 6.c specifically illustrates these high activity regions. Empty (white) regions on the main map (Fig. 6.a) represent areas where no vehicular traffic was detected during simulation. In such areas, even if any pedestrian is present, the road safety mobile app always works in low-rate mode ($r_{\text{fullrate}} = 0\%$). Examples of such areas are the Rhine River crossing the city, large parks and areas with no roads and inhabitants.

Fig. 7 also illustrates smartphone SOC of all urban regions. Battery SOC ranges between 57% and 87%. In other words,

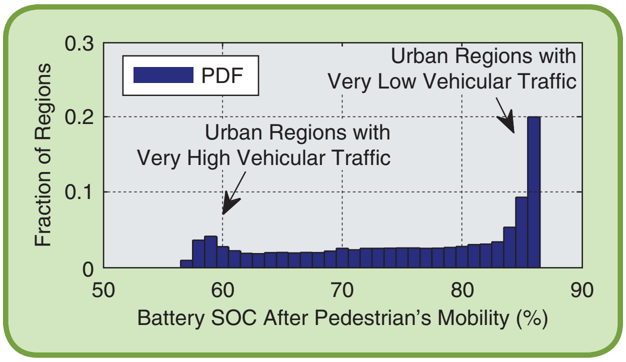


FIG 7 Smartphone battery state-of-charge (SOC) after pedestrian has been mobile in different geolocations (urban regions of the city) for 2 hrs during peak traffic.

worst-case battery discharge after 2 hrs mobility is 43%. SOC PDF has a binominal attribute which highlights regions of the city with high vehicular traffic (first peak on the chart) and regions with low vehicular traffic (second peak showing a much higher probability). In addition, CCDF of SOC values shows that in around half of urban regions, smartphone has more than 78% SOC ($p \approx 0.5$ for SOC > 78%). This concludes that road-safety app imposes only 10% overhead on SOC (SOC with road-safety turned off is 87% after 2 hrs). Fig. 8 denotes that total battery lifetime is more than 14 hrs 30 mins in around half of urban regions ($p \approx 0.5$ for $T > 14.5$) and worst-case battery lifetime 11 hrs among all regions. With a non-adaptive method, battery lifetime would be always around 11 hrs regardless of urban region.

As a result, using AMM road-safety system is *practical* as it is *not power-hungry*. Due to its small overhead (mean overhead is 13%) on battery SOC, it can be used during the whole mobility period while a practical battery charge also remains after mobility has ended.

In addition, Fig. 9 illustrates gain in battery lifetime using AMM in comparison to non-adaptive method. It shows the advantage of using AMM adaptive method as it increases the battery charge remained after mobility as well as the total battery lifetime. According to simulation results for 2 hrs of mobility, half of the regions achieve more than 40% increase in SOC as opposed to a non-adaptive approach. According to Fig. 9, the longer time person is mobile, the more is the advantage of AMM over non-adaptive approach.

This paper illustrated the results for Galaxy S3 assuming a battery with 2100 mAh capacity and typical lifetime of 16 hrs. However, we have also evaluated the results for different battery capacities ranging from 1000 mAh (used battery or older phones) up to 2100 mAh, implying different daily battery lifetimes (ranging from 7.5 to 16 hrs). Results show that with smaller battery capacity, the road-safety app imposes more overhead on SOC but on the other hand AMM achieves more gain over a non-adaptive method. As an example, for a low 1000 mAh capacity, mean overhead on SOC rises from 13% to 32% while mean gain improves from 34% to 418%.

It should be also noted that the road-safety app unregisters from smartphone's GPS sensing during *sleep mode* to save energy (thus GPS is turned off if no other app is using it) and registers back once switching to *low-rate* or *full-rate* modes. Doing so is possible because turn-on and turn-off delays are only a few seconds for Assisted GPS (A-GPS) in mainstream smartphones [47]. More details on energy-efficient and sensor-fusion-based positioning have been presented in [48], [49]. According to these works, power consumption of GPS sensing ranges between ~300 mW and ~600 mW depending on GPS and phone model. Power consumption by positioning is not included in baseline P_0 as we assume a typical user does not usually run applications that sense GPS. Instead, we explicitly add it to P_{mobility}. If we assume 450 mW for GPS sensing, we get mean battery lifetime of 12 hrs 20 mins instead of the ~14 hrs calculated above. Similarly, mean overhead of road-safety app over SOC rises from 13% to 26%. On the other hand, mean gain of AMM over a non-adaptive method improves from 34% to 42%.

B. Temporal

Fig. 10 illustrates power consumption of road-safety system from a temporal point of view for two randomly selected regions of city center. It demonstrates a 1 hr 30 mins (1.5 hours) long period of person's mobility including peak traffic time (06:00 to 07:30). As expected, this figure shows that average power consumption increases as time passes towards peak traffic.

Road-safety app frequently switches between full-rate mode and low-rate mode depending on risk level. Fig. 11 illustrates switch counts on Cologne city map. Number of switches can be significantly different among different locations in the city even if they have the same r_{fullrate} value.

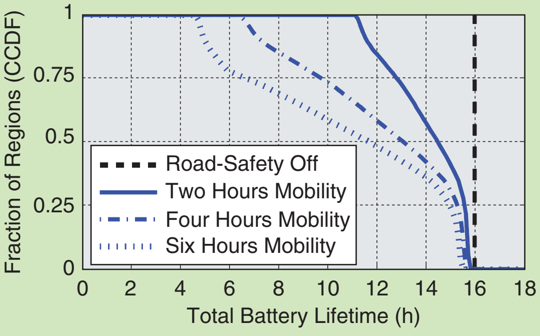


FIG 8 Smartphone battery lifetime (*T*) when pedestrian has been mobile in different geolocations (city regions) for different periods during a day. This CCDF shows probabilities of achieving longer battery lifetimes throughout the city.

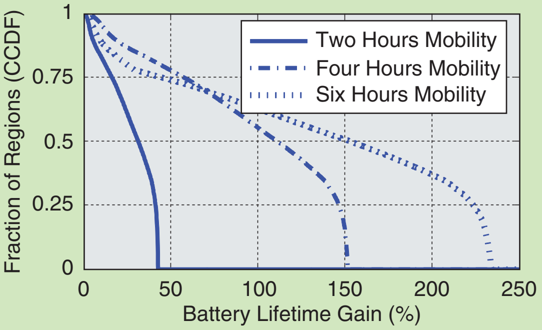


FIG 9 Comparing AMM with non-adaptive beaconing: Battery lifetime gain compared to a non-adaptive method, when pedestrian has been mobile in different geolocations (city regions) for different periods during a day. This CCDF shows probabilities of achieving more gain over a non-adaptive method throughout the city.

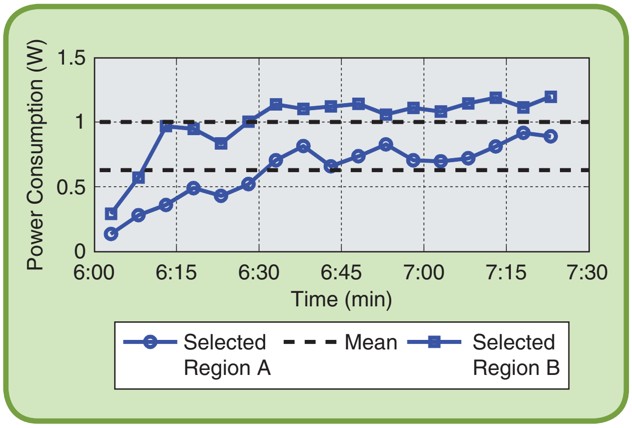


FIG 10 Changes in average power consumption of road-safety application during 1 hr 30 mins of pedestrian's mobility in peak traffic time, in two different regions of city center (A and B).

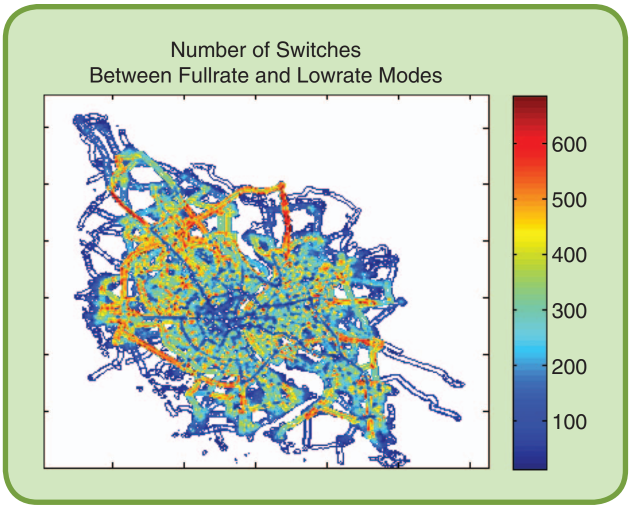


FIG 11 Temporal behavior of road-safety system: Number of switches between low-rate mode and full-rate mode in different urban regions illustrated on Cologne city map.

With very large switch count and high switch frequency (e.g., switch frequency larger than low-rate frequency), there is higher probability that it will not allow the system to stay in low-rate mode long enough to send the next beacon message (low-rate mode *interrupted* and *incomplete*). As an example, a selected region of city center has total number of 707 switches during 1 hr 30 mins long mobility, resulting in total 350 low-rate mode interrupts (total 9 min of low-rate mode duration is interrupted). In such interrupt scenarios, fewer beacon messages are sent during low-rate compared to what initially was expected, thus less power is consumed. This observation has been reflected in power consumption calculation and the results presented so far.

By comparing Fig. 11 with Fig. 6, it is observed that areas with minimum number of switches (more consistent messaging behavior) tend to be either regions with highest full-rate ratios (e.g. city center and main roads with high vehicle traffic volume) or regions with lowest full-rate ratios (e.g., suburbs). The reason is that in city center and main roads, traffic is so frequent that most of the times there is at least one vehicle approaching within the $w_{\max}$ high-risk range such that smartphone has few opportunities to switch to and stay in low-rate mode. Whereas, in low-traffic areas such as suburbs, system behavior is the opposite, mostly staying in low-rate mode. On the other hand, system's messaging behavior in areas with a medium full-rate activity is much less uniform (consistent) having much more number of switches.

VI. Server-side Experiment and Cloud Service Costs

This section examines practicality of the AMM road-safety system in terms of computation load and its associated cloud service costs for city-scale scenarios. We developed a server-side collision prediction prototype to measure execution times and delays in order to estimate compute demand and cloud costs. This section also explains how cloud servers can perform timely threat analysis on probe data being received from numerous road users. It also presents details of implementing such cloud system.

It is assumed that the road-safety system performs both V2V and V2P collision predictions. Geolocation, speed vector and other information of road users is updated maximum 10 times each second (10 Hz, frequency of full-rate mode communication). In each collision avoidance loop, server-side needs to perform pair-checks between road users moving in the city. Therefore, computation complexity of the whole collision prediction is $O(n^2)$ and computation load grows quadratically with the number of vehicles and pedestrians. However, to substantially reduce number of unnecessary V2V and V2P pair-checks, server-side divides the city into small divisions (regions) and groups all road users according to the region they reside in. Collision prediction is now performed only per each region. It pair-checks vehicles and pedestrians present in current region with each other as well as with vehicles in the adjacent neighboring regions. Duplicate region-to-region pairs are also avoided. Pair-check count for the whole city is $\mathrm{PC_{city}} = \sum_{\mathrm{region}=1}^{r} \mathrm{PC_{region}}$ where $\mathrm{PC_{region}}$ is the number of pair-checks for each region and r total number of regions in the city. When region width and height are both chosen equal to $w_{\max}$, and assuming the road users (N) are evenly distributed among regions (N/r road users in each region), we get:

$$\mathrm{PC_{city}} = 9N^2/2r - N/2 \qquad (11)$$

Grouping road users into regions and traversing neighbors only take $N + 5r$ (of O(n)). Compared to $N(N-1)/2$, Eq. 11 results in more than 97% (two orders of magnitude) reduction in pair-checks. In addition, as explained below, parallelizing the server-side application into several processes running on separate compute nodes can minimize the total execution time to a value between 100 ms and 500 ms.

Our prototype server-side application implements grouping of road users based on regions, pair checking and collision prediction as explained in this paper. High-risk range ($w_{\max}$) for V2P collision prediction is set to 200 m as explained before. In addition, $w_{\max}$ for V2V is extended to 500 m to support scenarios where both vehicles are moving at maximum 110 km/h (refer to Eq. (4)). For city-scale testing, the probe data records (road user beacons) including geolocation, speed and heading of vehicles and pedestrians are randomly generated on server-side and passed to road-safety application in every loop (emulating beacon messages arriving each 100 ms). These probe records are indexed in memory by road user unique ID. As the final output, server prepares list of colliding road users as well as pedestrians who need to switch their AMM mode. In future versions, a dedicated process will constantly read from these two queues and send relevant push messages (i.e., collision alert, switch command) to road users listed in the queue. The system has two tiers, and tier1 distributes the incoming probe records among all parallel processes of tier2.

We test our prototype application on Amazon Web Services (AWS) Elastic Compute Cloud (EC2) C4.8xLarge instance. For the sake of this experiment, both tiers are running as single process application on a single vCPU. This way, execution time of each complete server loop and other metrics such as inter-process data exchange overhead are measured. Furthermore, it is evaluated how the system achieves speed-up by adding more processes and instances, and how total execution time is reduced to meet timeliness requirements: Speedup $= (s+p)/(s+c+p/n)$ where s is execution time of sequential parts of the application, p execution time of parts possible to divide and parallelize, and c is inter-process networking overhead.

s is mainly the time spent for reading incoming probes and making the regions. After this step, each region together with its adjacent neighbors (a 'region-group') is an independent data unit. Pair-checking of road users within each region-group can be done in isolation from other region-groups. Therefore region-group records are distributed among n independent collision avoidance parallel processes. p denotes the time spent to compute all these records sequentially. Each vCPU runs one of system's parallel processes, except two vCPUs per instance reserved for miscellaneous system tasks. Therefore 34 out of total 36 vCPUs provided by each C4.8xLarge instance are taken into account (with 100% CPU utilization) in our estimation of total execution time. We also concurrently run several instances of our prototype application on different vCPUs to test performance of EC2 instance.

Values of s and p are obtained from our experiments running the application. The amount of inter-process data transfer is also obtained from experiments and value of c estimated accordingly as follows. While increasing number of processes and instances increases parallel execution speed, inter-process networking causes some overhead. Although for the sake of this experiment there is no network-based message passing between tier1 and tier2, we have considered the associated networking overhead towards total execution time. To maximize bandwidth and minimize latency, it is considered that instances are placed in the same placement group and running in a Virtual Private Cloud (VPC) in the same availability zone. Having above configuration, C4.8xLarge instances provide 10 Gbps maximum inter-process network throughput based on AWS's documentation [50] and at least around 5 Gbps consistent throughput based on third-party experiments [51], [52]. Similarly maximum inter-instance latency is considered 1 ms. We also maximize TCP/IP packet size to carry as large region-group data chunks as possible. Our estimation shows this will decrease data transmission time and thus increase the speedup. Finally, based on above values and the size of transmittable inter-process data obtained from experiments, networking overhead (c) is estimated as: $c =$ *latency + interprocess data transfer time.*

Table 2. Cloud service cost for road-safety system.

City Area (km × km)	Road Users	Density per km^2	Maximum resources required		Monthly Cost (USD)
			vCPUs	Instances	
10 × 10	1×10^4	100	1	1	763
	5×10^4	500	7	1	763
	5×10^5	5,000	464	14	10,682
30 × 30	5×10^4	55	1	1	763
	5×10^5	555	82	3	2,289
	1×10^6	1,111	493	15	11,445
40 × 40	5×10^4	31	1	1	763
	5×10^5	312	51	2	1,526
	1×10^6	625	313	10	7630

*C4.8xLarge reserved instance three-year term is 27,488 USD (763 USD per month).

*In each experiment, pedestrian population ranges from 20% to 80% of total road users. Because the high-risk range ($w_{\max}$) for V2V collision prediction is larger (500 m as opposed to 200 m for V2P), therefore larger vehicles/pedestrians ratios impose higher workloads. This table only shows results for the highest load in the range.

Table 2 summarizes experiment results for cities with different area, population and density of road users per km^2. It lists maximum resources required to predict collisions and define AMM mode switches for all road users within a certain time limit. For some workloads, meeting a short deadline of $t_{\text{comp}} = 100$ ms is not possible due to inter-process networking overhead. For example for 5×10^5 road users, it takes 98 ms to transmit the incoming probe records among tier2 nodes. In such scenarios, deadline is

Table 3. Network overhead of road-safety system.

	Network load evaluated for	Peak additional traffic caused by road-safety system	Capacity of cellular network per square km (peak)
Uplink	Individual User	8 Kbps	1,000 Mbps [1]
	Square km	115.2 Mbps	
Downlink	Individual User	16 Kbps	2,000 Mbps [1]
	Square km	81.6 Mbps	

[1] Assuming density of 20 base stations per square km [50], [51] and using LTE user equipment category 3 (LTE CAT 3).

extended to $t_{\mathrm{comp}} = 500$ ms and it still allows a safe margin of 2.33 s ($\approx$ 140 m). As shown in Table 2, monthly cost of running the road-safety system on cloud ranges between 763 USD and 11,445 USD.

However, this cost is much less compared to the DSRC-based approach where dedicated hardware is deployed on vehicles. To have a comparison, we consider a high density of 5,000 road users per km^2 in a city of total 5×10^5 road users. For the mobile cloud-based approach, it is assumed that 35% of all road users need to upgrade their phone to Internet-connected smartphone costing 500 USD (fairly expensive assumption). Therefore, there is an initial one-time cost of 87,500,000 USD for the whole city. In addition according to Table 2, estimated cloud service cost is 10,682 USD per month and ~128,000 USD per year for the whole city. For the classic DSRC-based approach, the cost of DSRC module installation is assumed only 200 USD for each road user. Therefore, estimated system deployment cost is 100,000,000 USD for the whole city. This deployment cost subtracted by initial cost of cloud-based approach is ~1.0×10^2 times more than the annual cloud service cost (equivalent to the cost of running the cloud-based road safety for around 100 years).

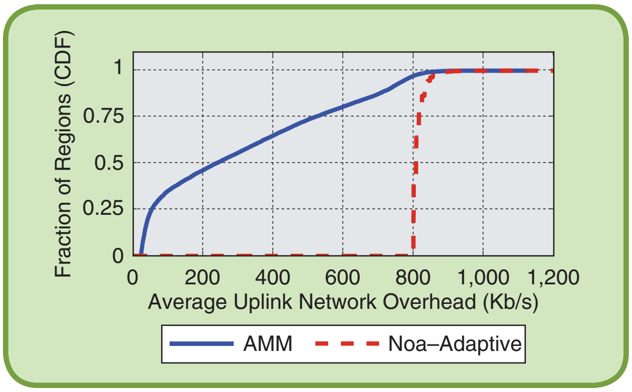


FIG 12 Average uplink load of road-safety system in different geolocations (city regions) throughout the city for total 2 hrs of peak traffic time. This figure also compares uplink overhead of AMM with uplink overhead of non-adaptive approach.

To sum it up, this section showed feasibility and cost-effectiveness of running road-safety system on mainstream cloud providers such as AWS EC2. It should be mentioned that there are several shared tenancy and dedicated hosting providers (e.g., Dediserv, Rackspace) on the market providing similar or better performance/cost ratio, and in addition cloud hosting prices are expected to keep dropping.

VII. Network Load Overhead

This section evaluates network load caused by the road-safety system and compares it against the capacity of cellular networks. Uplink overhead in each urban region is:

$$\mathrm{UL}_{\mathrm{region}} = (N_{\mathrm{ped}} \times f_{\mathrm{beacon}} + N_{\mathrm{veh}} \times f_{\mathrm{fullrate}}) \times S \tag{12}$$

Where N_{ped} is the number of pedestrians and N_{veh} is number of vehicles in each region. S is the size of each beacon message and f_{beacon} denotes average frequency of P2C messaging for pedestrians in the same region during a chosen mobility period (t_{mobility}):

$$f_{\mathrm{beacon}} = (t_{\mathrm{fullrate}} \times f_{\mathrm{fullrate}} + t_{\mathrm{lowrate}} \times f_{\mathrm{lowrate}})/t_{\mathrm{mobility}} \tag{13}$$

According to (12), uplink overhead grows linearly with the number of network nodes (i.e., vehicles, pedestrians) in the city.

Worst-case scenario (peak) uplink network load is when P2C communication for all pedestrians is performed at full-rate mode all the time ($f_{\mathrm{beacon}} = 10$ Hz). Simulation results show a maximum of 4,400 vehicles traveling in each square km simultaneously during one second, and it is assumed that a maximum of 10,000 pedestrians exist per km^2 in the most dense areas (Cologne average population per square km is only ~2,500). In addition, S equals 0.8 Kbit as estimated before in section IV.B. There are 100 urban regions per km^2 as explained in section V.A. Therefore, peak uplink BW overhead is calculated as 115.2 Mbps per square km. Uplink overhead for each network node (e.g., pedestrian's or driver's smartphone) is $\mathrm{UL}_{\mathrm{node}} = f_{\mathrm{beacon}} \times S$ and therefore its peak value is a small BW of 8 Kbps.

Furthermore, downlink overhead in each urban region is:

$$\mathrm{DL}_{\mathrm{region}} = (N_{\mathrm{colliding}} \times f_{\mathrm{alert}} + N_{\mathrm{ped}} \times f_{\mathrm{switch}}) \times S \tag{14}$$

Where $N_{\mathrm{colliding}}$ is the number of detected colliding road users within the urban region per second and f_{alert} is frequency of sending collision alert messages from cloud to those colliding nodes as long as the collision is not prevented. f_{switch} denotes frequency of sending mode switch messages from cloud to nodes in the region. It is assumed that collision alert and switch command have the same message size as the beacon message (S). According to (14), downlink overhead grows linearly with the number of network nodes in the city.

Worst-case scenario (peak) downlink load is theoretically when one mode switch command is sent in each server loop (every 0.1 s) to all pedestrians in the region

($f_{\text{switch}} = 10$ Hz). In addition, it is assumed that one collision is anticipated every second in each region ($N_{\text{colliding}} = 2$) and $f_{\text{alert}} = 10$ Hz. Therefore, peak downlink BW overhead is calculated as 81.6 Mbps per square km. Downlink overhead for each network node is $\text{DL}_{\text{node}} = (f_{\text{alert}} + f_{\text{switch}}) \times S$ and therefore its peak value is a small BW of 16 Kbps.

Table 3 summarizes peak network loads and compares it with capacity of network infrastructure [53], [54]. Moreover, Fig. 12 illustrates average uplink network load during 2 hrs based on realistic simulation results. It shows that half of regions have less than ≈ 0.2 Mbps uplink overhead (≈ 20 Mbps per km^2) and almost all regions have less than 0.8 Mbps overhead (80 Mbps per km^2). Therefore, cellular network infrastructure with commercial LTE base stations is capable of handling network load demand of road-safety system.

VIII. Conclusion

In this paper, we have proposed energy-efficient adaptive multimode (AMM) approach to enable development of cloud-based pedestrian road-safety systems that employ cellular technologies and mainstream smartphones rather than dedicated hardware and communication methods. In comparison to ad-hoc technologies (e.g. WiFi, IEEE 802.11 p), using cellular technology (e.g., 3G, LTE) together with smartphones is a better fit for pedestrian safety applications in terms of reducing user adoption costs and market penetration time. To run the road-safety system regardless of limited smartphone battery capacity, AMM adjusts mode and frequency of pedestrian-to-cloud beaconing depending on vehicle-to-pedestrian collision risk.

City-scale realistic mobility simulation concludes:

- AMM enables running wireless-based pedestrian road-safety systems with mainstream smartphones as it has a small overhead on battery lifetime (e.g., mean overhead is ~13% according to simulation for 2 hrs of pedestrian mobility). Road-safety mobile app can be used during the whole mobility period of pedestrian while a practical state-of-charge also remains after mobility has ended.
- Smartphone battery lifetime is practical regardless of pedestrian's geolocation within the city.
- AMM has advantage over non-adaptive methods. It increases smartphone SOC after pedestrian's mobility and extends the total battery lifetime.

Furthermore, server-side prototype experiment and evaluation of communication overhead conclude:

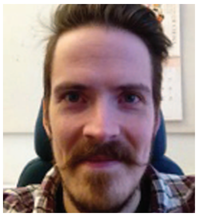
- Estimated computation load and its associated cloud service costs indicate practicality of running the pedestrian road-safety system on conventional cloud providers.
- It is feasible to run the system on conventional cellular networks.

Acknowledgement

This work was partly supported by Tekes (The Finnish Funding Agency for Technology and Innovation) project Everyday Sensing.

About the Authors

Mehrdad Bagheri is a PhD student at Aalto University working in joint projects by departments of Computer Science and Build Engineering focusing on smart traffic and green mobility. He obtained M.Sc. (software engineering and embedded systems) from Tartu University and Tallinn University of Technology in 2013. Before joining academia, he obtained experience in a wide range of projects in ICT industry for several years. His research interests are intelligent transportation systems, pedestrian active road-safety, IoT, cloud computing and mobile applications.



Matti Siekkinen obtained M.Sc. degrees in computer science from Helsinki University of Technology in 2003 and Ph.D. from Eurecom/University of Nice Sophia-Antipolis in 2006. He is presently a postdoctoral researcher at Aalto University. His current research focuses on mobile systems in which he combines techniques from multimedia computing, mobile networking, cloud computing, system analysis, data analytics, and HCI.

Jukka K. Nurminen is principal scientist at VTT and adjunct professor of computer science at Aalto University. In 2011-15 he spent five years as a professor at Aalto working on mobile cloud computing and energy-efficiency. He has a strong industry background with almost 25 years experience of software research at Nokia Research Center. Jukka's experience ranges from mathematical modeling to expert systems, from network planning tools to solutions for mobile phones, and from R&D project management to tens of patented inventions. Jukka received his M.Sc degree in 1986 and Ph.D. degree in 2003 from Helsinki University of Technology. His research interests are focused on systems and solution that are energy-efficient, distributed, or mobile.

References

[1] WHO. (2014, May 28). Decade of action for road safety 2011-2020: Saving millions of lives. [Online]. Available: http://www.who.int/violence_injury_prevention/publications/road_traffic/decade_booklet/en/

[2] T. Gandhi and M. M. Trivedi, "Pedestrian protection systems: Issues, survey, and challenges," *IEEE Trans. Intell. Transp. Syst.*, vol. 8, no. 3, pp. 413–430, Sept. 2007.

[3] D. Gerónimo, A. M. López, A. D. Sappa, and T. Graf, "Survey of pedestrian detection for advanced driver assistance systems," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 7, pp. 1239–1258, Jul. 2010.

[4] D. Fernandez Llorca, V. Milanes, I. Parra Alonso, M. Gavilan, I. Garcia Daza, J. Perez, and M. Sotelo, "Autonomous pedestrian collision avoidance using a fuzzy steering controller," *IEEE Trans. Intell. Transp. Syst.*, vol. 12, no. 2, pp. 390–401, June 2011.

[5] K. David and A. Flach, "CAR-2-X and pedestrian safety," *IEEE Veh. Technol. Mag*, vol. 5, no. 1, pp. 70–76, Mar. 2010.

[6] P. Ross, "Thus spoke the autobahn," *IEEE Spectr.*, vol. 52, no. 1, pp. 52–55, Jan. 2015.

[7] Y. L. Morgan, "Notes on DSRC & WAVE standards suite: Its architecture, design, and characteristics," *IEEE Commun. Surv. Tutorials*, vol. 12, no. 4, pp. 504–518, 2010.

[8] G. Karagiannis, O. Altintas, E. Ekici, G. Heijenk, B. Jarupan, K. Lin, and T. Weil, "Vehicular networking: A survey and tutorial on requirements, architectures, challenges, standards and solutions," *IEEE Commun. Surv. Tutorials*, vol. 13, no. 4, pp. 584–616, 2011.

[9] Y. Toor, P. Muhlethaler, A. Laouiti, and A. La Fortelle, "Vehicle ad hoc networks: Applications and related technical issues," *IEEE Commun. Surv. Tutorials*, vol. 10, no. 3, pp. 74–88, 2008.

[10] X. Wu, R. Miucic, S. Yang, S. Al-Stouhi, J. Misener, S. Bai, and W. Chan, "Cars talk to phones: A DSRC based vehicle-pedestrian safety system," in *Proc. IEEE 80th Vehicular Technology Conf.*, 2014, pp. 1–7.

[11] S. M. Tornell, C. T. Calafate, J.-C. Cano, P. Manzoni, M. Fogue, and F. J. Martinez, "Evaluating the feasibility of using smartphones for ITS safety applications," in *Proc. IEEE 77th Vehicular Technology Conf.*, 2013, pp. 1–5.

[12] C. Sugimoto, Y. Nakamura, and T. Hashimoto, "Prototype of pedestrian-to-vehicle communication system for the prevention of pedestrian accidents using both 3G wireless and WLAN communication," in *Proc. 3rd Int. Symp. Wireless Pervasive Computing*, 2008, pp. 764–767.

[13] GM Corporate Newsroom. (2014, May 26). GM developing wireless pedestrian detection technology. [Online]. Available: http://media.gm.com/media/us/en/gm/news.detail.html/content/Pages/news/us/en/2012/Jul/0726_pedestrian.html

[14] M. Liebner, F. Klanner, and C. Stiller, "Active safety for vulnerable road users based on smartphone position data," in *Proc. IEEE Intelligent Vehicles Symp.*, 2013, pp. 256–261.

[15] W. Vandenberghe, I. Moerman, and P. Demeester, "On the feasibility of utilizing smartphones for vehicular ad hoc networking," in *Proc. 11th Int. Conf. ITS Telecommunications*, 2011, pp. 246–251.

[16] G. Araniti, C. Campolo, M. Condoluci, A. Iera, and A. Molinaro, "LTE for vehicular networking: A survey," *IEEE Commun. Mag*, vol. 51, no. 5, pp. 148–157, May 2013.

[17] M. Whaiduzzaman, M. Sookhak, A. Gani, and R. Buyya, "A survey on vehicular cloud computing," *J. Netw. Comput. Appl.*, vol. 40, pp. 325–344, Apr. 2013.

[18] S. Luukkainen, M. Karjalainen, J. Winter, and M. Bagheri, "Business model analysis of mobile traffic safety services," *Emerald Info*, vol. 18, no. 1, pp. 56–66, 2015.

[19] Reviewed.com Cameras. (2014, Dec. 8). 4 Battery concepts that could change the world. [Online]. Available: http://cameras.reviewed.com/features/4-battery-concepts-that-could-change-the-world

[20] M. Bagheri, J. K. Nurminen, and M. Siekkinen, "Cellular-based vehicle to pedestrian (V2P) adaptive communication for collision avoidance," in *Proc. 3rd Int. Conf. Connected Vehicles and Expo*, Vienna, 2014, pp. 450–456.

[21] R. Schmidt, T. Leinmuller, E. Schoch, F. Kargl, and G. Schafer, "Exploration of adaptive beaconing for efficient intervehicle safety communication," *IEEE Netw.*, vol. 24, no. 1, pp. 14–19, Jan. 2010.

[22] K. Z. Ghafoor, J. Lloret, K. A. Bakar, A. S. Sadiq, and S. A. Ben Mussa, "Beaconing approaches in vehicular ad hoc networks: A survey," *Wirel. Pers. Commun.*, vol. 73, no. 3, pp. 885–912, May 2013.

[23] European Standard (Telecommunication Series). (2014, Mar. 12). ETSI EN 302 665: Intelligent Transport Systems (ITS); Communications Architecture. [Online]. Available: http://www.etsi.org/deliver/etsi_en/302600_302699/302665/01.01.01_60/en_302665v010101p.pdf

[24] European Standard (Telecommunication Series). (2014, Mar. 12). ETSI TS 102 637-2: Intelligent Transport Systems (ITS); Vehicular communications; Basic set of applications; Part 2: Specification of cooperative awareness basic service. [Online]. Available: http://www.etsi.org/deliver/etsi_ts/102600_102699/10263702/01.02.01_60/ts_10263702v010201p.pdf

[25] J. J. Anaya, P. Merdrignac, O. Shagdar, F. Nashashibi, and J. E. Naranjo, "Vehicle to pedestrian communications for protection of vulnerable road users," in *Proc. IEEE Intelligent Vehicles Symp.*, 2014, pp. 1037–1042.

[26] C. L. Huang, Y. Fallah, R. Sengupta, and H. Krishnan, "Adaptive intervehicle communication control for cooperative safety systems," *IEEE Netw.*, vol. 24, no. 1, pp. 6–13, Jan. 2010.

[27] M. van Eenennaam, W. K. Wolterink, G. Karagiannis, and G. Heijenk, "Exploring the solution space of beaconing in VANETs," in *Proc. IEEE Vehicular Networking Conf.*, 2009, pp. 1–8.

[28] C. Sommer, O. Tonguz, and F. Dressler, "Traffic information systems: Efficient message dissemination via adaptive beaconing," *IEEE Commun. Mag.*, vol. 49, no. 5, pp. 173–179, May 2011.

[29] SAE Int and DSRC Committee. Dedicated Short Range Communications (DSRC) Message Set Dictionary, SAE Standard J2735, 2015.

[30] A. Bujari, B. Licar, and C. E. Palazzi, "Movement pattern recognition through smartphone's accelerometer," in *Proc. IEEE Consumer Communications and Networking Conf.*, 2012, pp. 502–506.

[31] N. Roy, H. Wang, and R. Roy Choudhury, "I am a smartphone and I can tell my user's walking direction," in *Proc. 12th Annu. Int. Conf. Mobile Systems, Applications, and Services*, 2014, pp. 329–342.

[32] S. Hemminki, P. Nurmi, and S. Tarkoma, "Accelerometer-based transportation mode detection on smartphones," in *Proc. 11th ACM Conf. Embedded Networked Sensor Systems*, 2013, pp. 1–14.

[33] V. Raman, J. Shen, A. Kansal, V. Bahl, and R. R. Choudhury, "I am a smartphone and I know my user is driving," in *6th Int. Conf. Communication Systems and Networks*, 2014, pp. 1–8.

[34] A. Tang and A. Yip, "Collision avoidance timing analysis of DSRC-based vehicles," *Accid. Anal. Prev.*, vol. 42, no. 1, pp. 182–195, Jan. 2010.

[35] S. Kato, M. Hiltunen, K. Joshi, and R. Schlichting, "Enabling vehicular safety applications over LTE networks," in *Proc. Int. Conf. Connected Vehicles and Expo*, 2013, pp. 747–752.

[36] W. Hugemann. (2002). Driver reaction times in road traffic. [Online]. Available: http://www.calculamus.de/pdf/evu_2002_reaction_german.pdf

[37] Department for Transport (DfT)/Driver and Vehicle Standards Agency (DVSA). *The Official Highway Code: New Edition*. London, UK: Stationery Office Books, 2007.

[38] Euro NCAP. (2014, Mar. 14). Mercedes-Benz PRE-SAFE® Brake: For safer cars crash test safety rating. [Online]. Available: http://www.euroncap.com/rewards/mercedes_benz_pre_safe_brake.aspx

[39] Quartsoft. (2013, Sept. 26). How Honda's V2V and V2P technology uses smartphones to save lives? [Online]. Available: http://quartsoft.com/blog/201309/honda-v2v-v2p-technology-smartphones

[40] Pedestrian and Bicycle Information Center. (2015, Apr. 09). Pedestrian and Bicyclist Crash Statistics. [Online]. Available: http://www.pedbikeinfo.org/data/factsheet_crash.cfm

[41] M. A. Hoque, M. Siekkinen, J. K. Nurminen, S. Tarkoma, and M. Aalto, "Saving energy in mobile devices for on-demand multimedia streaming: A cross-layer approach," *ACM Trans. Multimed. Comput. Commun. Appl.*, vol. 10, no. 3, pp. 1–23, Apr. 2014.

[42] M. Behrisch, L. Bieker, J. Erdmann, and D. Krajzewicz, "SUMO: Simulation of urban MObility: An overview," in *Proc. 3rd Int. Conf. Advances in System Simulation*, 2011, pp. 55–60.

[43] Institute of Transportation Systems. German Aerospace Centre. (2015, Mar. 25). SUMO: Simulation of urban mobility. [Online]. Available: http://www.dlr.de/ts/en/desktopdefault.aspx/tabid-9883/16931_read-41000/

[44] S. Uppoor and M. Fiore, "Large-scale urban vehicular mobility for networking research," in *Proc. IEEE Vehicular Networking Conf.*, 2011, pp. 62–69.

[45] Institute of Transportation Systems at the German Aerospace Center (ITS-DLR). (2015, Mar. 25). Vehicular mobility trace of the city of Cologne, Germany. [Online]. Available: http://kolntrace.project.citi-lab.fr/

[46] OpenStreetMap. (2015, Mar. 25). [Online]. Available: http://www.openstreetmap.org/#map=5/51.500/-0.100

[47] M. B. Kjærgaard, J. Langdal, T. Godsk, and T. Toftkjær, "EnTracked," in *Proc. 7th Int. Conf. Mobile Systems, Applications, and Services*, 2009, p. 221.

[48] S. Bhattacharya, H. Blunck, M. B. Kjargaard, and P. Nurmi, "Robust and energy-efficient trajectory tracking for mobile devices," *IEEE Trans. Mob. Comput.*, vol. 14, no. 2, pp. 430–443, Feb. 2015.

[49] Z. Zhuang, K.-H. Kim, and J. P. Singh, "Improving energy efficiency of location sensing on smartphones," in *Proc. 8th Int. Conf. Mobile Systems, Applications, and Services*, 2010, p. 315.

[50] AWS. (2014, Sept. 5). Amazon Elastic Compute Cloud (EC2): Scalable Cloud Hosting. Available: http://aws.amazon.com/ec2/

[51] Server Density. (2014, Apr. 24). Network performance at AWS, Google, Rackspace and Softlayer. Available: https://blog.serverdensity.com/network-performance-aws-google-rackspace-softlayer/

[52] The Open Source Grid Engine Blog. (2014, Jan. 7). Enhanced Networking in the AWS Cloud: Part 2. Available: http://blogs.scalablelogic.com/2014/01/enhanced-networking-in-aws-cloud-part-2.html

[53] J. Markendahl and O. Makitalo, "A comparative study of deployment options, capacity and cost structure for macrocellular and femtocell networks," in *Proc. IEEE 21st Int. Symp. Personal, Indoor and Mobile Radio Communications Workshops*, 2010, pp. 145–150.

[54] CISCO. (2015, June 29). Network capacity for high-capacity suburban/urban small-cell network. [Online]. Available: http://docwiki.cisco.com/wiki/Wireless_Technologies#Table:_Network_Capacity_for_High-Capacity_Suburban.2FUrban_Small-Cell_Network