Measurement-based study of vehicle to Internet opportunistic communications

Ichrak Amdouni, Fethi Filali

Measurement-based Study and Analysis of Vehicle to Internet Opportunistic Communications

Ichrak Amdouni and Fethi Filali

EURECOM, Mobile Communications Department
2229 route des Crêtes, BP 193
F-06560 Sophia-Antipolis Cedex, France
E-mail: {amdouni, filali}@eurecom.fr

Abstract

Advances in wireless communication technologies as well as the low cost of wireless equipments have increased the ubiquity of network connectivity, illustrated by the large deployment of IEEE 802.11 access points (APs). Anytime, anywhere, users are being connected to networks and desire consequently more services on their mobile devices.

The Internet access from vehicles, in particular short lived connections, to roadside APs is a one of challenging services in this direction. Meanwhile, we wonder if those APs could offer Internet connectivity to vehicular networks especially for delay-tolerant applications. A preliminary requirement to make this possible is to provide the vehicle communication router with a list of available APs in its neighborhood.

In this paper, we introduce ScanCar, a layered measurement system for wireless infrastructure discovery, which is integrated in the vehicle's router to build and maintain a database of WIFI APs in the vehicle's range. Then, we report results of a real measurement study conducted on the data set ScanCar has provided through wardriving in different regions of the French Riviera. For instance, in Antibes Juan-les-Pins, which is a residential city, our findings show that the scan duration of the 2369 discovered APs, varies between 200 ms and 664 ms. Despite that among those APs only 8.2% were not protected, we found that on average, clients can potentially remain in at least one AP's coverage for 23.86 seconds which can be considered as a very promising result toward the development of services for vehicle to Internet opportunistic communication.

Keywords: WIFI access points, vehicle to Internet communication, wireless infrastructure discovery system, opportunistic communication, measurement-based study.

1 Introduction

In the recent few years there has been an explosive growth in the use of wireless LANs arising from the ease of installation and the low infrastructure cost of IEEE 802.11 networks. Wireless networking has become popular as many metropolitan areas in the world are equipped with an infrastructure to provide home networking and Internet access in hot spots (e.g., airports, cafes, restaurants) [1, 2].

According to Jupiter research, 65% of on-line house-holds have installed WIFI APs at home [3]. One could imagine that in areas with reasonably high density, such APs could provide Internet access to other users. This trend is likely to continue for the foreseeable future. Moreover, simple users' APs are often open or even underutilized, making possible the idea of sharing the Internet connectivity with other users. In this scenario, many legal, business and privacy issues arise, that is why different economic models try to legitimate and organize such phenomenon. A recent promising model is fon's [4], where users open their access points to all fon users that also open theirs. This model is becoming popular, and is now supported by established Internet Service Providers (ISP) such as British Telecom, Time Warner and Neuf. In addition to privately operated APs, emerging wireless mesh networks such as those operated by T-Mobile and others offer coverage in many parts of the world. It is exactly this tendency that motivates us to conduct this research. Regardless the aforementioned issues, how much could offering an unplanned Internet connectivity, or communication with any other network, to people in cars be possible and which performance this connectivity would have under delay tolerant applications?

This question has been addressed in several projects [5, 17, 19, 21, 20, 22, 23]. The main focus of these studies is the suitability of IEEE 802.11 standards for Vehicle to Internet (V2I) communications. Authors studied the distribution of duration of various connection establishment phases (association, IP address acquisition, etc.), the data throughput, the influence of the vehicle speed, the wireless conditions and the backhaul metrics and the different delays influencing the connectivity duration.

In this paper, we start by studying the environment that is destined to bear the opportunistic V2I communications. For this purpose, we have designed and developed ScanCar, a set of programs that embed on a computer equipping the car can periodically scan the environment and build a database about all APs from which the wireless interface can receive a frame during the course of the car's owner normal driving. We then process the collected information in order to study the characteristics of APs encountered in the road toward the development of an opportunistic communication architecture for V2I communication.

The remainder of this paper is organized as follows. In Section 2, we present a summary of our contributions. Section 3 is an overview about ScanCar: we recall its main objective and we expose some design choices that we have adopted. Section 4 is a description of the ScanCar communication stack and operations. In Section 5, we analyze the measurement results. Section 6 concludes the paper and outlines future directions.

2 Summary of the contributions

The main objective of ScanCar is to determine the characteristics of APs encountered in the road toward the development of an opportunistic communication architecture for Vehicle to Internet communication. These APs are stored in a local wireless infrastructure database that may be used to select the best one of them each time an opportunistic communication to the Internet is required. More precisely, ScanCar achieves its objectives as follows:

1. The collected data are recorded in a local database so that the car can use them without triggering scan operations each time it wants to find a usable (not protected) AP. In fact, Internet connectivity for moving vehicles can't exceed an intermittent connection due to travel speed [5] and the short range of IEEE 802.11. Considering the duration for which the vehicle is within an AP's coverage area, the shorter is the scan delay, the longer is the connectivity duration. Intuitively, the maintenance of this database should be studied, since the wireless environment can change. Thus, and in order to ensure that the collected information are up to date, ScanCar investigates this issue in the second point.

2. Using a GPS receiver, ScanCar can locate the vehicle it is running on when detecting each AP. This information can be exploited to build a map of APs in a specific region. This map can be then enriched by the time stamp of the AP detection (one of the details logged by ScanCar as we are going to explain later). Thus, when passing by a region, the system has a vision about the wireless infrastructure that it would opportunistically encounter. The information about the wireless infrastructure card can be exchanged between vehicles so that each one's local cache is enriched or updated. The scenario where a vehicle doesn't have much information about the wireless infrastructure in an area (a region that it visits for the first time for example) would best illustrate this utility.

Apart from promising cars an Internet connectivity, ScanCar provides a reusable platform that can be used to build many mobile sensing applications. A vehicle running ScanCar can be used as a mobile sensor collecting data about the wireless infrastructure in a given region. Besides, this tool has the advantage to be extensible with its layered architecture. Given the wireless database ScanCar has built, an infrastructure selection algorithm can be hold to select the best AP based on predefined criteria. This AP can be then the subject of association and connection procedures.

Scancar's design and its features presented in this paper have been fully implemented. The system has been tested in a car running at an average speed of 45 km/h. The vehicle was equipped by a GPS receiver, a 802.11g RangeMax card, and a gain antenna. The results reported in this paper are from the real world operation of this testbed. We have performed drives in different areas of Sophia Antipolis and Antibes Juan-les-Pins. We have found that in average, the vehicle still receiving beacons from an AP for about 23.86 seconds which constitutes a real opportunity for Internet connection offered by this AP. Our results show also that the choice of the measurement field is very important as it defines the density of APs and consequently the duration of scan that the system would spend looking for usable ones. Finally, we found that the scan duration varies between 200 ms and 664 ms.

In the next section, we will present an overall about ScanCar's architecture and AP-related information.

3 Overall architecture

3.1 Objective

The main motivation behind ScanCar is to allow people in vehicles to communicate with each other and to access to Internet via hotspots encountered opportunistically in the road. One first step to make this feasible is to offer to the user in motion the possibility to know which usable

APs are surrounding him, whenever he drives. ScanCar is a set of interacting software modules which run in a specific device in the vehicle to build and maintain a database of the APs in the car's range. ScanCar collects some useful information about discovered APs and records them in a local database. This information is then processed in order to study the Internet connection possibilities in the measurement field. This study focuses on the scan procedure (delay of scan, channel occupancy, distribution of APs' signal quality and the distribution of APs' encounter time).

In addition to the aforementioned functionality, the design of ScanCar has been done in a way which facilitates the incorporation of:

- A vehicle to AP connection procedure;
- An intelligent infrastructure selection algorithm for vehicular networks.

Thus, the vehicle can associate and connect to one best AP. These features are developed and experimentally evaluated using the above described functionality of ScanCar, but are not the scope of this paper which aims at presenting the ScanCar functionality and analyzing the obtained measurement results.

In the following, we describe our design choices setting that we adopted to achieve ScanCar goals. These choices concern the set of information to collect about each AP and the scan approach.

3.2 AP-related information

For each AP, we collected the following information:

1. Time stamp: Date of the detection of the AP;
2. GPS coordinates including: Altitude, longitude and latitude of the vehicle when discovering the AP;
3. Travel speed of the vehicle when discovering the AP;
4. BSSID: Unique identifier of 6 Bytes. In an infrastructure mode, it represents the MAC address of the AP;
5. ESSID: 32 characters identifier, it represents the network name;
6. Frequency channel number;
7. Encryption key: Set to "on" or "off" depending on whether the AP applies or not authentication mechanisms;
8. Signal quality;
9. Noise level;
10. Signal level.

This choice was not done at random since we have been guided by our functional requirements and our perspectives. As mentioned, the vehicle can proceed to an association and connection procedure once the ScanCar module provides it with the wireless infrastructure database. BSSID and ESSID information are the association procedure parameters, while the frequency channel number would accelerate this procedure if this AP is selected. Besides, in an unplanned wireless network context, discovering the security setting of an AP is a key information. Connecting to the Internet from an AP encountered opportunistically in the roadside without any previous authentication, suggests that this AP is not protected. Using the localization information, each car can have a visibility about usable APs in any region it has already driven in. It can also, exchange this information with other cars that, under an infrastructure announcement protocol, desire to share their local database, especially in a first visited area where all existing APs are not known.

To collect the AP-related information, ScanCar proceeds to an active scan as explained in the following.

3.3 Scan approach

In IEEE 802.11, there are two scan approaches: passive and active scan [7]:

- In the passive scanning, the mobile unit switches for a candidate channel and listens for periodic beacon packets generated by nearby APs to announce their presence (typically sent 10 times a second). This is done for all channels in sequence. The main drawback of this method is to optionally tune the time to listen to each channel. This time must be longer than the beacon period, which is unknown for the station until the first beacon is received. Moreover, the station has to wait for the whole beacon period because several APs of different WLANs can operate in the same channel. To reduce this delay, most IEEE 802.11 implementations handle active scanning [6].

- In the active scanning the mobile unit has to probe the AP, ESSID field is set to "ANY" when it is not looking for a particular AP. Each AP that activates broadcast option will answer this request via a probe response. The unit does not have to wait for beacons from APs. The issue, however, is that active scanning produces additional overhead in the network because of the transmission of probe and corresponding response frames.
  An active scan is mainly regulated by two parameters [8]:

  - MinChannelTime: It represents the amount of time to wait for the first response before declaring the channel empty (no APs).
  - MaxChannelTime: It represents the amount of time to wait for to collect potential additional probe responses from other APs. This value is meant to be configured based on an estimate of the number of overlapping APs and the load on the channel.

  Both timers are measured in Time Units (TU), which the IEEE standard defines to be 1024 microseconds. Exact values of these timers are not given in this standard, but previous empirical studies suggest values of roughly 11 ms in [6] and 40 ms in [9] for MaxChannelTime and values of 1 ms in [6], 7 ms in [9] for MinChannelTime.

In our work, ScanCar entrusts the Madwifi driver [10] to **actively** build the list of APs from which the car can receive a frame. In the following, we will detail the scan procedure.

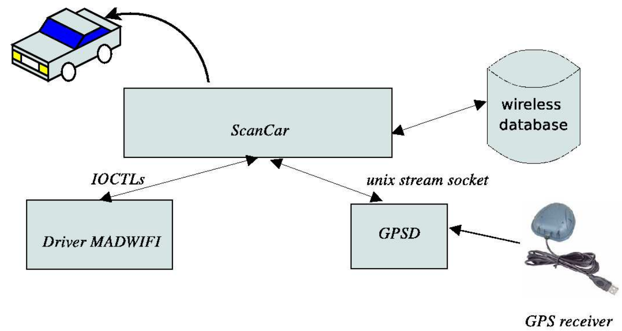


Figure 1: **The ScanCar communication stack**

4 ScanCar components and operations

ScanCar is a set of programs written in C. The script shell language was used to process collected data. We describe in the following subsections the main components and operations of this tool.

4.1 ScanCar components

Figure 1 illustrates ScanCar communication stack.

ScanCar has two interfaces. To collect data about all APs from which the car can receive a frame, it communicates in the kernel space with the Madwifi driver via ioctl calls [12]. The localization information of encountered APs are collected via the GPS daemon (GPSd) [11]. ScanCar-GPSd communications are performed in the user space via a unix stream socket.

4.2 Main operations of ScanCar

As shown in Figure 2, ScanCar loops through scanning the environment looking for APs and recording useful information about them.

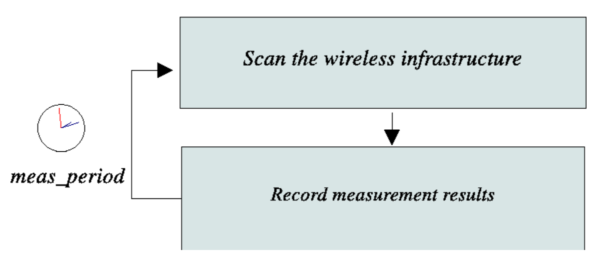


Figure 2: **ScanCar cycle**

4.2.1 Scan the wireless infrastructure

ScanCar utilizes its second interface to connect to the kernel to extract scan information via ioctls. The usual method in Unix to set and get parameters from a network device is through ioctl. Ioctl requests are usually operations performed on a file descriptor, but they also apply on network sockets. The ioctl is a kernel system call. Its arguments define the operations to be done, the parameters of these operations and the device they apply to. ScanCar uses some ioctls that have been defined under wireless extensions tools project [13]. To perform scan operations and collect data from the driver, ScanCar uses two ioctl' requests:

- SIOCSIWSCAN: Via this ioctl, ScanCar triggers active scan. The driver alternates consecutively between channels. *MinChannelTime* and *MaxChannelTime* are set respectively to 5 ms and 50 ms.
- SIOCGIWSCAN: Via this ioctl, the application is able to extract data returned by Madwifi about visible APs that have been discovered via the first ioctl.

GPS coordinates are collected via a GPS receiver with interaction with GPSd. Before launching ScanCar, we have run the daemon GPSd. This daemon probes each second the GPS receiver (which is the highest frequency of measurement supported by one GPS receiver [14]) via the command PAVD (position (P) including latitude and longitude, altitude (A), date (D) and car speed (V)) to return car position, velocity, and date.

Before starting each scan operation, ScanCar checks its connection to GPSd so that it reconnects to it if the connection was interrupted.

As shown in Figure 3, scan operations are periodically performed each $meas\_period$. The setting of $meas\_period$ depends on multiple factors:

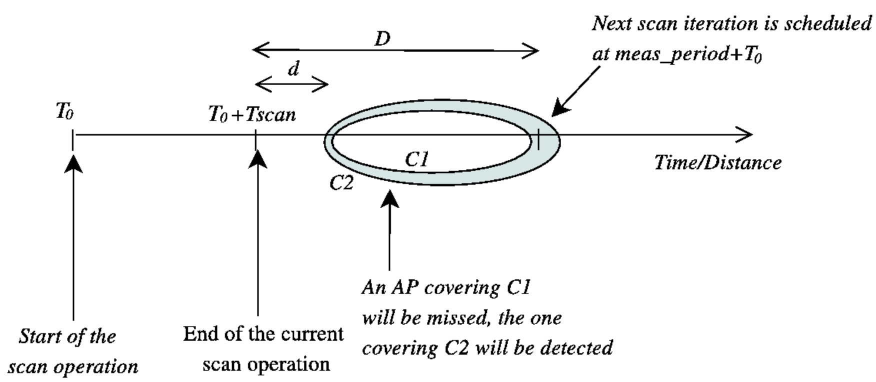


Figure 3: **The choice of the scan periodicity depends on several factors**

- First, ScanCar has to ensure the end of the last scan operation before triggering another one, regarding the scan duration itself. In our measurements, we have set the parameter *MaxChannelTime* to 50 ms, so at the presence of 11 channels, the upper bound of the scan period is 550 ms. To have the total scan overhead, we have to add to the time spent in scanning, an important parameter which is the cumulative channel switching delay. This latter overhead is a characteristic of the network interface design. It is composed of the time to switch to a new frequency, resynchronize and start demodulating packets. [8] has reported values of 19 ms for Intersil Prism2-based NICs and 5 ms for Atheros 5212-based NICs. This cost is per channel, so it adds considerable delay to the overall scanning process. Let $Tscan$ be the total scanning delay (AP detection and channel switching). In order to avoid the preemption of ScanCar from scanning all channels, $meas\_period$ should be set with the constraint (1).

$$meas\_period \geq Tscan \tag{1}$$

- Second, the *meas_period* should be short enough so that the moving vehicle wouldn't miss some APs after $Tscan$. To illustrate this constraint, we use the following notation:
  - V*:* The vehicle speed;
  - D*:* The distance the vehicle would move in $meas\_period - Tscan$ at the speed V*;*
  - C*:* The theoretical coverage area of an AP;
  - d*:* At the end of the present scan operation, the vehicle enters a region with no APs, or where all APs are configured on a different channel, let d be the distance the vehicle runs in this region.

  Using this notation, D is given by (2).

$$D = V * (meas\_period - Tscan) \quad (2)$$

  We consider now the following scenario. After having scanned a region during $Tscan$ and running d distance the vehicle encounters an AP. As illustrated in Figure 3, if this AP coverages the area $C1$, it would not be detected, on the other hand the AP covering $C2$ will be detected. To conclude, the AP having the theoretical coverage area C will not be detected unless

$$D - d < C \quad (3)$$

  We conclude from (3) and (1) that the *meas_period* timer should verify (4).

$$Tscan \leq meas\_period < Tscan + \frac{C + d}{V} \quad (4)$$

The vehicle should then periodically adjust *meas_period* to mach this equation.

4.2.2 Record measurement results

ScanCar performs recording data in a local database in specific format which can be easily post-processed. A record from such data set has the following format:

—>number of the scan operation:total number of discovered APs for this scan operation

Each record is then followed by these information:

time stamp: altitude (in feets): longitude (in degrees): latitude (in degrees): vehicle speed (kph): essid: bssid: channel number: encryption key (on/off): signal quality: noise level (dBm): signal level (dBm)

The analysis of the collected data set is the subject of the next section.

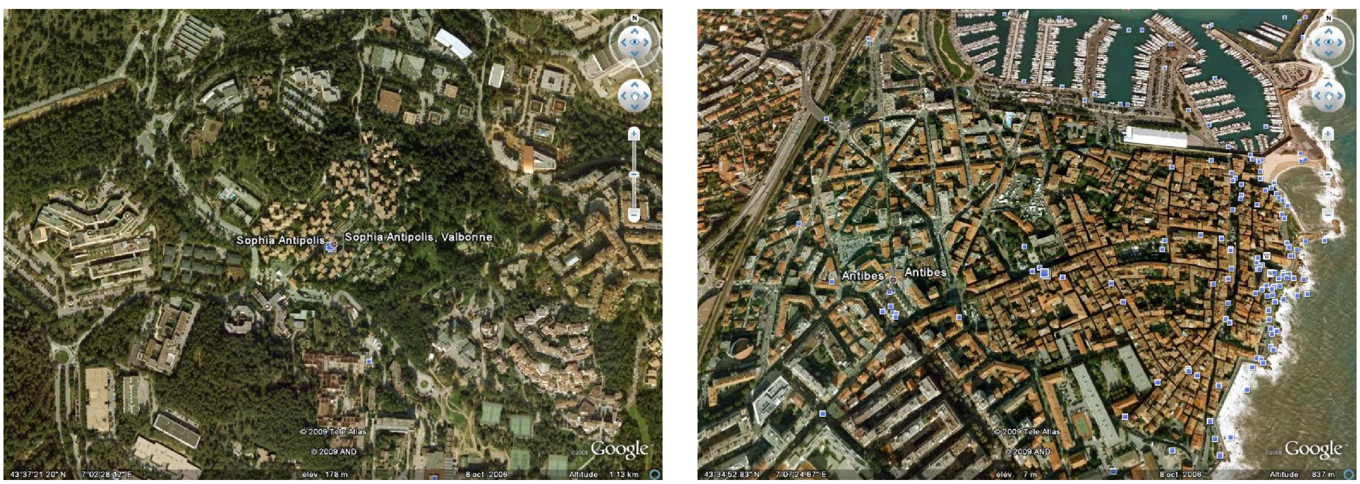
5 Results analysis

5.1 Experimental setting

Before describing our measurement study findings, we briefly summarize the characteristics of our experimental testbed and the equipment used.

We have conducted a measurement study using a car running at an average speed of 45km/h. The car was instrumented by a computer running Ubuntu 7.10 with an intel Core Duo 2,6 Ghz processor, 4 GB RAM and 100 GB hard disk. A Haicom HI-204E-USB GPS receiver was connected to USB port of the PC and fixed in the roof of the car. The computer has an IEEE 802.11g RangeMax wireless PC card (WPN511) and a wireless external gain antenna. The embedded computer draws power from the car, it launches ScanCar and GPSd.

Data reported in this paper derived from 2 driving hours in the regions of Sophia-Antipolis and Antibes Juan-les-Pins (see Figure 4). We aimed to generalize results and study the influence of buildings density and type, as well as road traffic conditions. The first field, Sophia Antipolis science park, is not very populated, it is a suburban area where located essentially sparse enterprises, schools and laboratories. According to [15], there are 1300 companies with 30000 employees, while Antibes Juan-les-Pins is a residential city (population density in 2006: 2863/km2 [16]), hence road traffic is more important.



(a) Sophia Antipolis

(b) Antibes Juan-les-Pins

Figure 4: **Field study, different regions of the French Riviera**

Note that we had no control over the vehicle mobility, car maintains a normal driving patterns and schedule during the course of our study. All the data collected and analyzed are therefore subject to the constraints of real traffic and network conditions.

Different experimental conditions and some results are described in Table 1.

After describing the experimental setting of our measurements, we report now the most pertinent results.

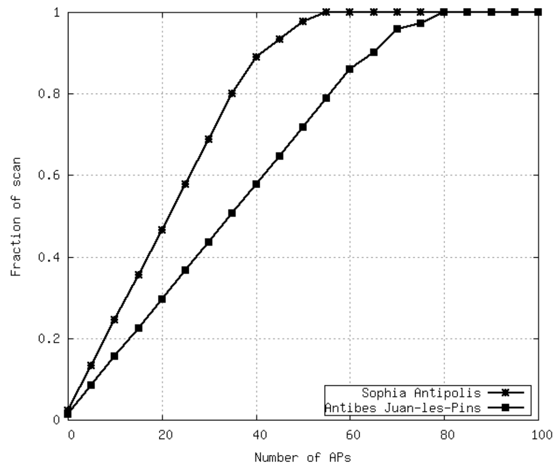


Figure 5: **CDF of the discovered access points over non empty scan operations**

5.2 Discovered access points

Figure 5 shows the number of discovered APs per non empty AP scans (ie. any scan that found at least one AP). We note that even APs with a small signal quality were discovered as shown in Figure 6.

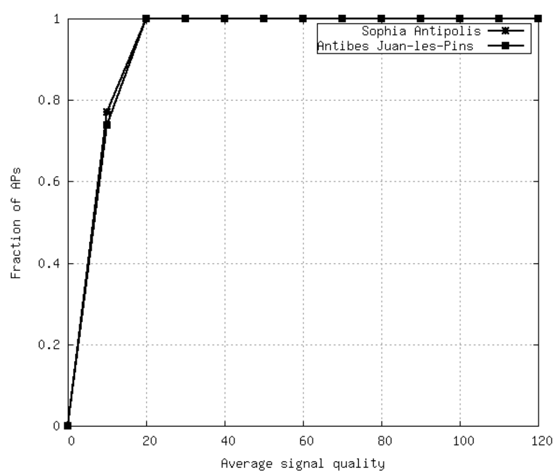


Figure 6: **CDF of average signal quality of discovered access points**

As depicted in Figure 5, almost 50% of scan operations detected a number of APs exceeding 35 in Antibes Juan-les-Pins and less than 25 in Sophia Antipolis. Difference between the discovered AP number in the two measurement fields is predictable, it is due to the low population density of Sophia Antipolis. This is confirmed by the number of empty scan, in Sophia Antipolis, 186 operations detected nothing, it is essentially the mountainous region where nothing than forests exists. In both areas, the car is often in the vicinity of multiple APs. Hence, an infrastructure selection algorithm is necessary. At the same time, this number still hold promise to offer Internet connectivity to mobile car users.

Nevertheless, only 8.2% of Antibes' APs are not protected, which means that the security information has to be carefully taken into account in the AP selection procedure. The remaining percentage of APs, those which are protected, represents significant missed connectivity opportunities. That's why it is important to provide users and service providers with incentives for opening their APs to other users in order to multiply Internet connection opportunities.

Over all drives, we encountered 27866 APs, among them 3066 are distinct, that is the scan operations report the same MAC address more than once. This suggests an important visibility duration which is confirmed by the average visibility CDF in Figure 7. Here we define the visibility duration as the time between the instant the driver has received the first frame from an AP and the instant of the reception of the last frame. As defined, the visibility duration is the upper bound of the potential connectivity duration because it does not include the time it might take for a client to actually achieve end to end connectivity (vehicle to AP association, vehicle's IP address configuration, etc). At an average speed of 45km/h, our results show that the minimum and maximum average visibility durations are respectively 4.28 s and 84.24 s. The medium visibility duration was reported to be 10 s in [18] and 6 s in [20] at 120km/h. These relatively large visibility durations confirm that a relative small number of APs could coverage a large area. It represents a usable connection opportunity for moving clients especially that the connection duration was reported to be 13 seconds in [17]. That said, it was noticed in [21] that as the vehicle's speed increases, the distance for which it is in range decreases, from 272 m at 5m/h to 143 m at 75m/h. This finding can explain the difference between our results and previous published results.

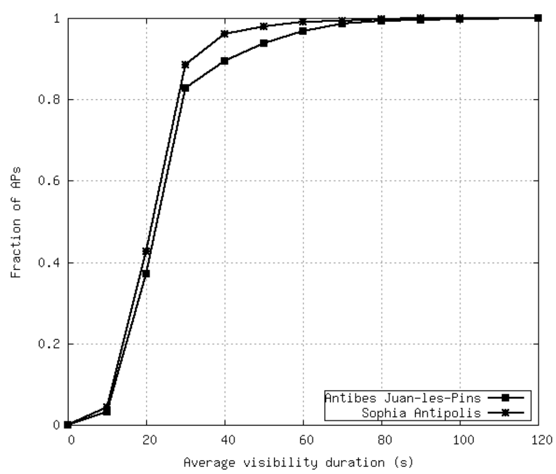


Figure 7: **CDF of the average visibility duration**

5.3 Channel occupancy

As it is shown in Figure 8, APs distribution over channels (as in our data set from Antibes Juan-les-Pins and Sophia Antipolis indifferently) is non-uniform, biased towards some channels: 1 (31%) , 6 (17%), 11 (23%) (results from driving in Antibes Juan-les-Pins). In IEEE 802.11g these channels are not overlapped, free of co-channel interference, users or device manufacturers take into account this information and carefully configure their APs. It can be seen from the graph that it makes sense to scan only in channels 1, 6 and 11 to obtain significant amount of data and reduce the scan duration (which is in-line with observations made in [19]), or guarantee larger listening periods to those channels in the scan procedure.

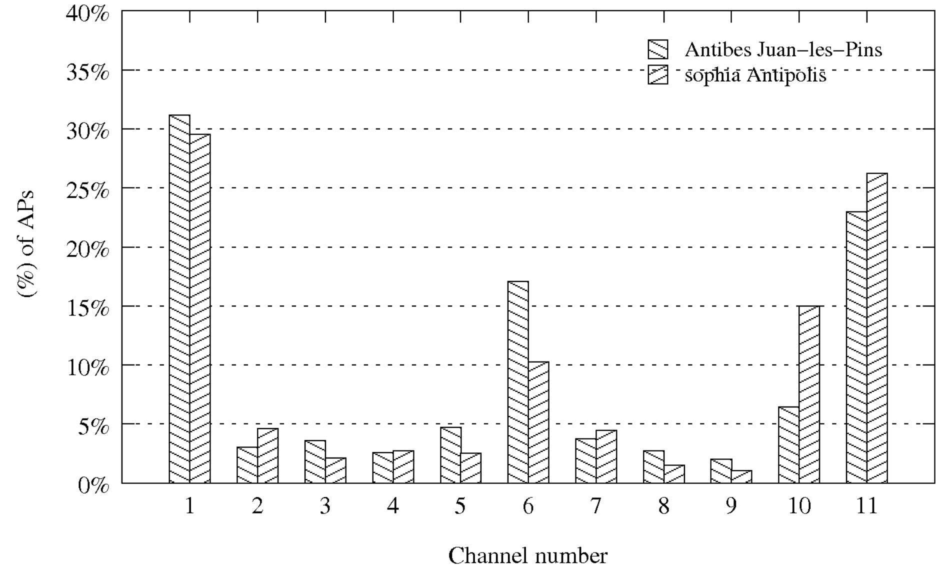


Figure 8: **Distribution of APs over channels**

5.4 Scan duration

The CDF in Figure 9 shows the distribution of scan duration. Scan delay varies from 200 ms and 664 ms (results from driving in Antibes Juan-les-Pins), almost 90% of scan operations last less than 300 ms. Those numbers are consistent with previously published benchmarks [17]. Note that this scan duration depends on the card and the antenna's performance [6].

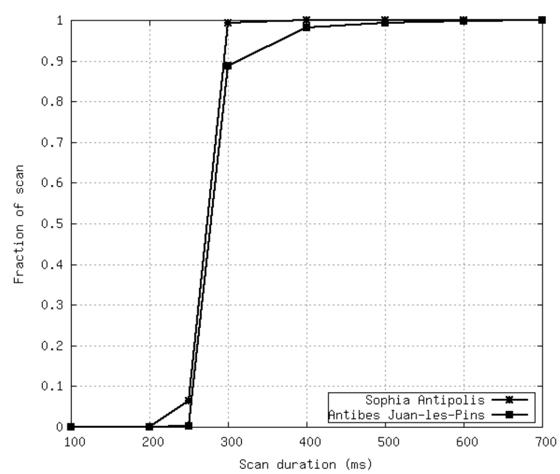


Figure 9: **CDF of scan duration**

6 Conclusions and future works

The main observation that encourages our work is the impressive penetration of WIFI-based networks illustrated by the large deployment of WIFI APs in many cities in the world. This deployment constitutes an opportunity for users who want cheap and high-quality Internet access from moving vehicles especially that cellular networks tend to be expensive.

This paper described ScanCar, a system for discovering wireless infrastructure for in-car users. ScanCar collects useful information about APs encountered opportunistically during drives in order to study the Internet connectivity possibilities for vehicles in a given region. We used a car equipped by a single radio, a GPS receiver and a WIFI gain antenna and running GPS daemon. Through drives performed in different areas in the French Riviera, ScanCar found that the medium scan duration was 432 ms and that APs' average visibility duration was 23.86 seconds (as in our experiments in Antibes Juan-les-Pins). These results show that APs cover a large area, and that the scan delays seem to be not penalizing the connection establishment process. With the proliferation of municipal, commercial and public WIFI initiatives, ScanCar is going to be efficient for many applications particularly ones that can tolerate some intermittent connectivity including mobile sensor networks, e-mail, messaging applications, web access, etc.

Giving the data it collects, ScanCar will be able to select the best AP according some criteria and proceed to association and connection procedures. After that, we can define parameters that influence the Internet connection for delay tolerant applications as far as the connection procedure is concerned. Our system will be extended to support IPv6 and ITS architectures (ETSI TC ITS, CALM, WAVE). We plan also to design and develop an infrastructure announcement protocol to allow vehicles to share their local infrastructure database.

References

[1] City-wide wi-fi rolls out in UK
http://news.bbc.co.uk/2/hi/technology/4578114.stm

[2] Cities Unleash Free Wi-Fi
http://www.wired.com/gadgets/wireless/news/2005/10/68999

[3] http://www.jupitermedia.com/corporate/releases/05.06.02-newjupresearch.html

[4] http://www.fon.com

[5] R. Mahajan, J. Zahorjan, and B. Zill, "Understanding Wifi-based connectivity from moving vehicles", in Proc. of IMC, October 2007.

[6] H. Velayos and G. Karlsson, "Techniques to reduce IEEE 802.11b MAC layer handover time". Technical Report TRITA-IMIT-LCN R 03:02, KunglTekniska Hogskolen, Stockholm, Sweden, April 2003.

[7] A Technical Tutorial on the IEEE 802.11 Protocol By Pablo Brenner Director of Engineering
http://sss-mag.com/pdf/802_11tut.pdf

[8] I. Ramani and S. Savage, "SyncScan: Practical fast hand-off for 802.11 infrastructure networks", in Proc. of INFOCOM '05: The 23rd Conference of the IEEE Communications Society, March 2005.

[9] A. Mishra, M. Shin, and W. Arbaugh, "An Empirical Analysis of the IEEE 802.11 MAC Layer Handoff Process", in Proc. of ACM Computer Communications Review, vol. 33, no. 2, April 2003.

[10] Madwifi driver
http://madwifi-project.org/

[11] GPS Daemon
http://gpsd.berlios.de/

[12] IOCTL function
http://www.opengroup.org/onlinepubs/009695399/functions/ioctl.html

[13] Wireless Extensions for Linux
http://www.hpl.hp.com/personal/Jean_Tourrilhes/Linux/Linux.Wireless.Extensions.html

[14] D. Hadaller, S. Keshav, and T. Brecht, and S. agarwal. "Vehicular opportunistic communication under the microscope", in Proc. of Mobile Systems, Applications, and Services (MobiSys'07), June 2007.

[15] http://www.sophia-antipolis.org/

[16] http://fr.wikipedia.org/wiki/Antibes#G.C3.A9ographie

[17] V. Bychovsky, B. Hull, A. Miu, H. Balakrisnan, and S. Madden, "A measurement Study Of vehicular Internet Access Using In Situ Wi-Fi Networks", in Proc. Mobicom'06, September 2006.

[18] J. Eriksson, H. Balakrishnan, and S. Madden, "Cabernet: Vehicular Content Delivery Using WiFi", in Proc. of the 14th ACM MOBICOM, September 2008.

[19] V. Navda, A. Prabhu Subramanian, K. Dhanasekaran, A. Timm-Giel, S. R. Das, "MobiSteer: using steerable beam directional antenna for vehicular network access", in Proc of MobiSys'2007, June 2007.

[20] J. Ott and D. Kutscher, "Drive-thru Internet: IEEE 802.11b for Automobile Users", in Proc. of INFOCOM'04, March 2004.

[21] R. Gass, J. Scott, and C. Diot, "Measurements of in motion 802.11 networking", in Proc. of Workshop on Mobile Computing System and Applications (WMCSA), April 2006.

[22] A. Balasubramanian, Y. Zhou, W. Bruce Croft, B. Neil Levine and A. Venkataramani, "Web search from a bus", in Proc. of the second ACM workshop on Challenged networks (CHANTS '07), September 2007.

[23] T. Arnold, W. Lloyd Jing Zhao and G. Cao, "IP Address Passing for VANETs", in Proc. of the 2008 Sixth Annual IEEE International Conference on Pervasive Computing and Communications (PerCom), May 2008.

Parameter	Antibes Juan-les-Pins	Sophia Antipolis
Number of cars	1	1
Duration of driving and measurement	1 hour	1 hour
Average car speed	45km/h	45km/h
Date	October 19th, 2008	January 27th, 2009
Total number of scan operations	919	909
Number of empty measurements (no scanned APs)	1	186
Number of detected APs over all scans	21403	6463
Number of distinct APs	2369	697
Number of non-protected APs	192 (8,2%)	192 (27,55%)
Number of protected APs	2177 (91,8%)	505 (72,45%)

Table 1: **Experimental setting and some results**