On the Selection of Scanning Parameters in IEEE 802.11 Networks

Nicolas Montavont*, Andrés Arcia-Moret† and German Castignani‡
*IMT/Telecom Bretagne, Irisa, nicolas.montavont@telecom-bretagne.eu
†T/ICT4D Lab, International Centre for Theoretical Physics, aarcia_m@ictp.it
‡University of Luxembourg / SnT, german.castignani@uni.lu

***Abstract*—In this paper, we study the IEEE 802.11 discovery process needed for handovers and propose an adaptive scanning strategy based on application requirements. The scanning process consists in actively probing the radio channels to gather access points information. We consider a well decoupled situation in which the scanning latency, the scanning failure rate and the number of discovered access points define the scanning performance. We model these scanning metrics by analytical expressions to represent the performance trade-off, i.e., finding the largest number of access points with a minimum latency. We present a novel approach based on a multi-objective optimisation approach to obtain the optimal number of channels to scan, the optimal channel sequence and its correspondent scanning timers. Finally, we compare one fixed and two adaptive scanning approaches by means of simulations. We show that our adaptive scanning strategies better manage the performance trade-off and allow different application profiles to match with specific scanning latency.**

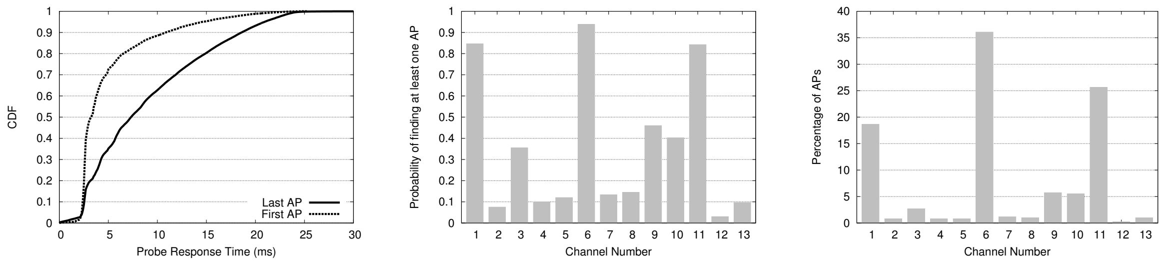
***Index Terms*—IEEE 802.11, Handover, Scanning**

I. Introduction

IEEE 802.11 is currently one of the most popular technologies providing broadband wireless access. Its low price, high data-rate and unlicensed band usage incite deployments in different environments (e.g., residential and metropolitan accesses). The Access Points (AP) deployment is thus decentralized, which leads to variable and unpredictable link performance. Due to the limited AP coverage, mobile users may need to roam between several APs to maintain their network connection. The process of changing AP, known as handover, should impact as little as possible on the client running-applications, since no data can be sent during a variable disconnection period.

A handover consists in discovering and selecting candidate APs, then a mobile station (MS) authenticates and associates with the chosen AP. AP discovery is performed through the *scanning process*, in which an MS sends a Probe Request management frame and waits for a Probe Response on each channel. This waiting time is managed by two timers in the scanning process, namely MinChannelTime ($MinCT$) and MaxChannelTime ($MaxCT$). If no Probe Response is received before $MinCT$ expires (because no AP is operating on the channel, the Probe Response is sent after the timer expiration, or there is a transmission error), the MS switches to the next channel and sends a new Probe Request. Otherwise, if at least one Probe Response was received, the MS waits for a longer timer, namely $MaxCT$, to receive more responses from other APs operating in the same channel.

The 802.11 standard does not specify the timer values nor the order in which the channels should be scanned (i.e., the channel sequence), while these parameters greatly impact the scanning performance [1]. Indeed there is a trade-off between keeping the latency short to minimize the impact on applications, and how many APs the MS is able to discover. In order to address this trade-off, we define the scanning performance by three metrics. The *scanning latency* is the elapsed time for scanning the whole set of channels. The *failure rate* is the probability of not finding any AP after completing the scanning. Finally, the *discovery rate* is the fraction of discovered APs over the total number of available APs. In our previous work [1], we observed that the level of congestion and the radio link condition affect Probe Responses delay. From this study we observed that there is not a single fixed optimal pair ($MinCT, MaxCT$) that always gives the best scanning performance in all deployments. The channel sequence is important, because an MS may stop scanning as soon as it finds an AP, as reported by Shin et al. [2]. So the sooner the discovery of an AP, the fewer the number of scanned channels and therefore the lowest the scanning latency. In this paper, we investigate the trade-off between these three performance metrics and determine how the scanning parameters ($MinCT, MaxCT$) and the channel sequence can be tuned to better fit the application's needs. Our contribution is to provide a method for applications that require adaptation, by imposing restrictions on the scanning performance. So, we use a multi-objective algorithm to find optimal scanning parameters to meet the application



(a) AP Response Distribution (b) Channel Use Probability (c) AP Channel Distribution

Fig. 1. Measurement study for model variables

needs and to optimize the scanning performance.

Section II proposes an analytical model for the scanning performance metrics and Section III proposes to resolve the scanning trade-off by a multi-objective optimization problem and compares three scanning strategies. In Section IV we present the related work on the scanning process. Finally, Section V concludes the paper.

II. Analytical model for scanning metrics

A. Model Definition

We propose a probabilistic approach to model the 802.11 scanning process. In this model we suppose that, when probing a channel $i \in [1, n]$, we are using a certain pair $(MinCT_i, MaxCT_i)$ that eventually will carry to one of the following events. If a Probe Response is received on channel i (A_i), then we have discovered at least one AP on this channel. Otherwise, if no Probe Response is received ($\bar{A}_i$), either there is no AP on the channel ($\bar{B}_i$) or there is at least one (B_i) that the MS was not able to discover. So, scanning channel i is a Bernoulli process with probability of success $p_i = P(B_i \cap A_i)$. Since events A_i and B_i are not independent, we have $P(B_i \cap A_i) = P(A_i|B_i) \cdot P(B_i)$. Let T_i be the delay of the first Probe Response on channel i. An AP is discovered if the MS receives a Probe Response before the expiration of $MinCT_i$, so $P(A_i|B_i) = P(T_i \leq MinCT_i)$. Finally, the Bernoulli process parameter for channel i is $p_i = P(T_i \leq MinCT_i) \cdot P(B_i)$. The following expressions model the scanning performance metrics:

$$L = \sum_{i=1}^{n}(MinCT_i + p_i \cdot MaxCT_i) \quad (1)$$

$$F = \prod_{i=1}^{n}(1 - p_i) \quad (2)$$

$$D = \sum_{i=1}^{n}(\rho_i \cdot \delta_i) \quad (3)$$

In Eq. 1, we express the expected value of the *scanning latency* (L) in terms of the timers ($MinCT_i$, $MaxCT_i$), p_i, and n the number of scanned channels. Recall that the MS waits for $MaxCT_i$ after $MinCT_i$ if at least one Probe Response is received. The *failure rate* F in Eq. 2 is expressed as the probability of n unsuccessful Bernoulli trials. The *discovery rate* is estimated by D in Eq. 3 as the probability of discovering all APs. ρ_i is the relative number of available APs on channel i. δ_i is the probability of finding all available APs on channel i using $(MinCT_i, MaxCT_i)$, derived from the probability distribution of the last probe response delay.

B. Model Instantiation

p_i, ρ_i and δ_i are statistical estimations that depend on the AP deployment. While some 802.11 deployments may be specific, common patterns are found across the world, such as the intensive usage of the three non-overlapping channels 1-6-11 [3]. In order to estimate p_i, ρ_i and δ_i we used empirical data from a measurement campaign we conducted to analyse urban AP deployments [4], [5]. We obtained traces from more than 6000 APs using Linux and Android-based MS performing scanning with long timers, ($50ms$, $200ms$) for $(MinCT, MaxCT)$, to maximize the number of discovered APs. Note that for an effective implementation of our approach, an exchange system with the cloud can assist MS to exchange those statistics in real time, as also suggested by Eriksson et al. [6].

Fig. 1(a) presents the first Probe Response time and last Probe Response time CDF. In a scanning, after an MS sends a Probe Request, several APs may schedule the transmission of a Probe Response. They are likely to access the medium at different time, and thus, an MS should receive all these answers one by one. We define first probe response time as the delay an MS observes between the probe request and the first probe response it receives. We define last probe response time as the delay between the probe request sent by an MS, and the last probe response received by an MS.

In order to obtain p_i, we compute the empirical first Probe Response delay distribution on each channel (Fig. 1(a)), so we can calculate the term $P(T_i \leq MinCT)$.

For $P(B_i)$, we use Fig. 1(b), showing the probability that at least one AP is deployed on a given channel. For δ_i we use the last Probe Response delay distribution of Fig. 1(a) and for ρ_i, we use Fig. 1(c) showing the average proportion of APs operating on each channel.

III. Scanning performance evaluation

A. Scanning algorithm

An ideal scanning algorithm seeks to discover the maximum number of APs in the shortest period of time. As in legacy implementations, an MS scans the full set of channels spending a constant amount of time on each of them. Clearly, this approach will lead to a suboptimal scanning latency or a suboptimal discovery rate. As we have presented in Section II, the scanning algorithm must find the best trade-off between the time spent to probe for APs (i.e., the scanning latency), the number of discovered APs (i.e., the discovery rate) and the discovery of at least one candidate AP after full scanning (i.e., the failure rate). Because these objectives can not be met simultaneously, we propose to fix one of these objectives and to leave to the applications the selection of the scanning behavior. For example, in some scenarios it is preferable to find the best candidate AP among all surrounding APs without time restrictions. In this case, the discovery rate is more important than the scanning latency. On the other hand, for real time applications such as VoIP, the scanning latency should be as low as possible to match the user expected quality of service.

In order to set up this adaptive behavior, we propose to instantiate the scanning variables, namely $MinCT$, $MaxCT$ and the number and order of scanned channels, according to a particular application profile. This strategy allows spending more time on certain channels to maximize the probability of finding a candidate AP.

We model the scanning parameters selection using a multi-objective optimization problem. We aim at finding the set of scanning parameters in the decision space (i.e., the channel sequence and the timers) that minimize L and F and maximize D in the objective space. The resolution of this problem provides multiple equivalent trade-off solutions along the Pareto-optimal front. With this set of solutions, one particular configuration can be selected by taking into account the application needs that may prioritize one of the objectives. We implemented this solution under the PISA framework [7] using the non-dominated sorting genetic algorithm (NSGA-2) [8]. Moreover, we fed our simulator with probabilistic models (explained in Section II) based on a large measurement campaigns [4], [5]. For comparison purposes, we propose three different strategies:

- **Fixed Timers Random Sequence (FTRS)** [9], in which all channels are scanned with single fixed timers and $MinCT_i = MaxCT_i\ \forall i$. We simulated for $MinCT_i$ and $MaxCT_i$ from 1 to 15 ms.
- **Lowest Response Delay Sequence (LRDS)**, where we use an ordered channel sequence based on the average Probe Response delay on each channel observed in the experiments [4], [5]. Thus, we order the sequence from the lowest to the highest response time, i.e., $\{1, 8, 13, 4, 9, 3, 5, 6, 11, 10, 7, 2, 12\}$.
- **Adaptive Best Sequence (ATBS)**, where the channel sequence gives priority to channels with higher probability of having at least one AP deployed (see Fig. 1(b)), i.e., $\{6, 1, 11, 9, 10, 3, 8, 7, 5, 4, 13, 2, 12\}$. For this strategy, we summarize in the right part of Table I the set of $(MinCT_i, MaxCT_i)$ that the multi-objective approach gives as output.

Note that in LRDS and ATBS, the MS may scan only a few channels in the sequence (skiping some of them), without scanning the whole sequence. For example, to obtain a scanning latency within 30 ms, the multi-objective algorithm indicates that the MS should scan channels 6 and 1 as indicated in Table I, with the following set of timers: channel 6 with $(12, 3)$ and channel 1 with $(8, 2)$. Both LRDS and ATBS use adaptive timers, i.e., different pairs $(MinCT_i, MaxCT_i)$ for each channel. After resolving the multi-objective problem, we obtain a set of Pareto-optimal[1] solutions that show the optimal trade-off for L, F and D.

B. Results

Table I shows a performance comparison for each of the scanning strategies (FTRS, LRDS and ATBS) measured by the three proposed metrics, i.e., the failure rate, the discovery rate and the latency. These results were obtained with an ad hoc simulator fed with 802.11 hot-spot data reported in [4], [5]. The Table also gives in the right part the pairs ($MinCT_i$, $MaxCT_i$) for the ATBS strategy for various latency levels, as calculated by the multi-objective optimization problem. A dash (-) replaces a timer value when the channel is not probed, for example, in ATBS, for a latency of 30 ms, only channels 6 and 1 are scanned.

Fig. 2 illustrates the Pareto-optimal set in the objective space (L, F and D) for the three proposed strategies configured with the multi-objective optimization output. F and D are represented in ordinates and L in abscissa. We clearly observe that ATBS strategy is the one that offers the best trade-off between L, F and D. It allows obtaining low latencies (very differently from FTRS) and, for a given value of L, ATBS attains higher D and lower F than the two other strategies. Moreover, ATBS either uses lower timers than the other two or

[1] A Pareto-optimal solution is one that cannot be improved in one of the objectives without degrading the others.

TABLE I

OPTIMAL OBJECTIVE TRADE-OFF AND SCANNING PARAMETERS CONFIGURATIONS FOR ATBS

L(ms)	FTRS		LRDS		ATBS		$(MinCT, MaxCT)$ values for ATBS												
	F(%)	D(%)	F(%)	D(%)	F(%)	D(%)	CH6	CH1	CH11	CH9	CH10	CH3	CH8	CH7	CH5	CH4	CH13	CH2	CH12
10	-	-	59.8	4.9	31.9	16.9	(5,3)	-	-	-	-	-	-	-	-	-	-	-	-
20	-	-	21.1	14.3	18.6	30.5	(9,10)	-	-	-	-	-	-	-	-	-	-	-	-
30	-	-	19.8	17.9	4.0	39.8	(12,3)	(8,2)	-	-	-	-	-	-	-	-	-	-	-
50	98.2	1.3	16.6	18.1	1.1	61.7	(15,5)	(7,3)	(7,7)	-	-	-	-	-	-	-	-	-	-
85	8.4	9.4	2.9	49.8	0.3	71.6	(15,13)	(15,11)	(7,7)	-	-	-	-	-	-	-	-	-	-
100	3.8	37.5	1.4	52.3	0.2	81.7	(15,13)	(15,13)	(15,13)	(8,8)	-	-	-	-	-	-	-	-	-
150	1.2	62.6	0.7	58.5	0.1	85.3	(15,13)	(15,13)	(15,13)	(12,12)	(4,4)	(8,8)	(12,8)	(4,4)	-	-	-	-	-
200	0.8	78.4	0.4	59.8	0.06	87.3	(15,13)	(15,13)	(15,15)	(15,15)	(15,10)	(11,11)	(7,3)	(7,7)	(6,4)	(6,3)	(3,3)	(3,3)	(3,3)

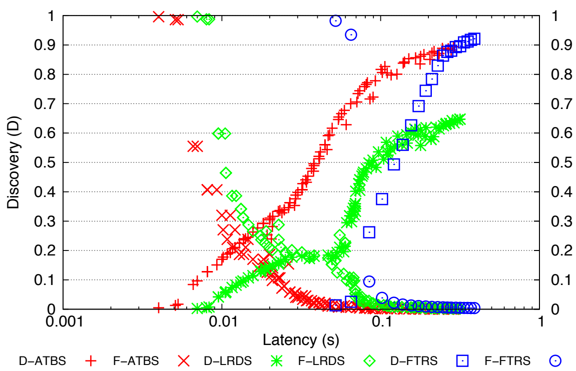


Fig. 2. Trade-off between scanning performance metrics

avoids scanning channels that are less likely to be used. Observe also that even when we carefully choose the sequence for LRDS, this approach does not give priority to highly populated channels (i.e., placing them at the beginning of the sequence) which gives a much better performance (specially for higher latencies). Note that FTRS badly manages the trade-off since it may start scanning channels that are not commonly used and without any capability to adapt timers.

As stated before, an MS may target different levels of latency, depending on its running applications. We can observe that the first configurations for ATBS in Table I involve low timers and a reduced channel sequence. These configurations may be adequate for an MS running real-time applications with low-latency requirements (e.g., VoIP, streaming). If an MS would like to keep a latency around $50\,ms$, it should scan the three first channels in the sequence (6, 1 and 11) using the timers given in Table I to discover in average the 61.7% of the available APs and failing on average the 1.1% of the cases. On the other hand, for elastic applications ($L \geq 100ms$), we can consider longer timers and more channels in the sequence. This configuration may be used by an MS running non real-time applications (e.g., web-browsing, e-mail) since the longer disruptions may be tolerated during a handover.

IV. RELATED WORK

The scanning process has been studied from different perspectives. However, to the best of our knowledge, we are particularly addressing a novel approach to improve the scanning latency. Differently from existing scanning algorithms based on fixed scanning parameters, our objective is to satisfy applications requirements, while considering appropriate failure and discovery rates. Consequently, we present the related work for the general optimization of the scanning process. We have identified strategies based on: timer values, scanning frequency and smart topology discovery. All of them with the single purpose of reducing the scanning latency without considering the application profile.

Strategies based on $MinCT$ and $MaxCT$ timer variations propose the simple adaptation of the timer's values. These values are normally pre-established within the devices. Velayos and Karlson [9] proposed, for the first time, theoretically estimated values for $MinCT$ (considering the 802.11 standard and the CSMA mechanism) and a $MaxCT$ proportional to $MinCT$. Lately, Wu [10] and Teng [11] improved the scanning using higher values for both timers. They show by means of an experimental testbed that, in ideal conditions, the $MinCT$ proposed in [9] is too low. The authors consider in their experiments the congestion of the channel which suggests an increased $MinCT$. In previous work [1], we have noticed that $MinCT$ should be calculated from the perspective of the OS kernel. At this point, the arrival of the control frames require more than the theoretical time to obtain scanning results.

Castignani et al. [1] present a dynamical strategy to adjust $MinCT$ and $MaxCT$ during the scanning process. The authors aim at reducing the scanning time in every channel, for example, reducing $MinCT$ and $MaxCT$ while discovering more APs. In a similar approach, the algorithm increases timers while the number of APs is considered low. This is called adaptive scanning.

Periodic scanning consist in grouping channels (whether adjacent or not) and alternate the scanning of few channels with the transfer and/or reception of layer-2 frames. This strategy avoids long interruptions on the networking service, instead, the system gives many small interruptions with an unnoticeable impact. Montavont et al. [12] describe a strategy in which an MS starts a scanning phase using two different periods. Authors differentiate these two periods depending on the current AP RSSI, in which the better the RSSI the higher the timers. This strategy improves the quality of the discovery process. Similarly, Wu [10] and Liao [13] present similar strategies (so called smooth scanning) looking for a minimal impact on the delay for the discovery process during a handover execution. The main strategy consist on discovering, with an active scanning, divided in several phases with variable duration allowing alternated data transfer. Finally, Nah et al. [14] and Park et al. [15], propose to dynamically adjust the duration of sub-phases of smooth scanning so as to make the interruption unnoticeable at user-level.

A different way to accelerate the active scanning consist in reducing the total number of channel to scan. This strategy is called selective scanning. As described by Shin et al. [2], there is a selection of the channels to review depending on the reported activity during the scanning, or simply there is a selection based on experiences from previous scanning results. For example, well-known non-overlapped channels 1, 6 and 11 have been reported with high probability for finding an AP and so they are privileged. Another strategy reported by Eriksson et al. [6], consists in a precomputed and stored probability of an AP operating in a channel to determine the scanning sequence.

V. CONCLUSION

In this paper, we have modelled the IEEE 802.11 scanning process and optimized the trade-off between the scanning performance metrics, namely the scanning latency, the failure rate and the discovery rate. In these terms, the objective of the scanning is to discover the highest number of APs in a minimum amount of time. We have obtained an analytical expression for each scanning performance metric and defined them using empirical data that have been gathered during a large measurement campaign of 802.11 urban AP deployments. Then, we introduced a new approach for AP scanning by considering the running applications needs. Depending on these requirements, we optimize the topology discovery while respecting the application latency constraints.

We have shown a reasonable approximation to an optimal performance for the scanning process, using a multi-objective optimization problem. This solution shows that by employing an optimal channel sequence and pre-configured timers on each channel, we can obtain a suitable trade-off between scanning performance metrics. Finally, results in this paper provide a guide to configure the scanning process highly adaptable to the application profile of the client. In a real implementation, the proposed analytical expressions can be instantiated using various methods, e.g., by an exchange between clients or an open data base that contains up-to-date information about the deployment.

REFERENCES

[1] G. Castignani, A. Arcia, and N. Montavont, "A study of the discovery process in 802.11 networks," *SIGMOBILE Mob. Comput. Commun. Rev.*, vol. 15, pp. 25–36, March 2011.

[2] S. Shin, A. G. Forte, A. S. Rawat, and H. Schulzrinne, "Reducing MAC layer handoff latency in IEEE 802.11 wireless LANs," in *MobiWac*, pp. 19–26, ACM, 2004.

[3] E. Giordano, R. Frank, G. Pau, and M. Gerla, "CORNER: a realistic urban propagation model for VANET," in *Wireless On-demand Network Systems and Services (WONS), 2010 Seventh International Conference on*, pp. 57 –60, feb. 2010.

[4] G. Castignani, L. Loiseau, and N. Montavont, "An evaluation of IEEE 802.11 community networks deployments," in *Int. Conference on Information Networking (ICOIN)*, 2011.

[5] G. Castignani, A. Blanc, A. Lampropulos, and N. Montavont, "Urban 802.11 Community Networks for Mobile Users: Current Deployments and Prospectives," *Mobile Networks and Applications*, vol. 17, pp. 796–807, 2012.

[6] J. Eriksson, H. Balakrishnan, and S. Madden, "Cabernet: vehicular content delivery using WiFi," in *14th ACM international conference on Mobile computing and networking, MobiCom '08*, (New York, NY, USA), pp. 199–210, ACM, 2008.

[7] S. Bleuler, M. Laumanns, L. Thiele, and E. Zitzler, "PISA - A Platform and Programming Language Independent Interface for Search Algorithms," in *Evolutionary Multi-Criterion Optimization (EMO)*, LNCS, (Berlin), pp. 494 – 508, Springer, 2003.

[8] K. Deb, *Multi-Objective Optimization Using Evolutionary Algorithms*. NY, USA: John Wiley & Sons, Inc., 2001.

[9] H. Velayos and G. Karlsson, "Techniques to reduce the IEEE 802.11b handoff time," in *Communications, 2004 IEEE International Conference on*, vol. 7, jun. 2004.

[10] H. Wu, K. Tan, Y. Zhang, and Q. Zhang, "Proactive Scan: Fast Handoff with Smart Triggers for 802.11 Wireless LAN," in *INFOCOM 2007. 26th IEEE International Conference on Computer Communications. IEEE*, may 2007.

[11] J. Teng, C. Xu, W. Jia, and D. Xuan, "D-scan: Enabling fast and smooth handoffs in AP-Dense 802.11 wireless networks," in *IEEE INFOCOM 2009 The 28th Conference on Computer Communications*, 2009.

[12] J. Montavont, N. Montavont, and T. Noel, "Enhanced schemes for L2 handover in IEEE 802.11 networks and their evaluations," in *Personal, Indoor and Mobile Radio Communications, 2005. PIMRC 2005. IEEE 16th International Symposium on*, vol. 3, sep. 2005.

[13] Y. Liao and L. Cao, "Practical schemes for smooth MAC layer handoff in 802.11 wireless networks," in *World of Wireless, Mobile and Multimedia Networks, 2006. WoWMoM 2006. International Symposium on a*, 2006.

[14] J.-W. Nah, S.-M. Chun, S. Wang, and J.-T. Park, "Adaptive handover method with application-awareness for multimedia streaming service in wireless LAN," in *Memorias del 23rd international conference on Information Networking, ICOIN09*, (Piscataway, NJ, USA), pp. 1–7, IEEE Press, 2009.

[15] J.-T. Park, J.-W. Nah, S. Wang, and S.-M. Chun, "Context-aware mobility management with energy eficiency for multimedia streaming service in wireless LAN," in *6th IEEE Conference on Consumer Communications and Networking Conference, CCNC09*, (Piscataway, NJ, USA), pp. 1332–1337, IEEE Press, 2009.