D-Fi: A Diversity-Aware Wi-Fi Using An OFDM-based Bloom Filter

Suchul Lee* and Chong-Kwon Kim†
Department of Computer Science, Seoul National University
Email: daxttt@gmail.com* and ckim@snu.ac.kr†

***Abstract*—To exploit frequency diversity in Wi-Fi channels, instantaneous channel quality must be estimated. However, there is a trade-off between acquiring channel quality information and improving protocol efficiency because channel estimation consumes time and frequency resource that ideally should be used for data transfer. In this paper, we present D-Fi (Diversity-aware Wi-Fi), a novel Wi-Fi PHY/MAC protocol, that capitalizes on frequency diversity gains while sustaining protocol efficiency. The D-Fi design allows to estimate channel quality while D-Fi is performing channel contention using an OFDM-based Bloom filter. To resolve the ambiguity caused by the Bloom filter, we adopt two methods: (i) An analysis-based multi channel backoff method enables to explore/exploit frequency diversity while reducing the occurrence of the ambiguity. (ii) Applying machine learning (ML) methods to the D-Fi PHY/MAC protocol corrects the ambiguity taken place already and makes our protocol reliable. We have shown the feasibility of D-Fi by implementing it on the USRP/GNURadio platform. Experiments and trace-driven simulations show that D-Fi successfully achieves frequency diversity gains without losing improved protocol efficiency.**

I. INTRODUCTION

Frequency diversity is one of the characteristics that should be considered in designing wireless communication systems, especially for ones that operate over a wide frequency band such as WiMAX [2] and 3GPP LTE [3]. In addition to spatial and temporal diversities, signals transmitted over a wide frequency band experience independent fluctuations across frequencies. This phenomenon is generally called “frequency selective fading” [4]. Frequency diversity is ignored in conventional WiFi systems because these systems use a channel as a whole. However, adoption of OFDM in 802.11 WLANs triggered recent research interests [8], [9], [12] in harvesting gains from frequency diversity. The importance of frequency diversity research becomes more important than ever as IEEE 802.11 working group (WG) is standardizing the use of wider channels. For example, 802.11n [1] can already use a 40MHz channel by Phased Coexistence Operation (PCO) and 802.11ac will provide up to a 160MHz channel. Accordingly, several Wi-Fi protocols [8], [9], [12] exploiting frequency diversity have already been proposed recently in academia.

To harness frequency diversity, a wireless communication system must provide a channel quality estimation functionality. Acquiring channel quality information consumes time and frequency resource that ideally should be used for data transfer. For example, many current wireless systems estimate channel quality using a training sequence (pilot) in a preamble or spend dedicated time only for the channel estimation purpose [1], [2], [3], [6], [8], [9], [12]. Moreover, for $N \times N$ Multiple-Input-Multiple-Output (MIMO) systems, N^2 channels have to be estimated [7] resulting in substantial protocol inefficiency. In this case, the high data throughput of a MIMO system cannot be achieved due to the large overhead of channel estimation. In short, there is a trade-off between frequency diversity gains and protocol efficiency.

The research approaches to achieve frequency diversity gains are categorized into two groups; (i) variants of Wi-Fi systems that improve the protocol efficiency [10], [11], [13] and (ii) frequency diversity aware protocols for various wireless networks such as WiMAX [2], 3GPP LTE [3], [31], [32], and Wi-Fi networks [8], [9], [12], [30]. However, none of them explore both of the conflicting objectives - i.e., reduction of channel estimation overhead and protocol efficiency - simultaneously. Most previous work emphasizes mainly one side of these [8], [9], [11], [12], [13] since the two objectives are considered as orthogonal to each other [10] (but it is not true as we have discussed above). Also, frequency diversity aware studies are highly theoretical rather than practical [30], [31], [32], i.e., these researchers solved the channel allocation problem assuming the perfect channel information is given.

We argue that satisfying two conflict objectives, achieving frequency diversity gain and protocol efficiency, boils down to acquisition of channel quality information with a minimum channel estimation cost. In this paper, we present D-Fi (Diversity-aware Wi-Fi), a novel Wi-Fi PHY/MAC protocol that exploits frequency diversity while sustaining protocol efficiency. Specifically, D-Fi collects channel information while resolving channel contentions using an OFDM-based Bloom filter without requiring a dedicated channel estimation mechanism. D-Fi can be combined with other protocols because it is orthogonal to those existing Wi-Fi proposals [10], [11], [13], [14] custom-tailored for improving protocol efficiency.

The D-Fi protocol has the following features.

- D-Fi channelizes a Wi-Fi band into several orthogonal subchannels based on the OFDM technique and uses each of them as a channel access unit. This channelized medium access amortizes MAC coordination burdens and hence improves overall MAC protocol efficiency. Moreover, it exploits frequency diversity inherent in a wide band by a frequency-aware subchannel allocation scheme.
- D-Fi estimates channel quality while performing contention based channel allocation. To do so, D-Fi adopts a Bloom filter based channel contention mechanism. Specifi-

cally, the D-Fi MAC protocol uses RTS/CTS-like Collision Resolution reQuest (CRQ)/Collision Resolution rePly (CRP) frames through a Bloom filter. A CRQ/CRP frame lasts only for a few OFDM symbols. The overhead of D-Fi is much smaller than that of the legacy RTS/CTS frame. Multiple stations (STAs) contend for subchannels simultaneously according to estimated subchannel quality as well as their traffic demands. An AP can estimate the uplink channel quality of the STAs using this synthesized CRQ frame without additional channel estimation overhead. After an AP perform frequency-aware subchannel allocation based on the channel estimates then it broadcasts a CRP frame to inform the STAs of the result of channel allocation.

- Bloom filter based channel contention incurs the ambiguity problem because of an intrinsic characteristic of a Bloom filter. D-Fi uses two methods to solve the problem. Firstly, an analysis-based multi channel backoff algorithm reduces the occurrence of the ambiguity while allowing D-Fi STAs to distributively explore/exploit frequency diversity. Next, applying machine learning (ML) algorithms to the D-Fi protocol resolves the ambiguity so that D-Fi can operate the MAC protocol reliably.

We implemented the OFDM-based D-Fi PHY/MAC on a testbed consists of four USRPs/GNUradios. The experiment shows the feasibility and practicality of the D-Fi PHY/MAC protocol. Further, we used detailed trace-driven simulation to evaluate the performance of D-Fi. Our results show that D-Fi has up to 3x and 1.5x better performance in terms of throughput compared to existing 802.11n and FICA [10], respectively.

In summary, this paper makes the following contributions. (i) We design and implement D-Fi, a Wi-Fi PHY/MAC protocol that exploits frequency diversity while sustaining the MAC efficiency. (ii) We provide a detailed analysis to address the ambiguity problem arisen from the use of a Bloom filter. Based on the analysis we propose a multi channel backoff algorithm that explores/exploits frequency diversity distributively while reducing the occurrence of ambiguity. (iii) We apply ML methods to the D-Fi PHY/MAC protocol and demonstrate the superior performance of ML methods in solving the ambiguity problem arisen in the D-Fi PHY/MAC protocol. (iv) We demonstrate the feasibility of D-Fi with our prototype implementation on the USRP/GNURadio platform and evaluate its performance using the detailed trace-driven simulation.

The rest of this paper proceeds as follows. Section II describes the design of the D-Fi PHY/MAC. We then provide a detailed analysis to deal with the ambiguity arisen from the use of a Bloom filter and propose a multi channel backoff algorithm in section III. Section IV describes ML methods applied to solve the ambiguity problem in the D-Fi protocol. We show the performance of ML algorithms and discusses some ML-related issues. Section V evaluates the D-Fi's performance using our experimentation and trace-driven simulation. Section VI reviews the related work. Finally, Section VII concludes the paper.

II. D-Fi Design

D-Fi is a CSMA-based Wi-Fi PHY/MAC protocol that performs wireless channel contention and channel quality estimation at the same time. Generally, channel quality estimation incurs overhead because extra estimation time and/or training sequences (pilot) are used for estimation. D-Fi acquires channel information while STAs are performing channel contention and no additional overheads are required. Based on the estimated channel quality, D-Fi exploits frequency diversity. In this section, we detail the design of the D-Fi PHY/MAC.

A. Channelization

Taking a large Fast Fourier Transform (FFT) window size means a long OFDM data symbol in time. Therefore, for the purpose of good protocol efficiency, it is desirable to choose a large FFT window. Although it is possible to choose any large FFT size theoretically, there are several practical concerns that prevent large FFT [4]: (i) Computational complexity increases as an FFT size increases since the theory tells us that the complexity of the N-points FFT(or inverse FFT) is $O(N \log N)$. (ii) The frequency separation between subcarriers is imperfect. These limitations are generally caused by mismatched oscillators, Doppler shift, or timing synchronization errors. And these factors eventually lead to lose orthogonality between subcarriers introducing non-negligible inter carrier interference (ICI) in practice. In D-Fi, to deal with such limitations, we choose the FFT size such that an OFDM symbol is 256/512 points in a 20/40MHz channel (subcarrier bandwidth is about 78.12KHz.).

Coherence bandwidth is a statistical measure of the range of frequencies over which the channel can be considered “flat”. Recent measurement studies [8], [12], [16], [17] have shown that the minimum coherence bandwidth over the industrial, scientific, and medical (ISM) license-free band (near 2.4/5GHz) is approximately 3MHz in indoor environments. Therefore, when a channel access unit (i.e., a subchannel) is narrower than 3MHz it can be considered as flat within a subchannel and is frequency-selective between subchannels.

These measurement results motivate us to develop D-Fi, a Wi-Fi protocol that exploits frequency-selectivity. We choose 17 contiguous subcarriers to form a subchannel (bandwidth is about 1.4MHz.). Among 17 subcarriers, 16 subcarriers are used for data transmission and one subcarrier is used as a pilot channel that tracks the subchannel quality while the data is being transferred. There are 14 orthogonal subchannels in a 20MHz band, and they are frequency-selective one another in typical indoor environments.

B. Protocol Overview

D-Fi uses Contention Resolution reQuest (CRQ) / Contention Resolution rePly (CRP) frame exchanges for channel contention (Fig. 1). Note that a CRQ/CRP frame lasts only for a few OFDM symbols and so its overhead is much smaller than that of the legacy RTS/CTS frame. If the medium is idle for more than distributed interface space (DIFS) STAs

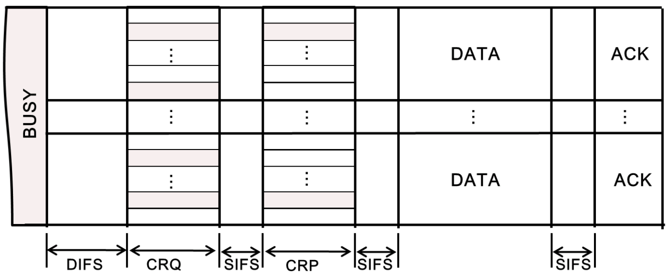

Fig. 1. The D-Fi MAC protocol overview.

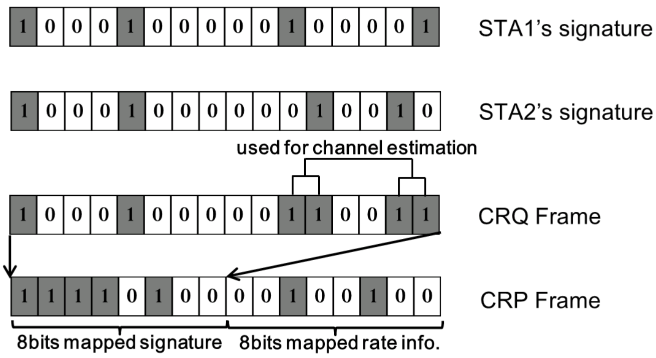

Fig. 2. The description for two Bloom filter based operations; inserting elements, i.e., signatures (CRQ) and testing membership (CRQ decoding). These operations are performed in one subchannel (i.e., one Bloom filter). Broadcast of a channel contention result (CRP) is also described at the bottom of the figure.

may transmit CRQ symbols simultaneously. Each STA selects K subchannels[1] likely to have good channel quality and modulates his own *signature* on each selected subchannel. Consequently, multiple CRQs sent from multiple STAs arrive at the AP. These CRQ symbols can be misaligned due to different propagation delay, sensing time (CCA), and RF RxTx switching delay. However the total misalignment has been shown to be tightly bounded [10]. In an OFDM system, as long as the misalignment is less than the cyclic prefix (CP), a receiver can decode misaligned signals [4]. We set the D-Fi CP length such that the maximum alignment is less than CP length.

An AP can extract STAs' uplink channel quality information from CRQ frames. Then the AP allocates subchannels to the STAs based on any channel allocation policy, for example, proportional fairness or throughput-optimum.

To inform STAs of the channel allocation results, the AP broadcasts a CRP frame. This frame conveys the signature of the contention winner and transmission rates for future data transmission.

C. Channel Contention and Estimation

1) Signature: A signature is a binary bit sequence of 16 bits. A STA receives a unique signature when it joins a Wi-Fi network. The rule for assigning a signature is as follows: First, divide a 16 binary bits sequence into four continuous-bits subsequences. Then choose one bit in each subsequence and mark chosen four bits (one bit from each subsequence) as "1" and the rest as "0". Therefore, 256 ($= 4^4$) possible signatures exist. Note that the number of STAs in a WLAN is typically not very large ($\leq$ order of tens and 256 is enough for unique allocation to all STAs).

A signature is carried over one subchannel; one bit over one subcarrier. We use binary amplitude modulation (BAM) to modulate a single bit on each subchannel. Specifically, BAM uses On-Off signaling that maps a binary "0" to zero amplitude and a binary "1" to a random complex number on the unit circle ($e^{j\theta}$) in a subcarrier. In other words, no signal is transmitted to modulate a binary "0" in a subcarrier and a fixed powered random complex signal is transmitted to modulate a binary "1" in a subcarrier. A receiver can easily detect a BAM symbol by measuring a signal power level on a subcarrier without demodulating an exact symbol.

[1]They are not necessary to be continuous.

STAs may join and leave dynamically. At the time of association, an AP allocates a signature to the joining STA. The allocated signatures among 256 possible ones are called "valid". If a STA is inactive for long time, its signature is taken back and set to be "invalid".

2) CRQ Frame: To facilitate simultaneous channel contention and estimation, D-Fi uses the Bloom filter [18]. A subchannel where signatures are transmitted can be considered as a Bloom filter consists of 16 bits. If only one signature is transmitted over a subchannel, then we can easily detect the signature. If two or more signatures collide, the AP uses the Bloom filter technique to resolve signatures. The process of identifying signatures from a Bloom filter is called "CRQ decoding" (Fig. 2).

In CRQ decoding, we should handle two types of ambiguity; the physical and logical errors.

Physical errors. One bit in a Bloom filter is actually one OFDM subcarrier. A STA will transmit a signal over some selected subcarriers representing its signature. Since the frequency separation between subcarriers is imperfect in practice a subcarrier suffers from so-called "spectral leakage" [19]. A signal spills over adjacent subcarriers. Since subcarrier-level signal detection is implemented by comparing between a signal power level and a threshold [10], [11] the signal can be falsely detected. We call this event "bitwise false positive (bitwise-FP)" and the event that the signal is falsely missed "bitwise false negative (bitwise-FN)". Both of the events are the physical errors. Careful and adaptive threshold adjustments can make them negligible. Our software radio implementation (explained in section V) as well as other implementations [10], [11] showed that the physical error rates are quite small.

Logical errors. An intrinsic characteristic of a Bloom filter is the logical error. During the CRQ decoding process, an AP falsely determines the signatures that are not actually requested. It is generally called "False Positive (FP)" of a Bloom filter. For example, two stations STA1 and STA2, whose signatures are "1000 1000 0010 0001" and "1000 1000 0001 0010", respectively, request to the same subchannel

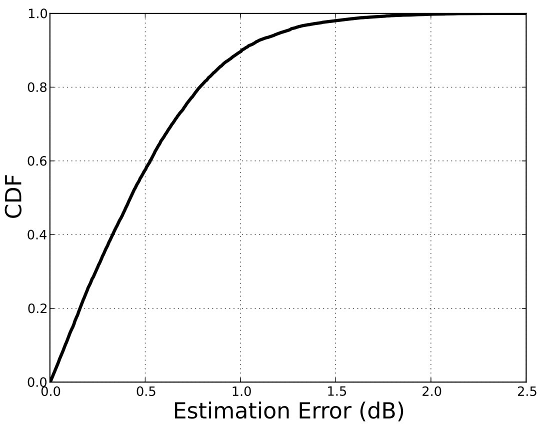

Fig. 3. The experimental result: the empirical CDF for the estimation error of our method.

resulting in a Bloom filter of “1000 1000 0011 0011”. The AP should decode “1000 1000 0011 0011” as a superposition of the signatures of STA1 and STA2. However, due to the inherent ambiguity, it may falsely decode it as “1000 1000 0001 0001” and “1000 1000 0010 0010 as well. Although D-Fi only considers STAs having valid signatures as channel contenders, there still is non-negligible FP rate.

We propose two methods to solve the ambiguity problem; an analysis based multi channel backoff algorithm and machine learning (ML) algorithms. The analysis based multi channel backoff algorithm aims to limit the number of channel requests for one subchannel. On average a STA will request K subchannels at once, and an appropriate value of K is determined by the analysis shown in section III. The multi channel backoff algorithm selects K preferable (i.e., high quality) subchannels in a distributed manner to exploit frequency diversity. On the other hand, ML-based CRQ decoding aims to reduce the probability of logical and physical errors in CRQ decoding (explained in section IV). In short, multi channel backoff prevents the logical errors while ML-based CRQ decoding corrects the physical and logical errors.

3) CRP Frame: To inform a STA of a channel allocation result, an AP broadcasts a CRP frame. This frame conveys the signature of a contention winner and data rate information for future data transmission for each subchannel. Since there are $256 (= 2^8)$ signatures, 8 bits are used for a signature and the rest are used for data rate information (Fig. 2).

4) Channel Quality Estimation: Assume that all stations use the same transmission power and the total transmission energy spreads evenly over each of four bits marked as “1” when sending a CRQ symbol. An AP can guess the channel qualities from the signal strength of unique bits. A unique bit is a bit that is transmitted by one station only. After CRQ decoding, we determine unique bits and use the average energy level of the unique bits belong to a signature as the channel quality (Fig. 2).

We have evaluated the channel estimation performance in terms of accuracy in our implementation. For most of the cases ($\geq$ 90%), the estimation error of our method is less than or equal to 1 dB (Fig. 3).

D. Proportional Fairness

Once channel quality estimates are available, an AP can allocate subchannels to STAs by the proportional fairness algorithm in [29]. Proportional fairness maximizes the sum of logarithmic throughput over the fixed number (W) of time slots. Let $T_i[n]$ be the throughput of a STA i in a time slot n, the throughput of a STA i during W time slots $T_i^{(W)}[n]$ is then:

$$T_i^{(W)}[n] = \frac{1}{W} \sum_{m=n-n_0}^{n+W-n_0-1} T_i[m] \tag{1}$$

where n_0 is the number of slots look back to the past, and $W - n_0 - 1$ is the number of slots in the future. With the equation (1), our objective function is written as:

$$\max \sum_i \log T_i^{(W)}[n] \tag{2}$$

By the Shannon’s theorem [4], the throughput can be further re-written as a function of estimated SNRs. The difference from the original problem is that we apply the proportional fairness algorithm to the reduced problem space since an AP can only estimate the channel quality of the STAs who have made a request. Even with this restriction, in subsection V-B, we will show that D-Fi has close to the optimal performance in terms of exploring/exploiting frequency diversity.

E. Why Bloom Filter?

Basically, a Bloom filter is a space-efficient data structure. Here, the space means the number of subcarriers constructing a subchannel. As we have described in subsection II-A, we cannot use large FFT windows. The price paid for this space-efficiency is probabilistic ambiguity inherent to a Bloom filter: it tells us that the element either *definitely* is not in the set or *may be* in the set. The term “*may*” means that a Bloom filter may generate ambiguity (i.e., false positives). In D-Fi, resolution of the ambiguity is particularly important because it estimates the channel quality based on the unique bits in signatures. Unfortunately, it is impossible to eliminate false positives completely and hence we turn our attention to find a method to mitigate the false positive probability. As we will see in section IV, machine learning algorithms (MLs) are good solutions to this problem. In addition, a careful choice of hashing functions (e.g., MD5 [18]) can further reduce the ambiguity. This application is left for future work.

III. Analysis

In this section, we analyze the false positive probability and the collision probability of the contention mechanism in D-Fi. Based on the analysis we propose a multi channel backoff method that enables a STA to explore/exploit frequency diversity distributively. It also reduces the false positive probability of the Bloom filter based contention mechanism.

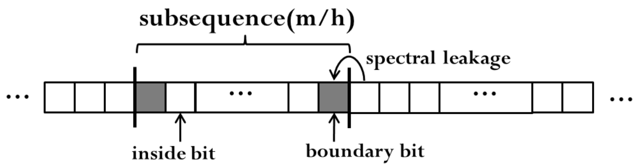

Fig. 4. The spectral leakage by the other hash function can only occur at the boundary of the subsequence.

We assume a WLAN consists of N STAs and C subchannels. As a Bloom filter is used for each subchannel C Bloom filters exist. A Bloom filter consists of m binary bits (i.e., subcarriers) and h hash functions[2]. A STA can request for K subchannels each time it contends for a channel. On average, r $(=\frac{N\times K}{C})$ STAs will select a certain subchannel. In other words, on average, r elements (signatures) will be inserted into a Bloom filter. Given that hash functions are uniform, the probability that a certain bit is selected by one of h hash functions is $\frac{h}{m}$.

Let us derive the probability that a subcarrier is set to be "1" taking into account the spectral leakage. An OFDM system suffers from high spectral sidelobes [19], and consequently, a subcarrier may accidentally be set to "1" because of the leakage of power from subcarriers nearby. Assume that the only adjacent subcarriers cause power leakage. Let P_{leak} be the probability of the spectral leakage. Then the probability that a certain bit is set to "1" because of the spectral leakage is $\frac{2hP_{leak}}{m}$.

Remind that each of our hash functions selects one bit from each of the non-overlapping subsequences (each subsequence is m/h bits long). The probability that an inside bit - a bit not adjoining to the subsequence boundary - is set to "1" is given as:

$$\frac{h}{m}+\frac{2hP_{leak}}{m} \tag{3}$$

While consecutive inside bits cannot be selected by two hash functions at the same time, two boundary bits can be set to "1" by two hash functions (Fig. 4). Therefore we have to subtract the probability of the event that two hash functions simultaneously set the bits at the boundary as "1" from the equation (3):

$$\frac{h}{m}+\frac{2hP_{leak}}{m}-(\frac{h}{m})^2P_{leak}(1+P_{leak}) \tag{4}$$

Combining equation (3) and equation (4), the probability that a certain bit is set to "1" is:

$$\begin{aligned} P^1_{positive} &= \frac{2(\frac{h}{m}+\frac{2hP_{leak}}{m}-(\frac{h}{m})^2P_{leak}(1+P_{leak}))}{\frac{m}{h}} \\ &\quad + \frac{(\frac{m}{h}-2)(\frac{h}{m}+\frac{2hP_{leak}}{m})}{\frac{m}{h}} \end{aligned} \tag{5}$$

Then the probability that a certain bit is set to "0" is $1-P^1_{positive}$.

[2]Each bit of a signature is chosen by each hash function.

Now we extend to the case of multiple requests onto a subchannel. If there are r requests to a subchannel, the probability that a certain bit is set to "0" is $P^r_{negative}=(1-P^1_{positive})^r$, and the probability that a certain bit is set to "1" is $P^r_{positive}=1-(1-P^1_{positive})^r$. Now consider a STA that does not contend for the subchannel. Even if the STA does not participate in contention, each of its h signature bits has non-negative probability of being "1". The probability that all h bits are "1", which would cause an AP to erroneously claim that a STA has requested for the subchannel, is given as:

$$P^{D-Fi}_{falsepositive}=(P^r_{positive})^h \tag{6}$$

For the collision probability of D-Fi, it is zero because an AP allocates a subchannel to exactly one STA.

For the comparison purpose, we also analyze the false positive and collision probabilities of the FICA [10] contention mechanism. In FICA, a STA transmits a request signal over one randomly chosen subcarrier within a subchannel. An AP selects one active subcarrier and all the STAs who have sent the signal on that subcarrier are allowed to use the subchannel for the next data transmission. In addition, even if FICA does not suffer from logical false positives, it may wrongly select inactive subcarriers due to the spectral leakage. Therefore, the probability of the false positive in FICA is:

$$\begin{aligned} P^{FICA}_{falsepositive} &= P(\text{A bit is set to "1" w/ spectral leakage}) \\ &\quad - P(\text{A bit is set to "1" w/o spectral leakage}) \\ &= (1-(1-(\frac{1}{m}+\frac{2P_{leak}}{m}))^r)-(1-(1-\frac{1}{m})^r) \end{aligned} \tag{7}$$

Since a collision occurs only when two or more STAs send their request signals on the same subcarrier, the collision probability in FICA is:

$$P^{FICA}_{collision}=1-(1-\frac{1}{m})^{r-1} \tag{8}$$

A. Remarks

To validate our analysis, we have performed simple simulations (Fig. 5). As anticipated, the false positive rate is significant when the number of requests for a subchannel is large. The D-Fi's signature based contention mechanism performs better than FICA's when the number of requests is less than 2.6; its collision probability and false positive probability are smaller than those of FICA. Even so, it is important to control the number of requests for a subchannel. In order to make the number of requests for a subchannel operate within an appropriate range, we propose a multi channel backoff method. Our multi-channel backoff method enables a STA to explore frequency diversity distributively while controlling the number of requests to a subchannel.

B. Multi Channel Backoff

We propose a multi-channel backoff method that distributively controls the number of subchannels a STA requests. Each STA maintains a vector, $[Pr(1), Pr(2), ..., Pr(C)]$,

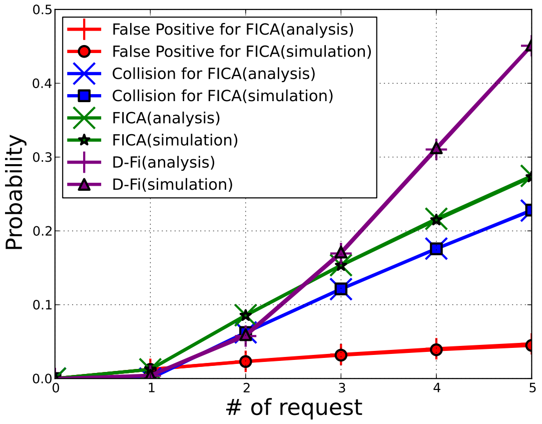

Fig. 5. The analysis and simulation results: the false positive and the collision probability of D-Fi and FICA. We have used 0.1 for the probability of the spectral leakage, P_{leak}.

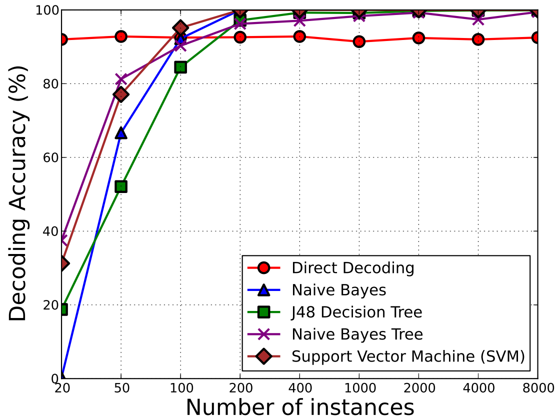

Fig. 6. The accuracy for the CRQ decoding with the ML algorithms.

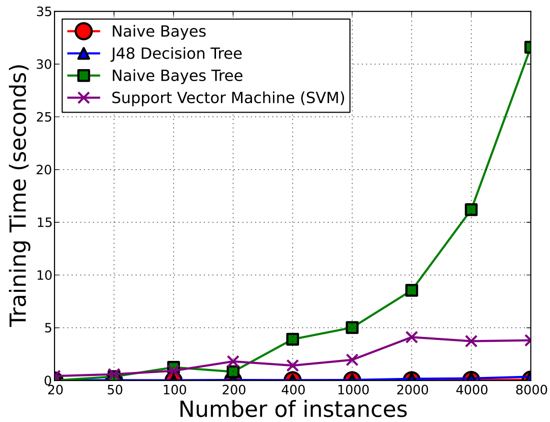

Fig. 7. The training time for ML models used for the CRQ decoding.

where $Pr(i)$ is how likely a STA requests for a channel i. Initially all $Pr(i)$ are set to be K/C. Based on the results of contention, we adjust $Pr(\cdot)$ according to the additive increase / multiplicative decrease (AIMD) manner. After hearing a CRP frame, a STA knows whether it is selected to use a subchannel or not. For each selected subchannel i, the STA increases the value of $Pr(i)$ by α. And for each non-selected subchannel i, the STA decreases the value of $Pr(i)$ by multiplying it with $\frac{1}{\beta}$. Afterwards $Pr(\cdot)$ is normalized in order that their sum is to be K.

Algorithm 1 shows the pseudo-code of the multi channel backoff algorithm.

On average, a STA requests K subchannels. Obviously, the optimal value of K depends on the number of active STAs (N) in a network. An AP estimates the number of active STAs in the network [20] and periodically broadcasts an appropriate K ($= \frac{r \times C}{N}$) value. We adjust r such that the false positive probability is not large (e.g., $\leq 10\%$) based on the analysis shown in section III.

One might argue that this multi-channel backoff mechanism cannot accommodate many STAs due to the high false positive probabilities. However, as we will see in the section IV, applying machine learning algorithms further eliminates the false positive probabilities and this allows D-Fi to accommodate many STAs (tens of STAs).

Algorithm 1 Multi channel backoff

for i := requested subchannel $i \in C$ **do**
  if subchannel i is requested & allocated **then**
    $Pr(i) \leftarrow Pr(i) + \alpha$
  else if subchannel i is requested & not allocated **then**
    $Pr(i) \leftarrow Pr(i)/\beta$
  end if
end for
$Pr(i) \leftarrow Pr(i) \times \frac{K}{\sum_{j \in C} Pr(j)}, \forall i \in C$

IV. Enhancement: Machine Learning

The multi-channel backoff controls to distribute requests over subchannels. However D-Fi still suffers from non-negligible false positives. We apply machine learning (ML) methods [22], [23] to further reduce the false positive probabilities.

To apply an ML method to the CRQ decoding process, we collect the dataset consisting of per-subcarrier RSSI readings. In our experiment, we assume that the maximum number of requests to a subchannel is three. We refer to a single set of 16 per-subcarrier RSSI readings as an *instance*. Since we know the STAs transmitting a CRQ frame in advance we can put a label (i.e., a list of the STAs transmitting a CRQ frame) on each *instance*. We can use this labeled set of *instances* to establish the ground truth. Now, we apply a supervised ML method to this set. Specifically, we train an ML model using this set of labeled *instances* and evaluate the trained ML model with the ground truth. ML models are evaluated with the cross-validation method provided by WEKA [21].

To visualize the CRQ decoding performance with the ML methods, in Fig. 6 we plot the accuracy of various ML algorithms. The applied algorithms[3] are Naive Bayes [24], Naive Bayesian tree [25], J48(C4.5) decision tree, and support vector machine (SVM) [27]. As shown in Fig. 6, all ML algorithms significantly outperform the direct CRQ decoding method (i.e., the method using the Bloom filter only) when the number of training instances is greater than 200. With sufficient training, ML algorithms correct the CRQ decoding errors almost completely ($\geq$ 99.9% accuracy). Fig. 7 shows the time required to train an ML model. The Naive Bayes algorithm, generally known as the simplest one, requires only tens of microseconds to be trained due to its low complexity. Moreover, an AP will take hundreds of milliseconds to collect 200 *instances* which are revealed to be sufficient to train a robust ML model. We next discuss several issues arisen when we apply ML methods to a real WLAN.

A. Getting The Set of Labeled Instances in a Real WLAN

To establish ground truth in a real WLAN, an AP has no choice but to label an *instance* through the direct CRQ decoding process. Then the false positives may happen and a subchannel can be assigned to a STA who actually does not request the subchannel. However the STA will not use the subchannel for the data transmission and the AP can infer the occurrence of a false positive and correct the label. Although it is hard for an AP to get the complete set of the labeled *instances* in a real WLAN, we believe that this corrected set of the *instances* will suffice to perform CRQ decoding robustly. An accurate performance evaluation of ML methods in a real world experimentation is our future work.

B. When an AP should train ML models?

To train a ML model, an AP needs a set consisting of at least 200 labeled *instances*, and this set must be evenly distributed over all possible labels. Note that our multi channel backoff algorithm tries to distribute requests evenly over all subchannels. Once trained, if no significant channel fluctuations exist, an ML model produces an accurate CRQ decoding output. We should re-train the ML model when a training set is outdated.

V. PERFORMANCE EVALUATION

A. Implementation

1) D-Fi Prototype and Experiment Setup: We implemented the D-Fi OFDM-based PHY/MAC on a small testbed of 4 USRPs [34] and GNU Software Define Radio (SDR) [35]. We adopt a simple Binary Amplitude Modulation (BAM) to modulate each bit of a signature used for a CRQ/CRP frame. In order to minimize the false positive of the subcarrier-level signal detection, the threshold used for the signal power level comparison is adaptively configured. Our experiment is conducted within a laboratory to show the feasibility of the D-Fi PHY/MAC protocol in a typical indoor wireless scenario.

[3]Naive Bayes and Naive Bayesian tree are generally known as simple and fast algorithms. And J48 tree and SVM are highly accurate shown in previous work [22], [23] although they are the results obtained from the area of an Internet traffic classification research.

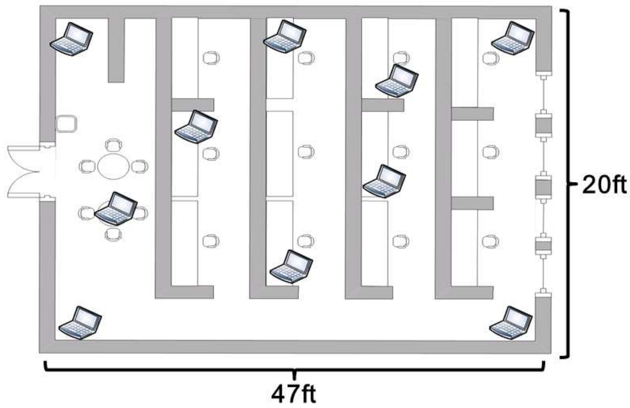

Fig. 8. The topology used in our experiment.

We depict the topology used in our experiment in Fig. 8. In Fig. 8, we have chosen four positions randomly, and let one node serve as an AP and the other three nodes be STAs associated with the AP. A rich set of the TX powers provided by the SDR is used, resulting in the 10dB difference between the min and max received signal strengths.

2) Results: Fig. 9 shows the feasibility of the OFDM subcarrier-level signaling. Since we have used three STAs for transmitting a CRQ symbol, multiple CRQ symbols combine at a receiver. The sum SNR of this synthesized CRQ symbol is plotted along the x-axis. We call the case that the signal strength difference among the individual CRQ symbols is smaller than 5 dB "similar case", and otherwise "different case". In the whole range of our experiment setups, the D-Fi's subcarrier-level signaling performs reliably. Occasional bitwise-FP and bitwise-FN may still happen, however, as we have shown in section IV, D-Fi successfully handles such occurrences with the ML algorithms. When this signature-level signal detection is applied to the CRQ decoding process, the accuracy of about 92% is achieved without the ML algorithms because of the logical errors. Applying the ML methods, however, the CRQ decoding process almost completely eliminates the logical errors and achieves the accuracy of about 99.9%.

We next show the accuracy of our channel estimation method. As shown in Fig. 3, for most of the cases ($\geq$ 90%), the estimation error is less than or equal to 1 dB. These two results show that the D-Fi's channel contention and estimation mechanisms are practically feasible in typical indoor environments where a WLAN operates.

B. Trace-driven Simulation

1) Simulation Setup: The above D-Fi prototype on the USRP is suitable for demonstrating the feasibility of the Bloom filter based channel contention and estimation method, but not the diversity exploration/exploitation performance of D-Fi. Since an USRP relies on software to process a signal, it experiences difficulty in processing a wide-band (20MHz) signal. Additionally, the supported data rate is not as high as that in hardware radios at a current development stage of

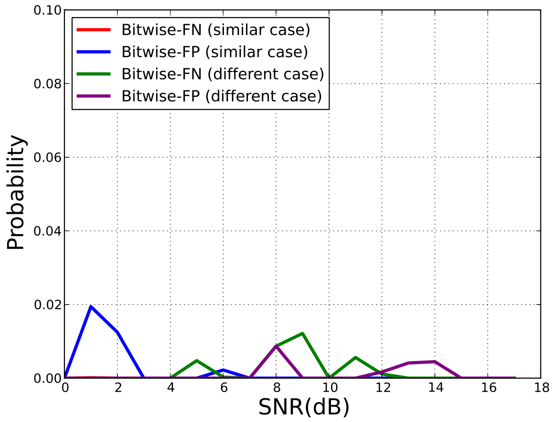

Fig. 9. The accuracy of the subcarrier-level signal detection.

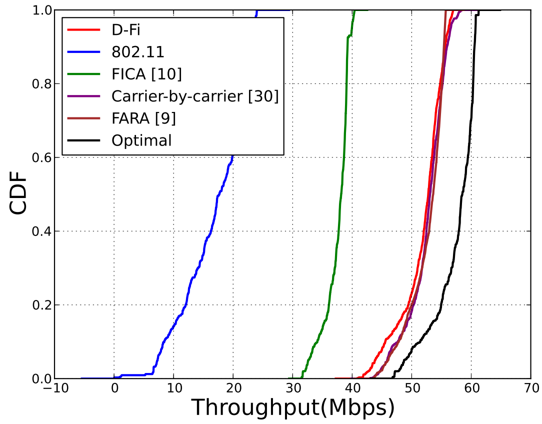

Fig. 10. The simulation results: the empirical CDF of the throughput for each scheme.

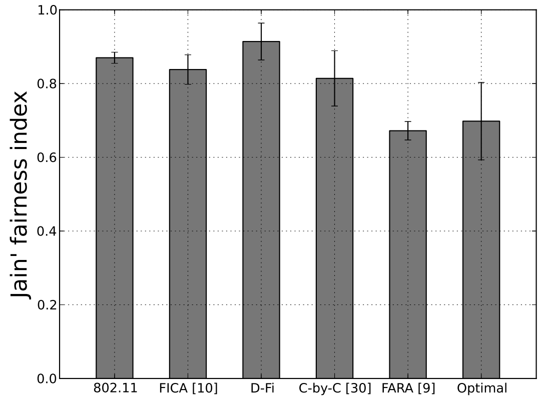

Fig. 11. The simulation results: fairness (Jain's fairness index).

an USRP. Therefore, we resort to trace-driven simulations to assess the diversity performance of D-Fi.

To conduct high fidelity emulation of real world setting, we have used the 802.11n data traces provided by the authors of [12]. The traces are obtained from commodity Intel Wi-Fi Link 5300 NIC and its modified driver [5]. The traces contain per-subcarrier (30 subcarriers for 20MHz) RSSI readings for both the 24 mobile and 30 static diverse links. With the 54 diverse links, we have set up to 50 nodes in our simulations.

We compare the diversity exploration/exploitation performance of 802.11n [1], FICA [10], Carrier-by-Carrier in turn algorithm (C-by-C) [31], FARA [9], D-Fi, and the throughput-optimal unit. For a fair comparison, we have modified C-by-C and FARA to use a subchannel as an access basis. For diversity-aware schemes such as C-by-C and FARA, we consider the same amount of the MAC protocol overhead with D-Fi to compare the performance due to the diversity exploitation capabilities. Multi channel backoff parameters for D-Fi (i.e., α and β) have been fine-tuned to assess the best performance of D-Fi.

2) Results: We first show the D-Fi's overall throughput against the other schemes. Fig. 10 presents the empirical cumulative distribution function (ECDF) of the throughput for each scheme. The D-Fi's throughput gains over legacy 802.11n and FICA are 3x and 1.5x, respectively. Because the 802.11n scheme does not channelize a 20MHz band and uses random access, it neither reduces the MAC overhead nor exploits the diversity. FICA, which uses the channelized random access, reduces the MAC overhead but fails to exploit the diversity. Now, let us compare the D-Fi's diversity exploration/exploitation performance with the other diversity-aware schemes in terms of throughput. Even though the proposed multi channel backoff algorithm requests only a subset of all the subchannels, the D-Fi's diversity exploitation/exploration performance is equivalent to those of the other diversity-aware schemes as shown in Fig. 10.

We next show the D-Fi's fairness performance. Since D-Fi allocates a subchannel based on the proportional fairness algorithm, it is expected to be fair as in random access schemes like 802.11n or FICA. To verify that, we compute Jain's fairness index with the throughput obtained by each STA. Fig. 11 presents the fairness index with all the schemes described above. It clearly shows that D-Fi offers high throughput while maintaining fairness comparable to random access schemes like 802.11n and FICA.

VI. Related Work

Related work falls in the following three areas.

A. Improving MAC efficiency

There has been a tremendous amount of work targeted towards improving MAC protocol efficiency [10], [11], [13], [15]. Among these, the most relevant to our work are [10], [11], [13]. FICA [10] tackles the inefficiency of the 802.11 MAC by redesigning both the PHY/MAC using OFDM-based fine-grained channelization. The authors of [11] have proposed Back2F that migrates the time domain backoff to the frequency domain. As the frequency domain backoff lasts for several OFDM symbols, it reduces relative MAC overhead and hence improves the MAC protocol efficiency. In addition, REPICK [13] have modified a receiver [11] to perform the frequency domain backoff instead of a transmitter and added the feature of ACK piggybacking, which further reduce the MAC overhead. These work share the similarity of using the OFDM technique to enhance MAC protocol efficiency. However, our work is different from them because D-Fi further exploits frequency diversity using the Bloom filter based channel contention/estimation while reducing overhead.

B. Frequency Diversity

Many theoretic classic proposals exploiting frequency diversity are well-summarized in the wireless communication textbook [4]. Some of them are currently being used by cellular systems like WiMAX [2], 3GPP LTE [3], etc. Recently, in academia, the theoretic studies applying proportional fair packet scheduling have been done in FDMA-based 3GPP LTE [31], [32] and CSMA-based OFDMA systems [29], [30]. Assuming perfect channel quality information by a training sequence (pilot), these work solved a resource allocation problem by mathematical modeling. However, our work proposes a WLAN protocol that practically considers channel estimation overhead. Many practical measurement studies have been conducted to show the existence of frequency diversity. Among these, the most relevant to our work are measurement studies in the 2.4GHz/5GHz ISM bands [8], [12], [16], [17]. Also, in WLANs, several frequency diversity-aware schemes have been proposed [8], [9], [12]. The authors of [8] introduced a practical rate adaptation scheme based on the effective SNR (eSNR) which is a channel metric that can consider frequency diversity. In FARA [9], a transmitter can send multiple packets to multiple receivers concurrently based on the OFDM technique. Thus it needs not consider the time-sync problem arisen when multiple packets combine at a receiver. FARA can be used in the downlink of a WLAN and is complementary to our work in that our work is mainly focusing on the uplink. Finally, the authors of [12] proposed a diversity-aware WLAN that uses an adaptive interleaver and an forward error correction (FEC) scheme based on per-subcarrier channel state information (CSI). It adopts different domain approaches such as a per-subcarrier FEC method and an interleaver and hence is orthogonal to our work.

C. Machine Learning

An explosive increase of digital data has led to spotlight in the use of machine learning (ML) techniques to extract engineering information from voluminous data [22], [23], [24], [25], [27], [28]. In the field of networking, Internet application traffic classification has been conducted using Naive Bayes [24], C4.5 decision tree or Naive Bayesian tree [25], and support vector machine (SVM) [27] algorithms. And [22], [23] have performed the scientifically grounded performance comparison among the several methods including well-known ML algorithms in terms of the classification accuracy. In the area of wireless networking, the authors of [28] have developed Airshark that extracts the signal-level features using the functionality provided by a Wi-Fi card and identifies multiple non-Wi-Fi signals like Zigbee, cordless phone, Bluetooth, etc based on the SVM [27] algorithm. To the best of our knowledge, our work is the first work that applies ML methods to a PHY/MAC WLAN protocol.

VII. Conclusion

We propose D-Fi, a novel Wi-Fi PHY/MAC protocol based on the OFDM technique. The proposed protocol performs channel contention and estimation at the same time using a Bloom filter to efficiently exploit frequency diversity. In addition, to address the ambiguity problem, an intrinsic weakness of a Bloom filter, we propose a multi channel backoff algorithm and apply machine learning algorithms to the D-Fi PHY/MAC protocol. We showed the feasibility of D-Fi by implementing it on a USRP/GNUradio testbed. Moreover, we have shown that the D-Fi PHY/MAC protocol can exploit frequency diversity with partial channel information through our trace-driven simulations.

Our future research directions include a full implementation of a D-Fi network using commodity devices for a thorough evaluation of our protocol and extending the D-Fi protocol to support a downlink or an ad-hoc mode operation.

Acknowledgment

We owe special thanks to the anonymous reviewers and Young-myoung Kang for their thorough and helpful feedback, which was tremendously vital to significantly improve the contents and presentation of this paper. This work was supported in part by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MEST) (No. 2010-0027410) and in part by Seoul R&BD Program (PA100062M0209722).

References

[1] IEEE Standard for local and metropolitan area networks Part 11: amendment 5: Enhancements for higher throughput. IEEE Std 802.11n-2009
[2] IEEE Standard for Local and metropolitan area networks Part 16: Air Interface for Fixed and Mobile Broadband Wireless Access Systems.
[3] 3GPP TS 36.201-820: Evolved universal terrestrial radio access (E-UTRA); long term evolution (LTE) physical layer; general description.
[4] A. Goldsmith. Wireless Communications. Cambridge Press 2005.
[5] D. Halperin, W. Huy, A. Shethz, and D. Wetherall. Tool Release: Gathering 802.11n Traces with Channel State Information. In ACM Compuer Communication Review (CCR) 2011.
[6] Y. Shen and E. F. Martinez. Channel Estimation in OFDM Systems. AN3059, Freescale Semiconductor, Inc., Jan. 2006.
[7] R. Crepaldi, J. Lee, R. Etkin, S. Lee, and R. Kravets. CSI-SF: Estimating Wireless Channel State Using CSI Sampling & Fusion. In IEEE Infocom 2012.
[8] D. Halperin, W. Hu, A. Shethy, and D. Wetherall. Predictable 802.11 Packet Delivery from Wireless Channel Measurements. In ACM SIGCOMM 2010.
[9] H. Rahul, F. Edalat, D. Katabi, and C. Sodini. Frequency-Aware Rate Adaptation and MAC Protocols. In ACM MobiCom 2009.
[10] K. Tan, J. Fang, Y. Zhang, S. Chen, L. Shi, J. Zhang, and Y. Zhang. Fine-grained Channel Access in Wireless LAN. In ACM SIGCOMM 2010.
[11] S. Sen, R. R. Choudhury, and S. Nelakuditi. No time to Countdown: Migrating Backoff to the Frequency Domain. In ACM MobiCom 2011.
[12] A. Bhartia, Y. C. Chen, S. Rallapalli, and L. Qiu. Harnessing Frequency Diversity in Multicarrier Wireless Networks. In ACM MobiCom 2011.

[13] X. Feng, J. Zhang, Q. Zhang, and B. Li. Use Your Frequency Wisely: Explore Frequency Domain for Channel Contention and ACK. In IEEE Infocom 2012.

[14] Y. Kang, J. Yoo, J. Lee, and C. Kim. A Distributed Message In Message aware Concurrent Transmission Protocol in IEEE 802.11 WLANs. To appear in EURASIP Journal on Wireless Communications and Networking 2012

[15] K. Wu, H. Li, L. Wang, Y. Yi, Y. Liu, Q. Zhang, and L. Ni. hJam: Attachment Transmission in WLANs. In IEEE Infocom 2012.

[16] G. F. S. Moya, and J. L. Z. Flores. Parameters Of A 2.4GHz Wide Band Radio Channel For WLAN applications. In proc. of the 14th International Conference on Electronics, Communications and Computers (CONIELECOMP '04)

[17] H. Macleod, C. Loadman, and Z. Chen. Experimental Studies Of The 2.4GHz ISM wireless Indoor Channel. In proc. of the 3rd Annual Communication Networks and Services Research Conference (CNSR '05)

[18] A. Broder and M. Mitzenmacher. Network applications of Bloom filters: A survey. Internet Mathematics, vol. 1, no. 4, pp. 485.509, 2005.

[19] H. A. Mahmoud and H. Arslan. Sidelobe Suppression in OFDM-Based Spectrum Sharing Systems Using Adaptive Symbol Transition. IEEE Communication Letters, vol. 12, no. 2, pp. 133.135, Feb 2008.

[20] G. Bianchi and I. Tinnirello. Kalman Filter Estimation of the Number of Competing Terminals in an IEEE 802.11 network. In IEEE INFOCOM 2003.

[21] WEKA tool. http://www.cs.waikato.ac.nz/ml/weka/

[22] H. Kim, K. Claffy, M. Fomenkov, D. Barman, M. Faloutsos, and K. Lee. Internet Traffic Classification Demystified: Myths, Caveats, and the Best Practices. In ACM CoNEXT 2008.

[23] S. Lee, H. Kim, D. Barman, S. Lee, C. Kim, Ted T. Kwon, and Y. Choi. NeTraMark: A Network Traffic classification benchMark. In ACM Compuer Communication Review (CCR) 2011.

[24] A. Moore and D. Zuev. Internet traffic classification using bayesian analysis techniques. In ACM SIGMETRICS, June 2005.

[25] D. M. Farid, N. Harbi, and M. Z. Rahman, Combining naive bayes and decision tree for adaptive intrusion detection, CoRR, vol. abs/1005.4496, Apr. 2010.

[26] N. Williams, S. Zander, and G. Armitage. Evaluating machine learning algorithms for automated network application identification. Technical Report 060401B, CAIA, Swinburne Univ., April 2006.

[27] K. Bennett and C. Campbell. Support vector machines: Hype or hallelujah? ACM SIGKDD Explorations, vol. 2, no.2, pp.1.13, 2000.

[28] S. Rayanchu, A. Patro, and S. Banerjee. Airshark: Detecting Non-WiFi RF Devices using Commodity WiFi Hardware. In ACM IMC 2011.

[29] Y. Zhang and S. Liew. Proportional Fairness in Multi-Channel Multi-Rate Wireless Networks - Part II: The Case of Time-Varying Channels with Application to OFDM Systems. In IEEE Transactions on Wireless Communications (TWC), Vol. 7, No. 9, pp.3457.3467. Sep. 2008.

[30] H. Kwon, S. Kim, and B. Lee. Opportunistic multi-channel CSMA protocol for OFDMA systems. In IEEE Transactions on Wireless Communications (TWC), Vol. 9, No. 5, pp.1552.1557, May 2010.

[31] S. Lee, I. Pefkianakis, A. Meyerson, S. Xu, and S. Lu. Proportional Fair Frequency-Domain Packet Scheduling for 3GPP LTE Uplink. In IEEE Infocom 2009.

[32] H. Yang, F. Ren, C. Lin, and J. Zhang. Frequency-Domain Packet Scheduling for 3GPP LTE Uplink. In IEEE Infocom 2010.

[33] V. Shrivastava, S. Rayanchu, J. Yoon, and S. Banerjee. 802.11n under the microscope. In ACM IMC 2008.

[34] Ettus Research, LLC. http://www.ettus.com.

[35] GNU Radio Project. http://gnuradio.org/redmine/wiki/gnuradio