Analog Bloom Filter: Efficient Simultaneous Query for Wireless Networks

Zhenghao Zhang
Computer Science Department
Florida State University Tallahassee, FL 32306, USA

***Abstract*—In this paper, we study the problem of supporting simultaneous query in wireless networks, where multiple nodes activate Orthogonal Frequency-Division Multiplexing (OFDM) subcarriers to announce the control information. Such simultaneous query can allow the Access Point (AP) to gather node state information in a single query and greatly improve the performance of wireless networks. We leverage the fact that the number of nodes that need to respond to the query is typically much smaller than the total number of associated nodes, such that nodes may be assigned with overlapping resources to reduce the query time. We propose a solution similar to the Bloom filer, called the Analog Bloom Filter (ABF), because it handles continuous analog signals. We propose an algorithm based on the idea of belief propagation which detects the binary states of the nodes according to the signal powers. We also propose to support multi-bit queries with error correction codes and a novel signaling scheme. We evaluate the proposed algorithms with simulations and the results show that they achieve similar or better performances than the existing query schemes while consuming much less resources.**

I. Introduction

In this paper, we study the problem of supporting simultaneous query in wireless networks. The network has a centralized controller, referred to as the Access Point (AP), which may be associated with as many as several hundred nodes. One example of such network is a future TV White Space cell, where each cell may cover an entire building [6], [14]. For such networks, the network efficiency can be significantly improved if the AP can obtain important information via a *simultaneous query* which will lead to simple Medium Access Control (MAC) protocols. For example, simultaneous query has been proposed for a polling-based MAC protocol [12] as well as for obtaining acknowledgments for broadcast messages [4]. This is because many MAC control frames actually carry very few bits of information; however, if the control frames are sent individually, each frame has to include overhead such as the physical layer header for synchronization, MAC header, etc. The simultaneous query mechanism amortizes the overhead over all nodes and achieves better efficiency.

To date, the proposed simultaneous query mechanisms are based on Orthogonal Frequency-Division Multiplexing (OFDM). For example, if the AP wishes to learn the buffer states of the nodes, each node may be assigned with a unique subcarrier. A node may transmit a signal on the subcarrier if its buffer is not empty and the AP can learn the buffer states by detecting energy levels on the subcarriers. Each node needs only transmit one or several OFDM symbols where each symbol can be as short as several microseconds.

In this paper, we also focus on OFDM-based simultaneous query. Existing query schemes assign *dedicated* subcarriers to the nodes. Clearly, challenges arise when the number of nodes in the network is large, especially when nodes wish to activate multiple subcarriers to combat *non-flat fading* or to announce multi-bit messages. One option is to partition the nodes into smaller groups, and let each group respond in turn. The disadvantage of this approach, clearly, is the prolonged response time. We note that the query interval will increase proportionally with the query response time to maintain the same network control overhead; therefore, an increased response time may lead to much larger packet jitters and may hurt the performance of real-time applications.

In this paper, we propose a novel query method inspired by the *Bloom filter*, called the *Analog Bloom Filter (ABF)*. With ABF, each node is assigned with a random set of subcarriers but only the active nodes will transmit; the AP runs an energy detection algorithm to determine if a node is active or to determine the type of messages sent by the node. The ABF is optimized for the type of queries in which the number of active nodes is much smaller than the total number of nodes. We note that this condition is often true, e.g., in the buffer state query, because the number of nodes with buffered data is typically much smaller than the total number of nodes. The existing method is basically a linear scan and is very conservative because every node is allocated with *dedicated* resources including time and the usage of subcarriers although many nodes are idle. The ABF, on the other hand, achieves better efficiency by opportunistically allowing a large number of nodes to share the resources, leveraging the fact that the idle nodes do not transmit signals and do not consume any resource. ABF is different from the Digital Bloom Filter (DBF) because it copes with energy readings which are real numbers, while the readings in some subcarriers may be very weak due to non-flat fading. We propose a belief-propagation algorithm to detect the binary states of the nodes. In addition, we also propose to use *error correction codes* and a novel signaling scheme to support multi-bit queries. We evaluate the proposed ABF with simulations driven by realistic channel models and the results show that ABF can significantly reduce the query time.

The rest of the paper is organized as follows. Section II discusses the related works. Section III discusses binary node state detection with ABF. Section IV discusses multi-bit detection with ABF. Section V evaluates ABF with simulations. Section VI discusses techniques to reduce the number of responders. Section VII concludes the paper.

II. Related Work

Signaling with ODFM subcarriers has been proposed in [4], [12], [11]. In particular, a polling-based MAC protocol was proposed and evaluated with simulation in [12], in which the AP sends polling queries and the nodes respond in different subcarriers to allow the AP to find the set of active nodes. In [11], a contention-based MAC protocol was proposed which uses OFDM subcarriers for more efficient contention resolution. A simultaneous acknowledgment scheme was implemented in [4] which allows multiple nodes to send acknowledgments simultaneously to the AP using different subcarriers. In this paper, we consider the problem of improving query efficiency when the number of active nodes is much smaller than the total number of nodes, and propose new solutions.

The Bloom filter was first proposed in [2] and there have been many works studying the Bloom filter such as [3], [5]. In this paper, we study the unique problems when the inputs are real values with possible errors, which are very different from the inputs to the typical Bloom filter.

The TV White Space has received much attention in recent years [1], [8], [10], [9]. We note that we consider the unique problem of efficient simultaneous querying, which is different from the typical problems studied for TV White Space such as channel selection, primary user avoidance, etc., and our solutions complement existing works.

III. Analog Bloom Filter for Binary Detection

We discuss the ABF for binary detection in this section. Binary detection allows each node to announce one bit as its state. A node that wishes to announce bit 1 is referred to as an *active* node and a node that wishes to announce bit 0 an *idle* node. With ABF, an active node activates n random subcarriers while an idle node remains silent, where n is a system parameter greater than 1. Note that multiple nodes may activate the same subcarrier and the signal at such subcarrier is the summation of the signals from all such nodes. The AP runs an algorithm based on the received signals to infer the states of the nodes.

The key differences between DBF and ABF include the following. First, DBF handles digitized input values while ABF handles real values as inputs. Second, DBF typically need not cope with errors in the input, while the inputs to ABF may be incorrect or noisy due to noise and non-flat fading. With non-flat fading, the received signal powers from the same node can be significantly different on different subcarriers, e.g., by 10 dB. It may happen that the signal power of a subcarrier assigned to an active node is very low which may lead to incorrect estimation of the node state.

For example, Fig. 1 shows the received signal according to the channel model in [13] when $n = 8$ and when there are 4 active nodes. The signals at the subcarriers assigned to one node is marked, where we can see that the signal strengths can vary significantly.

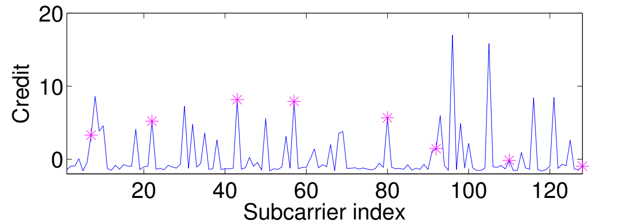


Fig. 1. The received signal at 128 subcarriers with 4 active nodes. $n = 8$. The signal from one node is marked.

A. Complexity Analysis

We first prove that the problem of determining the states of the nodes is NP-complete even under highly idealistic conditions, hence computationally challenging in general. To be more specific, we define the *Ideal ABF Binary Detection (IABFBD)* problem as detecting the binary states of the nodes when 1) the instantaneous channel states of all nodes are known to the AP and 2) the signal is much stronger than the noise such that the noise can be neglected. The nodes should be classified as active or idle, under the constraint that that the summation of the signals from the active nodes should be equal to the measured signals at each subcarrier.

Theorem 1: The IABFBD problem is NP-complete.

Proof: We reduce the Subset Sum problem [7] to the IABFBD problem. In the Subset Sum, a set of U integers and a number V are given, and the goal is to find a subset of integers adding up to V. Given any instance of the Subset Sum problem, we construct an IABFBD instance as follows. For each integer in the Subset Sum instance, create a node. If the integer value is x, let the node's signal amplitude at subcarrier 1 be x and the phase be 0. Let the node's signal strength at all other subcarriers be 0. Let the measured signal amplitude at subcarrier 1 be V. Clearly, a feasible solution to the constructed IABFBD instance is a feasible solution to the Subset Sum instance. ■

B. The ABF Binary Detection Algorithm

We develop a simple algorithm for binary detection. We note that the AP may not have the instantaneous channel states of all nodes, because the query response signals are simply a few OFDM symbols and do not include preambles for channel estimation. Therefore, the algorithm has to rely on non-coherent detection, i.e., using the *power level* on each subcarrier for detection. In addition, the algorithm must be simple because it may have to be implemented in hardware.

1) Preliminaries and Notations: We list the notations to be used in the following:

- We denote the total number of subcarriers as S and the number of subcarriers assigned to a node as n.
- We denote the set of subcarriers assigned to node v_i as Ω_i and the set of nodes assigned with subcarrier s_j as Φ_j.
- We denote the mean and the variance of the noise power as μ and σ, respectively, and assume they can be measured by the AP.

- We denote the received power at subcarrier s_j as ρ_j.
- We denote the *credit* at subcarrier s_j as y_j where $y_j = \rho_j - \mu - 2\sigma$. The rationale is that y_j should reflect the amount contributions subcarrier s_j can provide as evidences for the nodes to be active; hence, the evidence should be positive only if the received signal power is significantly larger than the noise power.
- We denote the total amount of credits node v_i receives as x_i. The amount of credits node v_i receives from subcarrier s_j is denoted as x_i^j; the amount of credits node v_i receives from subcarriers in Ω_i except s_j is denoted as $x_i^{\setminus j}$.
- We say a node v_i in Φ_j is in set Φ_j^+ if $x_i^{\setminus j} > 0$ and if its state has not been decided yet.

2) The Algorithm: Our proposed algorithm is described in Algorithm 1. Basically, it learns from the input signals and infers the amount of credits received by each node, and considers a node active if its credit is above a threshold. This is challenging because a subcarrier will likely be assigned to multiple nodes; the algorithm, without the channel state information, must distribute the credits to the nodes correctly such that the active nodes will likely receive higher credits than the idle nodes.

The algorithm will mark a node as idle or active; once such a decision is made the node is not reexamined. The algorithm may also mark a subcarrier as *used*, after which the subcarrier is not reexamined. The algorithm adopts two preprocessing steps. First, it marks a node v_i as idle if none of the subcarriers in Ω_i has power level greater than $\mu + 4\sigma$. This initial pruning removes the obvious idle nodes and can help reducing the error probability. Second, it evenly distributes the credit of subcarrier s_j to all nodes in Φ_j if $y_j < 0$. This is because the subcarriers with negative credits are almost always those only assigned to the idle nodes who should equally receive the negative credits.

The core of the algorithm is a loop in which it distributes the positive credits among the nodes. The idea is that a node v_i should receive more credits from a subcarrier s_j than other nodes also assigned to s_j, *if v_i receives more credits from other subcarriers.* The rationale is that the credits of the subcarriers assigned to the same active node will likely exhibit strong correlations; therefore, the amount of positive credits a node receives from other subcarriers can be viewed as the *belief* in this node from other subcarriers. In this sense, the algorithms is basically a *belief propagation* algorithm. It can be seen that the credits of a subcarrier will be given to a node proportional to the amount of positive credits received from other subcarriers; no credits will be given to a node if the amount of credits it receives from other subcarriers is negative.

An additional optimization step taken by the algorithm is to finalize the status of an active node once it has become apparent. In each iteration, it checks the node with the largest credit, denoted as v_u, to see if its credit has converged and is greater than $\eta n \sigma$ where $\eta = 1.5$. If yes, it will mark v_u as active and mark subcarriers in Ω_u as used. The rationale is that if it has become obvious that a node v_u is active, the credits of the subcarriers assigned to v_u should not be diverged to other nodes. This is extremely useful when the signal from v_u is strong, in which case even a small fraction of the credits belonging to v_u diverged to an idle node is sufficient to result in a false positive.

Algorithm 1 ABF Binary Detection Algorithm

Mark v_i as idle if none of the subcarriers in Ω_i has power level greater than $\mu + 4\sigma$.
for every subcarrier s_j with negative credit **do**
  for every $v_i \in \Phi_j$ **do**
    $x_i^j \leftarrow \frac{y_j}{|\Phi_j|}$
  end for
end for
while iterated no more than R rounds **do**
  for every unused subcarrier s_j with positive credit **do**
    for every $v_i \in \Phi_j^+$ **do**
      $x_i^j \leftarrow \frac{y_j x_i^{\setminus j}}{\sum_{t \in \Phi_j^+} x_t^{\setminus j}}$
    end for
  end for
  Let v_u be the node with the largest credit.
  if x_u has converged and $x_u > \eta n \sigma$ **then**
    Mark v_u as active. Mark subcarriers in Ω_u as used.
  end if
end while
for every node **do**
  Mark node v_i is active if $x_i > \eta n \sigma$.
end for

We note the following features of the algorithm. First, clearly, *the algorithm is very simple and can be implemented even in hardware.* It can be verified that with a naive implementation, the complexity of the algorithm is $O(RNn^2)$ where N is the total number of nodes. Second, *the credit of an idle node v_i will likely converge to a small value exponentially fast if Ω_i contains less active subcarriers than n.* This is because v_i has to share the credits of the active subcarriers in Ω_i with the active nodes. Initially, an active node will likely receive positive credits from n subcarriers. If Ω_i contains less active subcarriers than n, v_i will likely receive less credits than the active nodes. Therefore, it will get smaller shares of the credits from the subcarriers than the active nodes. Furthermore, getting less credits in the current round will lead to even less credits in the next round, because the amount of credits a node receives is proportional to the amount of credits it receives in the previous round. Therefore, the credits an idle node receives is likely only a fraction of credits it receives in the previous round and its credits will decay exponentially.

C. Parameters Selection and Subcarrier Assignment

1) The value of n: The optimal value of n should be such that $(\frac{nA}{S})^n$ is small where A is the number of active nodes. This is because $(\frac{nA}{S})^n$ is approximately the probability that the subcarriers assigned to an idle node are all active subcarriers;

in case of such an event, the algorithm may incorrectly mark an idle node as active which may lead to further errors because the amount of credits received by an active node may be reduced. As A is not known in advance, n should be selected such that it works reasonably well for a certain range of A. For example, we use $n = 8$ when $S = 128$ and when A is likely in $[1, 8]$.

2) Optimizing the Subcarrier Assignment: A naive assignment of the subcarriers to the nodes is a random assignment, i.e., selecting a set of n subcarriers uniformly among all possible combinations of n subcarriers. However, the core reason to assign multiple subcarriers to the nodes is to combat non-flat fading. Therefore, there is a need to spread out the subcarriers assigned to a node as much as possible. This, however, may be at the cost of increasing the collision probability, i.e., the probability that a subcarrier is assigned to multiple nodes.

We propose a *zone strategy* by which the subcarriers are divided into n zones of equal size (the zone sizes may differ by one if S cannot divide n) and a node will randomly select exactly one subcarrier in each zone. The advantage of this strategy is that it is easy to implement while ensuring that a node will always have subcarriers in each zone, which makes it unlikely that all subcarriers experience deep fading. It is also supported by the following theorem which states that the collision probabilities by the zone strategy and by the naive strategy are actually very close.

Theorem 2: Denote the maximum difference of the collision probabilities of the zone strategy and the naive strategy as g^*. We have

$$g^* \approx (1 - \frac{1}{S})^S - (1 - \frac{n}{S})^{\frac{S}{n}}$$

under two conditions 1) the subcarriers assigned to a node in the naive strategy are randomly selected with possible repeat (this approximation is valid if n is much smaller than S which is often true) and 2) both S and $\frac{S}{n}$ are reasonably large.

Proof: First, with condition 1, the probability that a particular subcarrier assigned to an active node is also assigned to another active node according to the naive strategy can be approximated as $1 - (1 - \frac{1}{S})^{(A-1)n}$ where A is the number of active nodes. The probability that a particular subcarrier assigned to an active node is also assigned to another active node according to the zone strategy is $1 - (1 - \frac{n}{S})^{(A-1)}$. The difference is thus

$$g(A) = (1 - \frac{1}{S})^{(A-1)n} - (1 - \frac{n}{S})^{(A-1)},$$

where we have written it as a function of A. Note that

$$\begin{aligned} g'(A) &= (1 - \frac{1}{S})^{(A-1)n} n \ln(1 - \frac{1}{S}) \\ &\quad -(1 - \frac{n}{S})^{(A-1)} \ln(1 - \frac{n}{S}) \\ &= \frac{n}{S}[(1 - \frac{1}{S})^S \ln(1 - \frac{1}{S})^S](1 - \frac{1}{S})^{(A-1)n-S} \\ &\quad -\frac{n}{S}[(1 - \frac{n}{S})^{\frac{S}{n}} \ln(1 - \frac{n}{S})^{\frac{S}{n}}](1 - \frac{n}{S})^{A-1-\frac{S}{n}} \end{aligned}$$

The two terms in the square parenthesis both converge to $-e^{-1}$; indeed, they are very close when both S and $\frac{S}{n}$ are reasonably large (e.g., when $S = 128$ and $\frac{S}{n} = 16$, the difference is 0.0002). Therefore,

$$g'(A) \approx \frac{n}{Se}[(1 - \frac{n}{S})^{A-1-\frac{S}{n}} - (1 - \frac{1}{S})^{(A-1)n-S}]$$

and it can be easily verified that $g'(A) > 0$ when $A - 1 < \frac{S}{n}$; i.e., $g(A)$ achieves the maximum when $A - 1 = \frac{S}{n}$. ■

Clearly, g^* is a very small value. For example, when $S = 128$ and $\frac{S}{n} = 16$, the collision probability is 0.17 when $A = 4$ according to the naive strategy while $g^* = 0.01$.

D. Multiple Antennas

We also propose to combine the signals from multiple receiving antennas to better cope with non-flat fading, because a subcarrier in deep fading at one antenna will unlikely suffer the same deep fading at other antennas. In our algorithm, we basically use the total received power from multiple antennas as the input power value to the algorithm.

IV. Multi-bit Queries

In many cases, the nodes may wish to announce multiple bits to the AP. For example, a node may maintain multiple queues of different priorities, and may wish to announce the index of the non-empty queue with the highest priority. One possibility is to perform multiple single bit queries to obtain multiple bits. Again, this may be suboptimal especially when the number of active nodes is much smaller than the number of associated nodes.

Extending the idea of the ABF in the binary case, we note that we may still assign a set of subcarriers to each node, while a node may activate the assigned subcarriers according to certain patterns to announce multiple bits. We may view the binary case as a special case where there is only one bit to be announced such that the pattern is either to activate all subcarriers or to activate no subcarrier. Clearly, the challenge with multi-bit queries is to design the patterns to activate the subcarriers such that the query is robust against non-flat fading and collisions in subcarriers.

Our solution is based on *Error Correction Codes (ECC)* and we first describe the basic idea with a simple query scheme.

A. A Simple Query Scheme with ECC

According to an (n, k, d) code, a codeword has n bits in which k are message bits; any two codewords differ in at least d bit locations. Each node is randomly assigned n subcarriers. To announce a k-bit message, a node first encodes the message into a n-bit codeword, then activates the assigned subcarriers corresponding to the '1's in the codeword and leave other assigned subcarriers idle. We define an active node as a node that sends non-zero messages; other nodes are idle nodes. The AP first runs the ABF Binary Detection Algorithm to determine the state of each node, i.e., whether a node is active or not. Then, for every active node, the AP checks the bit patterns of the subcarriers assigned to the node. If there is no collision in the subcarriers, the AP can decode the message bits correctly if the number of errors is less than $d/2$. If there are collisions, the AP can discard the bits involved in the collision, i.e., treat such bits as *erasures* in the codeword. The AP can still decode the message bits correctly if the number of errors in the remaining of the codeword is less than $d'/2$ where d' is the minimum distance of the codeword after removing the erasure bits.

B. Combining ECC with a Novel Signaling Scheme

The main challenge when applying ECC is to reduce the node state error probability to a desirable level. We note that the first step of the decoding is to find the set of active nodes. However, with the simple query scheme described earlier, an active node will only activate a subset of subcarriers assigned to it such that it will receive negative credits from the subcarriers it does not activate. The ABF Binary Detection Algorithm has to be modified; however, reducing the false negative probability is always at the cost of increasing the false positive probability.

We therefore propose a novel signaling scheme, in which a node uses *two adjacent subcarriers* to announce one bit. To be more specific, let γ be a constant less than 1. To announce bit '0' on subcarriers s_j and s_{j+1}, a node sets the transmission amplitude of subcarriers s_j and s_{j+1} to be γ unit and 1 unit respectively; to announce bit '1', a node sets the transmission amplitude of subcarriers s_j and s_{j+1} to be 1 unit and γ unit respectively. The AP can use the signals at subcarriers s_j and s_{j+1} to determine the value of the bit.

The advantage of this signaling scheme is that a node activates all assigned subcarriers hence the node states can be detected more accurately. A node must use two adjacent subcarriers because the adjacent subcarriers will experience very similar fading such that their relative signal powers can be used to infer the bit value; nonadjacent subcarriers may experience different fading and their relative signal powers may not reflect the correct bit value. The constant γ is chosen based on a tradeoff between sending enough energy on each subcarrier and the probability of detecting the bit value correctly; we set γ to be $1/2$ in our current implementation. Clearly, with this scheme, more subcarriers are used for each node and collision may occur more often. We note that the error correction code has resistance against collisions; in addition, it is expected that the multi-bit query will be used when only a small number of nodes may respond.

We make a further optimization according to a *soft* decoding scheme. Loosely speaking, the soft decoding scheme estimates the total amount of work the noise has to do to convert a particular codeword into the received vector, and picks the codeword that requires the minimum work. To be more specific, we denote codeword w as $[b_1^w, b_2^w, \ldots, b_n^w]$. Without loss of generality, suppose the assigned subcarriers are s_1 to s_{2n}. The *distance* at bit i with respect to codeword w is denoted as q_i^w and is calculated according to

$$q_i^w = \sum_{t=1}^{T} |Y^t_{2i-1+b_i^w} - \gamma Y^t_{2i-b_i^w}|^2, \tag{1}$$

where T is the number of antennas and Y_j^t is the complex signal value received at subcarrier s_j from antenna t. The *codeword distance* for codeword w is defined as $\sum_{i=1}^n q_i^w$. The decoding algorithm basically calculates the codeword distances of all codewords and picks the minimum. Note that such a linear search can be afforded because each node announces only a few bits and the number of codewords is small.

We now explain why the distance at bit i is calculated according to Equation 1. In the following we assume $b_i^w = 0$; the same arguments apply when $b_i^w = 1$. We note that the signal received by the AP at s_{2i-1} and s_{2i} from antenna t can be written as

$$Y^t_{2i-1} = a\gamma + v_1, Y^t_{2i} = a + v_2,$$

where a is determined by the channel state and v_1 and v_2 are the complex noise values. Reversely, if a is known, the likelihood that b_i^w is 0 monotonically decreases with $|Y^t_{2i-1} - a\gamma|^2 + |Y^t_{2i} - a|^2$ if the noise is white Gaussian. However, as there are three unknowns (a, v_1 and v_2) but only two observations (Y^t_{2i-1} and Y^t_{2i}), the exact value of a cannot be determined. Therefore, we adopt the policy which selects a to minimize $|Y^t_{2i-1} - a\gamma|^2 + |Y^t_{2i} - a|^2$. To be more specific, we basically solve the problem of minimizing $|v_1|^2 + |v_2|^2$ subject to the constraint that $Y^t_{2i-1} - v_1 = \gamma(Y^t_{2i} - v_2)$. Clearly, the objective function is minimized when $v_2 = 0$ and $v_1 = Y^t_{2i-1} - \gamma Y^t_{2i}$.

C. An Example Code

The exact code to be adopted depends on the number of message bits and the desired error correction capability. If to support 7 priorities, a $(7, 3, 4)$ linear binary code may be used with generating matrix as follows:

$$\begin{pmatrix} 1 & 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 \end{pmatrix}'$$

V. Evaluations

We evaluate the proposed ABF with Matlab simulations in a setting targeting at future TV White Space networks. We use the channel model initially developed for the indoor 802.11n channels [13]; we set the carrier frequency to be 700 MHz and the channel bandwidth to be 6 MHz to simulate the channels in the TV band. Two antennas are simulated. White Gaussian noise is added to the signal, and the Signal to Noise Ratio (SNR) is the average SNR at the active subcarriers. The number of associated nodes is 128 and the number of subcarriers is also 128.

A. Binary Detection

We compare with the Collision-Free Query (CFQ) method in which each node is assigned with one or multiple *dedicated* subcarriers. For a fair comparison, the CFQ also uses multiple antennas; a node is considered active if the total amount of power received in the assigned subcarriers is greater than $n(\mu + 4\sigma)$ where n is the number of assigned subcarriers. Unless otherwise specified, $n = 8$ for ABF. We use the False Positive (FP) ratio and the False Negative (FN) ratio

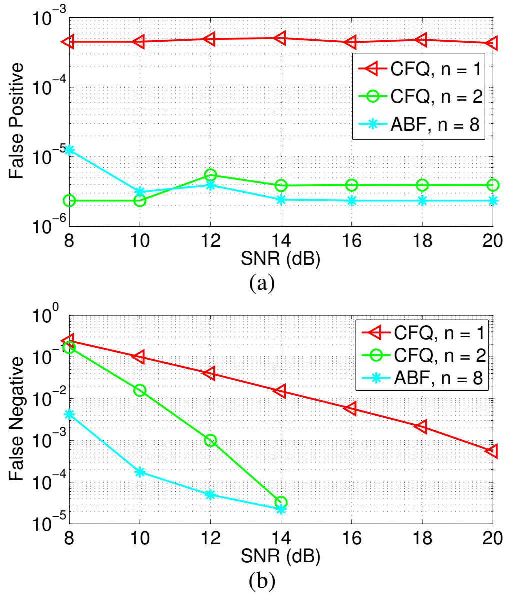


Fig. 2. The performance of CFQ and ABF as a function of SNR. $n = 8$ for ABF. (a). False positive. (b). False negative.

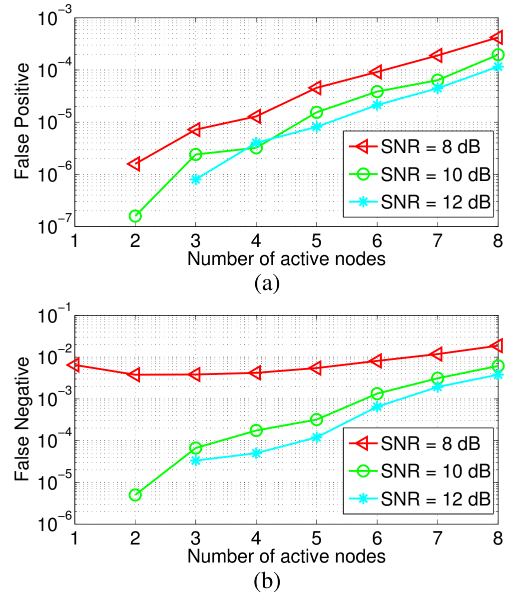


Fig. 3. The performance ABF as a function of the number of active nodes. $n = 8$ for ABF. (a). False positive. (b). False negative.

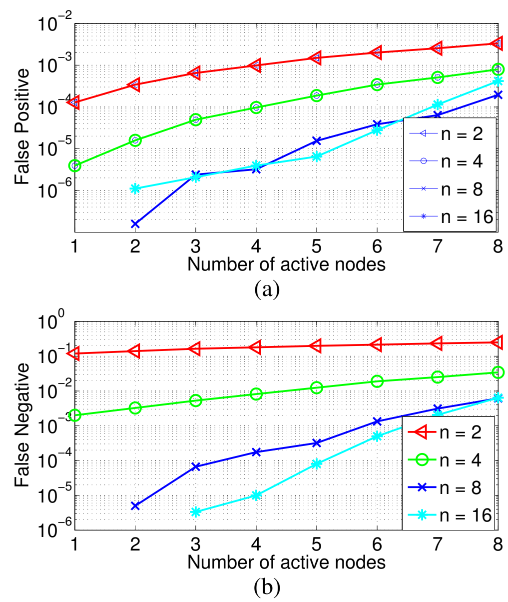


Fig. 4. The performance of ABF as a function of n. SNR = 10 dB. (a). False positive. (b). False negative.

as the performance metric, where an FP event occurs when an idle node is considered active and an FN event occurs when an active node is considered idle. We consider an FP ratio reasonable if it is around 10^{-5} or lower, i.e., if an FP event occurs around every 1000 queries or more. We consider a FN ratio reasonable if it is around 10^{-3} or lower. For a particular set of parameters, if the FP or the FN does not occur after 100,000 queries, the data point is not shown in the figure.

Fig. 2 shows the performance of CFQ and ABF when there are 4 active nodes and when the SNRs are between 8 dB to 20 dB. We can see that the performance of CFQ is poor when $n = 1$, but is reasonable when $n = 2$ with SNR 14 dB or higher. However, ABF achieves better or close performance as CFQ when $n = 2$ for CFQ. Therefore, ABF is able to cut the query time by factor of 2 in this simulation setting.

Unlike CFQ, ABF is sensitive to the number of active nodes because nodes share resources. Fig. 3 shows the performance of ABF as a function of the number of active nodes when the SNRs are 8 dB, 10 dB, and 12 dB, respectively. We can see that the performance is still acceptable up to 5 nodes when the SNR is 10 dB or higher.

We also test ABF for various values of n. Fig. 4 shows that the performance of ABF when there are 4 active nodes at SNR of 10 dB. We can see that the performances when $n = 8$ and $n = 16$ are significantly better than those when $n = 2$ and $n = 4$. It can also be seen that $n = 8$ and $n = 16$ lead to similar performances. $n = 8$ is more resilient to larger number of nodes which is why we recommend $n = 8$.

B. Multi-bit Detection

We also test the performance of ABF for multi-bit queries measured by the decoding error ratio. The example code in Section IV-C is used. Fig. 5 shows the decoding error ratio for 2 to 4 active nodes when the SNRs are between 8 dB to 20 dB. We can see that unsurprisingly, multi-bit queries require higher SNRs than single bit queries and are also more sensitive to the number of active nodes. Reasonable performance is achieved when the SNR is 14 dB or higher with 2 active nodes.

VI. Reducing the Number of Responders

We note that ABF critically depends on the number of responders being much less than the number of nodes. In this section, we discuss techniques that can be used to reduce the number of responders for certain type of queries.

A. Silencing the Known Active Nodes

At any time, the set of active nodes in the network may be categorized as *known* or *new*, where an active node is known if the AP is aware of its state and new otherwise. Clearly, only

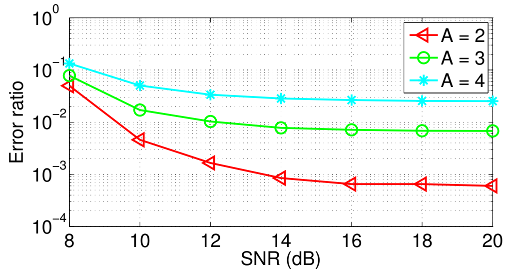
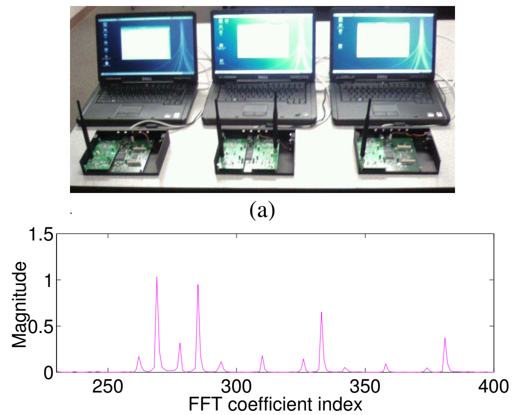


Fig. 5. The decoding error ratio of multi-bit query.



(a)

(b)

Fig. 6. (a). Devices. (b). The measured FFT coefficient magnitude.

the new active nodes should respond to the query, which will significantly reduce the number of responders. For example, a node may receive multiple packets from the upper layer. To send the first packet, it must respond to a buffer state query to request a transmission opportunity from the AP. It may piggyback its buffer state information with the first packet to inform the AP such that it does not need to respond to further queries. The problem of maintaining the coherence of the queue states between the AP and the nodes has been studied in [15] and the solutions can be adopted.

B. Dividing the Subcarrier into Multiple Frequency Slots

We may further divide the bandwidth of a subcarrier into multiple *frequency slots* such that a subcarrier can be used by multiple nodes without interfering with each other. We note that in the response message, the signal on a subcarrier does not change and hence occupies a very small bandwidth. A node can multiply its baseband signal with a fixed frequency offset to shift the signal in the frequency domain and avoid the signals from other nodes. We have verified this the GNU Software Defined Radio (SDR) experiments. We employ two senders assigned with the same 8 consecutive subcarriers, where one sender activates all 8 subcarriers with a smaller power and the other sender activates 4 subcarriers with a larger power. Fig. 6(a) shows the device we use. Fig. 6(b) shows the result after a 4,096 point FFT, where we can see that an active subcarrier occupies only several significant FFT coefficients and the signals of the two senders are clearly differentiable.

VII. Conclusion

In this paper, we study the problem of supporting simultaneous query where nodes activate random OFDM subcarriers as the responses to the query. We propose the Analog Bloom Filter (ABF) which exploits the fact that the number of nodes that should respond to the query is usually much smaller than the number of associated nodes, such that many nodes may share the OFDM subcarriers to reduce the query response time. We design an algorithm to detect the binary states of the nodes based on belief propagation. We also propose to use error correction codes and a novel signaling scheme to support multi-bit queries. We evaluate the proposed algorithms with simulations and the results show that ABF achieves similar or better performance compared to the existing collision-free query schemes with much less query time. Our future work includes incorporating ABF into MAC protocols combined with the techniques discussed in Section VI, and network performance analysis under the new protocol.

References

[1] P. Bahl, R. Chandra, T. Moscibroda, R. Murty, and M. Welsh. White Space Networking with Wi-Fi like Connectivity. In *ACM SIGCOMM*, 2009.

[2] B. Bloom. Space/time Trade-offs in Hash Coding with Allowable Errors. *Communications of the ACM*, pages 422–426.

[3] J. Bruck, J. Gao, and A. Jiang. Weighted Bloom Filter. In *IEEE International Symposium on Information Theory (ISIT'06)*, July 2006.

[4] A. Dutta, D. Saha, D. Grunwald, and D. Sicker. SMACK: a smart acknowledgment scheme for broadcast messages in wireless networks. In *ACM SIGCOMM*, 2009.

[5] A. Goel and P. Gupta. Small subset queries and Bloom filters using ternary associative memories. In *ACM Sigmetrics*, 2010.

[6] Kit Eaton. Google reveals secret hospital experiment to test whitespace wireless. http://www.fastcompany.com/1688950/google-already-testing-whitespace-broadband-gear-in-ohio.

[7] S. Martello and P. Toth. Subset-sum problem. *Knapsack problems: Algorithms and computer interpretations, Wiley-Interscience*, pages 105–136, 1990.

[8] R. Murty, R. Chandra, T. Moscibroda, and P. Bahl. SenseLess: A Database-Driven White Spaces Network. In *IEEE Symposium on Dynamic Spectrum Access Networks (DySpan)*, 2011.

[9] G. Nychis, R. Chandra, T. Moscibroda, I. Tashev, and P. Steenkiste. Reclaiming the White Spaces: Spectrum Efficient Coexistence with Primary Users. In *CoNEXT*, 2011.

[10] B. Radunovic, A. Proutiere, D. Gunawardena, and P. Key. Dynamic Channel, Rate Selection and Scheduling for White Spaces. In *CoNEXT*, 2011.

[11] B. Roman, I. Wassell, and I. Chatzigeorgiou. Scalable Cross-Layer Wireless Access Control Using Multi-Carrier Burst Contention. *IEEE Journal on Selected Areas in Communications*, pages 113–128, Jan. 2011.

[12] D. Saha, A. Dutta, D. Grunwald, and D. Sicker. PHY aided MAC: A new paradigm. In *IEEE INFOCOM*, April 2009.

[13] L. Schumacher. Matlab implementation of the IEEE 802.11 TGn Channel Model. http://www.info.fundp.ac.be/ lsc/Research/IEEE_80211_HTSG_CMSC.

[14] Todd Bishop. WiFi over TV spectrum: Microsoft tests white spaces in U.K. http://www.geekwire.com/2011/wifi-tv-signals-microsoft-testing-white-spaces-uk.

[15] S. Zhou and Z. Zhang. cMAC: A centralized MAC protocol for high speed wireless LANs. In *IEEE GLOBECOM*, 2011.