# Single-Radio Adaptive Channel Algorithm for Spectrum Agile Wireless Ad Hoc Networks

Liangping Ma, Chien-Chung Shen, and Bo Ryu

*Abstract*—**This paper presents the Single-Radio Adaptive Channel (SRAC) algorithm which enables dynamic spectrum access in multi-hop wireless ad hoc networks where each node has only one half-duplex radio (transceiver). Designed as a relatively independent module, SRAC can upgrade various existing single-radio legacy Medium Access Control (MAC) protocols to be dynamic spectrum access capable, achieving efficient use of the spectrum, relaxing their operating conditions, and naturally supporting multicast applications. The SRAC algorithm is characterized by three features: (a) dynamic channelization in response to jamming, primary spectrum users and channel load, (b) "cross channel communications", and (c) as-needed use of spectrum. We evaluate the performance of SRAC through analysis and QualNet simulations.**

*Index Terms*— **Medium Access Control (MAC), ad hoc networks, dynamic spectrum access, multicast, jamming, primary spectrum users.**

## I. INTRODUCTION

Spectrum availability imposes a significant constraint on the performance of wireless ad hoc networks. This is rooted in Gupta and Kumar's network capacity theory [1] and in Shannon's channel capacity theory [2]. The spectrum suitable for wireless communications has largely been allocated statically to the so-called primary spectrum users, and it has become more and more difficult to find unallocated spectrum for new wireless applications. On the other hand, recent studies show that the allocated spectrum is well under-utilized in most areas [3]. The discrepancy in spectrum allocation and utilization suggests a solution: spectrum availability will no longer be a problem if wireless nodes are allowed to access the allocated spectrum that is not currently being used by the primary spectrum users. As the primary spectrum users come and go, the spectrum used by wireless nodes becomes dynamic. DARPA's XG program [4] is one pioneering effort to promote this concept. Research has been done to develop mechanisms to detect the presence of primary spectrum users, and to access the idle spectrum in ways that conform to certain spectrum policies [5], [6], [7].

As a result of dynamic spectrum access, and due to the fact that different nodes in a wireless ad hoc network may be affected by different primary spectrum users that use different spectrum, there may be multiple dynamic channels being used in a wireless ad hoc network at any given point of time. In this sense, a dynamic spectrum access network is essentially a multi-channel network, except that the channels are dynamic. If each node chooses its own dyanmic channel disregarding other nodes' choices, the network might be partitioned into many small fragments, each on a different dynamic channel, thus hindering effective communications. To avoid this, coordination must be taken into account in dynamic channel selection. Efforts on this have been made by [8], [9], [10].

There exist many legacy MAC protocols designed for wireless ad hoc networks operating in static spectrum environments. They alone are inadequate to materialize dynamic spectrum access. However, with physical layer support for dynamic channelization and switching, and with the knowledge of the dynamic channel of the receiver, legacy MAC protocols can be enhanced to achieve dynamic spectrum access capability. In this paper, we focus on single-radio legacy MAC protocols, and propose the Single Radio Adaptive Channel (SRAC) algorithm to upgrade them to be capable of dynamic spectrum access.

The SRAC algorithm also handles jamming, assuming that the node has the ability to detect jamming. More importantly, the SRAC algorithm is efficient in supporting multicast (one to many) communications, which is often implemented as broadcast at the MAC layer[11]. Multicast is important to many applications, e.g., a group leader giving commands to group members in tactical environments[12], but it is not well supported by existing multi-channel MAC protocols. In some of these protocols, nodes are distributed on a number of different channels regardless of the capacity demand, and as a result, in order to make a broadcast reach all neighbors multiple transmissions are needed. In some other multi-channel MAC protocols, broadcast is carried out in a dedicated control channel. However, such a channel may not even exist due to the heterogeneity of spectrum availability across a dynamic

---

This work is supported in part by National Science Foundation under grants CNS-0347460 and CNS-0335302. Liangping Ma and Bo Ryu are with San Diego Research Center, Inc., San Diego, CA 92121 USA (e-mail: {lma,bo.ryu}@sdrcinc.net), Chien-Chung Shen is with the Department of Computer and Information Sciences, University of Delaware, Newark, DE 19716, (e-mail:cshen@eecis.udel.edu).

spectrum access network and even if it does it may lead to the control channel saturation problem[13].

The proposed SRAC algorithm addresses this issue while providing various additional support for dynamic spectrum access. In particular, SRAC offers the following features:

- a feasible adaptive channelization scheme,
- as-needed use of spectrum, which naturally results in efficient support for multicast,
- obviates the need for a transmitter and a receiver to be both interference free for the success of a transmission, thus relaxing the operating conditions,
- can use a single radio for everything, including spectrum sensing, control information exchange, and data transmission, and
- does not rely on network-wide time synchronization.

The rest of the paper is organized as follows. Section II briefly reviews related work and explains the motivation of this paper, Section III presents the details of the SRAC algorithm, Section IV provides the simulation results, Section V briefly discusses hardware implementations, and Section VI concludes the paper.

## II. RELATED WORK AND MOTIVATION

There are a number of MAC protocols recently proposed in the literature for dynamic spectrum access. In [10], a theoretically optimal MAC protocol based on the Partially Observable Markov Decision Process (POMDP) was proposed for time slotted network systems. It requires network wide synchronization, which, however, is not easy to achieve in a typical distributed wireless network in a cost effective manner [14][15]. In [9], the Dynamic Open Spectrum Sharing (DOSS) protocol was proposed to allow rather flexible use of the available spectrum, which, although eliminates the hidden and the exposed terminal problems in dynamic spectrum access environments, relies on a common control channel. Due to heterogeneity in spectrum use by jammers or primary spectrum users, different nodes may have different available spectrum, and a common control channel may not always exist. In addition, a fixed common control channel may result in the control channel saturation problem[13]. In [16], nodes are grouped according to local spectrum availability, and a distributed voting scheme is then used for each group to adaptively select its own common communication channel. This approach obviates the need for a network-wide common control channel, but it incurs group-wide communication overhead.

Many legacy multi-channel MAC protocols have been proposed for traditional static spectrum access environments. They are closely related to the dynamic spectrum access research because dynamic spectrum access essentially results in multiple channels for the nodes to access, although the channels are dynamic in nature. Adopting the terminologies in [17], multi-channel MAC protocols can be classified into four categories: *dedicated control channel, common hopping, split phase, and multiple rendezvous.* In *dedicated control channel*, two radios (transceivers) are used with one radio assigned to the dedicated control channel for control information exchange and the other one assigned to the data channels for data transmission. An example of this approach is the Dynamic Channel Assignment (DCA) algorithm [18]. The other three categories use only one radio. In *common hopping*, nodes hop together quickly, and a pair of nodes stop hopping if they agree to communicate and rejoin the common hopping pattern after the transmission ends. Channel Hopping Multiple Access (CHMA) [19] is an example of this approach. In *split phase*, time is divided into control periods during which all nodes tune to a common control channel for control information exchange, and data periods during which data transfer takes place on data channels. The Multichannel Access Protocol (MAP) [20] is an example of this approach. In *multiple rendezvous*, each node has a unique pseudo random channel hopping sequence, and in order to communicate the transmitter jumps to the receiver's hopping sequence. Slotted Seeded Channel Hopping (SSCH) [13] is an example of this approach. These legacy multi-channel MAC protocols share some of the shortcomings with the dynamic spectrum access MAC protocols reviewed above. The *dedicated control channel* protocols and *split phase* protocols have the same problem as DOSS. The *split phase* protocols and *multiple rendezvous* protocols have the synchronization problem as the one proposed in [10].

All these MAC protocols, either non-legacy or legacy, have various limitations in supporting dynamic spectrum access. First, the available spectrum is either not fully utilized, or is hard to use in practice. Except DOSS [9], all protocols suppose that a set of fixed non-overlapping channels are given, and a node can use only one of the channels at a time. However, if a node were allowed to dynamically combine available channels, better network performance would be achieved. For example, if two adjacent channels are available, they can be combined as a new channel to double the link capacity. This combination of channels is feasible with the Software Defined Radio (SDR) technology [21]. DOSS [9] improves on this by allowing the use of arbitrary segments of spectrum for communications, but the same scheme poses challenges to computation resources, thus making it difficult to implement in practice.

They are ineffective in supporting multicast [12], [11].

As mentioned early, to support multicast applications, the MAC layer only needs to support broadcast. The wireless medium is broadcast in nature, that is, a single shot of transmission could be potentially heard by all receivers. In single-channel environments, broadcast is done effectively, since all nodes are on the same channel. In multi-channel environments, however, nodes may be on different channels. The sender has to take one of the two approaches: (1) doing separate broadcast transmissions on each channel; (2) informing the receivers to go to a common channel by transmitting a short notice on every channel, and then doing broadcast on the common channel. In either case, it requires multiple transmissions, and the multicast efficiency is inversely proportional to the number of channels on which the receivers are. This motivates us to minimize the number of different channels being used in a network provided that there are enough number of channels to support desired network performance. Alternatively, broadcast can be done on a dedicated control channel, as done in the DCA protocol [18]. However, this may lead to the *control channel saturation* problem [13]. Nevertheless, if there is more spectrum available, we may increase the bandwidth via dynamic channelization to solve the *control channel saturation* problem.

They (except DOSS) hold an inappropriate precondition on the success of a wireless transmission, that is, not only the receiver but also the transmitter should be interference free. This pre-condition is unnecessarily stringent, however, since interference at the transmitter alone has no impact on the fate of a transmission. We do not need to require a channel to be idle when a transmitter transmits provided that the transmission does not disrupt primary spectrum users and other legitimate users. Otherwise, possible communications would not be allowed. To see this, consider an example in Fig.1, where nodes A and B try to communicate in the presence of two jamming sources. The shade indicates the area affected by jamming. There is no common spectrum where both node A and node B are free of jamming. None of the MAC protocols (except DOSS) will think communications between nodes A and B is possible. But if we notice the difference between the channel state requirements for the transmitter and the receiver, we can design a scheme to enable the communications. Specifically, node A transmits over channel 3, where node B can listen without interference, and node B transmits over channel 4, where node A can listen without interference. We call this scheme *cross channel communication* since the two-way MAC layer communications takes place across two channels, and we will see shortly that it plays an important role in dynamic spectrum access networks.

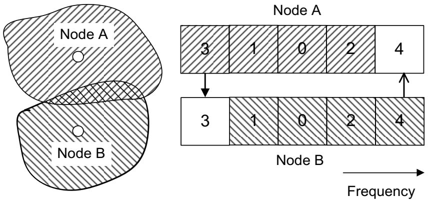


Fig. 1. Nodes A and B under hostile jamming can still communicate even if there is no common idle spectrum; the shades indicate the spectrum being jammed.

## III. THE SRAC ALGORITHM

### A. Assumptions and System Architecture

We make the following assumptions in the design of the SRAC algorithm:

- Each node has a single half-duplex radio (transceiver). When it transmits, it cannot receive, and vice versa.
- The radio can operate at multiple evenly spaced carrier frequencies, and can adjust its filters to pass a number of different bandwidths.
- The node is already capable of detecting the presence of jammers and primary spectrum users.

The first assumption is quite relaxed, as many commercially available radios used for data communications, such as WiFi cards, indeed operate in the half-duplex mode. The second assumption requires the radio to have dynamic channel switching capability, which is also quite relaxed and is in fact offered by many products such as WiFi cards and Software Defined Radios [21], although they may switch channels at quite different speeds. The last assumption is usually a difficult one to meet in practice, because there are various ways to jam a distributed wireless network [22]. For instance, a jammer can disrupt normal packet reception by either continuously sending a signal coded from random bits or sending fake legitimate packets. The detection of primary users is also difficult, because it is hard to estimate the impact of a transmission on potential primary spectrum users based on the observation at a transmitter alone [6][7]. In this paper, we do not address the methods of detecting jammers and primary spectrum users, which are discussed in details in [22], [6], [7]. Instead, we focus on the MAC protocol design, assuming that each node can reliably detect jammers and primary spectrum users.

The objective for the SRAC algorithm is to provide channel availability information to each node based only on local information so that existing legacy single-radio MAC protocols can be leveraged to provide other important MAC functionalities such as Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) [23] and

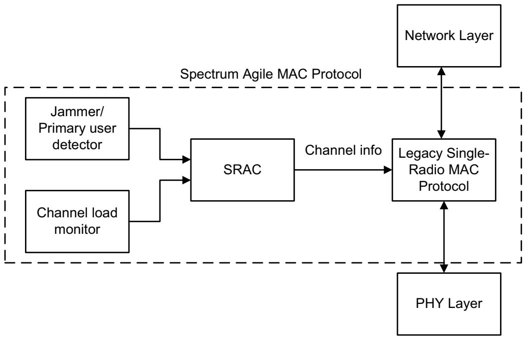


Fig. 2. The SRAC algorithm combined with a single-channel MAC protocol forms a spectrum agile MAC protocol.

Automatic Repeat Request(ARQ) [24] in a dynamic spectrum environment. Note that the legacy MAC protocols could be very different, some being asynchronous such as the IEEE802.11 DCF, and some being synchronous such as time-slotted Node Activation Multiple Access (NAMA) [25]. The SRAC algorithm is designed as a relatively independent module for broader applicability. The place of the SRAC algorithm in the system architecture is illustrated in Fig. 2. With the SRAC algorithm, a legacy MAC protocol is transformed to a new spectrum agile MAC protocol.

## *B. Key Features of SRAC*

### B.1 Adaptive Channelization

Adaptive channelization is a trade off between performance and practicality. In traditional approaches, a radio is pre-configured to use only a few fixed channels and can use only one at a time. This imposes a stringent constraint on the usable spectrum and hence a strong limitation on the network performance, as discussed early in Section II. On the other hand, ideally, any piece of spectrum is a permissible channel, meaning that spectrum can be channelized with essentially arbitrary values for carrier frequency and bandwidth. However, this scheme inevitably leads to challenges in designing the required radio transceivers. To overcome these problems, we propose an adaptive discrete channelization scheme, where a radio dynamically combines multiple fixed channels based on its needs to form a new channel.

Specifically, we define an *atomic channel* to be one that has the minimum bandwidth $b$ (Hz) of all permissible channels and takes a set of discrete values for carrier frequency. The bandwidth $b$ is set such that it is wide enough for an atomic channel to be used alone as a communication channel. Atomic channels can also be combined together to form a *composite channel*. Define channel $C_0$, whose carrier frequency is $f_0$ and bandwidth is $b$. The permissible carrier frequencies of composite channels are evenly spaced, and they can be represented by $f = f_0 + mb$, where $m = 0, \pm 1, \pm 2, ....$ The bandwidth of a permissible channel takes odd-number multiples of $b$, i.e., $kb$, where $k = 1, 3, 5, ....$ The reason of allowing only odd numbers is that otherwise, if $k$ were even the radio would need to support additional carrier frequencies $f_0 + (m + 1/2)b$. With these notations, any permissible channel can be represented by a pair of integers $(m, k)$, which is illustrated in Fig. 3.

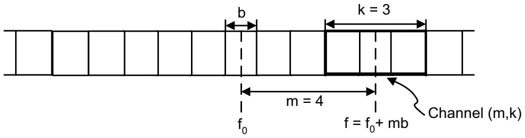


Fig. 3. The channel marked by the thick line is represented by $(m, k) = (4, 3)$, which designates carrier frequency $f = f_0 + 4b$ and bandwidth $3b$ (Hz).

Given $n$ atomic channels, a legacy MAC protocol views $n$ permissible channels, while SRAC sees a total number of $(n + 1)^2/4$ permissible channels if $n$ is odd, and $n(n+2)/4+1$ permissible channels if $n$ is even. The gain in the number of permissible channels is approximately $n/4$ times. Thus, SRAC can choose from a much larger pool of candidate channels, which include the channels viewed as permissible by legacy MAC protocols, to approximate the available spectrum, thus resulting in better network performance.

### B.2 As-Needed Use of Spectrum

Existing legacy multi-channel MAC protocols attempt to increase network capacity by utilizing multiple channels in parallel. This approach indeed improves unicast performance. However, as explained in Section II, this approach distributes nodes on different channels, and is thus difficult to take advantage of the broadcast nature of the wireless medium that could provide efficient support for multicast. Some legacy multi-channel MAC protocols resort to a common control channel for broadcast, but, as discussed early, such a channel may not exist across the network, and even if it exists, the control channel saturation problem may occur [13].

To keep the number of channels low for better multicast support while meeting the capacity demand, we propose that a node is not allowed to change its *receive channel* unless required by its own needs: to avoid jamming and primary spectrum users, or to provide more capacity in response to high channel load, where *receive channel* is defined as the channel over which a node receives. If initialized to share a common receive channel, nodes will tend to keep sharing that common *receive channel* to the extent possible. Be cautioned, the definition does not prohibit a node from transmitting on its own *receive channel*. The terminology *receive channel* is introduced to emphasize the following obvious but often overlooked fact.

**Observation 1:** For a wireless transmission to be successful, the transmitter does not need to be interference free. □

This is true because the success of a transmission is determined by the signal quality at the receiver, which is unnecessarily related to the interference at the transmitter.

Given the jamming, primary spectrum use and channel load conditions, a node will have a stable *receive channel*, which it can tell to its neighbors. To communicate, a neighbor can directly transmit a frame to this node over the *receive channel* without the need for channel negotiation over a common control channel to determine which channel is to be used for receiving. This results in significant reduction in communication overhead.

### B.3 Cross-Channel Communication

We have shown in Fig. 1 that "cross-channel communication" is important for enabling communications when there are multiple jamming sources and there is no common idle spectrum between the transmitter and the receiver. In such scenarios, communications are considered impossible by legacy MAC protocols. To further elaborate on this concept, we consider another example in Fig. 4. As was done in [16], we group nodes that share the same available spectrum together and call them a subnet. To make the channel selection rule completely distributed, we require that each subnet has a *single receive* channel, which only depends on the jamming and primary spectrum use conditions. In this example, all nodes originally use channel 0 as the *receive channel* (which is also the only channel for all transmissions). Then, two jammers start jamming the network, as indicated by the shades. Being jammed, subnet 1 and subnet 3, switch their *receive channels* to channels 4 and 1, respectively. The rest of the network, subnet 2, keeps using the initial *receive channel*. Without "cross-channel communication", subnet 2 can only communicate with either subnet 1 or subnet 3, but not both. The entire network is thus disconnected. However, with "cross-channel communication", three subnets can communicate, as indicated by the arrows in the figure.

The concept of "cross-channel communication" also turns out to be critical to the "as-needed use of spectrum" paradigm. For a meaningful discussion, consider a network that originally is a connected graph [24, p. 387], that is, there exists a path connecting any pair of nodes. We have the following result.

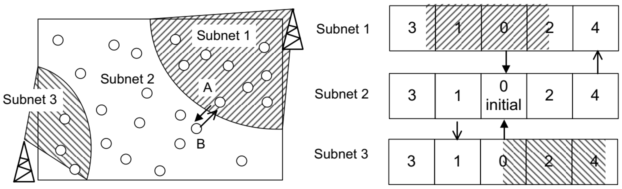


Fig. 4. The network remain connected after jamming by using "cross-channel communication".

**Proposition 1:** Suppose that in response to jamming, primary spectrum users, and high channel load, some nodes change their *receive channels* from the *initial receive channel*, and others keep using the initial *receive channel.*

(a) There is at least one node whose *receive channel* is different from that of at least one of its neighbors.

(b) If two nodes can communicate only if they share the same *receive channel*, then the network will be disconnected.

*Proof:* (a) Proof by contradiction. Suppose the conclusion is false, then it will end up with either all the nodes switching to a *receive channel* that is different from the initial *receive channel*, or all the nodes keep using the initial *receive channel*. In either case, this contradicts with the fact that, after the channel change, there are at least two *receive channels* being used in the network.

(b) Consider two nodes A and B. Node A changes its *receive channel*, and node B keeps the initial *receive channel*. By the definition of a connected graph, before the channel change, there is a path containing node A and node B. Now, take any of such paths, after the channel change, there will be at least one arc that disappears due to the assumption that two nodes can communicate only if they share the same *receive channel*. Therefore, there is no path containing both A and B. Thus the network after channel change is disconnected. □

It is important to note that Cross-Channel Communication is used for jamming only. In other words, if the cause of the problem is primary spectrum use, Cross-Channel Communication should not be used. A node can distinguish these two cases by using the output of the Jammer/Primary User Detector (see Fig.2), which analyzes the signals and waveforms received from the single radio. Additionally, the benefit of Cross-Channel Communication comes at the cost of frequent channel switching. For two-way traffic, a node has to transmit on one channel and receive on a different channel. Nevertheless, the benefit of Cross-Channel Communication outweighs its cost. In the case of jamming described above, if Cross-Channel Communication is not used, relay nodes must be used to forward the traffic in a multi-hop fashion. Otherwise, the network becomes disconnected. Either way, the benefit of Cross-Channel Communication outweighs its cost.

### *C. SRAC Algorithm Description*

The SRAC algorithm consists of two components: *receive channel* adaptation, and guidance to data transmissions. In *receive channel* adaptation, each node dynamically selects its *receive channel* in response to jamming, primary spectrum users and channel load. It also keeps track of changes in the *receive channels* of its one-hop neighbors. The collected *receive channel* information is then used to guide ordinary data transmissions.

#### C.1 Receive Channel Adaptation

We consider receive channel adaptation for jamming and primary spectrum use separately.

**Case 1: Jamming**

There are two key aspects in the design of the receive channel adaptation procedure. First, the selection of the receive channel must be well coordinated to facilitate future communications. This is done by providing a channel selection rule that exploits the similarity in jamming, primary spectrum use and channel load between nodes in close geographical proximity. The similarity is justified by various radio propagation pathloss models [26] and the fact that (1) wave-length scale strong signal strength variation (fading) can be mitigated by averaging over different locations, i.e., by mobility, (2) when a node detects jamming and primary spectrum users it may broadcast that information to its neighbors, and (3) traffic transmission can be overheard by the neighbors. Second, the change in receive channel must be conveyed to neighbors reliably. This is achieved by introducing an ARQ [24] mechanism.

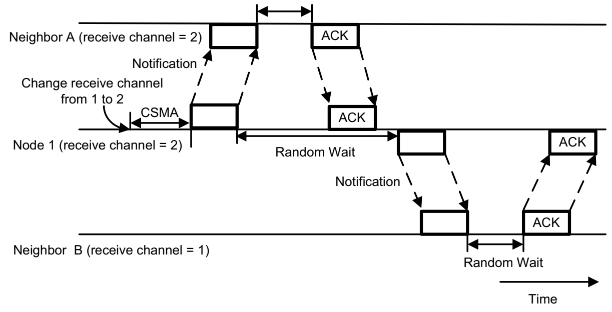


Fig. 5. Channel change notification and acknowledgement in response to jamming. Node 1 changes its receive channel from channel 1 to channel 2, and it notifies neighbors on the new channel (channel 2) and neighbors on the old channel (channel 1).

A pre-configured common channel is set as the initial *receive channel* $C_0$ for each node upon power up. The nodes then follow the following steps.

1. Select a candidate channel for the new receive channel. Each node monitors the presence of jamming/primary users and the channel load on the channel where it is currently on, and feeds this information to the SRAC algorithm, as shown in Fig. 2. Based on this information, the node makes a decision on whether it needs to change its current receive channel and how much the bandwidth it needs. If it decides to change, it refers to the channel information that it has collected, and chooses an idle channel whose carrier frequency is closest to the carrier frequency of the initial receive channel $f_0$. Depending on the channel activities, this distance criterion may result in two candidates, which are on either side of the carrier frequency $f_0$. If such a tie occurs, SRAC chooses the one with a lower carrier frequency.

2. Evaluate the candidate channel. The node switches to the candidate channel and evaluates it. If the candidate channel's condition meets its needs, the node will set it as the new *receive channel*. On the other hand, if it does not, the node will try the next candidate channel as done in step 1.

3. Receive channel change notification and acknowledgement. When a node changes its *receive channel*, it is obliged to inform all of its neighbors of the change so that future traffic destined to it will be sent over the new *receive channel*. To do this, it creates a notification frame which contains information about its new receive channel. Each node maintains a channel table which records the latest receive channel for its one-hop neighbors. Each entry is in soft state, that is, it will age out unless it is updated in time. The notification frame is sent over the receive channels in the table via multiple broadcasts. Upon finishing one of such broadcasts, the node changes back to its own receive channel to wait for acknowledgements. After receiving a notification frame, a neighbor waits a random time before sending back an ACK frame over the new receive channel indicated in the notification frame, and the ACK contains information about this neighbor's receive channel. Thus, the ACK frame serves two purposes: acknowledging the notification and updating the *receive channel* of the neighbor that sends the ACK. This way, both the node and its neighbors obtain each other's latest receive channel information. Figure 5 shows the timing diagram of this message exchange. Node 1's receive channel is channel 1 initially, and it then changes to channel 2. It first broadcasts a notification frame on channel 2, and the nodes that hear the

notification frame will sense the channel via CSMA and then send back an ACK. We use CSMA to reduce collision rather than pure Aloha style random access since the former generally performs better than the latter under the same conditions [27, p. 400]. When the ACK from node A arrives, node 1 updates the channel information for node A. Node 1 then broadcasts a duplicate notification frame on channel 1, which, in this case, is node 1's previous receive channel. After receiving the notification frame, node B waits a random time and sends an ACK over channel 2, which is node 1's current receive channel. After some random time, node 1 look ups its channel table for remaining listed receive channels. If it finds one, it will send a notification frame, just as done on its current and previous receive channels.

This notification and acknowledgement process will be carried out for every receive channel in the channel table. If a node receives an ACK from each node on a certain receive channel in the channel table, it considers the notification procedure for that channel has been successful. Otherwise, it will retry that channel until all neighbors have acknowledged or it hits a maximum retry limit.

Note that the medium access scheme on a node's own receive channel is different from that on the receive channels of its neighbors. This is due to the fact that, by the receive channel selection rule, its own receive channel allows the node to successfully listen, while other receive channels can not ensure this. Thus, we resort to the "random wait before access" scheme for those "other receive channels" to minimize collision. In the example shown in Fig. 5, CSMA is used for the medium access on node 1's own current receive channel, while "random wait before access" is used for the other receive channels in node 1's channel table.

We now consider the reliability of the notification procedure through a simple analysis. Let the probability that a notification frame is lost at node $A$ be $p_A$, the probability that an ACK gets lost at node 1 be $p_1$, and the maximum number of retries be $K$, then the success rate of the notification process between node 1 and node $A$ is $p_s = 1-[1-(1-p_A)(1-p_1)]^K = 1-(p_A+p_1-p_Ap_1)^K$. Since the notification and ACK frames are short and the propagation delay is small, the frame loss rate $p_A$ and $p_1$ will be generally very small [27]. For $p_A = p_1 = 0.2$ and $K = 7$, we have a success rate $p_s = 0.9992$, and as the frame loss rates increase to $0.5$, we still have $p_s = 0.8665$.

4. Background channel probe. Even if a node has completed the notification process, it still needs to check the current status of other channels. With this information, when the node needs to change its current receive channel, it can immediately find a potential candidate. The probe is done as follows: when the transceiver is not in transmission, a node randomly picks a short time interval and switches to an *atomic channel* to do passive listening. The channels to be probed are limited to the atomic channels between the initial receive channel and its current receive channel. More precisely, suppose carrier frequency of the current receive channel is $f_n$, and recall that the carrier frequency of the initial receive channel is $f_0$. The node will probe all atomic channels that satisfy $|f_i - f_0| < |f_n - f_0|$. If the node finds an acceptable channel in its background probe, it will choose that channel as a candidate receive channel and go to step 1. By doing this, the network intends to concentrate around the initial receive channel, thus reducing the number of receive channels to the extent where the channel conditions permit.

**Note:** (1) As discussed before, the methods of detecting jamming and primary spectrum users are out of the scope of this paper. As to channel load, it can be estimated by locally observing the percent of time during which the channel is in active use. For instance, a node periodically polls its channel status, and sets a binary number $s_n$ to 0 if the channel is idle, and to 1 otherwise. The observation is then used to update an exponential weighted moving average (EWMA) of $s_n$ by: $\bar{s}_n = (1-a)\bar{s}_{n-1} + as_n$, where $0 < a \ll 1$, to reduce statistical variation. If $\bar{s}_n$ is greater than a threshold, the node looks for a new receive channel with larger bandwidth, otherwise, it keeps the current receive channel. (2) More up-to-date channel information statistics can be obtained if nodes are put in the promiscuous mode to overhear other frames, which are modified to contain the receive channel information of the frame source. (3) Putting the channel entry as soft state enables SRAC to adapt to mobility. A stale entry will be aged out if the corresponding neighbor moves away. (4) When a node moves into another node's communication range, they will discover each other by the back ground probe process.

The receive channel adaptation process has two properties. The first one analyzes the impact of background channel probe on the equivalent link quality, and the second one points out an overhead-free implicit coordination in receive channel selection.

**Proposition 2:** Suppose a node randomly picks $\lambda$ portion of the time to leave its current receive channel for background channel probe, and let the frame loss rate without the probe at this node be $p_l$. Then with the probe, the frame loss rate is increased by $\lambda(1-p_l)$.

*Proof:* While the node is absent from its current receive channel for a probe, a frame destined to it will be lost. Since the absence interval is chosen randomly, the probability that a frame is successfully received becomes $p_l' = 1-(1-\lambda)(1-p_l)$. Thus $p_l' - p_l = \lambda(1-p_l)$. □

It is clear that as $\lambda \ll 1$, the impact is equivalent to a negligible degradation in link quality, which can be easily compensated by link layer or upper layer retransmissions. Since the wireless link is lossy in nature anyway, this degradation does not dramatically affect the network performance.

**Proposition 3:** If we quantize the jamming, primary spectrum use and channel load conditions into discrete levels, then those nodes that fall in the same levels will switch to the same new receive channel without communication overhead.
*Proof:* It follows from the receive channel selection rule in steps 1 and 2. □

**Case 2: Primary Spectrum Use**
In the case of jamming, a transmission on the channel being jammed is acceptable since it does little harm. However, in the case of primary spectrum use, a transmission on the channel newly occupied by the primary spectrum users may disrupt the primary spectrum users, and thus should be prohibited. Since the only possible transmissions that causes such disruption are the notification packets sent over the old receive channel (See Fig.5), we only need to disable these transmissions. As a result, it takes longer for the nodes to know the changes in receive channel of its neighbors.

### C.2 Guidance to Data Transmissions

The channel information collected in the receive channel adaptation procedure is then used to guide the transmission of data and related control messages.

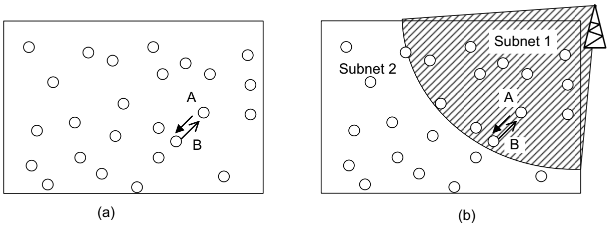
There are only two types of transmissions that SRAC need to support at the MAC layer, namely, unicast and broadcast. Multicast, is often a network layer concept, and is supported through MAC layer broadcast and network layer multicast membership check up [11].

To send a unicast frame, the sender first looks up the receive channel for the destination and then sends the frame by following the legacy MAC protocol being used. Consider an example where node A wants to send a unicast frame to node B, and the legacy MAC protocol is CSMA/CA. Node A first separately sends an RTS frame over Bs' receive channel and other receive channels in its channel table. Any node that hears the RTS frame should refrain from sending frames to node A. Upon receiving the RTS frame, which contains the receive channel of node A, node B will reply with a CTS frame over A's receive channel as well as the receive channels in B's channel table. Any node that hears the CTS frame should refrain from sending frames to node B. Upon receiving the CTS frame, node A will send a data frame over B's receive channel. This example again shows the merit of SRAC: even if node A and node B do not share a common channel where they can both transmit and receive, they can still communicate.

To send a broadcast frame, the sender checks its channel table and transmits the frame on every tabulated distinctive receive channel. Suppose node A wants to send a broadcast frame to its neighbors $B_1, B_2, B_3, B_4$, in which nodes $B_1$ and $B_2$ share a receive channel $C_1$ with node A, and nodes $B_3$ and $B_4$ share receive channel $C_2$. Node A broadcasts the frame separately on channels $C_1$ and $C_2$. It takes SRAC two transmissions instead of 4 transmissions to complete the multi-channel broadcast.

### *D. SRAC in Operation*

The advantages of the SRAC algorithm are best exhibited through a few examples. The one in Fig. 6 shows how nodes experiencing the same jamming and primary use conditions coordinate in their receive channel adaptation. Two nodes originally communicate over the initial receive channel, and upon detecting jamming or primary spectrum users, they switch to the same new receive channel by following the receive channel selection rule described early in Section III-C.1. They can communicate over the new receive channel immediately. Note that no communication is necessary.

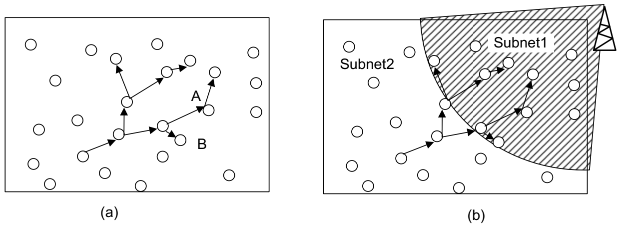


(a) (b)

Fig. 6. Two SRAC nodes experiencing similar jamming/primary spectrum user conditions switch to the same receive channel and can communicate immediately.

The example in Fig.7 illustrates how the SRAC algorithm supports a multicast application. Originally, all the multicast nodes use one receive channel. Upon detecting a jammer, some of the nodes (Subnet 1) switch to a new receive channel, while the others stay with the initial receive channel. With "cross channel communication", the two subnets can still communicate, and within each subnet ordinary broadcast is being used.

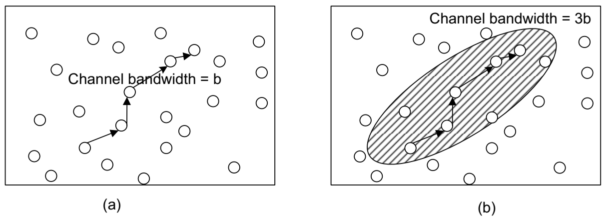
The example in Fig. 8 shows how the SRAC algorithm adapts to channel load. In Fig.8(a), the channel load is



(a) (b)

Fig. 7. SRAC nodes under the attack of a jammer support network layer multicast through “cross channel communication”.

beyond a desired threshold. Having observed the high channel load, the nodes involved (within the shade) triple the initial channel bandwidth. Note that in order to minimize inter-channel interference, it is desirable to ensure that the new channel ($3b$) does not overlap the old channel ($b$). With more bandwidth, the channel load is reduced and better network performance is achieved.



(a) (b)

Fig. 8. SRAC adapts channel load by tripling the channel bandwidth.

## IV. SIMULATION RESULTS

We first briefly discuss SRAC implementation in the QualNet network simulator [28] and then present the simulation results.

### A. *Implementation*

We implemented the SRAC algorithm as a relatively independent module, and integrated it with the CSMA/CA MAC protocol in QualNet 3.8, which captures the major features of IEEE 802.11 Distributed Coordination Function (DCF) while not getting into unnecessary complex issues such as dynamic rate control in the current QualNet 802.11 DCF code. In addition, we implemented a module to simulate jammers and primary spectrum users.

### B. *Simulations*

The CSMA/CA MAC protocol employs CSMA and RTS/CTS mechanisms to reduce collision. The radio propagation pathloss is modeled as Two-Ray, and the physical layer raw data rate is 1Mbps for RTS/CTS and SRAC specific control frames, and 2Mbps for other transmissions (per atomic channel).

#### B.1 Unicast

The network topology is shown in Fig. 9, where node 1 sends CBR traffic to node 3 via relay node 2, starting from time 0.5 second and ending at time 30 seconds. The jammer starts at time 10 seconds and lasts to the end of the simulation. The routing protocol is AODV.

The CBR packet size is 512 bytes. Figure 10 shows that at CBR sending rate 100 packets/s, without SRAC, the CBR traffic flow stops shortly after time 10 seconds when the jammer starts, while with SRAC, it continues. The ripple in the throughput for the case of SRAC is due to the communication overhead caused by notification and acknowledgement message exchange. We next increase the CBR sending rate to 200 packets/sec. The throughput as a function of time is now plotted in Fig. 11. The drop in the throughput after the start of the jamming is because the network already reaches its capacity, and there is no spare capacity, as in Fig. 10, to completely compensate for the artificial packet loss (i.e., due to a node not being always on a channel).

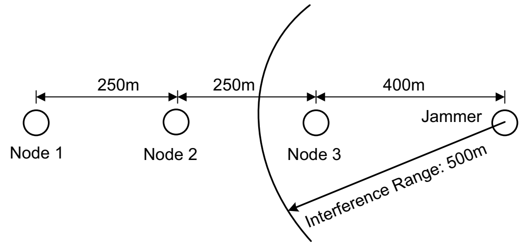


Fig. 9. Network setup for the unicast simulations.

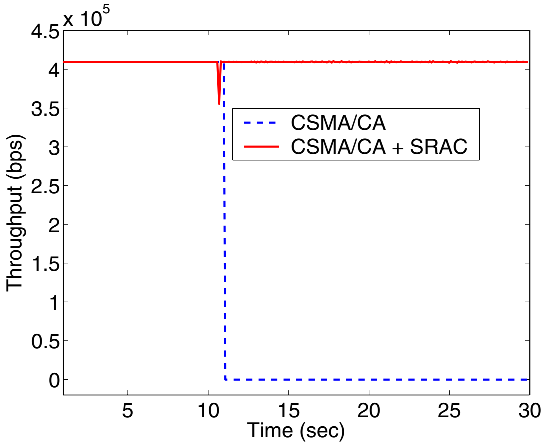


Fig. 10. Throughput for the CSMA/CA MAC protocol with and without SRAC for 100pkts/sec CBR traffic generator.

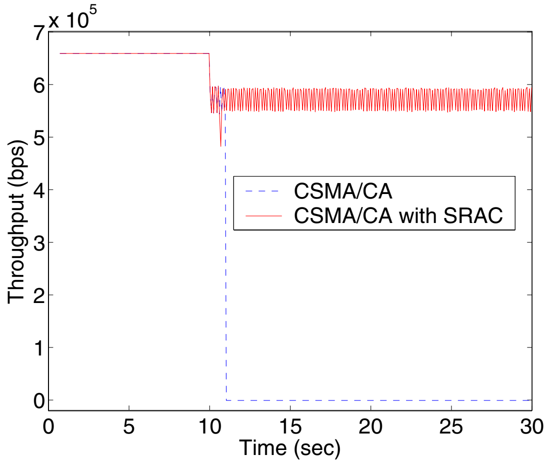


Fig. 11. Throughput for the CSMA/CA MAC protocol with and without SRAC for 200pkts/sec CBR traffic generator.

### B.2 Multicast

The multicast routing protocol is On-Demand Multicast Routing Protocol (ODMRP) [11], and the multicast application is Multicast CBR (MCBR). For clarity, we first focus on a simple network topology, as shown in Fig. 12, where the nodes are evenly spaced. Nodes 5, 6, 4, 7, 8, 12 form a multicast group. Node 5 sends MCBR traffic to the other nodes in the multicast group. The MCBR application starts at time 20 seconds, and stops at time 100 seconds, with data rate of 2 packets/sec. The jammer starts at time 50 seconds. The physical layer parameters such as transmission powers are set the same as those in the unicast simulations. We run the same scenario 10 times with different random seeds to obtain an average. It is seen in Fig. 13 that with SRAC, the average total number of packets received by the MCBR receivers are 94.61% higher than that without SRAC. Note that for those nodes that are affected by the jamming, i.e., nodes 4, 7, 8 and 12, the gain is even higher. In fact, if there were no jamming, 800 packets would be received in total at most by the MCBR receivers. With SRAC, a total number of 793 packets are received under the jamming attack.

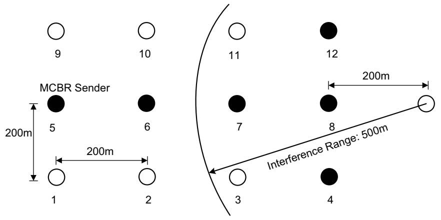


Fig. 12. Network setup for the MCBR multicast application, where solid black circles represent the nodes in the multicast group.

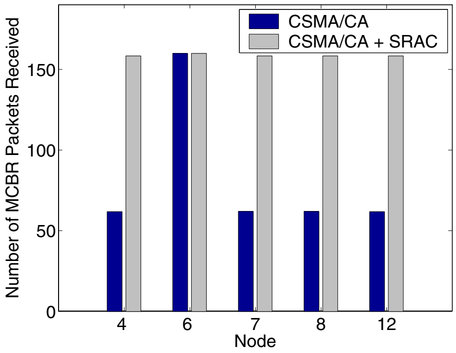


Fig. 13. The number of MCBR packets received by each multicast group member, where node 6 is not affected by jamming while others are.

We now consider a more complicated scenario, where 28 nodes and a jammer are randomly placed in a 1500 × 1500 $m^2$ terrain. The jamming range is 896 m, and the 28 nodes each have a communication range of 754m. Five nodes are selected to form a multicast group, with two under direct attack from the jammer. One of the five nodes sends MCBR traffic to the other four nodes, starting from time 20 seconds and ending at time 100 seconds. The jammer starts at time 50 seconds and continues until time 100 seconds. We calculate the gain of SRAC in the number of received MCBR packets over that of the case where SRAC is not used. For each MCBR sending rate, we calculate the average number of received MCBR packets and the 95% confidence interval based on 10 random seeded simulations. We see from Fig. 14 that the gain decreases as the MCBR sending rate increases, which agrees with Proposition 2 since the artificial packet loss introduced by SRAC increases as a node spends more time switching channel to relay traffic across different channels. However, even at high traffic load the gain is still significant. Note that, if we exclude the two nodes that are not directly affected by the jammer in the calculation, the gain will be much higher.

## V. DISCUSSIONS

The SRAC algorithm allows combining individual *atomic channels* for a *composite channel* that may have a rather wide bandwidth. Traditional modulation schemes may not be applicable considering the difficulty in short time synchronization at the receiver. Alternative approaches do exist, however, such as the Orthogonal Frequency Division Multiplexing (OFDM) modulation [26]. Also, an OFDM transmitter can easily adjust the signal bandwidth by using a subset of a large number of subcarriers. Software Defined Radio (SDR) is a promising economic approach to providing physical layer support for the SRAC algorithm.

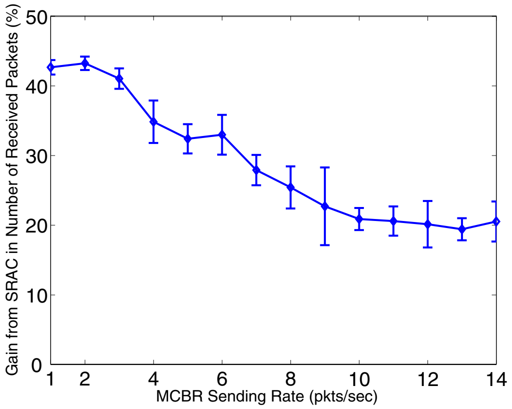


Fig. 14. The gain of SRAC in the number of packets received by the MCBR receivers changes with the MCBR sending rate, where the bars denote 95% confidence intervals.

There is a penalty for frequent channel switching in the SRAC algorithm. Capacity is wasted when a radio transitions from one channel to another. The exact capacity loss depends on the implementation. State-of-the-art implementation of 802.11a/b/g radios [29] take only about 30 $\mu$s to switch channel, as opposed to about 200 $\mu$s by commercially available 802.11b radios [30]. The impact of channel switching delay is negligible for low data rates, and becomes significant for high data rates.

Emission policies for nodes in the presence of primary spectrum users may be different from the one assumed in the paper. Instead of ensuring zero disruption, the policy may allow minimal disruption. Thus, the proposed SRAC algorithm is conservative in conforming emission policies.

## VI. CONCLUSIONS

We proposed the Single-Radio Adaptive Channel (SRAC) algorithm to upgrade existing legacy single-radio MAC protocols to be dynamic spectrum access capable. It provides a feasible dynamic channelization mechanism to make the best of the available spectrum, relaxes the radio communication conditions to enhance network connectivity, and exploits the broadcast nature of the wireless medium to provide efficient multicast support. The performance of SRAC is evaluated through analysis and QualNet simulations.

## REFERENCES

[1] P. Gupta and P. R. Kumar, “The capacity of wireless networks,” *IEEE Transactions on Information Theory*, vol. IT-46, pp. 388–404, March 2000.

[2] J. G. Proakis, *Digital Communications*. McGraw-Hill, Inc., 3rd ed., 1995.

[3] FCC Spectrum Policy Task Force, “Report of the spectrum efficiency group,” Nov., 2002.

[4] Advanced Technology Office (ATO), Defense Advanced Research Projects Agency (DARPA), “Overview of the next generation program.” [Online]. Available: http://www.darpa.mil/ATO/programs/XG/index.htm.

[5] L. Berlemann, S. Mangold, and B. Walke, “Policy-based reasoning for spectrum sharing in cognitive radio networks,” in *1st IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks (DySPAN)*, (Baltimore, MD), pp. 1–10, Nov. 8–11 2005.

[6] M. McHenry, “The probe spectrum access method,” in *1st IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks (DySPAN)*, (Baltimore, MD), pp. 346–351, Nov. 8–11 2005.

[7] B. Wild and K. Ramchandran, “Detecting primary receivers for cognitive radio applications,” in *1st IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks (DySPAN)*, (Baltimore, MD), pp. 124–130, Nov. 8–11 2005.

[8] L. Cao and H. Zheng, “Distributed spectrum allocation via local bargaining,” in *Second Annual IEEE Communications Society Conference on Sensor and Ad Hoc Communications and Networks (SECON)*, (Santa Clara, California, USA), Sep. 2005.

[9] L. Ma, X. Han, and C.-C. Shen, “Dynamic open spectrum sharing mac protocol for wireless ad hoc networks,” in *1st IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks (DySPAN)*, (Baltimore, MD), pp. 203–213, Nov. 8–11 2005.

[10] Q. Zhao, L. Tong, and A. Swami, “Decentralized cognitive MAC for dynamic spectrum access,” in *1st IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks (DySPAN)*, (Baltimore, MD), pp. 224–232, Nov. 8–11 2005.

[11] S. H. Bae, S.-J. Lee, W. Su, and M. Gerla, “The design, implementation, and performance evaluation of the on-demand multicast routing protocol in multihop wireless networks,” *IEEE Network Magzine*, vol. 14, pp. 70–77, Jan./Feb. 2000.

[12] J. C. Ramming, “Control-based mobile ad-hoc networking (CBMANET) program motivation and overview,” Aug. 2005. [Online]. Available: http://www.darpa.mil/ato/solicit/CBMANET/briefs/CBMANET_Overview-Ramming.pdf.

[13] P. Bahl, R. Chandra, and J. Dunagan, “SSCH: Slotted seeded channel hopping for capacity improvement in IEEE 802.11 ad-hoc wireless networks,” in *MobiCom’04*, (Philadelphia, PA), Sept. 2004.

[14] F. Zhao and L. J. Guibas, *Wireless Sensor Networks: An information processing approach*. Morgan Kaufmann Publishers, 2004.

[15] S. Ganeriwal, D. Ganesan, H. Sim, V. Tsiatsis, and M. B. Srivastava, “Estimating clock uncertainty for efficient duty-cycling in sensor networks,” in *The 3rd ACM Conference on Embedded Networked Sensor Systems (SenSys)*, (San Diego, California, USA), Nov. 2005.

[16] J. Zhao, H. Zheng, and G.-H. Yang, “Distributed coordination in dynamic spectrum allocation networks,” in *1st IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks (DySPAN)*, (Baltimore, MD), pp. 259–268, Nov. 8–11 2005.

[17] J. Mo, H. W. So, and J. Walrand, “Comparison of multichannel MAC protocols,” in *Proceedings of the 8th ACM international symposium on Modeling, analysis and simulation of wireless and mobile systems*, 2005.

[18] S.-L. Wu, C.-Y. Lin, Y.-C. Tseng, and J.-P. Sheu, “A new multichannel MAC protocol with on-demand channel assignment for

multi-hop mobile ad hoc networks," in *Int'l Symposium on Parallel Architectures, Algorithms and Networks (I-SPAN)*, 2000.

[19] A. Tzamaloukas and J. Garcia-Luna-Aceves, "Channel-hopping multiple access," in *Proceedings of IEEE ICC 2000*, (New Orleans, LA), June 2000.

[20] J. Chen, S. Sheu, and C. Yang, "A new multichannel access protocol for ieee 802.11 ad hoc wireless lans," in *Proc. of PIMRC 2003*, pp. 2291 – 2296, 2003.

[21] W. Tuttlebee, ed., *Software Defined Radio: Enabling Technologies*. West Sussex, England: John Wiley & Sons, Ltd, 2002.

[22] W. Xu, W. Trappe, Y. Zhang, and T. Wood, "The feasibility of launching and detecting jamming attacks in wireless networks," in *MobiHoc'05*, (Urbana-Champaign, Illinois, USA.), May 25–27 2005.

[23] L. L. Peterson and B. S. Davie, *Computer Networks: A system approach*. San Francisco, CA: Morgan Kaufmann Publishers, second ed., 2000.

[24] D. Bertsekas and R. Gallager, *Data Networks*. Upper Saddle River, NJ, USA: Prentice Hall, second ed., 1992.

[25] L. Bao and J. Garcia-Luna-Aceves, "A new approach to channel access scheduling for ad hoc networks," in *Seventh Annual International Conference on Mobile Computing and Networking (Mobicom)*, (Rome, Italy), pp. 210–221, July 2001.

[26] G. L. Stuber, *Principles of Mobile Communications*. Kluwer Academic Publishers, 2nd ed., 2001.

[27] L. Kleinrock, *Queueing Systems*, vol. 2. John Wiley & Sons, Inc., 1976.

[28] Scalable Network Technologies, Inc., "Qualnet network simulator." [Online]. Available: http://www.scalable-networks.com.

[29] M. Zannoth, T. Rhlicke, and B.-U. Klepser, "A highly integrated dual-band multimode wireless LAN transceiver," *IEEE Journal of Solid-State Circuits*, vol. 39, pp. 1191–1195, Jul. 2004.

[30] Maxim Integrated Products, Inc., Sunnyvale, California, USA, *MAX2820, MAX2820A, MAX2821, MAX2821A 2.4GHz 802.11b Zero-IF Transceivers Data Sheet rev. 04/2004*.