WiFlex: Multi-Channel Cooperative Protocols for Heterogeneous Wireless Devices

Jiwoong Lee [(a)], Jeonghoon Mo [(b)], Tran Minh Trung [(b)],
Jean Walrand [(a)], and Hoi-Sheung Wilson So [(a)]

Abstract—[1] **In ISM bands, many wireless protocols proliferate such as 802.11, bluetooth, and ZigBee. However, these incompatible protocols create complex coexistence and connectivity problems. If the same trend continues, similar interference and performance problems will continue to exist in future unlicensed bands. As new unlicensed bands open up, one can take a different approach to spectrum sharing. Instead of proposing a new MAC protocol for each type of application, we propose a family of parameterized MAC protocols called WiFlex that can tailor to different application needs ranging from wireless sensors to media center. Yet, these protocols within this family are compatible with each other to allow communication and spectrum-sharing coodination among different types of devices. We envision this family to be based on an OFDM-like multichannel physical layer.**

The contribution of this paper includes the discovery of an *asynchronous split-phase* protocol with dynamic priority support. This protocol enables powerful devices to achieve a high throughput and protects low power devices with urgent but only occasional transmissions. It is distributed and data collision-free. Moreover, it can support low delays for real time applications.

The performance of the protocols are evaluated using an extension of NS-2. The results demonstrate the coexistence of devices with disparate radio characteristics and the support of applications with different requirements with good performance.

I. Introduction

The standard practice in the design of wireless protocols is to customize them for a specific class of device. For instance, WiFi supports a high data rate without stringent energy limitation; Bluetooth is for low-bit rate devices communicating over a shorter distance; Zigbee is for low-power devices with light traffic; WiMax is for long-range transmission at a high bit rate.

This segmented approach enables the design of protocols that are suitable for specific devices. However, the approach also results in systems that (i) cannot communicate directly and (ii) interfere with one another. A costly bridge device with two types of radios are necessary to allow different kinds of devices to communicate. In addition, incompatible devices sharing the same spectrum degrade each other's performance, as reported in the literature [2], [3], [4], [5], [6], [7]. Some devices may even starve others. Overall, this problem arises due to the lack of medium access coordination between heterogeneous devices sharing the common spectrum band.

It is not straightforward to define fairness among heterogeneous devices. A high-end device with a large spectral footprint and complex modulation should achieve a higher throughput than a low-end device. Also, a low-end device often is associated with less traffic demand. Thus, it is not desirable to equalize the throughputs of all devices. Priority access for service differentiation should be supported. It is desirable to have an integrated and flexible framework to support various kinds of applications.

A. Design Goals

WiFlex has the following design goals:

1) **Connectivity and coexistence of Heterogeneous Devices:** Under the same PHY/MAC architecture, WiFlex supports communication among heterogeneous devices with different physical capabilities such as accessible frequency range and transmission rate. WiFlex facilitates the negotiation of these physical capabilities between the potential sender and the receiver before data exchange.
2) **Fairness and Priority Access:** WiFlex treats devices with the same access priority equally. Moreover, WiFlex also supports different levels of premium access mechanism. Premium access should be distinguished from basic access in terms of individual throughput and medium access delay. At the same time, one type of devices (e.g., laptops) do not starve others (e.g., sensor nodes).
3) **Efficient Utilization of Spectrum:** WiFlex does not waste spectrum resources for devices to schedule access. WiFlex data transmission is collision-free.

Note that WiFlex devices differs from cognitive radios which focus on the reuse of idle medium by secondary users. In WiFlex, the spectrum is unlicensed and hence there are no primary users. Also, explicit communication among all kinds of devices are supported.

We are not aware of any existing MAC protocol which tries to support wireless devices of vastly different capabilities. However, for the purpose of performance benchmarking, one can think of straight-forward extensions of existing multi-channel protocols that can support some goals of WiFlex. Our study does not aim to compare the performance of WiFlex to all previously proposed multichannel protocols, nor to achieve backward compatibility with commercially available ISM band technologies such as IEEE 802.11, Bluetooth, or Zigbee. WiFlex is designed to be a greenfield platform supporting *IEEE 802.11-like* high end devices and *Zigbee-like* low end devices simultaneously.

Finally, we point out that transmission power control for heterogenous devices opens up another dimension of communication issues which are not yet addressed in this paper and left open for future research.

[1]This work was supported in part by the National Science Foundation under Grant CNS-0435478 and in part by the KOSEF, Korea. The authors are with [(a)] Department of Electrical Engineering and Computer Sciences, University of California, Berkekely, CA 94720, USA and [(b)] School of Engineering, Information and Communications University, Yusong, Daejon, 305-600, Korea. Corresponding author, Jiwoong Lee, porce@eecs.berkeley.edu

II. WiFlex: Cooperative Coexistence Protocol

To realize the aforementioned design goals, WiFlex has the following main features:

- OFDM-like multichannel based physical layer
- Common control channel
- Three phase asynchronous split phase
- Fairness and Priority Access algorithm
- Efficiency improvement algorithms.

A. OFDM based Physical Layer

WiFlex adopts an OFDM-like multichannel physical layer. An OFDM system divides frequency into multiple orthogonal subcarriers, each of which is a distinct frequency within the available spectrum. A group of adjacent subcarriers combine to form a channel. In this paper, we assume that the smallest unit of spectrum allocation is a channel which consists of a fixed number of consecutive subcarriers. Therefore, the available spectrum is divided into disjoint channels. Since these channels do not overlap, we assume they not interfere with each other for simplicty. In reality, a small guard band may be necessary among different channels.

Low end devices may be limited to only operate on a small subset of channels and that it can only modulate a smaller number of channels at a time. This allows low-end devices to consume less power and cost less than higher end devices. High-end devices can use more more channels simultaneously when communicating with other high end devices, but they can also use one or two channels when talking to lesser devices.

With the advent of reconfigurable hardware technologies, it is possible to design hardware which can transmit data over a variable number of channels. In [9], Poon introduces a reconfigurable architecture which allows the same hardware to be used for processing FFT and inverse FFT of different numbers of points. FFT is the fundamental operation behind OFDM. This configurable architecture can be scaled to handle larger FFT (i.e., OFDM communication over more channels) by either increasing the number of processing elements or by running the same hardware at a higher clock frequency).

Therefore, it is possible to use the same basic hardware design to build a family of WiFlex devices ranging from low-end to high-end devices. A high-end device would either include more replicates of the same hardware blocks or run the same chip at a higher clock speed.

While we have selected OFDM for the physical layer of WiFlex, other choices are possible. The key features we require from the physical layer is that it should be able to support devices with different cost, transmission rate, and adjustable transmission bandwidth.

In short, each WiFlex device i is parameterized by:

1) f_i, the maximum number of consecutive channels the device can modulate. f_i represents the *maximum bandwidth* of radio i in units of channels. A more powerful radio can modulate/demodulate over a wider band for each packet. Device i may choose to transmit using $1, 2, \ldots,$ up to f_i channels for each individual packet depending on the maximum bandwidth of the intended receiver j and the current set of idle channels.

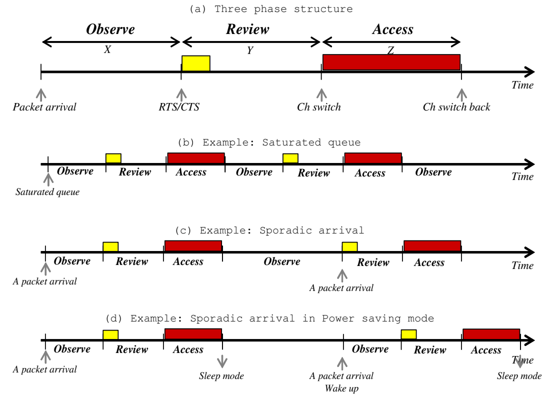

Fig. 1: Asynchronous Split Phase

2) D_i, the subset of channels C the device can access. D_i determines the range of frequencies over which that i can operate.
3) M_i, the set of modulation schemes (e.g. PSK, ASK, and N-QAM etc.) that device i can use. We assume that there is a common basic modulation that is understood by all receivers. This determines the basic rate of any two WiFlex devices.

B. Common control channel

We selected a MAC protocol based on a common control channel in order to provide compatibility to devices which can use only a single channel. The WiFlex MAC protocol uses channel 0 as a common control channel. Devices use the channel to agree on the duration, modulation, and set of channels they use for exchanging data. They use RTS/CTS exchange to make an agreement. Multiple pairs of devices may select disjoint sets of channels that they use simultaneously. The asynchronous split phase mechanism is more efficient than the more familiar synchronous split phase.

To accommodate low end devices, the transmission rate in the control channel cannot be higher than what the low end device can support. This creates the control channel bottleneck. This congestion problem has been addressed for multi-channel MAC with a large number of channels and homogeneous devices in [12]. For WiFlex, we will show that a packet aggregation technique can alleviate this problem.

C. Three Phase Asynchronous Split-Phase (ASP) protocol

WiFlex adopts a three phase asynchronous Split-Phase (ASP) MAC algorithm to achieve both efficiency and fairness goals. The asynchronous algorithm is adopted for better utilization of available spectrum and three phase algorithm is proposed for fairness and efficiency. For example, the second phase is introduced to achieve fairness and priority goals at the cost of efficiency, as will be explained.

WiFlex does not require synchronization among devices, which is the main difference between the synchronous and the asynchronous versions of split phase [2] In the ASP, there are no agreed phases of control or data among different devices. Rather, each device independently has its own phases and associated durations

[2]Refer to [10] for the details of synchronous split phase algorithm.

even can vary over time. Transmissions are performed according to a certain rule, which will be explained shortly. It turns out that WiFlex achieves better spectrum utilization by avoiding unused spectrum of control phase.

The data transmission of WiFlex consists of three phases: *Observe*, *Review*, and *Access* as is shown in Figure 1 (a). Observe phase is for information collection to avoid the potential conflict in channel/time selection. Review phase is again for conflict avoidance by obeying fairness and priority rule. The first two phases correspond to the control phase of SSP and the *Access* phase corresponds to the data phase. A special case of WiFlex with no Review phase and $f_i = 1$ becomes the same as in the proposal of [11]. Figure 1 (b), (c), and (d) show three exemplary realization of three phases.

In the description that follows X, Y, and Z are system parameters that represent the duration of three phases, and shared by all the devices in the system.

- The *Observe* phase is the starting phase of WiFlex access algorithm. A device is required to observe the control channel for at least a fixed duration X before exchanging a reservation request or RTS/CTS. In this phase, a device monitors others' RTS/CTS handshakes. As we show below, if X is large enough, the device can precisely figure out which channels are to be idle and can be used without collisions.
- The *Review* phase starts by broadcasting an RTS message to the neighbor and lasts for a fixed duration Y except in the case of fast overriding. The RTS message includes receiver ID, reservation request channels, and reservation duration. Based on the others' requests that is monitored in the Observe phase, the reservation channels and duration are determined not to overlap others' reservation unless the device decides intentionally to override others. (See following sections for the discussion about overriding.) If the device cannot find non-overlapping channels and time for itself and cannot override others, it randomly backs off and restarts the ASP. Otherwise it sends out the request.
  Upon request, the receiver device may send back CTS message indicating whether it accepts the request. After the RTS/CTS exchange, the device keeps monitoring of control channel to see if others send out reservation requests overriding its own until the Review phase ends.
- If the device's request was overridden by others, it randomly backs off and restarts the ASP. Otherwise it enters the *Access* phase at the end of *Review* phase. On entering, it can switch its radio to the reserved data channels and start to access the medium. The access time is limited by Z. The sender is also expected to show up at the reserved channels and time. After the data exchange, the sender/receiver pair come back to the control channel and repeat the process if necessary.

Collision-free Access Recall that ASP is asynchronous and there is always possibility that not all the devices share the reservation information. Indeed, some devices in general are tuned to data channels while others are tuned to the control channel. We show that special relationships of X, Y, and Z enable collision-free data transfer even if all the devices are neither synchronized nor share the common knowledge on scheduling.

Proposition 1. *Assume that all devices are in a single collision domain and access is non-preemptive. If*

$$X \geq Z,$$

the ASP algorithm is collision free irrespective of the value of Y.

Proof: Consider a device which just returns to the control channel after completion of data transmission and define this time to be 0. The earliest time it can start the next transmission is $X + Y$. Now classify all other devices into two groups: those that have already issued requests or the others that have not. A device in the first group finishes transmission by time $Y + Z$ at most. Hence, if the monitoring duration X is longer than the maximum Access duration Z, the device does not need to worry about other devices in the first group. The second group device has not issued request yet. Therefore, their activities can be monitored by the device and it can avoid collision with devices in the second group. □

The following proposition holds when the overriding is allowed.

Proposition 2. *Assume that all devices are in a single collision domain and access is preemptive. If* $X \geq Z$ *and* $Y \geq Z$*, the ASP algorithm is collision free.*

Proof: If a collision occurs, there are two possible cases: the preemptee could not hear overriding claim of preempter, which corresponds to the case $Y < Z$, or the preempter actually could not hear preemptee's original claim and happened to override, which corresponds to the case $X < Z$. □

Throughout the paper we assume $X \geq Z$ and $Y \geq Z$ unless explicitly state differently.

D. Fairness and Priority Access Algorithm

The primary role of the *Review* phase is to provide the additional *degree of freedom* to control the priority access and fairness of scheduling. Note that we still can achieve collision-free scheduling algorithm when $Y = 0$ by not allowing preemptions (as in [11]). However in that case, the scheduling cannot support priority access scheme. Moreover, if some device generates packets more often than others, there is no explicit way to regulate throughput fairness in the system.

Preemption in the ASP protocol works as follows: At a given time, every device manages its own access score. A device with higher score may override another with lower score. Assume that device A sends its request to reserve channels and another device B with higher score observes A's request. Device B can *override* A's request when its Observe phase is just over, A is still in its Review phase, and B has no other option but to override. When B sends out an overriding reservation request over control channel, device A yields. (Otherwise they would collide in the data channels.) Section II-E1 provides more details on the overriding mechanism.

By regulating each device's score over time, both priority access and fairness of scheduling can be achieved.

One example of scoring is based on the distributed version of weighted proportional fairness algorithm, which is a modification of PF algorithm used in the IS-95 downlink HDR system. In that

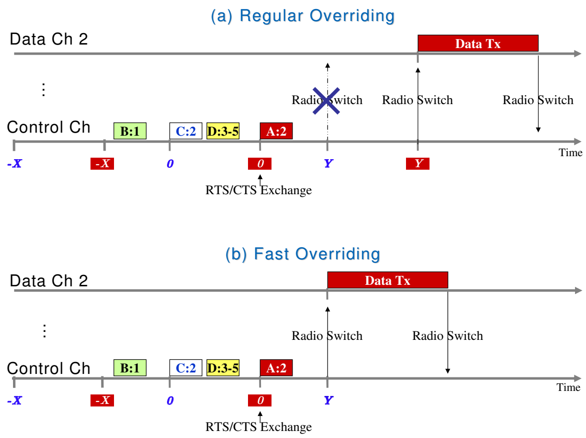

Fig. 2: Overriding

scheme, the priority of device i at discrete time t is determined by

$$\frac{w_i(t)}{T_i(t)}, \quad \forall i$$

where w_i is the weight of device i and $T_i(t)$ is the average transmission rate of device i which can be computed as:

$$T_i(t) \quad = \quad \alpha T_i(t-1) + (1-\alpha) R_i(t)/P_i \qquad (1)$$

where α is a value between 0 and 1, and $R_i(t)$ the instantaneous goodput rate of device i at time t which is positive or 0. The weight w_i of device i is a function of a device type. P_i is the device priority provided from an exogenous way, such as access service contract. The computed score is quantized into an integer value of n_q bits (say, 8) and broadcast with a request message.

In the long run, the average of score of device i is higher if P_i is larger. Among the devices with the same P_i, the average score remains the same. The score is directly related to the access chance, and in turn, to access delay and individual throughput. Therefore, the priority access and fairness of scheduling is realized by controlling P_i. Representative experimental results are provided in the next section.

One advantageous feature of this generic preemption architecture is in that it does not require a long term memory; it only needs to know current pending requests over the Observe phase when it has a frame to send. When it does not have a frame to send, it can skip monitoring. In the same line, recall that the device is not required to monitor more than X seconds for its Observe phase. This feature is helpful for a power saving mode.

E. *Efficiency Improvement Algorithms*

1) Overriding: A WiFlex device may override others' reservation if it has a higher access score than theirs. Overriding is a key mechanism to regulate both the fairness and priority access. The algorithm is as follows: Before sending out reservation request message, a device searches for overridable resource, i.e., others' reservation channels and time which was broadcast with lower score during its Observe phase. The device searches for the chance of *Fast overriding* first and *Regular overriding* next. If none of them is possible, it gives up overriding and uses, if any, empty resource only.

Under *Fast overriding*, a higher priority device usurps the reservation of a lower score device and uses it. The overriding device does not have to wait for Y fully. Instead, it may wait for less than Y, say, y. As a result shorter access delay is obtained. Though $y < Z$, despite Proposition 2 interestingly, the access is still guaranteed to be collision-free because the reservation to be overridden was originally collision-free. Usurping collision-free reservation does not introduce collisions.

If Fast overriding is neither feasible nor beneficial, the device tries *Regular overriding*. Under *Regular overriding*, any device is forced to wait for Y before accessing the data channel. As a result, the reservation channels and time of the overriding device partially overlaps that of the overridden one. The non-overlapped portion of previous reservation resource is wasted. Spectrum underutilization here is the price of fairness and priority support.

2) MAC Frame Aggregation: A WiFlex device i can aggregate MAC frames addressed to the same receiver only if the overall transmission time does not exceed Z. A MAC frame aggregation is called a Jumboframe. Suppose that the length of any frame is L, the transmission rate is R per channel, and the number of data channels to use is $n \le f_i$. Then the transmission time becomes $\frac{L}{nR}$. The device is allowed to transmit up to i_J frames successively without releasing the channel if $z := \frac{i_J L}{nR} \le Z$. The benefit of frame aggregation is twofold: first, it helps reduce the frame access delay, which is partially incurred by the Observe/Review phases, yielding enhanced individual throughput; second, it counteracts the potential control channel bottleneck problem. Note that a WiFlex design goal is to facilitate the coexistence and compatibility between heterogeneous device classes. Devices generally support different transmission rates. Since the common control channel information needs be shared among all the devices, RTS/CTS should be exchanged at the minimum rate. Frame aggregation efficiently counteracts the potential common control channel bottleneck problem.

III. Performance Evaluation

In this section, we evaluate the efficiency and fairness of the WiFlex ASP protocol in a single collision domain. We modified the NS-2 simulator by adding multi-channel carrier-sensing, channel corruption, and three phase ASP protocol operations. WiFlex supports any type of application traffic including TCP and UDP. The control channel to exchange RTS/CTS follows CSMA.

One can think of various kinds of preemption algorithm. In our simulation, each device behaves in a greedy, but no more greedy than necessary, way. That is, the device looks for the *best-fit* resource in a selfish sense. So whenever possible and beneficial to itself, it overrides others' reservation.

A. *Coexistence*

In this scenario, we mix two types devices, high-end and low-end devices. The high-end devices model the behavior of WLAN while the low-end devices model the behavior of small sensors using Zigbee or Bluetooth. There are 24 high-end connections, each of which generates a stream of packet at the rate of 620Kbps. The low-end device generates traffic at the rate of 1.6Kbps. We varied the number of low-end connections from 2 to 100. Figure 3 shows the fraction of time used to transmit data on data channels. We see that high-end devices with higher source rate do not starve low-end devices and WiFlex protocol provides *fair* chance of medium access. The balancing between high-end and low-end devices also can be controlled further by priority assignment.

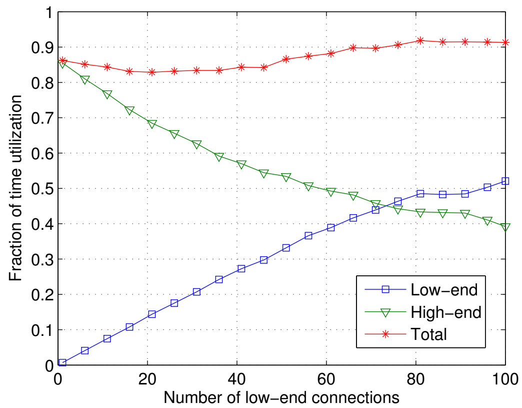

Fig. 3: Coexistence scenario

B. Common Control Channel Congestion

WiFlex requires the transmission rate at control channel to be commonly supported by all devices in the network. If there are low-end devices supporting only slow transmission rate, the control channel can be the bottleneck of the system performance.

To see the effect of control channel rate, we vary it from 0.15Mpbs to 2Mbps with 24 standing CBR connections of source rate 620Kbps each. Transmission rate at data channels is fixed as 2Mbps.

Figure 4 shows that the system throughput is immensely impaired when the control channel rate is low with Jumbo frame feature turned off. When Jumbo frame is used however, a fair amount of system throughput is largely obtained suggesting that the control channel is no longer the bottleneck even at the very low rate. From this result, we may claim that low system performance is not an unavoidable price of heterogenous device support but can be lessened by countermeasures.

C. Fairness and Priority access

To test the fairness and priority access algorithm, we generate three different scenarios with 50 TCP connections. We set thes value of α to 0.95 and $f_i = 8$. In the first scenario, *Proportional Fairness with premium service*, the priorities of the first ten connections are $P_i = 10$, and the others have $P_i = 1$. In the second scenario, *PF without premium service*, we set the priorities of all connections equal to 1. Finally, we evaluate the system performance *without PF* by using scenario 3, in which we turn off the fairness algorithm. Figure 5 portrays results from these three scenarios. In PF with premium service scenario, the first ten connections show 2.5 times larger individual throughput than others, which is expected by design. In the second scenario, the individual throughput of all connections are well regulated and achieves the *fairness index* 0.99. We observe an interesting phenomenon when fairness algorithm is intentionally turned off; while overall throughput is enhanced, the fairness index deceases to 0.94. In WiFlex speak, overriding provides fair access chances to devices at the price of resource waste incurred by overriding action. This is a typical example of tradeoff of throughput and fairness.

D. Priority service to VoIP

Under WiFlex, VoIP packets mingles well with TCP packets. We have five standing TCP connections with large packet size and vary the number of VoIP sessions from 6 to 26. VoIP connections have higher priority(10) than that of TCP connections(1). Figure 6 shows the delay distribution of TCP and CBR VoIP connections. We can observe that the delay of VoIP packets mostly falls within 50msec, which is an appropriate delay budget for the first hop access network. High Priority access of VoIP packets tends to slow down of TCP access with low priority. The balance between two types of accesses can be controlled further by priority assignment.

E. Efficiency

Figure 7 shows the maximum throughput that we can get with WiFlex based on two different scenarios; CBR connection only and TCP connection only. In each scenario, number of connections vary from 1 to 95. In case of CBR, each source generates fixed length packets(2318 Bytes) with constant bit rate 620Kbps. In TCP scenario, source rates vary according to TCP congestion control but packet size itself is the same. In both cases we assumed that there are 9 channels of 2Mbps each. One of them is used as the control channel while the others are used as data channels. Therefore, the maximum throughput is 16Mbps. Throughput is measured at transport layer. Figure 7 shows that the max throughput in the CBR scenario is 15.3Mbps, while it is 12.4Mbps in the TCP scenario. This difference is caused by TCP ACK message overhead that does not contribute throughput while taking time/frequency resource.

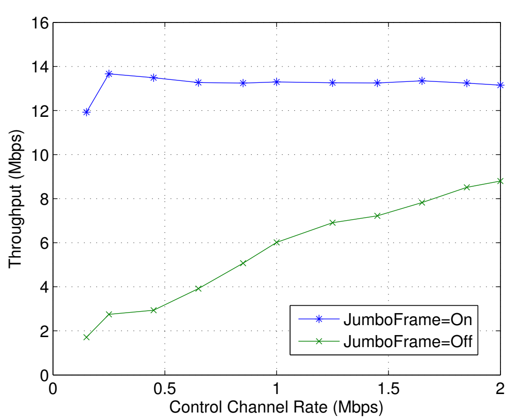

Fig. 4: Effect of control channel rates(8 data channels of 2Mbps)

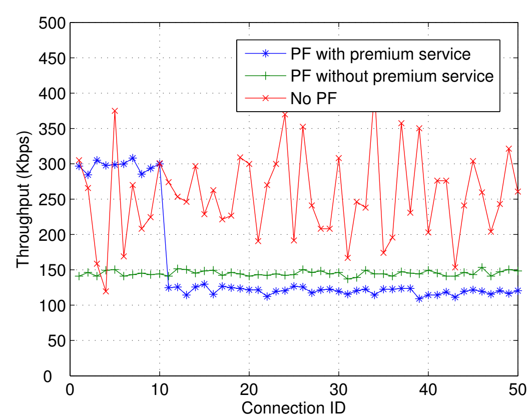

Fig. 5: Fairness test (throughput of individual connections)

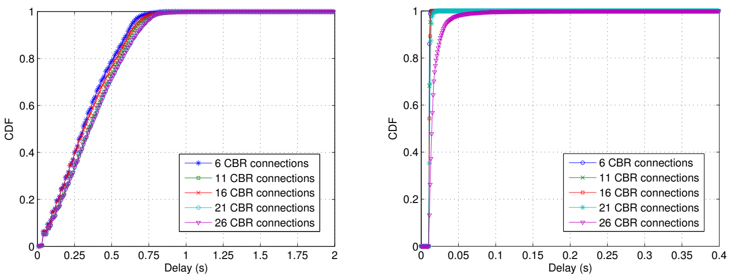

Fig. 6: Delay CDF of TCP (top) and CBR (bottom) (8 data channels of 2Mbps)

F. Sensitivity of Review Phase

The duration of Review phase, Y is one of the critical WiFlex design parameters since it directly affects collision frequency, the access delay and system throughput. Figure 8 shows the system throughput with different values of Y up to 20msec, 20 pairs of devices exchange backlogged traffic, Z = 10msec and $f_i = 8$. Notably, when fast overriding is not used, no data collisions occur for $Y \geq Z$ while collisions are the primary cause of performance drop for $Y < Z$. This experimental results confirms Proposition 2. When fast overriding is used, no data collisions occurs for the whole range of Y as designed. In terms of system throughput, jumbo frame and fast overriding is certainly beneficial. Sensitivity of other phases and effects on priority/fairness are a future research topic.

IV. CONCLUSIONS

The paper proposes a new family of cooperative multichannel protocols. The design goals were to enable the coexistence of devices with widely different characteristics and of applications with different requirements. These protocols are not backward compatible.

The protocols are based on an OFDM-like PHY and an asynchronous split phase MAC. Some devices can transmit only on one OFDM channel while others can transmit on a large number of channels. When they are done transmitting, the devices tune to a common control channel where they go through an Observe then a Review phase. The Review phase enables to implement priorities where a device with a higher score overrides prior reservations. By suitable adjustment of the score of the devices, one can implement a proportionally fair scheduler,

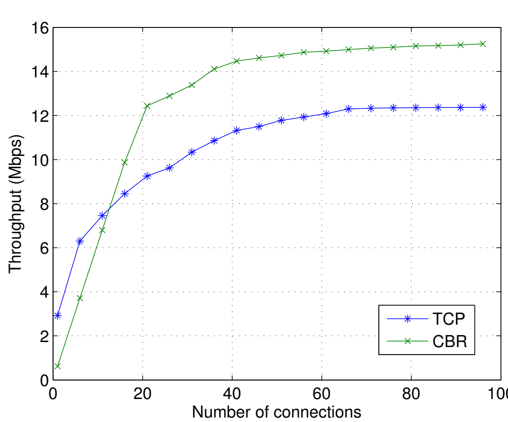

Fig. 7: System throughput(1 control 8 data channels of 2Mbps)

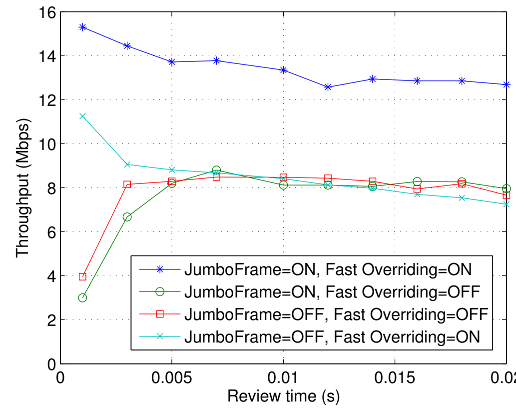

Fig. 8: Review sensitivity

guarantee that real-time applications face a low delay jitter, and that streaming applications get the required throughput whenever possible. The performance characteristics of the protocols were evaluated on detailed simulation experiments based on a multi-channel extension of NS-2 with a refined model of the physical layer.

There are still more issues to be explored in WiFlex. One of the key assumption in our simulation is that the devices are within a single collision domain. Another issue can be impact of different transmission range of heterogeneous devices. Devices with lower transmission power may have smaller transmission range. They are research topics of future research.

REFERENCES

[1] Specification of Bluetooth System, November 2003

[2] Y. Kwok, and M.C.-H. Chek,"Design and evaluation of coexistence mechanisms for Bluetooth and IEEE 802.11b systems," *In Proc. of PIMRC 2004*, Vol. 3, pp. 1767-1771, September 2004.

[3] A. Conti *et al.*, "Bluetooth and IEEE 802.11b coexistence: analytical performance evaluation in fading channels," IEEE JSAC, Vol. 21(2), pp. 259-269, February 2003.

[4] L. Ophir,Y. Bitran, and I. Sherman, "Wi-Fi (IEEE 802.11) and Bluetooth coexistence: issues and solutions," *In Proc. of PIMRC 2004*, Vol. 2 pp. 847-852, September 2004.

[5] Qixiang Pang, and V.C.M Leung,"Improved Channel Classification and Scheduling for Non-collaborative Bluetooth/ WLAN Coexistence," *In Proc. of VTC-Spring 2006*, Vol. 3 pp.1303-1307, 2006.

[6] K. Shuaib, M. Boulmalf, F. Sallabi, and A. Lakas, "Co-existence of Zigbee and WLAN - a performance study," *In Proc. of WOCN 2006*, April 2006.

[7] A. Sikora, and V.F. Groza, " Coexistence of IEEE802.15.4 with other Systems in the 2.4 GHz-ISM-Band," *In Proc. of IMTC 2005*, Vol. 3, pp. 1786 - 1791, May 2005.

[8] L. Berlemann, C. Hoymann, G. Hiertz, and B. Walke, "Unlicensed Operation of IEEE 802.16: Coexistence with 802.11(A) in Shared Frequency Bands," *In Proc. of PIMRC 2006*, pp. 1-5, September 2006.

[9] A. S. Y. Poon, "An energy-efficient reconfigurable baseband processor for flexible radios," *In Proc. of IEEE Workshop on Signal Processing Systems*, Banff, AB, Canada, Oct. 2006.

[10] Jungmin So and Nitin Vaidya, "Multi-Channel MAC for Ad Hoc Networks: Handling Multi-Channel Hidden Terminals Using A Single Transceiver," *In Proc. of ACM MobiHoc*, May 2004.

[11] J. Shi, T. Salonidis, and E. Knightly, "Starvation Mitigation Through Multi-Channel Coordination in CSMA based Wireless Networks," *In Proc. of ACM MobiHoc 2006*, Florence, Italy, May 2006.

[12] J. Mo, H.W. So, and J. Walrand, "Comparison of multi-channel MAC protocols," *In Proc. of MSWIM 2005*, pp. 209-218, 2005

[13] H. W. So, J. Walrand and J. Mo, "McMAC: A Multi-Channel MAC Proposal for Ad-Hoc Wireless Networks", *In Proc. of* IEEE WCNC 2007, Hongkong, China, March 2007.