

# Experimental Implementation of Optimal WLAN Channel Selection Without Communication

D.Malone, P. Clifford, D.Reid, D.J.Leith  
Hamilton Institute, National University of Ireland, Maynooth, Ireland

**Abstract**—[1] proposed a simple decentralised algorithm for channel allocation that is provably correct and requires no message passing or common administrative control between interfering WLANs. In this paper we implement this algorithm using a standard 802.11 hardware testbed and demonstrate that it does indeed offer the potential for effective channel allocation in realistic environments. This includes environments with complex, spatially varying noise and channel dependent propagation behaviour and with time-varying load.

## I. INTRODUCTION AND TESTBED SETUP

For the Communication Free Learning (CFL) algorithm itself, a full list of references and a survey of related work please see [1].

The testbed consists of 10 PC-based embedded Linux boxes based on the Soekris net4801, 5 boxes configured as APs in infrastructure mode and 5 as client stations. We also use 5 PCs acting as monitoring stations to collect measurements to ensure that there is ample disk space, RAM and CPU resources available for collection of statistics. These machines are setup as five WLANs (denoted WLAN A - WLAN E) located in a university office space as shown in Figure 1. All systems are equipped with an Atheros AR5004G 802.11a/b/g mini-PCI card with an external antenna. All nodes use a Linux 2.6.16.20 kernel and the MADWiFi wireless driver. All of the systems are also equipped with a wired Ethernet port, which is used for control of the testbed. Specific vendor features on the wireless card, such as turbo mode, are disabled. Channel scanning is also disabled as we use the CFL algorithm for channel selection. Unless otherwise stated, all of the tests are performed using the 802.11a physical transmission rate of 18 Mbps with RTS/CTS enabled and the channel number explicitly set. With this PHY rate and using 1500 byte packets, the achieved throughput in an isolated WLAN is measured to be approximately 13 Mbps.

To generate wireless network traffic and to measure throughput we use mgen. While many different network monitoring programs and wireless sniffers exist, no single tool provides all of the functionality required and so we have used a number of common tools including tcpdump. Network management and control of traffic sources is carried out using ssh over the wired network.

## II. IMPLEMENTATION OF CFL ALGORITHM

The CFL algorithm requires no special hardware support and, in addition to avoiding message passing, does not require

This work was supported by Science Foundation Ireland grant IN3/03/I346.

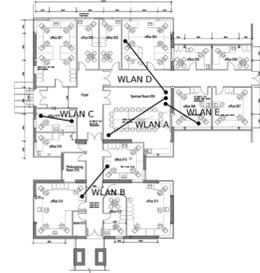


Fig. 1. Plan showing wireless node locations.

clock/slot synchronisation between interfering WLANs. The algorithm is implemented as a user-space perl script that runs on each WLAN AP. WLAN-wide channel switching is achieved by a broadcast instruction from the AP that is received by a user-space script running on each WLAN client station, which then uses the *iwconfig* command to change channel.

The CFL algorithm requires a measure of channel quality. We initially investigated using the RSSI value returned by the AP wireless NIC. However, we found this value to be unreliable – when channel quality is degraded due to interfering WLANs it is quite possible for the background noise level to be low yet for the frame error rate to be high due to colliding transmissions. We therefore use a direct measure of frame error rate as our channel quality metric. Channel quality is estimated from the average frame error rate measured over a 10 second interval; this duration was chosen experimentally.

To allow scripting entirely within user-space we took advantage of RTS/CTS. Using *tcpdump* to monitor packets transmitted, over 10 second intervals we collected statistics on (i) RTS transmissions for which no CTS handshake was received, (ii) transmissions for which the RTS/CTS handshake was successful but the data packet transmission was not ACKed, and (iii) transmissions with successful RTS/CTS and data/ACK handshakes. We label (i) as CSMA/CA collisions, (ii) as frames lost due to interference and (iii) as successful transmissions. The first of these labels is approximate as RTS/CTS handshakes may be lost due to interfering transmissions or noise. However, the CFL algorithm only requires a coarse good/bad measure of channel quality and we find that measuring channel quality by the percentage of type (ii) events and thresholding at 10% is effective. We are also investigating other measures [2].

### III. NATURE OF INTERFERENCE ENVIRONMENT

First we attempted to characterise the interference environment in our testbed.

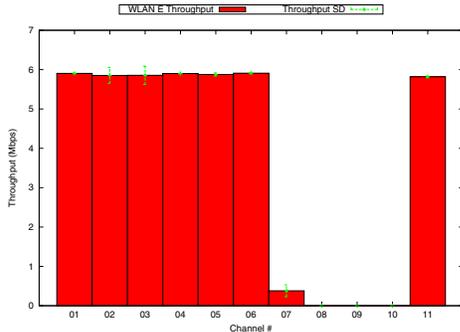


Fig. 2. Baseline throughput for WLAN C versus channel number in 2.4GHz band (no other WLANs active).

The testbed hardware supports operation both in the 802.11a 5GHz band and in the 802.11b 2.4GHz band. Spectrum analyzer measurements revealed little external interference in the 5GHz band (a noise floor of around -80dB being typical), significant external interference was observed in the 2.4GHz band. Figure 2 shows measured throughput versus channel number in the 802.11b band for WLAN C – none of the other WLANs active here, so there is no testbed related interference. It can be seen that there exists significant background noise on channels 7-10. We note that the level of external interference is strongly location dependent and is essentially negligible for WLANs B and E which are located approximately 10m further than WLAN C from the interference source.

Figure 3 shows measurements of the mean rate of successful transmissions versus channel number when a single WLAN is active (WLAN E). Measurements are repeated about an hour apart. The time-varying nature of the channel quality is evident – e.g. compare channels 48 and 153.

Also marked on Figure 3 are error bars that indicate the standard deviation of the error time history measured over a period of 50s. It is evident that variations in channel quality also occur on shorter time-scales. This is shown in more detail in Figure 4 which shows an example time history of measured channel quality over a period of approximately 60 minutes. It can be seen, for example, that the error rate rises to around 15% for a period of about 10 minutes early in this experiment, then falls to around 3% after approximately 30 minutes.

Our measurements indicate that the level of interference between WLANs can be strongly channel dependent. Figure 5 shows the measured interference level between WLANs B and C as the channel number is varied. We found this effect to be particularly pronounced in the 5GHz band, with a significantly lower level of channel dependence measured in the 802.11b 2.4GHz band.

#### A. Spatial Reuse

To investigate the level of spatial reuse feasible in our testbed, we measured the frame error rate between pairs of

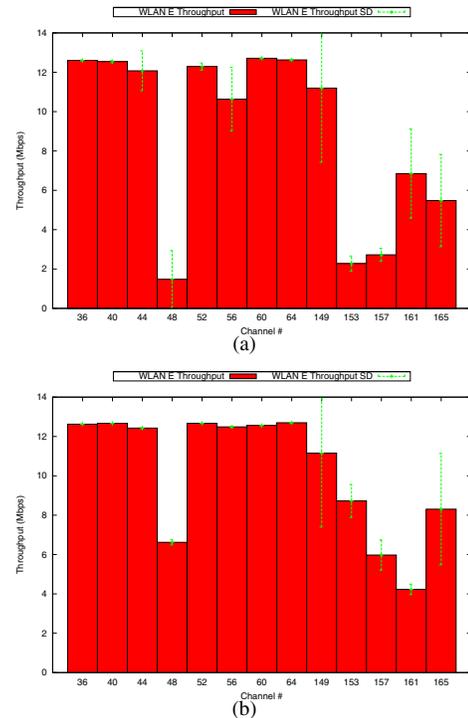


Fig. 3. Measured throughput with a single WLAN active (no interfering WLANs). Measurements are shown for WLAN E over the range of 802.11a channels. The upper and lower plots are about 1 hour apart. Observe the substantial variation in throughput both with channel number and time.

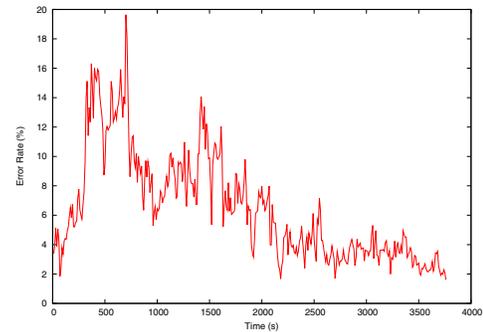


Fig. 4. Example of time-varying channel quality.

WLANs as the channel used by one WLAN was varied. Initially we consider the behaviour when the 802.11a 5GHz band is used. Figure 6(a) shows the measured throughputs of WLANs A and E when WLAN E is held fixed on channel 36 while the channel used by WLAN A is varied between channel 36 and channel 64. Figure 6(b) shows the corresponding measurements for WLANs C and E. Note that unlike in 802.11b/g, 802.11a channels are not numbered consecutively i.e. channels 36 and 40 are in fact adjacent. Observe from Figure 1 that WLANs A and E are located adjacent to each other whereas WLANs C and E are located approximately 10m apart. We therefore expect that a larger separation in channels is needed between WLANs A and E than between WLANs C and E and indeed our measurements support this prediction.

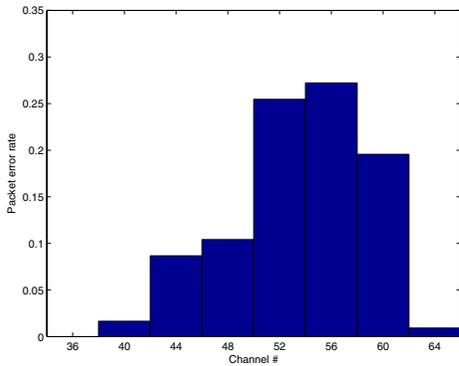


Fig. 5. Measured interference induced error rate versus channel number in 5GHz band. Here WLANs B and C both transmit CBR traffic on the same channel. Plot shows measured packet error rate at WLAN C as the channel number used for transmission is varied (with WLANs B and C always sharing the same channel)..

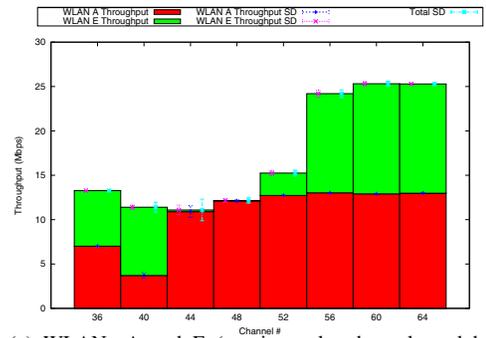
It can be seen that when WLAN A is located on channel 56 and above, the aggregate network throughput is 26Mbps which is approximately the maximum combined capacity that can be achieved by two independent WLANs for the 802.11a settings used here. Observe also that both WLANs achieve throughputs i.e. network capacity is allocated equally. However, when WLAN A is on a channel that is closer to that of WLAN E we have that (i) the aggregate network throughput falls substantially and (ii) the WLANs can experience dramatically different throughputs (e.g. when WLAN A uses channels 44 or 48 it achieves a throughput close to zero, while WLAN E achieves throughput close to 12Mbps). The latter unfairness is associated with hidden node type effects that occur when the WLANs operate on channels that are sufficiently close for their transmissions to interfere yet not so close that they can successfully decode each others transmissions. When the WLANs operate on the same channel, they can decode each others transmissions since the WLANs are located near to each other and thus the 802.11 CSMA/CA operation fairly allocates the available bandwidth. However, the aggregate network throughput is half that achieved when the WLANs operate on orthogonal channels.

This behaviour can be contrasted with that of WLANs C and E. It can be seen from Figure 6(b) that even when WLANs C and E use adjacent channels the aggregate network throughput is nevertheless close to 26Mbps. Note that WLANs C and E are located only 10m apart, yet the attenuation due to walls etc when combined with the attenuation between adjacent channels is sufficient to effectively yield orthogonality of transmissions.

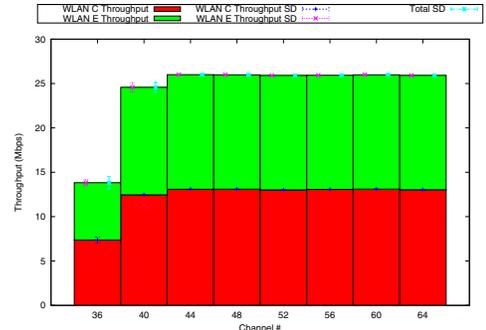
#### IV. COMMUNICATION-FREE CHANNEL ALLOCATION ALGORITHM EXPERIMENTAL RESULTS

##### A. Convergence to non-interfering channel allocation

To demonstrate the CFL algorithm for channel selection, we simultaneously generated traffic between the nodes on each



(a) WLANs A and E (x-axis marks channel used by WLAN A).



(b) WLANs C and E (x-axis marks channel used by WLAN C).

Fig. 6. Measuring potential for channel reuse. Using the 802.11a 5GHz band, WLAN E is held fixed on channel 36 while the channel used by second WLAN is varied. Measurements are shown for WLANs A and E and for WLANs C and E. Height of histogram indicates aggregate throughput of both active WLANs. Light shaded area marks throughput of WLAN E and dark shaded area marks throughput of second WLAN. Also marked on the histogram are the standard deviations of the throughput, which give a measure of throughput variability – it can be seen that the standard deviations are consistently low. WLANs A and E are located adjacent to each other whereas WLANs C and E are located approximately 10m apart.

of the five WLANs. To create a relatively demanding channel allocation task, the channel allocation algorithm was restricted (via scripting) to the use of four 802.11a channels. Initially, all WLANs are started on the same channel.

Figure 7 shows traces of the channel selection time histories for each of the five WLANs as we run the CFL algorithm. Throughput significantly increases once a non-interfering channel allocation is selected, yielding a substantial increase in network capacity: the aggregate throughput from 50-60 seconds is approximately 51 Mbps compared with 11.31 Mbps when the WLANs all use the same channel. That is, we obtain approximately a factor of four increase in network capacity through appropriate channel selection.

##### B. Convergence Rate

Figure 7 shows that the network converges to a non-interfering channel allocation in approximately 20 iterations. The duration of an iteration is determined by the time required to sense channel quality and is set to 10s in our tests yielding an overall convergence time of 200s. Of course, during this convergence period the network continues to achieve a signif-

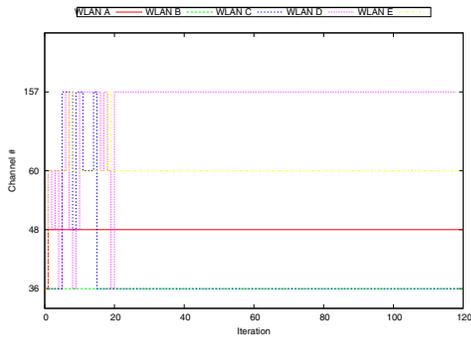


Fig. 7. WLAN channel time histories. Five WLANs, four available channels. Note that in this example the network settles on only three channels.

icant level of throughput. Hence, the cost of the convergence period in terms of throughput is limited.

Secondly, the simulation analysis in [1] indicates that the CFL algorithm converges rapidly under a wide range of conditions and this is confirmed in our experimental tests. For example, the mean convergence time measured over 10 tests is five iterations with five WLANs and four available channels.

### C. Controlling local channel reuse

Observe in Figure 7 that WLAN B and WLAN D settle on the same channel. It can be seen from Figure 1 that these WLANs are located near to each other and on closer inspection of packet traces we find that the nodes in these WLANs are visible to each other (no hidden nodes). That is, both nodes involved in a collision are able to detect that the collision occurred, thus the 802.11 CSMA/CA MAC is able to schedule transmissions properly and the frame error rate (i.e. packet losses not associated with CSMA/CA collisions) is low. Since our objective here in allocating channels is to avoid hidden node and interference related problems, this behaviour is as expected. Indeed, it seems desirable in dense deployments as it increases the level of channel reuse. That is, channel reuse is possible not only between WLANs located so far apart that their transmissions do not interfere, but also between WLANs located close together so that CSMA/CA operates correctly.

It is straightforward to force nearby WLANs to use different channels by observing beacons; this has been verified experimentally.

### D. Impact of external/channel dependent interference

Our measurements of the testbed interference environment highlight the presence of external interference sources in the 2.4GHz band, and the channel dependent nature of the level of interference between WLANs.

Returning to the channel dependent interference between WLANs B and C noted in Figure 5, we recorded statistics on the channels selected by these WLANs over a series of 10 tests. In line with Figure 5 we find that, as expected, the CFL algorithm settles on either channel 36,40 or 64 and avoids the lower quality channels. Similarly, in the case of WLAN E it can be seen from Figure 3 that the quality of certain channels can be strongly time-varying. We can also observe in Figure

3 that certain channels are consistently of good quality, e.g. channels 36-44 and 60-64. Our measurements confirm that the CFL algorithm automatically adapts to channel dependent interference by avoiding the low quality channels and settling on the good quality channels.

### E. Time-varying network conditions

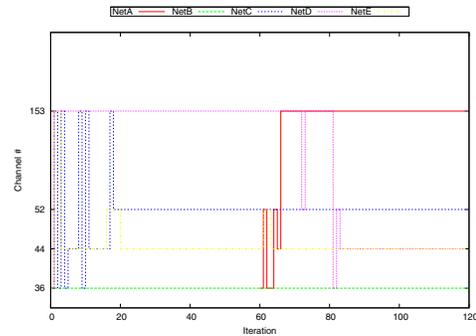


Fig. 8. Example of a new WLAN becoming active. Four WLANs active initially, with fifth WLAN beginning at time 60.

The level of interference between WLANs is dependent on the traffic load on each WLAN. In particular, when a WLAN carries no traffic and therefore generates essentially no interference. Importantly, when a WLAN that has been inactive becomes active, we require to allocate a channel to that WLAN and this may require reconfiguration of the channel allocations used by other nodes. Since the CFL algorithm is convergent (i.e. stays settled on a non-interfering channel allocation once it has found one), it can be left running at all times. Changes in the network, such as a previously dormant WLAN becoming active, that create new interference will then automatically activate the CFL algorithm to adapt the channel allocation to restore a non-interfering allocation. This is illustrated in Figure 8. Here, we start with four WLANs which quickly settle on a non-interfering channel allocation. At iteration 60 of the CFL algorithm, a fifth WLAN is activated (i.e. begins transmitting traffic). It can be seen that the network automatically reconfigures its channel allocation to accommodate this new WLAN and quickly settles on a new non-interfering configuration. At time 50 the total throughput was 50.1Mbps; at time 120 this had increased to 60.3Mbps.

## V. CONCLUSIONS

[1] discusses valuable theoretical properties of the CFL algorithm; in this paper we have demonstrated its ease of implementation and its practicality in a wide range of challenging real world conditions. The algorithm is flexible enough to perform well even when not operating in the regime assumed by the theory.

## REFERENCES

- [1] Leith,D.J., Clifford, P., "A Self-Managed Distributed Channel Selection Algorithm for WLANs". *Proc. ACM/IEEE RAWNET*, Boston, 2006.
- [2] Malone, D.W., Clifford,P., Leith,D.J., 2006, MAC Layer Channel Quality Measurement in 802.11. *IEEE Communications Letters*, February, 2007.