Virtualization for Load Balancing on IEEE 802.11 Networks

Tibério M. de Oliveira[1], Marcel W.R. da Silva[1], Kleber V. Cardoso[2], and José Ferreira de Rezende[1]

[1] Universidade Federal do Rio de Janeiro - Rio de Janeiro, RJ, Brazil
{tiberio,marcel,rezende}@gta.ufrj.br
[2] Universidade Federal de Goiás - Goiânia, GO, Brazil
kleber@inf.ufg.br

Abstract. In IEEE 802.11 infrastructure networks composed by multiple APs, before a station can access the network it needs to make a decision about which AP to associate with. Usually, legacy 802.11 stations use no more than the signal strength of the frames received from each AP to support their decision. This can lead to an unbalanced distribution of stations among the APs, causing performance and unfairness problems. This work proposes a new approach that combines the number of associated stations and the current load of each AP plus the virtualization of client wireless interfaces. In this approach, stations frequently switch of association among APs and stay on each one of them for a time interval that is calculated based on the number of associated stations and the channel current load. Simulation results confirm the improvement obtained in the load balancing and fairness on network capacity allocation, while keeping the maximum network utilization.

Keywords: Wireless networks, scheduling, 802.11, association control.

1 Introduction

Nowadays, there is a large number of IEEE 802.11 access points (APs) available in both private and public access networks. Before a client station (STA) can have access to the data transmission service provided by such networks, it has to follow procedures of association and authentication with one of the APs in its transmission range. Initially, the STA detects APs in its vicinity by scanning wireless channels and collecting responses (probe responses and/or beacon frames) from them. Then the STA authenticates and associates with the AP from which it received frames with the highest RSSI (Received Signal Strength Indicator).

As presented in [1], this association metric does not ensure efficiency in the resources usage and may lead to a poor performance due to the unbalanced number of associated stations among the APs. Alternative approaches have been proposed [1,2,3], which perform a load balancing among APs by including load conditions in the frames in order to allow the STA to select the least loaded AP.

P. Sénac, M. Ott, and A. Seneviratne (Eds.): MobiQuitous 2010, LNICST 73, pp. 237–248, 2012.

However, most of the proposals assume that the current load is given by the number of stations associated with the AP, not taking into account the amount of traffic generated by them.

Many 802.11 manufacturers also provide proprietary solutions for load balancing [16,17,18]. In general, there is not enough information about the algorithms applied in these solutions, so their performance can not be evaluated. Every proprietary solution is based on the AP hardware from that specific manufacturer which allows the use of extra and nonstandard features and protocols. This brings a well-known and undesirable drawback: cross hardware manufacturer incompatibility.

Our proposal takes into consideration the effective throughput achieved by the stations and uses wireless network interface virtualization [4] to perform load balancing. The virtualization scheme allows a single physical interface to offer simultaneous connectivity to more than one AP [5]. Only standards-compliant resources are employed in our solution. Through a simulation study, we show that this proposal outperforms the standard RSSI-based association control and other approach for load balancing known as DLBA (Dynamic Load Balancing Algorithm) [3].

The rest of this paper is organized as follows. In Section 2, we briefly present some important related works. Section 3 exposes some important concepts about load balancing and IEEE 802.11 network virtualization. Section 4 presents our proposal of load balancing through network virtualization. In Section 5, we show the results in comparison with some other approaches. Finally, in Section 6, we present our conclusion.

2 Related Work

Recently, virtualization has become an important tool in several areas, such as operating systems [6], faults detection and diagnosis [7]. In wireless networks, it has been applied in the handover process [8,9] and network/interfaces virtualization [10,11,4,5].

The IEEE 802.11 physical interface virtualization allows an STA to associate with multiple APs simultaneously. This capability can be used with the purpose of allowing the concurrent access to multiple networks or virtually increasing the connectivity in a unique infrastructure network [5]. In this last scenario, it is also required that STAs constantly change their association among the APs in order to announce their presence. Currently, the traditional association approach takes into account only the RSSI measured by STAs from multiple APs. The main drawback of this approach is that it can lead to an unbalanced distribution of STAs among the APs, which can drastically reduce network performance [3,1,9,12].

To circumvent this problem, some metrics for AP association that define a relation between RSSI and the amount of associated stations with an AP were proposed [3,1,2]. In this work, we propose a load balancing mechanism based on IEEE 802.11 physical interface virtualization, which uses a metric derived from the amount of associated STAs and channel load to determine how long an STA will stay associated with each AP. This proposal uses a virtualization scheme called Frequency Hopping (FH) and prioritizes the AP that has the lowest channel occupation.

3 Background

3.1 IEEE 802.11 Network Virtualization

The main virtualization focus is to allow a single substrate to execute multiple virtual experiments. The wireless medium can be virtualized by different techniques as presented in [4]. The most common are: FDMA, TDMA, Combined TDMA and FDMA, CDMA, SDMA, and Frequency Hoping (FH).

The FH scheme is a dynamic version of the Combined TDMA and FDMA scheme. It allows experiment partitioning by allocating a unique sequence of frequency and time slots for each virtual experiment. However, differently from the Combined TDMA and FDMA scheme, it allows that the same experiment uses different sequences. It is the most flexible and complete scheme to be used with IEEE 802.11 networks.

Our mechanism uses the FH scheme combined with an algorithm to choose the sequences, i.e. channels and time slot sizes to be used by each STA. The algorithm employs a metric based on media occupation and the amount of associated STAs with each AP. Our proposal also makes use of IEEE 802.11k and 802.11r standards in following manner. The proposed metric requires some data to be collected by the 802.11k, and the wireless interface virtualization employs 802.11r for fast handover.

3.2 IEEE 802.11k and IEEE 802.11r Standards

The IEEE 802.11k is an amendment to IEEE 802.11-2007 standard for radio resource management, which aims to increase the physical layer and medium access availability. For this purpose, it defines a sequence of measurements requests and reports about radio and network information that can be used by upper layers in different ways. It presents low overhead of control messages and low processing requirements, and provides enough accurate measurements for our proposal.

The radio measurements in wireless networks help applications to adapt automatically to the dynamic medium conditions, facilitating the management and maintenance of a WLAN. The standard specifies a generic pair of *Radio Measurement Request* and *Radio Measurement Report* frames, which can be specialized to acquire measurements such as channel load, STA statistics, number of neighbors, etc.

When an STA switches between APs in a handover process, it starts a reassociation with the new AP. During this process, the IEEE 802.11r implemented in the APs takes care of changing forwarding tables of L2 devices by issuing gratuitous ARP messages on the distribution system. It also allows the involved APs to exchange authentication information, which drastically reduces the time spent in the handover process.

4 Load Balancing by Using Virtual Interfaces

In general, a suitable load balancing provides an efficient use of resources, which in most cases improves fairness in their allocation. Thus, by properly distributing STAs across APs, the aggregate network throughput is increased and the fairness in the network capacity allocation is improved. The DLBA association control algorithm balances the number of STAs associated with each AP by using the RSSI. However, since STAs may present different traffic profiles, this is not enough to proper balance the network load, and hence, it is not suitable for providing throughput or delay fairness.

In this context, we propose a new load balancing mechanism based on the time slot scheduling of a frequency hopping virtualization technique. This mechanism is completely distributed and it is only needed in the STAs, therefore not requiring changes in the APs. Also, its implementation relies on standards (IEEE 802.11r and 802.11k).

In the proposed mechanism, the STAs manage associations with multiple APs using one virtual wireless interface for each AP. An STA stays associated with each AP for a time interval that is dynamically adjusted by the scheduling algorithm, which takes into account measurements of channel occupation and number of STAs associated with each AP. This time interval is always a fraction of a scheduling cycle and is called active time of a virtual interface.

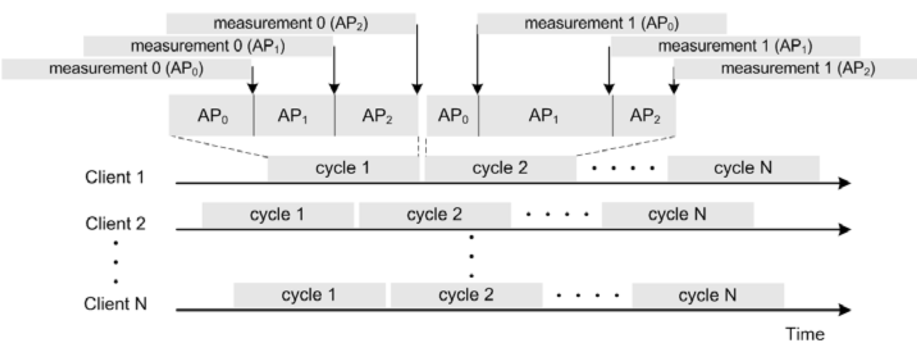


Fig. 1. Scheduling cycle and active times of virtual interfaces

Figure 1 illustrates the scheduling process in which an STA periodically chooses a different AP to associate with. When the APs in the STA's coverage area operate in different channels[1], the scheduling process becomes a client-based FH virtualization scheme.

[1] Our mechanism does not depend on an efficient channel allocation, but this is a common assumption in infrastructure networks with centralized control.

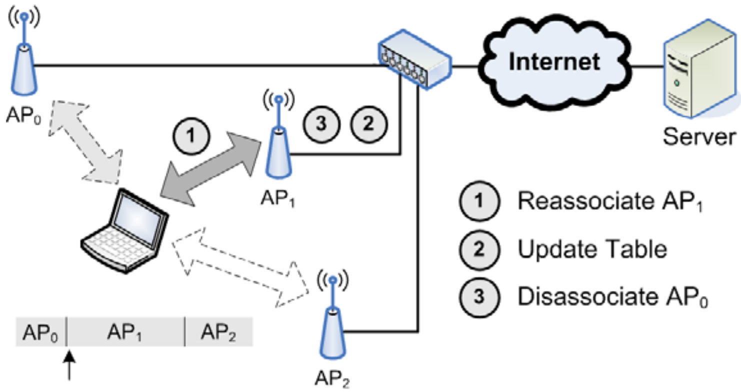


Fig. 2. Fast handover during virtual interface switching

Just before the end of an active time, the STA collects information from the current AP concerning channel occupation and number of associated STAs during the present active time. At the end of every scheduling cycle, the STA uses the collected measurements to compute fractions (or weights) for the calculation of the active time of each virtual interface in the next scheduling cycle. The measurements are performed and made available by the APs using IEEE 802.11k.

In Fig. 1, STA 1 has three virtual interfaces since it is in the coverage area of three APs[2]. To avoid synchronization of active times of different STAs on the same AP, every active time is randomly varied by 10% of its current computed value.

In order to keep the unawareness of the APs concerning the load balancing, our mechanism employs the IEEE 802.11r protocol to implement a fast handover process with authentication. It allows an uninterrupted forwarding of downlink traffic during virtual interface switching. As illustrated in Fig. 2, when an STA switches to a new AP (AP_1), it performs a fast authentication before reassociating, and then initiates an IEEE 802.1X authentication. After that, the IEEE 802.11r 4-way handshake is performed between the STA and the new AP. If this operation is successful, the new AP updates the L2 forwarding tables of network devices of the distribution system (DS). The whole process of migrating from one AP to another takes a non-negligible time. In IEEE 802.11r, this time is around 40 and 50 ms (excluding the scanning time) [13].

4.1 Active Time Computation Algorithm

The performance of the proposed load balancing mechanism is closely tied to the active time duration of each virtual interface. After each cycle, these durations are recomputed to be used in the next cycle. These updates are based on channel load and the number of associated STAs per AP. Initially, the mechanism computes an estimate of the idle time for each STA associated with $AP(i)$, as described by Equation 1:

[2] This number may vary from cycle to cycle.

$$\begin{cases} s \in AP(i), & \overline{I_{(i)}}(s) = \dfrac{1 - Ch_{Load}(i)}{NSt(i)}, \\ s \notin AP(i), & \overline{I_{(i)}}(s) = \dfrac{1 - Ch_{Load}(i)}{NSt(i) + 1} \end{cases} \tag{1}$$

where $\overline{I_{(i)}}(s)$ represents the average amount of time the $AP(i)$ has stayed idle per associated STA, $Ch_{Load}(i)$ is the last channel load reported by the $AP(i)$, and $NSt(i)$ is the number of STAs associated with $AP(i)$. When the STA s is not yet associated with $AP(i)$, the equation includes the STA s as if it was already associated with $AP(i)$ in order to artificially account for its contribution to the load.

The sum of all time slots is equal to the duration of a scheduling cycle, and the number of time slots is equal to the number of APs (N_{APs}). Time slot durations are determined by a weight that is assigned to each time slot. Hence, the weights ($W_{(i)}(s)$) computed by an STA s are normalized by the scheduling cycle duration to obtain a percentage of the cycle. This computation is described by Equation 2:

$$W_{(i)}(s) = \frac{\overline{I_{(i)}}(s)}{\sum_{k=1}^{N_{APs}} \overline{I_{(k)}}(s)} \quad . \tag{2}$$

Since the active time duration is adaptive, it allows our mechanism to dynamically track the network load. In other words, an STA allocates time slots proportionally to the amount of channel load and number of stations reported by APs. For example, an STA allocates larger time slots to APs that have a lower amount of load per associated STA. Therefore, the proposed virtualization mechanism provides a dynamic load balancing across the APs while performing a fair resource sharing among STAs.

To retain association with an AP, the amount of maintained state is low as it can be see in [5], which implements virtual interfaces at the driver level. Our proposal only adds the active time per AP to the state because this information summarizes the decision of the algorithm. Besides, the number of simultaneous association is not high, because infrastructured networks commonly have a deployment plan that tries to avoid too much superposition among APs coverage. This approach is taken because superposition in excess degrades performance and wastes resources.

5 Results

The ns-2 simulator, with a set of modifications, was used to evaluate the proposed load balancing mechanism. At the network and MAC layers, we added interface queues to handle the traffic of each virtual interface. Figure 3 depicts an example where STA 1 has three virtual interfaces. Each virtual interface is modeled as a transmission queue of a few packets capacity. In this example, the first virtual interface is at its active time, and is associated with AP 1 using channel 1. Modifications were also needed to provide 802.11k measurements

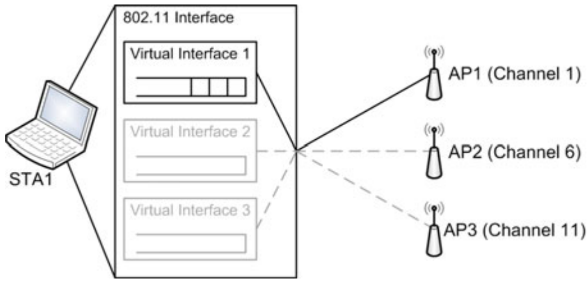


Fig. 3. Virtual interfaces implementation

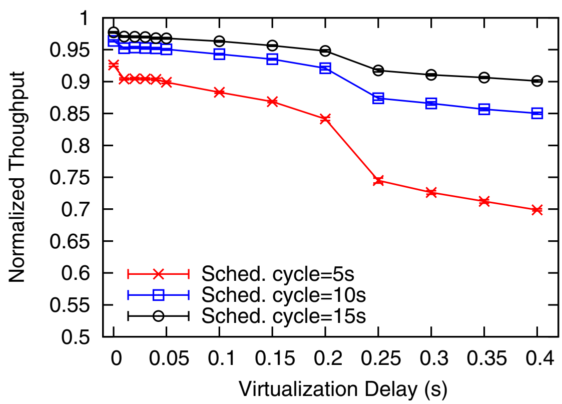


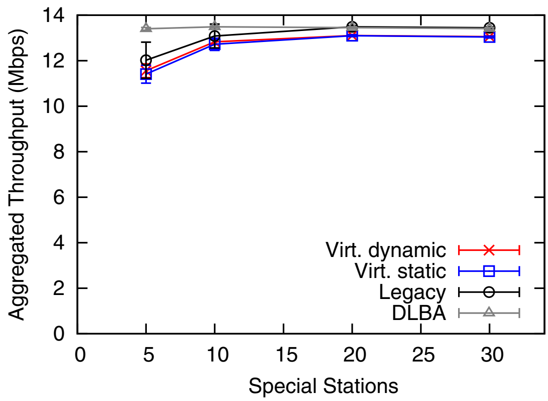
Fig. 4. Virtualization overhead

of the number of associated STAs, channel occupation and RSSI. Moreover, we have implemented the proposed load balancing mechanism and other association control mechanisms to make performance comparisons.

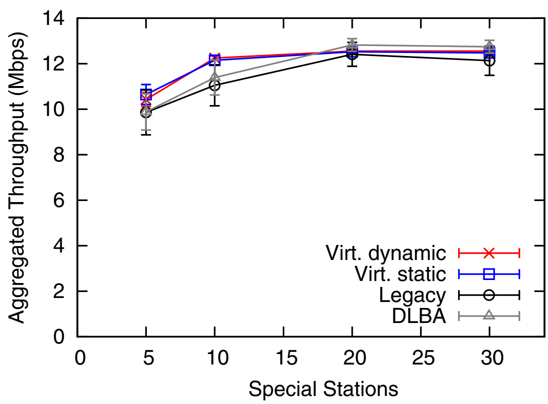
Before evaluating the proposed mechanism in commonly used IEEE 802.11 infrastructure scenarios, we have performed some experiments to assess the overhead imposed by the virtualization. The delay incurred when an STA virtualizes to a new AP, called here of virtualization delay, is due to channel switching, re-association, authentication and L2 forwarding tables update. During this time, all in transit packets can be lost, degrading flows performance. To do this evaluation, we run simulations with only one virtualized STA in the range of three APs interconnected by a unique L2 switch. This STA performs an FTP download during all the experiment. Figure 4 shows the FTP flow average throughput normalized in respect to the throughput obtained by simulations when no virtualization is used. The normalized average throughput is plotted as a function of the virtualization delay for different scheduling cycle durations. According to [9], typical values for the virtualization delay are not larger than 40 ms. Results show that the throughput decreases by at most 10% in the range of 0 and 40 ms. The smaller is the duration of the schedule cycle, the more affected is the throughput since the STA virtualizes more often. When a channel scanning overhead, which has a typical value of 350 ms, is added to the virtualization delay, the performance degradation is severe. However, our mechanism does not require channel scanning since each STA already knows which APs it is virtualizing with.

To evaluate the performance of the proposed mechanism, the next simulations involve an infrastructure network composed by three APs disposed at the center of a square area, respecting a minimum distance of 120 meters among them. Each AP is configured to use one of the non-overlapping channels (1, 6 and 11).

Two classes of STAs were positioned in the communication range of these three APs: legacy and special STAs. Legacy STAs download Web traffic and use a traditional association method. Special STAs downloads FTP traffic during all simulation. In the simulations, special STAs use one of the following association control methods:



(a) Without legacy stations, only special stations



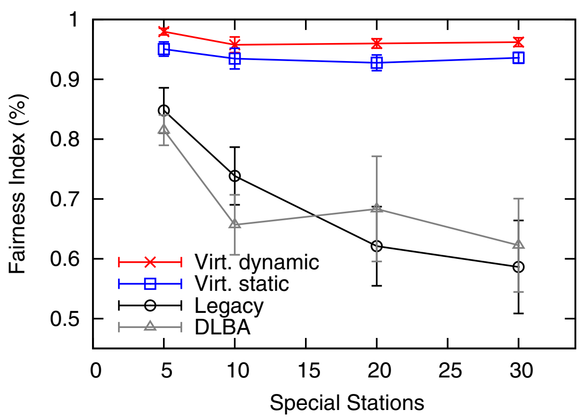
(b) 30 legacy stations with Web traffic

Fig. 5. Aggregated throughput of the FTP flows generated to special STAs as a function of the number of special STAs

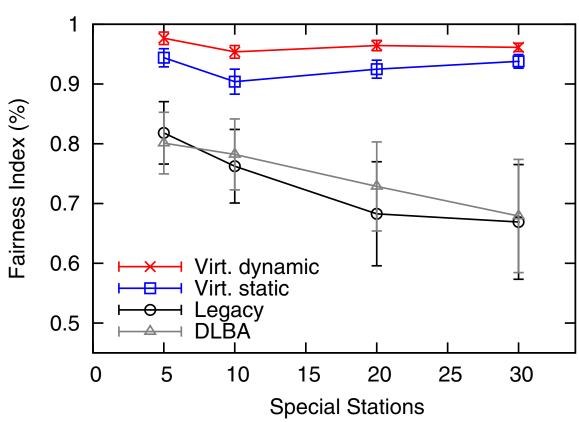
- **Legacy**: Traditional association method, i.e. the same used by the legacy stations [14];
- **DLBA**: The DLBA association mechanism [3];
- **Static Virtualization**: The proposed association mechanism with static time slot duration;
- **Dynamic Virtualization**: The proposed association mechanism with dynamic time slot durations.

Each simulation run lasts for 200 s and 30 different STAs positioning scenarios were generated for each configuration set. The results presented are the mean of values for the 30 scenarios with confidence interval bounds at a confidence level of 95%.

Figure 5 presents the aggregate throughput of special STAs in two scenarios, without and with 30 legacy STAs. The aggregate throughput is given by the



(a) Only special STAs



(b) 30 legacy STAs with Web traffic.

Fig. 6. Fairness index

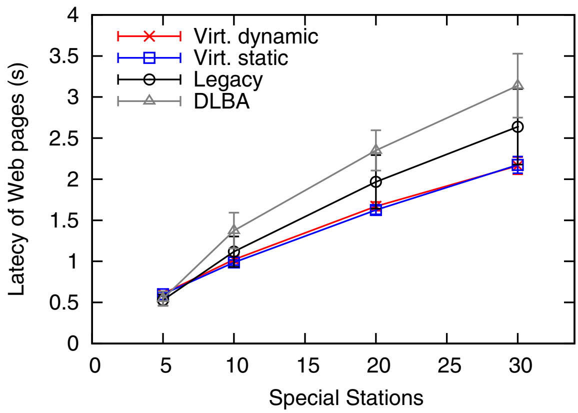


Fig. 7. Mean latency of web pages for legacy STAs

sum of the throughput of all FTP flows. Results show that with a small number of STAs the maximum network capacity is reached in all scenarios. This demonstrates that virtualized STAs are able to use all available capacity even subject to virtualization overhead.

Figure 6 shows the fairness index, as defined in [15], which was calculated over the throughput values obtained by the special STAs. It represents how fair the network capacity is distributed among the STAs. The closer this index is to 1, the fairer is the distribution. According to Fig. 6, the fairness index of virtualized stations is the best among all association algorithms. This demonstrates that our proposal provides a better load balancing, ensuring a fairer sharing of network resources. The difference between static and dynamic virtualized stations is related to the fact that the static algorithm keeps bad associations for a longer time: a large number of STAs may be associated with the same AP simultaneously. On the other hand, the dynamic algorithm is able to adapt to the time duration that the STA is associated with each AP according to the load measurements in each channel. This implies better fairness in resource sharing through load balancing in each AP. DLBA and Legacy association algorithms, which rely on RSSI measurements, present poor fairness and larger variance.

Other important performance metric is presented in Fig. 7. This figure shows the web page average latency obtained by legacy STAs when the amount of special STAs increases. This metric allows evaluating how friendly the association algorithm is to the legacy stations. Once again, DLBA and Legacy STAs present a lower performance with a significant increase in the average latency to visualize web pages. The virtualized methods have a low impact in the average latency of web pages as the amount of special STAs increases.

6 Conclusions

This work has presented a new mechanism that uses IEEE 802.11 network interfaces virtualization as a way to perform load balancing. Through simulations experiments, our proposal has been evaluated and compared with a traditional approach and another load balancing proposal that takes into account only the RSSI. The results show that the virtualization overhead is negligible and the full aggregate capacity of the network can be achieved. However, the most important result of the evaluation is that the proposed mechanism improves the fairness in the network capacity sharing. Considering the throughput experienced by client stations, our proposal provides to all clients equal average throughput. Additionally, our mechanism avoids performance degradation of legacy user communications.

The performance of the two presented virtualization techniques differs only in one aspect: the fairness index results. Slight differences in the fairness index [15] results represent large differences in the throughput distribution experienced by the clients. Thus, as the main objective of the proposed mechanism is to provide better load balancing to the clients, the dynamic virtualization technique is the best choice when compared to the other mechanisms evaluated. The improvements come at a low cost in complexity, since the measurements are simple and based on an IEEE standard. Besides, virtualization of network interface is a well-known mechanism that is becoming available natively in modern operating systems, including the ones that run on mobile devices.

As future works we intend to compare the performance of the dynamic virtualization to other mechanisms in the literature. We also intend to develop a prototype of the proposed virtualization mechanism using off-the-shelf 802.11 network devices supported by the open source MadWifi driver.

References

1. Bejerano, Y., Han, S.-J., Li, L.E.: Fairness and load balancing in wireless LANs using association control. In: MobiCom 2004: Proceedings of the 10th Annual International Conference on Mobile Computing and Networking, pp. 315–329. ACM, New York (2004)
2. Villegas, E.G., Ferre, R.V., Aspas, J.P.: Cooperative load balancing in IEEE 802.11 networks with cell breathing. In: IEEE Symposium on Computers and Communications, ISCC 2008, pp. 1133–1140 (July 2008)
3. Sheu, S.-T., Wu, C.-C.: Dynamic Load Balance Algorithm (DLBA) for IEEE 802.11 wireless LAN. Tamkang Journal of Science and Engineering 2, 45–52 (1999)
4. Paul, S., Seshan, S.: Technical document on wireless virtualization. GENI: Global Environment for Network Innovations, Tech. Rep. (September 2006)
5. Bahl, P., Bahl, P., Chandra, R.: MultiNet: Connection to multiple IEEE 802.11 networks using a single wireless card. Microsoft Research, Redmond, WA, Tech. Rep. (August 2003)
6. Barham, P., Dragovic, B., Fraser, K., Hand, S., Harris, T., Ho, A., Neugebauer, R., Pratt, I., Warfield, A.: Xen and the art of virtualization. In: SOSP 2003: Proceedings of the Nineteenth ACM Symposium on Operating Systems Principles, pp. 164–177. ACM, New York (2003)
7. Adya, A., Bahl, P., Chandra, R., Qiu, L.: Architecture and techniques for diagnosing faults in IEEE 802.11 infrastructure networks. In: MobiCom 2004: Proceedings of the 10th Annual International Conference on Mobile Computing and Networking, pp. 30–44. ACM, New York (2004)

8. Wang, W.-C., Hsu, C.-H., Chen, Y.-M., Chung, T.-Y.: SCTP-based handover for VoIP over IEEE 802.11 WLAN using device virtualization. In: The 9th International Conference on Advanced Communication Technology, vol. 2, pp. 1073–1076 (February 2007)
9. Ramine, I., Savage, S.: SyncScan - Practical fast handoff for 802.11 infrastructure network. In: INFOCOM 2005, March 13-17, vol. 1, pp. 675–684 (2005)
10. Smith, G., Chaturvedi, A., Mishra, A., Banerjee, S.: Wireless virtualization on commodity 802.11 hardware. In: WinTECH 2007: Proceedings of the Second ACM International Workshop on Wireless Network Testbeds, Experimental Evaluation and Characterization, pp. 75–82. ACM, Montreal (2007)
11. Mahindra, R., Bhanage, G.D., Hadjichristofi, G., Seskar, I., Raychaudhuri, D., Zhang, Y.Y.: Space versus time separation for wireless virtualization on an indoor grid. In: Next Generation Internet Networks, NGI 2008, pp. 215–222 (April 2008)
12. Athanasiou, G., Korakis, T., Ercetin, O., Tassiulas, L.: Dynamic cross-layer association in 802.11-based mesh networks. In: 26th IEEE International Conference on Computer Communications, INFOCOM 2007, pp. 2090–2098. IEEE, Los Alamitos (2007)
13. Bangolae, S., Bell, C., Qi, E.: Performance study of fast BSS transition using IEEE 802.11r. In: IWCMC 2006: Proceedings of the 2006 International Conference on Wireless Communications and Mobile Computing, pp. 737–742. ACM, New York (2006)
14. Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications, IEEE Std. 802.11 (August 1999)
15. Jain, R.K., Chiu, D.-M.W., Hawe, W.R.: A Quantitative Measure of Fairness and Discrimination for Resource Allocation in Shared Computer System. Digital Equipament Corporation, Maynard, MA, USA, DEC Research Report TR-301 (September 1984)
16. Cisco, Aggressive Load Balancing on Wireless LAN Controllers (WLCs), http://www.cisco.com/application/pdf/paws/107457/load_balancing_wlc.pdf (last access: July 25, 2010)
17. Aruba, Optimizing Aruba WLANs for Roaming Devices, http://www.arubanetworks.com/pdf/technology/DG_Roaming.pdf (last access: July 25, 2010)
18. Trapeze, Client Load Balancing, http://www.trapezenetworks.com/solutions/client_load_balancing/ (last access: July 25, 2010)