An Architecture for Extensible Wireless LANs

Rohan Murty[†], Alec Wolman[‡], Jitendra Padhye[‡], and Matt Welsh[†]
[†]Harvard University, [‡]Microsoft Research

1 Introduction

Today's wireless LANs are a mess. The current 802.11 family of WLANs involves a jumble of competing standards, and a slew of implementations with varying degrees of interoperability and conformance to those standards. This situation has arisen in part from the need to innovate and evolve these networks over time, driven by new applications and increasing load. In the future we expect this situation to get worse, given the shift towards wireless as the dominant access network. Driving these changes are new devices such as Wi-Fi enabled VOIP handsets and audiovisual equipment, as well as new services such as Apple's Time Capsule which performs background backups over wireless. In the longer term, we anticipate WLANs will become the *default* access network and will support filesystem and server traffic as well.

Currently, it is not unusual for wireless LAN users to experience performance and reliability problems. A significant factor is the scarcity and poor utilization of the wireless spectrum, which suffers from a "tragedy of the commons". Scaling up WLANs to meet new traffic demands, especially time-critical applications involving audio, video, or sensor telemetry data, is proving to be difficult. This is in spite of underlying innovations at the PHY layer, which largely address the need for more throughput, but not how that throughput is managed. Moreover, enterprises often have an interest in imposing customized policies on WLAN traffic, for example, prioritizing time- and safety-critical traffic over large file downloads.

Existing wireless LANs make poor use of the wireless spectrum, largely due to the "intelligence" which is hard-coded into the vendor-specific software and firmware of wireless LAN clients. For example, WLAN clients control the decisions for AP associations, transmit power control, and physical data rate adaptation. The 802.11 standards specify the mechanisms, yet the policy is left entirely up to vendor-specific implementations. As a result, vendors view these areas as an opportunity to innovate and compete with each other. However, the end result of these attempts to innovate are limited, and we argue that is primarily an architectural limitation: by viewing the client as a stand-alone entity that solely uses local information about the devices it is interacting with, many important opportunities for improving the behavior of these algorithms cannot be realized.

The current approach to innovation and evolution in wireless LANs is primarily through standardization, which has resulted in an alphabet soup of protocols within the 802.11 family. Certain Wi-Fi vendors offer vendor-specific WLAN extensions such channel bonding, or non-standard data rates to support communication with weak signals. Such extensions only work when both the AP and the client are using the same brand of Wi-Fi chipset and software drivers, which prevents widespread adoption. The downside to standardization is primarily the glacial progress in deploying new protocols. The standards process takes a very long time to reach agreement, and even after standards are ratified it takes a long time to replace and/or upgrade the wide variety of client equipment utilizing the infrastructure.

We argue that to move away from the current mess, we need to rethink the basic architecture of wireless LANs. Our focus is not on changing the fundamental building blocks such as PHY-layer coding schemes or the CSMA nature of the MAC. Rather, we are interested in a developing an architecture that allows for extensibility, to ensure WLANs can adapt to meet future needs. We are guided by two key design principles:

- *Whatever we design today will be wrong in five years.* If history is any guide, we cannot anticipate all future uses for wireless LANs. Furthermore, the best way to evaluate innovations is through actual deployments with real users. With current WLANs, deploying new hardware and upgrading NICs and drivers for all of the affected clients is an expensive proposition, not to mention the management and personnel costs involved. We argue that an extensible WLAN can adapt to new uses, and can allow rapid deployment and evaluation of experimental designs.
- *Infrastructure should manage the wireless spectrum.* Networks can make the best use of resources by shifting much of the responsibility for managing the wireless spectrum (such as associations, power control, channel assignment, and physical layer rates) to the infrastructure, away from the individual clients. This has the additional benefit of making it easier to evolve the system because clients take much less of the responsibility for spectrum management. This approach also allows administrators to customize the network's policies for handling different traffic demands.

This paper describes *Trantor*[1], a new architecture for wireless LANs. Trantor's architecture is based on global management of channel resources, taking this responsibility explicitly away from clients and moving it into the infrastructure. To provide extensibility, the interface between the infrastructure and clients is simple and relatively low-level. Clients implement a small set of relatively simple commands which allows the complicated logic of the algorithms to exist primarily within the infrastructure. The commands fall into two categories: *measurement commands* allow the infrastructure to instruct clients to gather local information on channel conditions, such as RSSI from visible APs, and to report this information periodically; and *control commands* allow the infrastructure to control the behavior of clients, such as setting the transmit power or instructing a client to associate with a specific AP. Each client still implements a basic CSMA MAC for individual packet transmissions, but is otherwise not responsi-

[1] Named after the ruling planet of the first Galactic Empire as described by Isaac Asimov in the *Foundation series*.

Proposed Standard	Year Proposed	Year Incorporated
802.11m	1999	2007
802.11d	2001	2007
802.11h	2003	2007
802.11k	2003	2008
802.11r	2004	2008
802.11T	2004	No
802.11v	2005	No

Table 1: **Proposed amendments to 802.11, dates of first task group meeting and dates of incorporation into the standard**

ble for most aspects of wireless management.

Trantor takes its cue from recent work on centralized management of resources in DenseAP [12], yet it pushes this approach much further by adding support for controlling physical data rates, transmission power, and clear-channel assessment. This approach can yield tremendous gains in efficiency, and also can provide a control point for the infrastructure to impose policies for shaping certain classes of traffic, prioritizing individual users, and so forth. Trantor's architecture is inherently evolvable to support new classes of applications, and supports global policy decisions to manage traffic load. Yet another key benefit of Trantor is that it allows the infrastructure to collect and use historical information (e.g. observations of client behavior over long time scales) to customize the behavior of a WLAN to the characteristics of its particular environment.

2 Background and Motivation

There is mounting evidence that wireless LAN architectures cannot keep up with demands of new classes of applications. Wireless LANs are already the most popular access network in homes and hotspots, and are rapidly becoming dominant within the enterprise. Apart from laptops, wireless interfaces are now found in a wide range of consumer devices, including smartphones, PDAs, media servers, set-top boxes, and network-attached storage appliances. Indeed, we anticipate that wireless LANs will dominate on desktops and even some servers, leading to increased channel load and new traffic patterns (such as filesystem traffic).

Industry is scrambling to respond to this challenge by introducing ever more sophisticated upgrades to the 802.11 standard, including 802.11e (QoS support), 802.11k (association protocol), 802.11d (regulatory domains), and many others. The need for interoperability between clients and APs, as well as backwards compatibility with previously-deployed technologies, mandates standardization which is an inherently slow process. Table 1 shows some of the 802.11 amendments currently in task group, along with the year the task group started. This situation has gotten to the point where we have "metastandards" designed to manage the documentation (802.11m) and testing (802.11T) of other standards. To illustrate the difficulty in deploying new improvements, consider the time from when the first security flaws in WEP were discovered to the present. Although today WPA and WPA2 have significant use, there are still a surprisingly large number of APs using WEP.

Another factor leading to the current situation is the heavy reliance on clients to make decisions about their use of the wireless spectrum. Even if such decentralized decision-making were optimal (which it is not), differences in vendors' implementations of the standards can lead to interoperability problems and inefficient use of the spectrum. One possible solution is to add more coordination to the client-AP interaction. For example, 802.11e provides support for QoS by allowing certain classes of traffic to be prioritized over others. However, this requires changes to both the clients and APs, slowing innovation. Ideally, we should be able to achieve the same goal without having to upgrade the client logic.

A third confounding factor is the commodity nature of 802.11 hardware, intended to drive down costs for NICs as well as home access points. A laptop must work equally well in a simplified 802.11b installation at home or in a sophisticated enterprise network comprising multiple standards. It is difficult to optimize an 802.11 implementation in such a milieu, requiring more and more complexity to be pushed into the OS drivers.

Our goal is to lift wireless LANs out of this quagmire of standards and develop a simple and extensible network architecture that supports: heterogeneous traffic demands; fine-grained control over network management; customized policies for traffic shaping and prioritization; and rapid innovation. The key idea is to strip most of the complexity away from the client and push it into the infrastructure, which we argue is better suited to managing the wireless spectrum and performing network management.

Some commercial efforts are a small step in this direction, although they are hampered by the need to maintain backwards compatibility with existing 802.11 networks. Cisco, Meru Networks, and others support intelligent radio resource management for enterprise WLANs, collecting measurements on interference, channel utilization, and other metrics to optimize network capacity. Research projects such as DenseAP [12] and MDG [6] have investigated approaches for managing 802.11 resources centrally to increase capacity. Similarly DIRAC [15] explored managing APs centrally. However, these systems are still limited by client-side behavior that may interact poorly with the goal of network-wide optimization. On the other hand, in [13] the author proposes equipping APs with analog-to-digital converters such that they are oblivious to the PHY/MAC layers being used at the client. As a result, all intelligence in the network is pushed to the clients.

Of course, stripping complexity from the clients is not without its challenges and potential pitfalls. One key question is how much complexity *can* be removed from the client without losing control and efficiency. At one extreme, *all* aspects of radio channel management of the client, including the PHY layer, modulation scheme, and the MAC, could be relegated to the infrastructure. However, in this work we decided to base our system on the core pieces of the existing 802.11 standards rather than pushing to this extreme. One reason is that this makes it much easier for us to implement and experiment with our architecture.

Nevertheless, we can experiment with many other aspects of wireless management, including channel assignment, AP associations, power levels, PHY rates, and channel bandwidths,

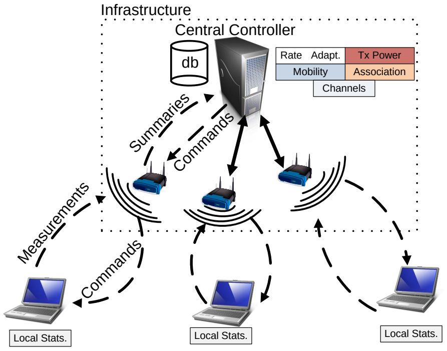

Figure 1: Proposed Trantor architecture. APs send summaries to the Central Controller (CC) about local information as well as responses from clients.

because these knobs can be tuned at coarser time scale. We also retain base 802.11 functionality (associations, authentication, etc.) to support legacy clients and hardware. Hence, Trantor builds on top of an underlying CSMA MAC and base 802.11 functionality.

Related to this issue is a key question: at *what point* should one stop modifying the client? Is it at the application layer, the MAC, the PHY, etc.? One approach is to realize the SDR vision by using an analog-to-digital converter at the client which downloads the entire PHY and MAC layers from the infrastructure. This is an extreme design point as it offers the highest degree of extensibility. However, moving to a complete SDR side steps the issue of what functionality should be implemented at the clients and what responsibility should be given to the infrastructure? For example, should clients continue to decide when to transmit and which modulation schemes to use? At the heart of these issues is a debate on the ideal separation of functionality between the client and the infrastructure, given that we want to be able to enable rapid innovation in the wireless network without sacrificing performance. For the rest of this paper we address this debate in the context of 802.11 only but as part of our ongoing and future work we are also actively exploring how PHY/MAC layers weigh in on these questions.

The second key challenge is determining how much information the client should collect and report to the infrastructure to assist in management decisions. Given complex and dynamic environments exhibiting interference, multipath, and node mobility, the number of variables that affect an individual client's link quality can be extremely large. Ideally, each client could report measurements on the observed channel occupancy, RSSI from multiple APs (and other clients), and variation in channel conditions over time. However, collecting this information could involve high overheads and interact poorly with power-saving measures at the client.

In the following sections, we outline the Trantor architecture and describe our approach to these challenges.

3 Trantor Architecture

In this section we first outline the Trantor architecture and then describe the various management aspects of the system.

In a typical WLAN architecture (such as Aruba [1] or Meru [2]) deployed in the enterprise, the infrastructure consists of APs managed by a central controller (CC). In this context, management entails channel assignment and transmit power control. A controller manages the channel and transmit power for each AP. Clients on the other hand make many decisions independent of the infrastructure. In some small networks, there is no CC and each AP acts independently [3].

The Trantor architecture is illustrated in Figure 1. The main difference between current architectures and Trantor is that various decisions clients currently make have been shifted to the CC. In Trantor, clients receive commands, collect measurements, and report back to the infrastructure. The CC manages all APs and clients in the system. Each AP periodically sends summaries consisting of its own local measurements as well as the local measurements collected by each associated client. Therefore, the CC receives information from every wireless node in the system, be it an AP or a client. The CC does not receive reports from a client not associated with the network.

Based on the summaries received from APs, the CC executes various algorithms and can do one of the following: (i) send a command to an AP to execute a particular action (for example, change channels, disassociate a particular client, etc.), or (ii) send a command to a client (via an AP). Later in this section we elaborate on the commands the CC can send to the client. Note that this design does not require true centralization: a "logical" central controller can be implemented in a decentralized manner across all the APs in the system. The key point however is that most of the intelligence resides in the infrastructure and not with the clients.

This architecture is extensible because policy changes can be easily introduced into the system at the CC. Since clients and APs report summaries to the CC, the infrastructure also has global knowledge of the system. The CC also utilizes a database to store received summaries to main historical knowledge of the system.

Trantor is intended for an enterprise environment where there is one administrative domain that manages all APs. Academia and industry have examined centralizing the data plane and certain decisions such as channel assignments. DenseAP [12] examined centralizing associations. Trantor pushes this further by building an extensible architecture where those decisions that can benefit from global and historical knowledge have been moved from clients to the infrastructure. This approach distinguishes Trantor from prior work. In the future, as wireless networks evolve and potentially newer decision-making aspects of clients are introduced, we envision moving more such decisions to the infrastructure. However, note that decisions such as going into power saving mode do not necessarily benefit from global knowledge and hence are retained at clients. Also, in this paper, we only focus on infrastructure mode in clients and not on ad-hoc mode since the former remains the dominant use of wireless networks in the enterprise.

We now describe the design of clients and the infrastructure in the system.

3.1 Client Design

Clients (and APs) are dumb agents controlled by the infrastructure. Table 2 outlines the various commands the CC sends to APs and clients in the system. Most commands are common to

$ListNodes()$	Report list of $< mac, rssi >$ heard
$ReportLoss(n)$	Report loss-rate when sending to n
$ReportReTrans(n)$	# of retransmissions when sending to n
$ReportAirTime(n)$	Report air-time utilization
$TxPackets(x, s, n)$	send x packets, each of s bytes to n
$Associate(ap)$	Associate with AP ap
$SetTxLevel(p)$	Set transmit power to p
$SetCCA(t)$	Set CCA threshold to t
$SetRate(r)$	Transmit all future packets at rate r
AcceptClient (c)	AP lets client c associate with it
Handoff (ap, c)	Handoff client c to AP ap
EjectClient (c)	Disassociate c from the network

Table 2: **Sample set of commands the CC sends to clients and APs. n is a wireless node. Commands in bold are applicable to APs only.**

both APs and clients. We first focus on how these commands are used by the CC when dealing with clients.

Collecting Measurements: The infrastructure can use the commands listed in Table 2 to estimate packet losses, retransmissions, RSSI of packets from APs, other clients in the vicinity, and channel utilization, all as seen by a client. Our working hypothesis is that such information is fundamental to all macro-level decisions such as associations, handoffs, power control, and rate-adaptation [6, 12]. Clients collect measurements over a measurement window w, which is selected by the infrastructure. w may be changed over time to permit finer- or coarser-grained measurements from each client.

A challenge for the infrastructure is to normalize measurements reported by different clients which may be using different radio chipsets. For example, raw RSSI values reported may vary across clients due to variance in receiver sensitivity.

Active Probing: Using the $TxPacket$ command, the CC can instruct clients to perform active measurements. Active probes can be used to directly ascertain link quality, congestion, and other conditions that can be difficult to derive from passive measurements alone. They can also assist in diagnosing performance issues in the network; we discuss this further in Section 4.

Measurements collection and active probing by clients are unique to the Trantor architecture and represent two fundamental primitives to support extensibility. By combining these mechanisms, the CC can collect detailed measurements of the network state and factors that affect client performance, such as traffic patterns, interference, and channel congestion. Such an approach can potentially reduce the need for a dedicated wireless monitoring infrastructure [8, 5].

Collecting this information from clients achieves three key goals of the Trantor architecture. (i) the infrastructure can optimize the overall performance observed by clients in the network, by tuning many aspects of individual clients' use of the radio channel. (ii) the infrastructure can impose policies to manage competing uses of the radio channel. (iii) Trantor can automatically diagnose and remedy performance problems through centralized observation and control.

We briefly present two example uses of the measurements collected by the infrastructure.

- *Conflict graph construction:* Using information collected from clients, Trantor constructs a conflict graph of the set of clients currently interfering with each other on the same channel, whether or not those clients are currently associated with the same AP [4]. This information can be used to mitigate interference by tuning channel assignments and transmission power control of individual clients.
- *Active AP selection:* Trantor can leverage active probing measurements between clients and APs to optimize client-AP associations. If the loss-rate between a client and its AP rises above a given threshold, rather than relying strictly on client RSSI measurements (as is currently done), the CC initiates active probing between the client and multiple nearby APs to determine the best association.

3.2 Infrastructure Design

In Trantor, the infrastructure bears the additional responsibilities of managing client-AP associations, channel assignment, power control, and rate-adaptation.

There is a mutual interdependence between these various aspects of wireless channel management. For example, managing associations affects the the number of clients on a given channel (since clients are assigned to APs fixed on a single channel) which in turn has the potential to increase interference. Reducing transmit power levels of interfering nodes can mitigate this problem, but it also affects the reception rate for a given data rate (since the probability of successfully decoding a packet for a data rate is determined by a SNR threshold). Performing a joint optimization of these decisions is a non-trivial problem. However, in Trantor since the infrastructure has global and historical knowledge of the performance of the wireless network as well as control over client behavior, it has the potential to address this problem. This is an aspect of the system we are actively exploring. Prior WLAN architecture proposals have lacked such information and hence it has been harder for them to address this problem. We briefly describe possible techniques to address these management decisions.

Client-AP Associations: Using $Associate(ap)$, the CC has the ability to control which AP a client can associate with. It can also use **AcceptClient** and **RejectClient** to prevent a client from associating with an AP. As prior work [6, 12] has shown, client-AP association decisions must take into account load at the AP as well as the quality of the client-AP connection. As mentioned earlier, Trantor can leverage active probing measurements for this purpose. Furthermore, historical information can also help improve association decisions. For example, prior work [7] has observed clients in certain locations in an office building in spite of receiving strong signals from an AP, experience heavy packet losses due to a poor wireless channel. Such information can be used to quickly converge on a client-AP association decision.

Transmit Power Control: Prior work [6] has shown how coordinated power control can lead to an increase in overall network capacity. We adopt a similar approach. The CC also has control over each client and AP's CCA threshold since it is required to set the appropriate power level at these nodes.

Rate Adaptation: Prior work on rate-adaptation has focused on clients adjusting rates based on local information such as packet loss or RSSI of received packets. Packet losses at a client commonly occurs due to one of the following reasons: (i) collisions caused by hidden terminals, (ii) local channel noise. The remedy to (i) is to increase or fix the current data rate. The remedy to (ii) is to lower the current rate in order to improve the SNR of the signal. Hence, it is important to distinguish between the two cases when determining the next course of action for rate adaptation. Most prior work in this space suffer from the lack of additional information that can help distinguish between these two cases.

Prior work has shown that some cooperation between clients and the infrastructure can help a client pick better rates [10]. In Trantor, the availability of global and historical knowledge can facilitate rate adaptation further. We argue data rates must be adjusted based on a longer term view of the network rather than just the recent few packets. Hence, based on reports from nearby APs and clients (global knowledge) and observing the behavior of the network over long periods of time (historical knowledge), the CC can potentially ascertain the reason behind significant packet losses in the network [8]. Based on the measurements received from APs and clients, the CC constructs a conflict graph and uses a probabilistic analysis to determine if an AP or client is experiencing loss due to a hidden terminal like problem or due to channel noise. Using this analysis it instructs each node precisely which data-rate to transmit at. A node continues to transmit at the same rate until its told to change its transmission rate by the CC.

There is a tradeoff between using global knowledge for centralized rate adaptation and the timescale over which this can be performed for each wireless node. A slow rate adaptation can result in an AP or client temporarily experiencing poor performance. Our working hypothesis is that nodes do need to change data rates but *not* as often as prior work has come to expect. In other words, we do not expect the wireless medium to be choppy on a sustained basis and therefore we prefer choosing a "correct" rate slowly than an "incorrect" rate quickly.

Mobility: Since the infrastructure handles associations it is must also handle mobility. While a client can be instructed to explicitly associate with a different AP (hence the actual cost of the handoff is negligible), delays might be incurred by the infrastructure in gathering measurements, analyzing them, and determining if a handoff should take place. This is where historical knowledge of the wireless network is key for improving performance. In an office deployment, clients typically move along corridors or hallways. The infrastructure can observe such patterns and predict the trajectory that a client will take. This can help reduce the time taken to determine when a client must switch APs. To enable handoffs, the CC uses $Associate$ to inform the client to switch associations and **Handoff** to inform the source AP to send the association state and buffered packets to the destination AP.

Classifying Clients: The ability to offer differentiated services to clients is a key gain the Trantor architecture has to offer. To achieve this the system must be able to quickly classify clients based on their traffic. Such classification is important because it impacts the association and handoff decisions. For example, VOIP clients tend to suffer when contending with bulk transfer clients for the same part of the spectrum. Therefore, in Trantor, we can associate clients to different APs on different channels based on their traffic classification. This entails clustering VOIP clients together when performing associations or handoffs. Furthermore, it is also important the system classifies a client quickly since this can impact the handoff latency. We address the classification problem using a lightweight technique whereby each AP monitors the flow of packets to/from a client. Using a technique similar to one proposed in [11], observing a few samples of packet size, port number, and inter-packet arrival time, the AP classifies the client's traffic as (i) latency sensitive or (ii) a bulk transfer. However, there can be cases when a client simultaneously starts a Skype call (VOIP) and also begins a file download. We currently classify such clients as being bulk transfer agents.

4 Benefits of Trantor

In this section we discuss several tangible benefits that the Trantor architecture provides.

Traffic differentiation: Trantor permits the network administrator to impose local policies on the network to prioritize certain clients over others based on their traffic. For example, hospital environments may want to prioritize data from hospital instruments over standard WiFi usage, and companies may want to limit large media downloads during the day. This could entail various approaches such as grouping client associations based on their traffic type or rate limiting certain clients more than others.

Site-specific policies: A direct consequence of extensibility is the ability to customize the behavior and performance of the wireless network based on the context. Prior work has shown wireless traffic patterns fluctuate by time of day as well as location [14]. For example, a large auditorium or conference room might experience heavy spikes of traffic congestion during meetings, whereas dormitories may experience heavier loads at night. To deal with such situations, the infrastructure needs to dynamically provision the spectrum based on the client traffic mix. We present two example policies and briefly describe how they impact the various decisions made by the infrastructure.

- *All APs on the corridor in the 2nd floor must prioritize VoIP traffic over other kinds of traffic between 9 a.m. and 4 p.m..* Because VoIP clients are sensitive to packet losses and jitter, we want them to be able to use lower data rates and at the same time not have to contend for the medium with other clients performing bulk transfers. Hence, the infrastructure would impose an association policy that prioritizes associations and handoffs from VoIP clients, assigns higher data rates and power levels to VoIP clients, and hands off mobile clients more aggressively.
- *Based on the number of web clients on the 1st floor, the first floor wireless network must devote X% of APs to interactive traffic.* In this case the infrastructure would enforce an association policy that only permits interactive traffic clients to

use certain APs. This policy could also entail more aggressive rate-adaptation.

Fault Diagnosis: Based on client measurements reports, the infrastructure can detect, resolve, or at least shed more light on performance anomalies and outages in the network. Two typical examples of such diagnosis are as follows.

- *Rate Anomaly*: The rate anomaly problem arises due to the "worst client" impacting the performance of other wireless clients in the vicinity, and prior work has shown this can significantly reduce WLAN capacity [9]. In Trantor, because the infrastructure controls the transmission rate for each client, it can now detect such rate imbalance situations and either increase the data-rate for the offending client or change its association to a different AP.
- *Reasoning about client losses*: Using global and historical knowledge, the infrastructure can attempt to ascertain why a client experiences significant losses. This impacts rate adaptation as well as transmit power (at the APs). However, persistent losses (despite rate and power changes) could be used to diagnose whether a particular area of the building suffers from poor channel conditions.

5 Discussion and Conclusions

We present Trantor, an extensible architecture for WLANs. The fundamental tenet of the Trantor architecture is to move wireless management decisions from clients into the infrastructure when such decisions can benefit from global and/or historical knowledge. As part of this we proposed centralizing various wireless management aspects. Trantor is also able to provide better customization of a WLAN according to an environment. We now outline some key challenges we plan to investigate related to the Trantor architecture.

Dealing with malicious clients: It is relatively easy to detect violations where a client does not follow the infrastructure's instructions. For example, based on reports from APs and other clients, it is easy to detect such violations and disassociate the offending client from the network. However, it is a much harder problem to determine if a malicious (or faulty) client is sending spurious reports. One potential way to address this problem is to verify such reports with reports from other APs and clients in the neighborhood, but this remains an open issue.

Scalability: The scalability of the Trantor infrastructure depends on a host of factors including: the rate at which each client is polled for measurements; the size of the measurements reports; and the ability of the CC to make quick decisions during handoffs and change rates quickly when necessary. To be effective in enterprise settings, the central controller must be designed to handle a large network consisting of thousands of APs and clients. One strategy to scale gracefully is to use a zoned approach in which separate controllers are assigned to distinct physical zones in the network (such as different buildings, or floors of a building), with the assumption that limited sharing is required across zone controllers to make effective network management decisions.

Presence of other interfering networks: An argument for clients retaining their decision making abilities is for them to react to interference from other competing wireless networks in the vicinity. However, since clients are always reporting measurements to the infrastructure, such events can be dealt with effectively in our proposed infrastructure as well. The infrastructure can profile which other competing networks are operating in the vicinity and use this information when determining policies for clients.

Security: Trantor's extensibility can make it easier to deploy new security mechanisms. This is critical to the operation of a wireless network because in the event a security mechanism is found to be flawed (WEP, for example), without depending on new standards to be adopted, it is easy to implement and push out new security mechanisms quickly in Trantor. For example, WPA2 was proposed as a replacement for WEP via 802.11i and it did not require hardware changes. Such updates can be easily rolled in Trantor but it would require expanding the interface listed in Table 2.

Responsiveness: One open question is whether the clients need to adapt their behavior more rapidly than can be easily accommodated by the Trantor architecture, with its cycle of collecting data, analyzing it centrally, and then sending out commands to cause the clients to adapt.

References

[1] Enterprise solutions from aruba networks, http://www.arubanetworks.com/solutions/enterprise.php.

[2] Meru networks, http://www.merunetworks.com.

[3] Xirrus, http://www.xirrus.com/products/.

[4] N. Ahmed and S. Keshav. SMARTA: A Self-Managing Architecture for Thin Access Points. In *CoNEXT*, 2006.

[5] P. Bahl, R. Chandra, J. Padhye, L. Ravindranath, M. Singh, A. Wolman, and B. Zill. Enhancing the Security of Corporate Wi-Fi Networks Using DAIR. In *MobiSys*, 2006.

[6] I. Broustis, K. Papagiannaki, S. V. Krishnamurthy, M. Faloutsos, and V. Mhatre. MDG: Measurement-driven Guidelines for 802.11 WLAN Design. In *MobiCom*, 2007.

[7] R. Chandra, J. Padhye, A. Wolman, and B. Zill. A Location-based Management System for Enterprise Wireless LANs. In *NSDI*, 2007.

[8] Y.-C. Cheng, M. Afanasyev, P. Verkaik, P. Benko, J. Chiang, A. C. Snoeren, G. M. Voelker, and S. Savage. Automated Cross-Layer Diagnosis of Enterprise Wireless Networks. In *SIGCOMM*, 2007.

[9] M. Heusse, F. Rousseau, G. Berger-Sabbatel, and A. Duda. Performance anomaly of 802.11b. In *Infocom*, 2003.

[10] G. Judd, X. Wang, and P. Steenkiste. Efficient Channel-aware Rate Adaptation in Dynamic Environments. In *MobiSys*, Uppsala, Sweden, 2008.

[11] A. W. Moore and D. Zuev. Internet traffic classification using bayesian analysis techniques. In *SIGMETRICS*, 2005.

[12] R. Murty, J. Padhye, R. Chandra, A. Wolman, and B. Zill. Designing High-Performance Enterprise Wireless Networks. In *NSDI*, San Francisco, CA, April 2008.

[13] S. Singh. Challenges: Wide-Area wireless NETworks (WANETs). In *MOBICOM*, 2008.

[14] D. Tang and M. Baker. Analysis of a Local-Area Wireless Network. In *MobiCom*, Boston, MA, August 2000.

[15] P. Zerfos, G. Zhong, J. Cheng, H. Luo, S. Lu, and J. J.-R. L. DIRAC: a software-based wireless router system. In *MOBICOM*, 2003.