IBM Research Report

On Quantifying Multiplexing Gains in a Wireless Network Cloud

Mukundan Madhavan
IBM Research India
Bangalore - 560 045. India.
Email: *mukmadha@in.ibm.com*

Parul Gupta
IBM Research India
Bangalore - 560 045, India
Email: *parulgupta@in.ibm.com*

Malolan Chetlur
IBM Research India
Bangalore - 560 045. India.
Email: *mchetlur@in.ibm.com*

IBM Research Division
Almaden - Austin - Beijing - Delhi - Haifa - T.J. Watson - Tokyo - Zurich

Abstract

The Wireless Network Cloud (WNC) [1] is a novel network architecture where wireless base-stations are implemented as software modules and multiple base-stations are consolidated to a single centralized computing platform. Due to the time-varying and random nature of base-station traffic, consolidation leads to multiplexing of statistically-varying basestation loads on a common hardware platform. In turn, this can lead to significant hardware reduction in the consolidated platform as compared to the distributed network. This paper represents the first analysis of this consolidation gain. Through traffic simulation experiments, we quantify the extent and variation of this multiplexing gain in a WiMAX base- station network in different traffic conditions. We show experimentally, that the obtained gain increases linearly with network size (number of base-stations). Further, we also show that the consolidation gain is higher when the consolidated base-stations face higher traffic intensity.

1 Introduction

Today's wireless networks have a distributed base-station architecture, where the base-band processing for each base-station is performed on a dedicated proprietary platform at the antenna site (See Figure 1a). The Wireless Network Cloud (WNC) [1] is an alternate system architecture where base-stations are implemented as software stacks (also called Software Defined Radio) and consolidated to a common central location. This central processing platform, called the wireless network cloud, is connected to Remote Radio Heads (RRH) in different cells through high-speed links (See Figure 1c). Apart from reducing the operational costs of the network and facilitating physical layer co-operative communication techniques, consolidation is expected to provide significant reduction in the already large and increasing infrastructural costs. In this paper, we will analyse the causes and the extent of these infrastructural gains (also called multiplexing gains) for an intermediate cloud architecture that is relevant in the immediate future. In the intermediate architecture, the base-station MAC layers are consolidated in a cloud and the physical-layer processing is retained close to the cell antenna-sites (Figure 1b). This architecture has the advantage of being more relevant in the immediate future when technical challenges still exist in consolidating physical layer processing. On the other hand, this intermediate architecture also multiplexes random loads on to one platform, thus lending itself to the study of multiplexing gains.

Briefly, the rationale behind the expected infrastructural gain is as follows. User traffic is bursty in nature and is usually described using random processes. Since the base-station load is an aggregate of many such independent user flows, it is also random. Consequently, different base-station loads do not peak simultaneously. Therefore, when base-station loads are multiplexed, the peak of the cumulative load is much lesser than the sum of the individual peak loads. When carrying out peak hardware provisioning, this reduced peak load in the cloud results in lower computational needs. Peak-provisioning is assumed for base-stations due to the strict real-time constraints on their processing deadlines that must be met at least 99.99% of the times. In this paper, we determine the extent of multiplexing gains in the considered architecture and characterize different properties of this gain through experiments on an already implemented WiMAX base-station system. By conducting traffic experiments on these base-station MACs using VoIP traffic, we show that the cloud CPU requirement can be as small as half the requirement in a distributed network.

Towards this goal, we first characterize the variation of the CPU requirements in both the cloud and standalone networks separately for different network sizes and show that the hardware requirement in a cloud grows at a slower rate than in a distributed network. We also calculate the multiplexing gain achieved through consolidation and show that the gain is significant and also that it increases with the size of the network. Further, we study the multiplexing gain as the incoming traffic per base-station increases significantly, and show that higher gains can be obtained at higher traffic levels. Using the above experiments, we also demonstrate that the network hardware requirements in a cloud increases much more slowly when compared to a conventional distributed network for the same amount of increase in traffic, implying fewer hardware upgrades over time. This is especially significant in the light of the expected explosion in the user-base of wireless services [2]. Finally, we analyse the contribution of individual factors like traffic burstiness and variation in intra-day base-station load patterns to this gain, and use this analysis to derive initial insights about more generic settings. In summary, the contribution of this paper is that we analyse and quantify the multiplexing gains in a network cloud. Also, we use this analysic to show that a network cloud architecture is more robust and scalable in the face of rapid traffic increase and network growth.

This paper is organized as follows. In Section 2, we briefly describe the WNC network architecture the concept of multiplexing gain as used in this paper. We also give a brief description of the

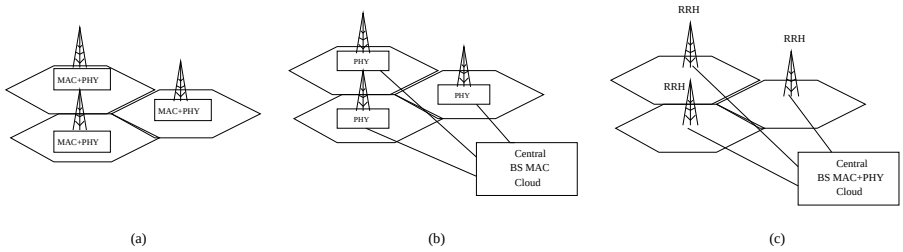


Figure 1: Access Network Architectures: (a) Distributed Architecture (b) Intermediate WNC Architecture (c) Complete WNC Architecture

WiMAX standard's MAC layer. In Section 3, we explain the experimental setup used to estimate the multiplexing gains, including some of the important characteristics of the MAC layer and the experimental methodology used to estimate the multiplexing gains. In Section 4, we provide the multiplexing gain results obtained from the experiments and identify the contribution of individual factors such as traffic burstiness and base-station traffic patterns to this gain. Finally, Section 5 concludes the paper.

2 Brief Introduction to Related Background

2.1 The WNC Architecture

Figure 1 shows the existing wireless network architecture as well as the WNC architecture proposed in [1]. Every wireless network is split into smaller coverage cells, each covered by a base-station. The base-station processing, which comprises of MAC and physical layer processing, is located at the site of the transmission antenna of each cell in a traditional network.

The main difference in the WNC architecture is that the MAC and physical-layer processing of all base-stations are consolidated to a centrally-located processing platform instead of being in their respective coverage cells. To pass transmission data between the remote antennas (RRH) and this central cloud, they are connected through high-speed links like optical fiber cables (OFC). In this paper, we consider an intermediate architecture between the distributed and WNC architectures for our experiments. In our system, the base-station MAC layers are consolidated on a cloud, whereas the physical layers are retained at the antenna site. This architecture is a step towards a full-fledged WNC and is better realisable in the near future, since consolidation of physical layer modules faces practical issues such as synchronization between cloud and RRHs as well as the lack of ubiquitous optical fiber infrastructure.

Multiple factors affect the costs and benefits of the wireless network cloud. We briefly list a few main factors below.

- A significant part of a base-station's cost is caused by non-transmission related costs like air-conditioning, security and real-estate. Consolidation can lead to significant savings here.
- The cost of high-speed links to be laid between the central cloud and the RRHs.
- The WNC architecture makes physical layer co-operative communication techniques more feasible. These techniques need large volumes of data to be shared between base-stations at high-speeds, and are less practical in the current network architectures. Though this does not directly affect the cost of the system, it is a significant benefit that can improve the spectrum utilization [3].
- Finally, as we shall see, the statistical variations in base-station loads can be exploited to reduce the network hardware required to support base-stations in a cloud.

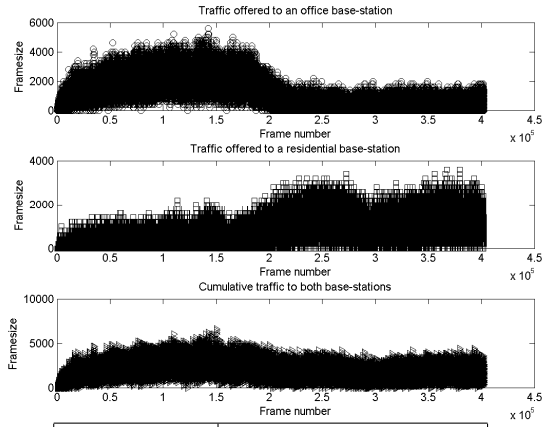
In this paper, we focus on analysing the infrastructural gains that can be achieved from the cloud architecture.

2.2 Multiplexing Gains due to Consolidation

User traffic is typically variable in nature and is usually modelled as a random process [4, 5]. Hence, the total load offered to a base-station is a combination of many independent user flows and is also

random. This means that all base-stations in a network are unlikely to be processing their peak loads simultaneously. When provisioning for a distributed network, each platform has to support the peak load of one base-station instance. On the other hand, a cloud platform has to support many base-station instances, and provisioning must be done for the peak of the sum loads of all these instances. Since different base-station loads are unlikely to peak simultaneously, the peak of this sum load is likely to be lesser than the sum of the individual peak loads. Therefore, lesser hardware can support the same network on a cloud than in a distributed network.

We will now illustrate this with the help of Figure 2. Consider two base-stations A and B, with A serving an office-area and B serving a residential locality. Typically, offices function during the day and one would expect A to face higher traffic during the day than at night. Similarly, one would expect B to face lesser traffic in the day than at night. Figure 2 shows the aggregate traffic faced by A and B from all its users over every $5ms$ interval (This is a WiMAX frame's duration, which forms constraints on processing latency) when VoIP traffic is generated to these base-stations according to the above assumptions. Briefly, the initial half of the experiment simulates day-time traffic to both base-station, followed by a transition into night-time traffic for the remaining experiment duration. This setting is used so as to simulate the traffic pattern over an entire 24-hour period. The table



Base-station	Maximum Bytes input to base-station
Office-area	5600
Residential	3600
Sum load	6600 (= 4600 + 2000)

Figure 2: Example Illustrating Multiplexing Gains

adjoining Figure 2 shows the peak traffic offered to each individual base-station over all frame-durations as well as the peak of the cumulative load of the two base-stations. This cumulative load is obtained by summing the time-plots of the individual loads of the two base-stations. We note that standalone implementations of each base-stations must be provisioned to meet its respective individual peak, whereas the cloud should be provisioned to meet the processing deadlines of both the base-stations simultaneously at the peak of their combined load. From the table, we can see that each individual base-station's contribution to the cumulative peak load is lower than its respective peak in the standalone case. So, in a cloud, it is sufficient to provision such that the two base-station instances are able to simultaneously process these reduced component loads within their frame deadlines, rather than provision for their individual peaks.

We will show in Section 3.1 that the WiMAX processing latency increases monotonically with the input byte-size offered to it. This leads us to conclude that since the traffic components of the cloud's peak load are lower, so is the hardware requirement for a cloud. Two factors contribute to this lower cumulative traffic. They are: **(i)** variation in the user arrival-departure patterns of different base-stations, seen over a larger time-scale (like a day), and **(ii)** traffic burstiness, which affects the load at more fine-grained time-scales. Depending on the traffic type, the variability of the packet enqueuing times can be low (VoIP traffic) or high (video traffic). In this paper, we assume a VoIP traffic model while investigating multiplexing gains. However, we note that since VoIP has less variability in its flow compared to other traffic types such as video, the gains obtained here is likely to be lesser than in a mix of, say, video and voice traffic. Therefore, we expect our results to indicate a lower bound on the achievable multiplexing gains.

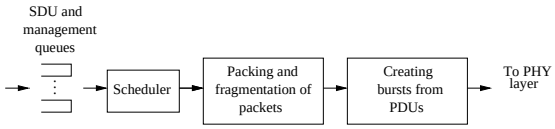


Figure 3: Simplified Diagram of the WiMAX MAC Layer Downlink

2.3 WiMAX MAC Layer

WiMAX, defined in the IEEE 802.16e [6] standard, is a wireless broadband telecommunication system. It is an OFDM [6]-based system where transmission occurs in the form of OFDM symbols. Each OFDM symbol spans a certain pre-defined duration in time, and an allocated frequency band. A WiMAX "frame" is a time-frequency matrix used to represent a set of consecutive OFDM symbols. Each cell in the matrix, referred to as a slot, represents a group of frequency sub-carriers and OFDM symbols. The WiMAX frame has a fixed duration of the order of milliseconds (typically $5ms$), and hence, a fixed number of OFDM symbols per frame. Figure 3 shows a simplified view of the processing modules in the WiMAX MAC layer for downlink transmission. The MAC layer maintains a packet queue for each associated connection, and packets arriving from the higher layers are enqueued into these queues. The MAC layer performs the following functions periodically for each downlink frame.

- **Scheduling** : The MAC layer allocates frame slots to different connections based on constraints such as QoS of each connection, queue backlogs and wireless channel conditions.
- **Packing and Fragmentation** : The packets in the input queues can be either fragmented into smaller components or multiple packets can be packed into a bigger unit, depending on factors like channel conditions and connection characteristics.
- **Burst Creation** : After packing and fragmentation, the resultant Protocol Date Units (PDUs) are concatenated to form "bursts". A burst is a single continuous logical region in the frame which has a constant modulation and coding scheme applied to it.
- The MAC layer then passes these bursts and some management information to the physical layer for transmission.

Importantly, the processing for each frame must be completed within the frame-duration. The frame-duration forms a hard deadline which must be met at least 99.99% of the times. Therefore, hardware provisioning for the base-station must be such that the peak processing latency is smaller than the frame deadline. In the next section, we will discuss the MAC layer processing latency characteristics and its implications on provisioning in greater detail.

3 Experimental Setup for Measuring Multiplexing Gain

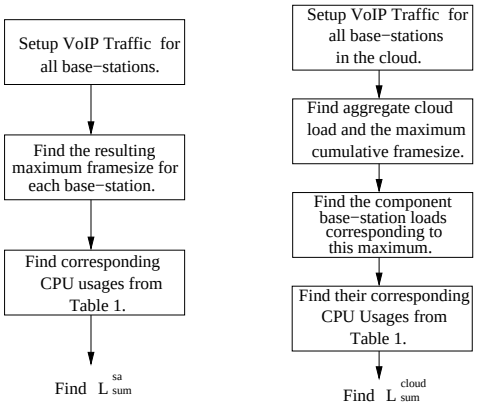


Figure 4: Experimental Methodology

As explained earlier, the frame duration forms a hard deadline for the base-station's MAC processing latency. Consequently, hardware provisioning for the network should be such that the peak processing latency is less than this deadline. Keeping this and the CPU-intensive nature of the

Cumulative Input Size	Frame-processing Latency (μs)	Hardware Needs (CPU Cycles/frame)	Hardware needs (CPU Cycles per second)
0	20	6×10^4	1.2×10^7
600	40	1.2×10^5	2.4×10^7
1200	60	1.8×10^5	3.6×10^7
2400	80	2.4×10^5	4.8×10^7
3000	100	3×10^5	6×10^7
5000	160	4.8×10^5	9.6×10^7
6000	180	5.4×10^5	10.8×10^7
7000	220	6.6×10^5	13.2×10^7
8000	240	7.2×10^5	14.4×10^7
9000	260	7.8×10^5	15.6×10^7
10000	280	8.4×10^5	16.8×10^7

Table 1: Variation of MAC Stack's CPU utilization with input size

MAC layer in mind, we express hardware provisioning for a given platform in terms of the processor frequency required in a single processor core to support the MAC instance. We estimate C, the processor occupancy of the MAC instance in cycles per second (CPS), at peak load. The quantity C is indicative of the extra processing power, in addition to the system processing requirements, that must be provisioned in the platform to support the MAC. The general idea in the experimental setup is to study the effect of multiplexing of base-stations on the WiMAX frame-sizes and to relate the reduction in CPU requirements due to the reduced frame-sizes. thus indicating gain. Though it might be possible to have more complex metrics for characterizing hardware requirements other than CPS, we have chosen this metric for its simplicity.

3.1 Latency Characteristics of the MAC Layer

In this section, we will measure the WiMAX MAC layer's processing requirements as a function of its aggregate input packet size. The measurement was conducted using a modified single-thread version of the MAC described in [7] on an Intel Xeon 3GHz processor. Although the CPS values provided below are dependent on the Intel x86 processor architecture, we feel that this approach is reasonable given the popularity of Intel architectures.

We first measure the MAC layer's per-frame latency as a function of the input size. Then, we convert this latency to the corresponding CPU CPS value using the CPU frequency. Due to lack of measurement tools at the time-scale of frame-processing (a few hundred microseconds), we use an averaging approach to measure the per-frame latencies. We input a constant input byte size to the MAC layer over a period of 5 seconds, and divide the cumulative latency in this window by the number of elapsed frames (1000). Measurements were made using the Linux tool "top". We repeat this procedure for different input sizes.

Table 1 shows the measured per-frame processing latencies and the corresponding number of cycles consumed per-frame, which is the product of the processing latency and the CPU frequency. Column 4 translates the CPU cycles per-frame to an equivalent CPS measure by scaling them by the frame duration. As we can see, the processing latency of the MAC layer is an increasing function of the frame-size of the WiMAX MAC assuming that packets are processed upon arrival. Since the peak frame-size of the MAC corresponds to the peak latency, a reduction in the cumulative frame-size as seen in the table of Figure 2 implies a reduction in the CPU requirement for the base-station system.

3.2 Multiplexing Gain Experiments on the WiMAX Stack

Let us consider a set of N base-stations, half of which are located in an office-area and the rest serve a residential zone. As explained before, the office and residential base-stations have significantly different traffic arrival patterns. To estimate the multiplexing gain that can be derived by pooling these base-stations in a cloud, we simulate traffic sessions in these base-stations as described below.

User traffic to each base-station is assumed to follow a Poisson arrival-departure process [8]. Let T_I denote the mean Poisson inter-arrival time between successive users to a base-station and T_S denote the mean session-duration. By varying the value of T_I appropriately, it is possible to simulate the day-night traffic variations. Consider two different values for the office base-station's Poisson inter-arrival time T_I during the day (T_O^D) and at night (T_O^N) such that $T_O^D < T_O^N$. For the office base-station, by fixing $T_I = T_O^D$ during the day and $T_I = T_O^N$ at night, we can simulate higher traffic levels at day and lower traffic levels at night. Similarly, to simulate a residential station's load, one can use $T_I = T_R^D$ and $T_I = T_R^N$ during the day and night respectively with $T_R^D > T_R^N$. The average

number of users associated with a base-statoin for a given value of T_I can be found as [9]

$$U = \frac{T_S}{T_I}. \tag{1}$$

By varying the set $(T_O^D, T_O^N, T_R^D, T_R^N)$ suitably, different traffic settings can be simulated to this office-residential base-station pool. We assume a mean session duration of $T_S = 120$ seconds in our experiments.

During each session, users enqueue VoIP packets to the base-station queue. VoIP traffic is described by an ON-OFF traffic model [5]. During an ON-period, the user enqueues packets of constant size at uniform intervals. No packets are enqueued during the OFF-time. The durations of ON-OFF periods are exponentially distributed with means of 350 ms and 650 ms respectively [10]. Assuming a G.711 codec for the voice source, the constant inter-packet delay in the ON-time is $20ms$ and each packet is 200 bytes large. We point out that Figure 2 was generated using this setup using parameter values $T_O^D = 1, T_O^N = 5, T_R^D = 5, T_R^N = 2$ (seconds). From (1), this corresponds to an average number of users of $(U_O^D, U_O^N, U_R^D, U_R^N) = (120, 24, 24, 60)$.

T_S	Mean session-duration of each user.
T_I	Generic variable used to denote the mean Poisson inter-arrival time.
T_O^D	Value of T_I in an office base-station during day time.
T_O^N	Value of T_I in an office base-station at night.
T_R^D	Value of T_I in a residential base-station during the day.
T_R^N	Value of T_I in a residential base-station at night.
U	The average number of users associated to a base-station (See Eq. (1)).
U_O^D	Value of U in an office base-station during day time.
U_O^N	Value of U in an office base-station at night.
U_R^D	Value of U in a residential base-station during the day.
U_R^N	Value of U in a residential base-station at night.

Table 2: Summary of Important Notations

3.2.1 Measuring CPU Requirements in a Distributed Architecture

: Let n_{max} denote the number of WiMAX frames spanned by each traffic experiment involving the N base-stations. We denote the size of the n^{th} frame in the i^{th} base-station as $B_i(n)$. Further, let $n_{i,sa}^*$ denote the frame with peak processing latency among all frames processed by the i^{th} base-station. Then, since latency increases monotonically with frame-size, we have

$$n_{i,sa}^* = \arg\max_n B_i(n); 1 \leq n \leq n_{max} \tag{2}$$

Using Table 1, we can find the CPU CPS corresponding to $B_i(n_{i,sa}^*)$. This is the peak processing requirement of the i^{th} base-station in the distributed setting. Let this CPS value be denoted as L_i^{sa}. We define

$$L_{sum}^{sa} = \sum_{i=1}^{N} L_i^{sa}. \tag{3}$$

The measure L_{sum}^{sa} indicates the cumulative CPU peak CPS requirement of all the standalone MAC processing systems in the distributed setting.

3.2.2 Measuring the Cumulative CPU Cycles spent in a Cloud

: Since all the base-stations are multiplexed on a single computing platform in the cloud, this platform must be provisioned to meet the peak of the sum load instead of the individual peaks. To find the peak CPS utilization in the cloud, we first find the sum

$$B_{sum}(n) = \sum_{i=1}^{N} B_i(n); \forall n \in \{1, ..., n_{max}\}. \tag{4}$$

Next, we find the frame with maximum cumulative load. This is defined as

$$n_{cloud}^* = \arg\max_n B_{sum}(n); 1 \leq n \leq n_{max}. \tag{5}$$

For the frame n^*_{cloud}, we find the corresponding individual base-station loads $B_i(n^*_{cloud})$, $\forall i \in \{1, ..., N\}$. These individual base-station loads must simultaneously be provisioned so that all of them finishes processing within their individual frame deadlines. Therefore, the CPS requirement of the cloud platform is the sum of the individual CPS requirements corresponding to each $B_i(n^*_{cloud})$. So we first find L_i^{cloud}, the corresponding CPS requirements to the frame-sizes $B_i(n^*_{cloud})$ using Table 1. Then, the sum

$$L_{sum}^{cloud} = \sum_{i=1}^{N} L_i^{cloud}. \tag{6}$$

gives the CPU CPS requirement for the consolidated platform. We define the cloud multiplexing gain due to consolidation to be

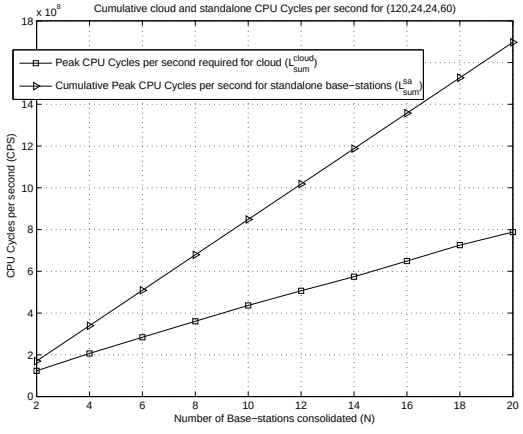
$$G = L_{sum}^{sa} - L_{sum}^{cloud} \tag{7}$$

The gain G gives an indication of the cumulative savings in peak CPU CPS obtained over all base-stations as a result of consolidation. In the further sections, we will quantify G for different traffic settings and discuss its implications.

4 Experimental Results

4.1 Extent of Multiplexing Gains in a Cloud Network

Figure 5 shows the average values of L_{sum}^{sa} and L_{sum}^{cloud} for an increasing N in the traffic setting $(U_O^D, U_O^N, U_R^D, U_R^N) = (120, 24, 24, 60)$, averaged over 300 runs of traffic experiments. Also shown is a linear equation corresponding to each curve, obtained using basic curve-fitting. These linear equations are a close fit on the plotted curves, with negligible error residues (error between the fit and the observed value is less than 1% of the observed value at any point in the curve).



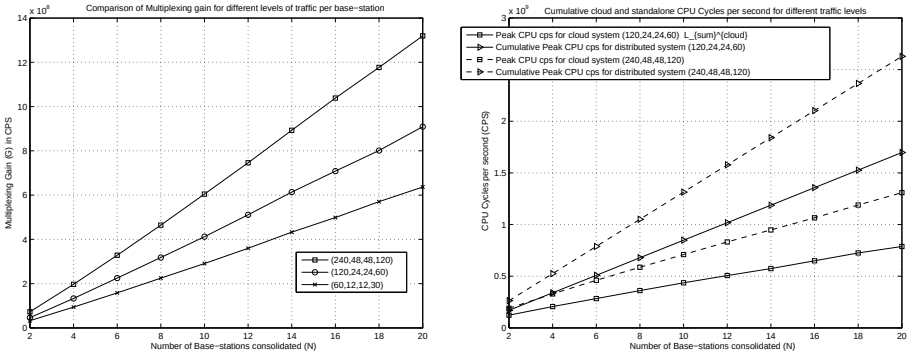
$$L_{sum}^{sa}(N) = 3.682 \times 10^7 N + 6.047 \times 10^7$$
$$L_{sum}^{cloud}(N) = 8.484 \times 10^7 N + 6.865 \times 10^4$$
$$G(N) = 4.802 \times 10^7 N + 6.04 \times 10^7$$

Figure 5: Cloud and standalone utilizations for different network sizes

As expected, L_{sum}^{sa} increases almost linearly with the network size. Interestingly, the cloud utilization L_{sum}^{cloud} also increases linearly, but at a much slower rate. Consequently, the multiplexing gain G also increases linearly with the network size. This resultant gain is shown in Figure 6(a) (middle curve). From this curve, we see that consolidating network base-stations to a cloud always reduces the computational needs of the network, and that this reduction, which is the multiplexing gain, increases linearly with the size of the network. The main implication of this result is that as networks grow in size, the cloud architecture requires a more gradual increase in hardware provisioning than a distributed architecture. This prevents the need for frequent upgrades to the network infrastructure, making the cloud architecture much more preferable for networks in growing markets.

4.1.1 Insights into Cloud Architecture:

The multiplexing gain experiments also provide insight on the optimal size for the cloud. If we consider the $(120, 24, 24, 60)$ plot in Figure 6(a) for $N = 6$ and $N = 12$, we see that the gain at $N = 6$ is 2.25×10^8 CPS, whereas the gain for $N = 12$ is 5.11×10^8 CPS. This value of the multiplexing gain G at $N = 12$ is more than twice the gain at $N = 6$. This shows that by consolidating double



(a) Multiplexing gains for different levels of traffic

(b) Cloud and standalone utilizations for different levels of traffic

Figure 6:

the number of base-stations, the gain G more than doubles, leading us to conclude that a larger cloud gives better infrastructure gain. However, the trade-off with larger clouds is the larger cost of laying high-speed links to the cloud location. Hence the optimum size of the cloud can be obtained as one giving the best combination of the two costs.

4.2 Robustness of Architectures to Traffic Growth

We now look at the trends in multiplexing gain as the traffic level in each base-station increases. Figure 6(a) also shows multiplexing gain curves when the parameter-set $(U_O^D, U_O^N, U_R^D, U_O^N)$ changes by an order of magnitude from (120,24,24,60) to (240,48,48,120) and (60,12,12,30). We observe that the multiplexing gains increase significantly with an increase in the traffic level per base-station. To further understand this trend, let us compare the values of L_{sum}^{sa} and L_{sum}^{cloud} corresponding to the $(120, 24, 24, 60)$ and $(240, 48, 48, 120)$ traffic settings, as shown in Figure 6(b). As expected, L_{sum}^{sa} increases as the traffic grows because each base-station has to support larger frame-sizes now. However, it is interesting to see that the cloud utilization L_{sum}^{cloud} increases at a much slower rate than L_{sum}^{sa}, and this is reflected as an increased multiplexing gain. Thus, when network traffic increases, statistical multiplexing causes the network hardware requirement to increase more gradually in a cloud than in a distributed network. This reduced requirement for hardware upgradation makes the cloud much more preferable in the future, when wireless traffic is expected to increase rapidly.

4.3 Contribution of Individual Sources to Multiplexing Gain

In this section, we will isolate the contributions to multiplexing gain from each of the identified causes - traffic burstiness and difference in base-station traffic patterns. Towards this, we modify our experiment setup slightly. Instead of consolidating office and residential base-stations together, we now consolidate only office base-stations and measure the resultant computational needs and multiplexing gain. Since base-station variety is now absent in this setup, the obtained multiplexing gain is contributed only by traffic burstiness. Therefore, by finding the difference between the multiplexing gain in Figure 6(a) and this gain, we can isolate the gain due to differences in diurnal traffic patterns of the office and residential base-stations. Figure 7 shows the gain curve obtained by consolidating only the office-base-stations with $U_O^D = 120$ and $U_O^N = 24$ along with the complete $(120, 24, 24, 60)$ multiplexing gain curve, reproduced from Figure 6(a). Surprisingly, we notice that these curves closely follow each other, implying that the contribution due to diurnal traffic variations is negligible. This is because of the very small difference in the residential base-station's day and night traffic levels. On the other hand, when the office base-stations with the same traffic loads are consolidated with residential base-stations having $U_R^D = 24$ and $U_R^N = 120$, the increased diurnal variation in the residential station's traffic complements the office base-stations resulting in additional gain. From the above experiment, we see that a careful choice of base-stations for consolidation can lead to increased gain. Another important observation here is that traffic burstiness has a significant contribution towards the gain even in the case of not-too-bursty VoIP traffic. This is a promising trend for the cloud since other traffic types such as video are much more burstier, leading us to expect even higher gains.

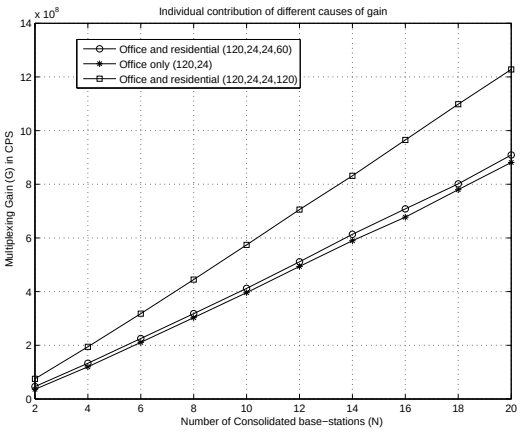


Figure 7: Individual Contributions to Multiplexing Gain

5 Conclusion

In this paper, we have considered a cloud architecture where base-station MACs are consolidated and shown that significant hardware gains can be derived by multiplexing base-station loads. These multiplexing gains increase with both network size as well as traffic level. Using multiplexing gain experiments, we have also shown that the cloud architecture is more suitable than today's networks in the face of growing traffic loads. Finally, we have also identified the contribution of traffic burstiness in this gain, leading to the promising discovery that traffic burstiness contributes significantly to overall gain even in a VoIP traffic setting. These experiments quantify and give insights into the infrastructural gains of consolidating base-stations.

In the future, it would be interesting to see a similar analyses for a complete WNC with the physical-layer also being consolidated. Also, it would be interesting to see similar experiments for more generic traffic mixes and confirm our expectations of even higher gains in those settings.

References

[1] Y. Lin, L. Shao, Z. Zhu, Q. Wang, and R. K. Sabhikhi, "Wireless network cloud: architecture and system requirements". IBM J. Res. Dev. 54, January 2010, pp 38-49.

[2] Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update, 2009-2014.

[3] P.Gupta, A.Vishwanath, S.Kalyanaraman, Y. Lin, "Unlocking Wireless Performance with Co-operation in Co-located Base Station Pools", *Proceedings of* COMSNETS 2010, Invited Paper.

[4] D. Staehle, K. Leibnitz, P. Tran-Gia, "Source Traffic Modeling of Wireless Applications", Report No. 261, Universitt Wrzburg Institut fr Informatik Research Report Series, June 2000.

[5] P.T.Brady, "A Model for Generating ON-OFF Speech Patterns in Two-Way Conversations", Bell System Technology Journal, Vol. 48. September 1969.

[6] J.G. Andrews, A. Ghosh, R. Muhamed. "Fundamentals of WiMAX: Understanding Broadband Wireless Networking". Prentice Hall Communications Engineering and Emerging Technologies Series, March 2007.

[7] M. Chetlur, U. Devi, P. Dutta, P. Gupta, L. Chen, Z. Zhu, S. Kalyanaraman, Y. Lin, "A software WiMAX medium access control layer using massively multithreaded processors". IBM J. Res. Dev. 54, January-February 2010.

[8] M. Mitzenmacher, E. Upfal, "Probability and Computing : Randomized Algorithms and Probabilistic Analysis", Cambridge University Press, 2005.

[9] J.D.C Little, "A Proof of the Queueing Formula $L = \lambda W$", Operations Research, 9, 383-387, 1961.

[10] S. Deng, "Traffic Characteristics of Packet Voice", IEEE International Conference on Communications, Vol. 3, pp 1369-1374, 1995.