Baseband Signal Compression in Wireless Base Stations

Aida Vosoughi
ECE Department, Rice University
Houston, Texas 77005
vosoughi@rice.edu

Michael Wu
ECE Department, Rice University
Houston, Texas 77005
mbw2@rice.edu

Joseph R. Cavallaro
ECE Department, Rice University
Houston, Texas 77005
cavallar@rice.edu

***Abstract*—To comply with the evolving wireless standards, base stations must provide greater data rates over the serial data link between base station processor and RF unit. This link is especially important in distributed antenna systems and cooperating base stations settings. This paper explores the compression of baseband signal samples prior to transfer over the above-mentioned link. We study lossy and lossless compression of baseband signals and analyze the cost and gain of each approach. Sample quantizing is proposed as a lossy compression scheme and it is shown to be effective by experiments. With QPSK modulation, sample quantizing achieves a compression ratio of 4:1 and 3.5:1 in downlink and uplink, respectively. The corresponding compression ratios are 2.3:1 and 2:1 for 16-QAM. In addition, lossless compression algorithms including arithmetic coding, Elias-gamma coding, and unused significant bit removal, and also a recently proposed baseband signal compression scheme are evaluated. The best compression ratio achieved for lossless compression is 1.5:1 in downlink. Our simulations and over-the-air experiments suggest that compression of baseband signal samples is a feasible and promising solution for increasing the effective bit rates of the link to/from remote RF units without requiring much complexity and cost to the base station.**

I. Introduction

A base station consists of a base station processor (BSP) which is connected to a radio unit via copper wire or fiber optic serial data link. This link transfers baseband signal samples. Common Public Radio Interface (CPRI) [1] is an industry standard that specifies a data transfer protocol for the serial data link connecting BSP and RF. In traditional base stations, BSP and RF units are collocated and the link connecting them is relatively short (e.g. connecting BSP box near antenna tower to the top of the tower). Distributed antenna system is another example of a base transceiver station which distributes signal data from a main antenna at base station to multiple remote antennas (100s of meters away). In such a setting, since several remote antennas are connected to a single BSP, the characteristics of BSP-RF links become even more important.

The high data rate requirements of evolving wireless standards have led to a need for increasing the data rates on BSP-RF links. However, in both traditional base stations and the ones with distributed antennas, increasing the rates of transmission or increasing the number of physical links largely adds to the hardware complexity and cost. While 3G required lower bit rates to carry W-CDMA carriers to/from baseband, 4G standards require bit rates that are multiple times higher (in Gbps range). As a result, more complicated fiber optic transceivers and integrated circuits are needed at both ends of the BSP-RF link. Cloud-based radio access network (C-RAN) which is a new paradigm for base stations also requires high data rates over the links between remote radio heads and the base station server [2] [3]. On the other hand, it requires the radio heads to be designed as simple as possible. These requirements make the BSP-RF link critical.

Recently, compression of data prior to transfer over the BSP-RF serial links has been proposed and attracted much interest. Compression is a pathway to increase the effective bit rate of the existing links without adding to the resource usage or link complexity. Since the baseband signals are usually oversampled and the samples are represented with more bit resolution than needed, compression can reduce some of the redundancy. Therefore, compression can help to simplify the link between remote/local RF units and BSP, and reduce the power consumption and cost. Compression can also eliminate the need to upgrade the existing slow links. Compressing and decompressing signal samples have overhead too. In order to be cost-effective, compression/decompression schemes must have reasonably low latency and use minimum resources. In addition, the distortion caused by a lossy compression has to be tolerable.

Figure 1 illustrates a base station architecture that applies compression in both downlink and uplink. In uplink (from RF to BSP), the digital signal samples of each baseband channel are first compressed at the RF unit. Each compressed sample includes a compressed I and Q sample. The compressed samples are then transfered from RF to BSP and finally are decompressed in BSP, where the I and Q samples are retrieved and further processed. The compression process in downlink is the reverse of uplink.

In this paper, we explore different lossy and lossless approaches to the compression of baseband samples traveling between BSP and RF in a base station. The ultimate goal is to find a solution which balances the compression performance, complexity, and distortion. Over-the-air experiments help us to compress actual baseband signals and observe the effect of compression on channel bit error rate.

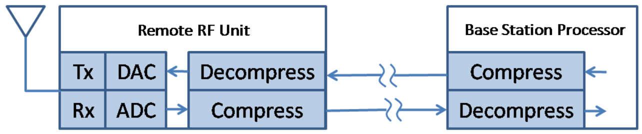

Fig. 1. Base station architecture with compression/decompression

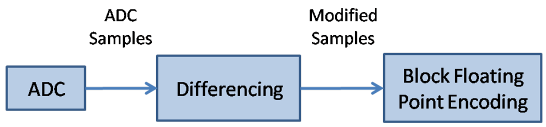

Fig. 2. Compressing ADC samples in [5]

II. Related Works

As mentioned before, compression of sampled high speed analog signals, as in wireless infrastructure, has not been studied greatly before. One reason for this can be the broad variations in signal parameters including center frequency, bandwidth, and signal-to-noise ratio [4]. On the other hand, well-studied lossy audio and video compression algorithms are not useful for compressing communications signals. These algorithms take advantage of human hearing and vision limitations and moreover they require significant signal processing effort which is not desirable for our purpose.

Wegener [5] proposes compressing baseband signals over the BSP-RF serial data link of the base transceiver station. Their proposed compression method is sample-by-sample differencing followed by block floating point encoding. The proposed encoding in [5] divides a sequence of samples into groups of fixed length. The required number of bits to represent the largest sample is determined for each group. This is called the exponent of that group. All the samples in the group are then encoded using the same number of bits indicated by the group's exponent. The author in [5] also suggests that the exponent differences be encoded with a Huffman code [6] to achieve additional compression. Since the exponents can be highly correlated, Huffman codes can help to decrease the total number of bits required to represent them. The proposed compression algorithm is in fact composed of two blocks: a redundancy remover (a series of sample-by-sample difference operations), and a bit packer (block floating point encoder). Figure 2 describes the algorithm schematically. It is stated in [7] that implementation of the proposed compressor in a digital signal processor requires about 50 instruction per sample and compression ratios fall between 1.3:1 and 2:1 on baseband signals.

Wegener [8] describes real time sensor compression for medical transducers such as CT scanners to reduce the costs of the medical imaging data acquisition, transport, and storage infrastructure. Medical imaging sensor signals are usually oversampled, they have high peak-to-average ratios, and they are sampled by imperfect ADCs (that is effective number of bits is less than resolution) [8]. As a result, there are redundancies that can be removed. A method for analog compressed sensing is introduced in [9]. In this method, analog bandlimited signals are sampled at rates lower than Nyquist, without loss of information. This allows compression in the sampling stage. The paper [9] considers the case of multiple narrowband transmissions with arbitrary and unknown carrier frequencies.

Recently, several compression techniques have been proposed to improve the performance and power efficiency of serial links in network on chip (NoC) architectures. Ogg et al. [10] propose Unused Significant Bit Removal (USBR) as a real-time compression technique for NoC. The proposed technique is aimed at data streams where the most significant bits are less likely to change compared with the least significant bits. They report the results of compression on MPEG1 coded picture data as an average bit reduction of 17% to 47%. Hong-Sik et al. [11] propose a lossless data compression scheme for NoC, based on Rice-Golomb coding [12]. The compression unit includes a compression prediction unit. Based on the predicted value for each data packet, the compressor decides whether or not to compress the packet. Their experimental results show around 15% improvement in throughput. Frequent-pattern compression [13] has been used for cache compression recently. It compresses some frequent patterns such as consecutive 0s and 1s in data streams.

While all of the above schemes are lossless, we propose a lossy compression scheme for baseband samples in this paper and compare its performance with some of the schemes explained in this section. We show that the proposed lossy method can have a significantly better performance compared to the lossless ones.

III. Proposed Compression Schemes

We consider both lossless and lossy compression schemes. Lossy methods have better performance but they cause distortion as some of the samples are corrupted or lost. Lossless compression is preferred where the data loss tolerance is limited. On the other hand, the compression ratio of lossless compression schemes is normally smaller than that of lossy schemes. In summary, we try to find a balance between compression performance and data loss by exploring different schemes. Finally, the implementation complexity (e. g. gate count or silicon area) and latency of compression and decompression modules that must be added to base stations is of great importance. These factors directly affect the cost and performance of a base station.

Although, at first sight it may seem that there is no redundancy in modulated symbols traveling between BSP and antenna, there are a couple of processing steps done in base stations that generate correlation between baseband signal samples. One of these processes is the upsampling and interpolation of the modulated symbols by pulse shaping filter. This process removes the drastic changes in the random-looking baseband signal samples and therefore the consecutive samples will follow a smoother trend. As a result, these samples are much more compressible than the samples before upsampling and filtering. Figure 3 shows the effect of square root raised

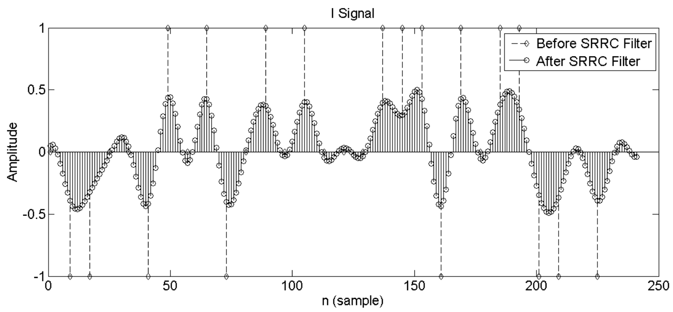

Fig. 3. Pulse shaping filter increases the correlation between adjacent samples.

cosine filtering with an upsampling rate of 8 on baseband I symbols (modulated with differential QPSK). Exploiting the existing correlation between consecutive samples is a key factor in compression of baseband signal samples. In addition, analog to digital conversion of baseband samples may generate digital values that have higher bit-resolution than needed.

As shown in Figure 1, in the downlink direction the compression must be done in BSP and the decompression is done at the other end of the link in RF unit. In uplink, the compression is done in RF and the decompression is done in BSP. There is an important difference between downlink and uplink compression process and that is in downlink, where the base station is transmitting, the signal to be compressed is not noisy. On the contrary, in uplink the signal to be compressed has traveled over the air and hence is affected by the noisy channel. As a result, we expect uplink compression to be more challenging.

A. Lossless Compression

Among several other lossless compression schemes that we studied and implemented, in this paper we discuss the following schemes which appear to have the best performance in compressing baseband signals: 1. Adaptive arithmetic coding (AC) [14], 2. Elias-gamma coding [15], and 3. Unused significant bit removal (USBR) [10]. These algorithms have never been proposed in the BSP-RF link compression. AC is a variable length entropy encoding that encodes the entire sequence of symbols into a single number, a fraction belonging to the interval [0,1). Assume symbol alphabet is $\{A, B\}$ with distribution (p_A, p_B), where p_A and p_B are the probabilities of the source generating A and B, respectively. AC starts with the interval [0,1) and for each input symbol it shrinks the interval based on the probability of the received symbol. The compressor output will be the shortest binary representation of a real number in the final interval. This description can be generalized to alphabets with larger size. Adaptive AC is an improved version of AC in which the values of p_A, p_B are updated over time based on the observation of the generated symbols. Elias gamma code is a prefix-free universal code which encodes integer numbers by appending a number of zeros before the binary representation of each number. The number of zero bits appended is one less than the binary length of the number. USBR [10] which was mentioned in Section II, is a lossless compression technique that groups a fixed number of samples together and extracts the common most significant bits of all of the samples in each group. Then it outputs the common most significant bits once instead of repeating them for every sample. The baseband signal lossless compression scheme proposed in [5] is also implemented for comparison purposes.

Differencing the consecutive samples decreases the range of the values to be compressed and accordingly it may improve the compression performance as we show in our results later. We propose to use a differencing step before compressing with each of the above lossless schemes. The differencing process is presented in a recursive form in (1). If $\mathbf{x}_0$ is the original input sample vector to the compressor then we define $\mathbf{x}_i$ as the ith derivative of $\mathbf{x}_0$. For differencing order of i, $\mathbf{x}_i$ is computed from $\mathbf{x}_{i-1}$ by calculating the differences between consecutive elements of the vector $\mathbf{x}_{i-1}$. Note that when differencing a sequence, the first element of the sequence must remain unchanged for reference. The differencing operation does not have much overhead since it needs only simple subtractions. Clearly, a higher order differencing is more costly.

$$\begin{aligned} \mathbf{x}_i &= [x_{i0},\ x_{i1},\ x_{i2},\ x_{i3},\ ...] \\ \mathbf{x}_{i+1} &= [x_{i0},\ x_{i1}-x_{i0},\ x_{i2}-x_{i1},\ x_{i3}-x_{i2}, ...] \end{aligned} \tag{1}$$

TABLE I
COMPRESSION SCHEMES COMPLEXITY (W: WINDOW SIZE, n: MAX # OF BITS PER SAMPLE)

Compression scheme	Required operation	Timing
[5]	bit shift, bit compare, 1 subtract and 1 table lookup per W samples	$O(n.W)$
USBR	bit shift, bit compare	$O(n.W)$
Adaptive AC	bit shift, bit compare, multiply	$O(n)$
Elias-gamma	bit shift, append bit	$O(n)$

In Table I, we describe the operations needed in each of the lossless compression schemes. In this table, W is the window size used in USBR and [5]; and n is the maximum number of bits per sample. The timing complexity of each algorithm is also listed in the third column. USBR and [5] have the highest timing complexity since they consider a window of samples for compression. Since [5] also encodes exponents of each block with Huffman code, it requires one subtraction and one table lookup per block. Adaptive AC has the most complex operations among all since it needs multiplication which is costly in hardware. Another complication in adaptive AC is keeping track of the symbol probability distributions. The over-the-air experimental results for lossless schemes are presented in Section V.

B. Lossy Compression

We propose scalar quantization or least significant bit removal for compressing baseband samples. This lossy compression scheme shifts out a number of least significant bits (LSB) of each sample in the transmitter and shifts in zeros instead in the receiver. Therefore, using this technique, has the same effect as quantization when the transmitted and received samples are compared. In the equations in (2), A is the original n-bit sample, A' is the sample after removing l LSBs, and A'' is the decompressed version. Equation (3) shows the compression ratio (CR) of quantization and the maximum error magnitude (maxErr). Clearly, by using this scheme, less number of bits are transmitted for each sample. However, replacing a number of LSBs with zero introduces inaccuracy. Note that in uplink as the received symbols are noisy, one can expect that a number of LSBs are already in error, therefore, simply not transmitting noisy bits and replacing them by zero bits in the receiver does not increase BER considerably.

$$\begin{aligned} A &= a_{n-1} \; .. \; a_{l+2} \; a_{l+1} \; a_l \; .. \; a_2 \; a_1 \; a_0 \\ A' &= a_{n-1} \; .. \; a_{l+2} \; a_{l+1} \; a_l \\ A'' &= a_{n-1} \; .. \; a_{l+2} \; a_{l+1} \; a_l \; 0 \; 0 \; .. \; 0 \end{aligned} \tag{2}$$

$$\text{CR} = \frac{n}{n-l} \;, \qquad \text{maxErr} = 2^l - 1 \tag{3}$$

Quantization only increases the probability of error if it forces the received constellation points to pass the detector's boundaries. As we show in simulation and experimental results in the next sections, the number of LSBs that can be removed without causing distortion depends on the modulation type. With higher size constellations, the Euclidean distance between constellation points are smaller, therefore in order for the detector to detect correctly, more resolution is needed. As a result less LSBs can be ignored in modulations with more symbols. In other words, since higher size modulations are more sensitive to noise and quantizing I/Q samples is a type of added noise, we expect to be more sensitive to quantization.

For example, when QPSK (Quadrature Phase-Shift Keying) modulation is used, I/Q samples can take only two values each (-1,+1). This means that the detector only needs to know whether the sample is negative or positive. Therefore, if our ADC generates 16-bit samples in the receiver, then we expect to be able to drop a large portion of LSBs without affecting BER. With 16-QAM (Quadrature Amplitude Modulation), I/Q samples take four values each (-3, -1, +1, +3). As a result, while still there may be a lot of redundancy in each 16-bit sample, a larger number of bits is needed to accurately represent I/Q values without causing error.

Our simulations and over-the-air experiments, presented in the next sections, confirm this reasoning. The proposed lossy scheme can be implemented efficiently in hardware as it only needs bit shift operations. Its timing complexity is $O(n)$, where n is the maximum number of bits in each sample.

IV. Simulations Results

In this section, the performance of the lossy compression scheme is explored via simulations. For our simulations, we assume a two-node system: the source and destination. In uplink, the source is the mobile station and the destination is the base station and it is the opposite way in downlink. We further assume that the nodes are equipped with one antenna each, therefore this is a SISO (single input single output) system. The input bit vector is chosen uniformly at random and the wireless channel is modeled as a Rayleigh fading channel. The channel model can be expressed as: $\mathbf{y} = h{\cdot}\mathbf{x}+\mathbf{n}$, where $\mathbf{x}$ is the upsampled, upconverted, and modulated transmit symbol vector, $\mathbf{y}$ is the received vector, h is the Rayleigh channel response and $\mathbf{n}$ is additive white Gaussian noise. The vector $\mathbf{y}$ is downconverted, downsampled, and demodulated in the receiver to retrieve the original bit vector.

Figure 4 shows lossy compression simulation results for differential QPSK (DQPSK), 16-QAM, and 64-QAM modulation types in uplink and downlink. Three BER curves are shown in each figure including the original (no compression) case. The other two curves show two different cases of compression with different number of LSBs truncated. The case with lower LSB in each figure represents the maximum number of LSBs that can be ignored such that the BER remains very close to the original case. Note that the full precision in downlink and uplink is assumed to be 16 and 14 bits, respectively. As it is clear from the results, a relatively large portion of bits can be truncated without affecting the BER significantly. This results in a reasonable compression ratio without appreciable increase in error rate. We further explore lossy compression performance by over-the-air experiments in Section V.

V. Over-the-Air Experimental Results

A. Experimental Setup

To conduct our over-the-air experiments, we use Rice university's WARP hardware [16] and WARPLab framework [17]. WARP is a programmable wireless platform which includes FPGA, radio, ADC/DAC, and other components to prototype a wireless physical layer. Using WARPLab, we can interact with WARP nodes directly from the MATLAB workspace. Our experimental setup includes two WARP nodes, one as the transmitter and the other one as the receiver. We make use of the WARPLab SISO MATLAB reference design and add our compression and decompression functions to it. The reference design includes generating a bitstream, modulating, upsampling, pulse shape filtering, transmitting the signal over a wireless channel using WARP radio boards, filtering the received signal with a matched filter, downsampling, demodulating and recovering the transmitted bitstream.

The radio boards do the digital to analog and analog to digital conversion with a sampling rate of 40 MHz; they also do the upconversion/downconversion to/from RF. The ADC and DAC resolutions are 14 and 16 bits, respectively. The carrier frequency is 2.4 GHz. Upsampling and downsampling rates are 8 samples per symbol and the channel bandwidth is assumed to be 5 MHz. We use an indoor wireless line-of-sight channel. Both of the WARP nodes are connected to a desktop computer via an Ethernet switch.

In this setup, for downlink, the workstation plays the role of the base station processor, the transmitter node can be considered as the radio unit in base station, and the receiver

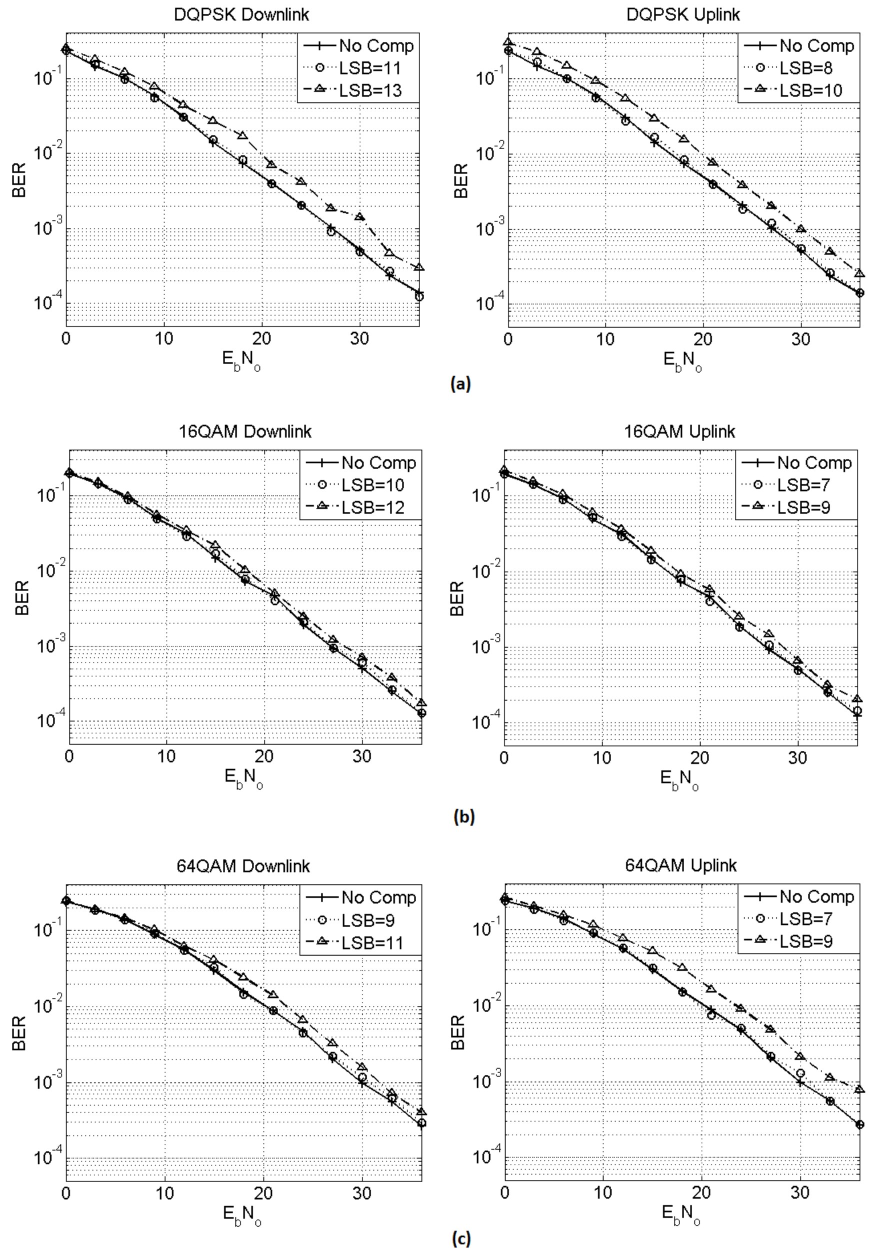

Fig. 4. Lossy compression simulation results for: (a) DQPSK (b) 16QAM (c) 64QAM. LSB shows the number of least significant bits that are ignored from the 16 bit samples in the downlink and the 14 bit samples in the uplink.

node is emulating the mobile station. In uplink, the transmitter can be assumed as a mobile station. The signal is then transmitted over-the-air to the receiver node which plays the role of the base station radio unit and finally the signal is retrieved by the computer which is emulating the base station processor. This architecture is described in Figure 5. Since the link between BSP and RF is wired, it can be assumed a perfect channel without noise. Therefore, for the sake of simplicity of experiments, we do both compression and decompression in the MATLAB script running in the workstation.

B. Lossless Compression Results

The performance results of five different lossless compression schemes in uplink and downlink directions are shown in Table II. All of these algorithms consist of a differencing step before compression. For this experiment, the input bitstream is chosen at random, the modulation is differential QPSK, the measured average BER of the channel is 0.0009, and 4000 samples have been compressed in each trial. In downlink, [5] results in the best compression ratio (which is 1.5:1). This compression ratio has been achieved with a differencing order of 3 while [5] proposes differencing up to 2. The next best compression ratio in downlink is for USBR and

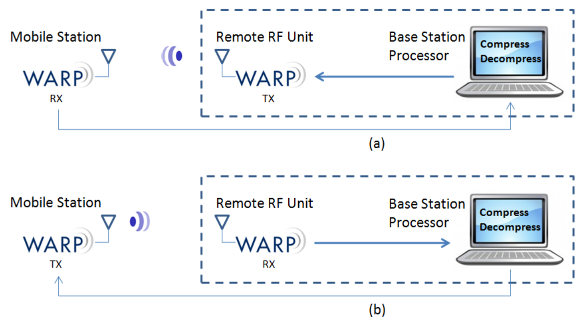

Fig. 5. Base station emulation with WARPLab in (a) downlink (b) uplink

TABLE II
LOSSLESS COMPRESSION RATIO RESULTS, DQPSK MODULATION

Algorithm	TX	RX
[5]	1.51	1.07
USBR	1.41	1.17
Elias-gamma+Adaptive AC	1.40	0.84
Adaptive AC	1.26	0.95
Elias-gamma	1.22	0.72

Elias-gamma+adaptive AC. We found that the combination of Elias-gamma coding and adaptive AC has better compression performance than each of them alone.

As it is clear from Table II, in all cases, the compression ratio is noticeably lower in uplink. This is because the noise of the wireless channel corrupts the smoothness of the signal samples trail, achieved by the pulse shaping filter in the transmitter. Therefore, the correlation between the consecutive samples in the receiver is decreased. In fact, in most of the cases, the RX compression ratio is even lower than 1 which suggests that lossless compression in uplink expands the data as opposed to compressing it. USBR has the best compression ratio in uplink.

Another important observation that we made is that increasing upsampling rate improves the compression ratios in TX direction but has no effect on compression ratios in the RX direction. The improvement in compression ratio in TX is because when the upsampling rate is greater, the samples are closer and more correlated. The fact that the compression ratio is not improved for RX, shows that, in presence of noise, even the closer sampling cannot increase the correlation or compressibility of samples. We observed that, in uplink a differencing order of 1 results in the best compression ratio and further differencing is not beneficial. But, in downlink, a differencing order of 4 is the best for the majority of algorithms.

C. Lossy Compression Results

To see how the inaccuracy introduced by quantization affects the average BER of the receiver, we measured the BER for different number of LSBs ignored both in downlink and uplink. Note that, ADC and DAC resolution are 14 bits and 16 bits, respectively. We measured the average BER for the case of no compression for comparison. The experiment was repeated for five transmit power levels. Similar to the previous experiments, the input bitstream is random, and the upsampling rate is 8. We tried two modulation types: DQPSK and 16-QAM.

TABLE III
LOSSY COMPRESSION RATIO RESULTS

	DQPSK	16-QAM
TX	4	2.3
RX	3.5	2

Our experimental results show that for DQPSK modulation, 12 and 10 LSBs can be ignored in downlink and uplink directions, respectively, without affecting BER. See Figures 6 and 7. These quantization levels result in a compression ratio of 4:1 and 3.5:1 for downlink and uplink respectively (driven from equation (3)). Figures 6(b) and 7(b) show that ignoring greater number of LSBs does affect the BER and thus must be avoided. For 16-QAM modulation, 9 and 7 LSBs may be removed without BER being affected in downlink and uplink respectively. See figures 8 and 9 for BER curves comparisons. This results in a compression ratio of 2.3:1 in downlink and 2:1 in uplink. Table III summarizes the best lossy compression ratios for both modulation types. The over-the-air experimental results for lossy compression performance (number of LSBs) closely match the simulation results shown in Section IV with few disparities. The differences may come from the dissimilarity of channel model with the over-the-air channel and/or additional experimental noise sources that were not modeled.

Other than BER, there is another factor that resists excessive amount of quantization and that is the quantized baseband signal must have no out of band frequency components. That means the effect of the pulse shaping filter must not be neutralized completely. We observed our results with the above-mentioned levels of quantization in the frequency domain and made sure this was not the case.

VI. CONCLUSIONS

In this paper, we explored lossless and lossy compression of baseband signal samples. Compression of baseband signals is beneficial in decreasing the bit rates on the serial link between base station processor and antenna in base transceiver stations. We proposed quantization as a lossy compression method. Quantization offers better compression ratio compared to lossless schemes. We analyzed a number of lossless compression schemes including adaptive arithmetic coding, Elias-gamma coding, unused significant bit removal, and a recent scheme proposed specifically for baseband signals. Lossless compression does not seem to be as promising as lossy compression for our purpose; particularly, lossless schemes are less effective in uplink. While lossless methods have the advantage of not introducing distortion at all, we showed that quantization can have better compression performance without adding noticeable distortion. The performance of a quantization method is highly dependent on several factors

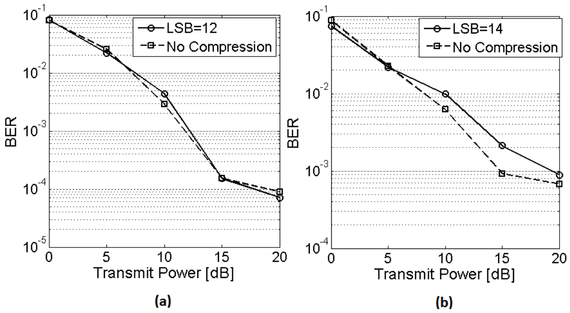

Fig. 6. Comparing BER in downlink when (a) 12 LSBs (b) 14 LSBs are ignored. Modulation is DQPSK.

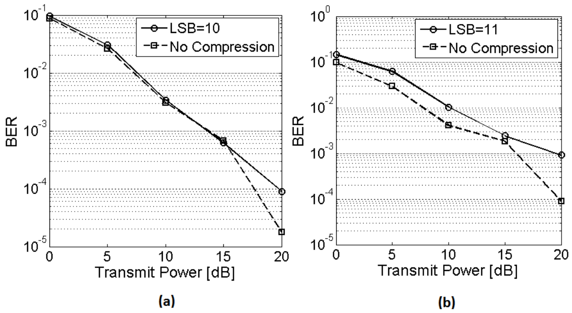

Fig. 7. Comparing BER in uplink when (a) 10 LSBs (b) 11 LSBs are ignored. Modulation is DQPSK.

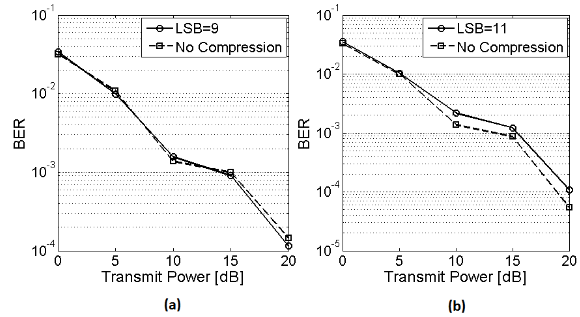

Fig. 8. Comparing BER in downlink when (a) 9 LSBs (b) 11 LSBs are ignored. Modulation is 16-QAM.

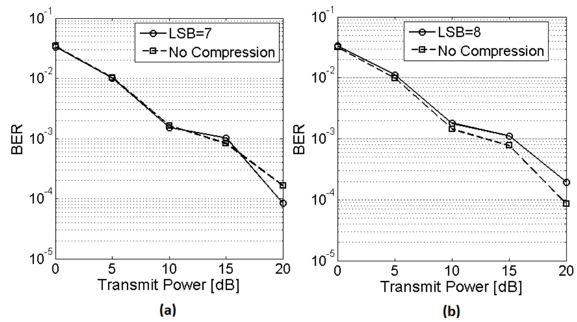

Fig. 9. Comparing BER in uplink when (a) 7 LSBs (b) 8 LSBs are ignored. Modulation is 16-QAM.

including modulation type, channel bandwidth and noise level. Quantization has a very good compression performance for modulation types with low constellation size and since it is fairly cheap to implement in hardware it can be a good option for compression in base stations.

The feasibility of an adaptive compression method has not been studied before and can be future work. For example, different compression algorithms may be deployed based on a set of defined parameters. Since the RF module is generally desired to be simpler compared to base station processor, an asymmetric compression approach can be beneficial. In addition, the link is asymmetric by nature since the input of compressor is noisy in uplink but noiseless in downlink.

Acknowledgments

This work was supported in part by the National Science Foundation under grants EECS-0925942 and CNS-0923479.

References

[1] "Common Public Radio Interface (CPRI): Interface Specification V4.2," 2010. [Online]. Available: http://www.cpri.info

[2] D. M. Dicke and P. Cameirao, "lightRadio™ baseband processing and backhauling," Sept 2011, accessed 30-Mar-2011. [Online]. Available: http://www2.alcatel-lucent.com/blogs/techzine/2011/lightradio-baseband-processing-and-backhauling/

[3] H. Guan, T. Kolding, and P. Merz, "Discovery of cloud-RAN (Nokia Siemens Networks)," Apr 2010, accessed 30-Mar-2011. [Online]. Available: http://www.thecom.co.il/files/wordocs/article_download.pdf

[4] A. Wegener, "Adaptive compression and decompression of bandlimited signals," *US Patent 7,009,533*, March 2006.

[5] ——, "Compression of baseband signals in base transceiver systems," *US Patent 8,005,152*, August 2011.

[6] D. Huffman, "A method for the construction of minimum-redundancy codes," *Proceedings of the IRE*, vol. 40, no. 9, pp. 1098 –1101, Sept 1952.

[7] A. Wegener, "Algorithm delivers lossless compression to ADC samples," 2010, accessed 24-Nov-2011. [Online]. Available: http://electronicdesign.com/article/analogmixedsignal-design/algorithm_delivers_lossless_compression_to_adc_samples.aspx

[8] ——, "Compression of medical sensor data [exploratory DSP]," *Signal Processing Magazine, IEEE*, vol. 27, no. 4, pp. 125 –130, Jul 2010.

[9] M. Mishali and Y. Eldar, "Xampling: Analog data compression," in *Data Compression Conference (DCC), 2010*, Mar 2010, pp. 366 –375.

[10] S. Ogg and B. Al-Hashimi, "Improved data compression for serial interconnected network on chip through unused significant bit removal," in *19th International Conference on VLSI Design*, Jan 2006, p. 5 pp.

[11] H.-S. Kim, Y. Jung, H. Kim, J.-H. Ahn, W.-C. Park, and S. Kang, "A high performance network-on-chip scheme using lossless data compression," *IEICE Electronics Express*, vol. 7, no. 11, pp. 791–796, 2010.

[12] S. Golomb, "Run-length encodings (corresp.)," *IEEE Transactions on Information Theory*, vol. 12, no. 3, pp. 399 – 401, Jul 1966.

[13] A. R. Alameldeen and D. A. Wood, "Frequent pattern compression: A significance-based compression scheme for l2 caches," Computer Sciences Department, University of Wisconsin-Madison, Tech. Rep., 2004.

[14] G. Langdon and J. Rissanen, "Compression of black-white images with arithmetic coding," *IEEE Transactions on Communications*, vol. 29, no. 6, pp. 858 – 867, Jun 1981.

[15] P. Elias, "Universal codeword sets and representations of the integers," *IEEE Transactions on Information Theory*, vol. 21, no. 2, pp. 194 – 203, Mar 1975.

[16] "Rice University WARP project." [Online]. Available: http://warp.rice.edu

[17] "WARPLab framework overview." [Online]. Available: http://warp.rice.edu/trac/wiki/WARPLab