Improved LPC-Based Fronthaul Compression with High Rate Adaptation Resolution

Leonardo Ramalho, Igor Freire, Chenguang Lu, Miguel Berg and Aldebaro Klautau

***Abstract*—This paper presents a fronthaul signal compression scheme based on linear prediction coding (LPC) adapted to orthogonal frequency division multiplexing (OFDM) signals. The proposed method is capable of providing fine tuning of the compression factor, which is an alternative to legacy compression methods that tune the compression factor by changing the discrete number of bits of the quantizer and, consequently, are only able to do so with coarse resolution.**

***Index Terms*—LTE signal compression, C-RAN, fronthaul, OFDM , LPC, Huffman.**

I. Introduction

RADIO access networks (RAN) are evolving to meet the increasing data transport and flexibility demands. An attractive architecture is the centralized RAN (C-RAN), where centralized baseband units (BBUs) communicate to remote radio units (RRUs) over the fronthaul (FH). This FH transport, however, is currently challenged by a limitation of the current protocol, i.e. based on CPRI [1], in which in-phase (I) and quadrature (Q) samples are transported between BBUs and RRUs. This leads to FH rates growing rapidly with the air-interface's bandwidth and number of antennas [2].

There are two main alternatives to alleviate such FH rates: different functional split options [2] and fronthaul signal compression (FSC). This work focuses on the latter, as it requires minimum changes in the existing BBU and RRU design. There are currently a variety of FSC schemes exploiting different techniques such as resampling [3], vector quantization [4], linear predictive coding (LPC) [5] and many others. Most of them are lossy and partially rely on the use of re-quantization, hereafter called quantization.

The main limitation of such methods that is addressed by the present work is their coarse compression factor resolution. This can be a problem when considering future FH interfaces supporting multipoint-to-multipoint operation over packetized networks, rather than dedicated FH links between BBUs and RRUs, for example, as specified in eCPRI [1]. In this case, since the FH network will rely on statistical multiplexing, the specific deployments in which FH traffic is allowed to coexist with others variable-rate traffics will be subject to eventual network congestions. In such scenario, a rate adaptive FH network can reconfigure the FSC modules to adapt the FH rates and avoid service disruption. Then, for improved performance in the adaptation, the FSC scheme would ideally allow fine tuning of the compression factor.

A finely-tunable FSC scheme can also provide improved flexibility in terms of the number of RRUs served over the FH. For example, if the FH capacity is R_{max} and the uncompressed stream of LTE IQ samples of each RRU requires a rate of R_F, the compression factor F applied for all k RRU streams must be $F \geq \frac{kR_F}{R_{\text{max}}}$. Then, to serve an additional RRU, the compression factor adopted by all RRUs could be increased by no much more than $\Delta F = \frac{R_F}{R_{\text{max}}}$, if the resolution allows.

Nonetheless, current FSC methods more commonly vary the compression factor by changing the number b_Q of quantization bits. For example, a combination of resampling and vector quantization (VQ) is adopted in [4], which reports compression factors of 5.5, 4.5 and 3.8 for downlink (DL) LTE signals when b_Q is varied between 5, 6 and 7 bits/sample, achieving error vector magnitudes (EVMs) of approximately 4.2%, 2.1% and 1%, respectively. Meanwhile, [3] uses resampling, block scaling and quantization and, for the same range of b_Q, varies its compression factor over 3.9, 3.4 and 2.9 with EVMs of 4.6%, 2.3% and 1.15%, respectively. In both cases, only coarse changes of the compression factor and distortion are achieved.

To overcome this limitation, we propose an FSC scheme that allows adjustments to the compression factor by changing not only b_Q, but also a loading factor γ, which is a continuous variable. The new FSC scheme derives from an earlier work [5] and is based on a combination of LPC adapted to orthogonal frequency division multiplexing (OFDM), being further improved here with adjustable scaling. It will be shown that the scaling creates a mechanism to achieve finer resolution in the adjustments to the compression factor and distortion.

II. Proposed Method

The proposed method is based on the scheme of [5], which encodes real (s_i) and imaginary (s_q) baseband components independently. This work assumes the RRU is capable of extracting the symbol timing synchronism and the cyclic prefix (CP) can be conveniently removed. The I or Q samples of the non-prefixed time-domain OFDM are denoted as $x[n]$, where $0 \leq n \leq N-1$ and N is the adopted FFT length. Subsequently, these samples are encoded using LPC and Huffman and the result sent over the FH. Lastly, at the decompression unit, all operations are reversed. The process is then repeated for each OFDM symbol. Fig. 1 shows the proposed FSC scheme, which inserts adjustable gains into the scheme of [5]

This work was supported in part by the Innovation Center, Ericsson Telecomunicações S.A., Brazil, CNPq (processs 201493/2017-9/PDE) and the Capes Foundation, Brazil, and by the European Union through the 5G-Crosshaul project (H2020-ICT-2014/671598).

L. Ramalho, I. Freire and A. Klautau are with the Computer and Telecommunications Engineering Department, Federal University of Para, Belem 66615-170, Brazil (e-mails: {leonardolr, igorfreire, aldebaro}@ufpa.br).

C. Lu and M. Berg are with Ericsson Research, Kista 164 80, Sweden (e-mails: {chenguang.lu, miguel.berg}@ericsson.com).

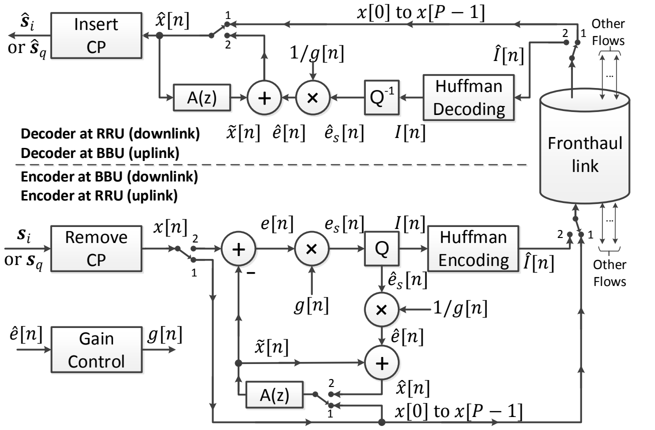

Fig. 1. Proposed method based on the OFDM-adapted LPC including the adjustable scaling of prediction error.

with the goal of introducing an additional parameter other than b_Q for controlling the rate and distortion of the FSC scheme.

As proposed in [5], the P initial samples of each OFDM symbol $x[n]$ are sent uncompressed over the FH (switches of Fig. 1 in position 1). This allows the decoder to initialize the P taps of the predictor $A(z)$ (see [5]) before starting the compression and, by doing so, avoid increased prediction errors at the OFDM symbol boundaries. In the end, the effective average compression factor becomes $F = \frac{b(N+N_{\text{CP}})}{bP+L(N-P)}$, where b is the number of bits used to represent the uncompressed I or Q components, N_{CP} is the CP length in samples and L is the average number of bits per Huffman codeword.

The adjustable scaling stages are placed before and after the uniform scalar quantizer Q. First, the scaled prediction error ($e_s[n] = g[n]e[n]$) is quantized, so that it is represented by a quantized value $\hat{e}_s[n]$ and a binary index $I[n]$. Next, the quantizer output ($\hat{e}_s[n]$) is multiplied by $1/g[n]$, which yields the rescaled and quantized version of the prediction error $\hat{e}[n]$.

After quantization, the index $I[n]$ is Huffman encoded to a variable-length codeword $\hat{I}[n]$ that is ultimately transmitted over the FH. The resulting average number of bits per Huffman codeword can be expressed as $L = \sum_i L_i P(e_s^i)$, where L_i is the length of the i-th codeword in the Huffman dictionary, $P(e_s^i)$ is the probability of $e_s[n]$ being represented by the i-th quantizer level and $i = 1, \ldots, 2^{b_Q}$. The proposed adjustable gain, then, provides a way to give fine changes in this average codeword length, as explained in the sequel.

In the proposed scheme, the Huffman dictionary is fixed (set during a training stage) and so are the lengths L_i, but the adjustable gain can change the distribution $P(e_s^i)$ and ultimately the average rate L. Since the gain $g[n]$ is a continuous variable, $P(e_s^i)$ can be changed in fine steps and consequently L too. Of course, there is an optimum Huffman dictionary that would give the lowest L for the scaled distribution, but here the sub-optimal fixed dictionary is preferred in order to avoid re-training during runtime and, consequently, simplify the implementation.

In addition to the flexible rate L achieved by the method, the proposed adjustable scaling also allows the control of the distortion levels resulting from the quantization process. The following sub-sections clarify how this is accomplished.

A. Analysis of the Quantization and Clipping Noise

The distortion introduced by the quantization process can be decomposed into two main effects [6]: (granular) quantization noise $n_q[n]$ and clipping noise $n_c[n]$. Hence, at the transmitter side, the prediction error after the scaling of $g[n]$ and the quantization can be modeled as: $\hat{e}_s[n] = e_s[n] + n_q[n] + n_c[n]$. The overall quantization, then, introduces a distortion whose variance (σ_d^2) can be modeled as the sum of the variance of both components [7], namely $\sigma_d^2 = \sigma_q^2 + \sigma_c^2$, where subscript q refers to quantization (granular) and c to clipping noise.

Based on Fig. 1 and the model of $\hat{e}_s[n]$ given above, the recovered prediction error after re-quantization and rescaling by $1/g[n]$ at the decoder side can be expressed as:

$$\hat{e}[n] = \frac{\hat{e}_s[n]}{g[n]} = e[n] + \underbrace{\frac{n_q[n] + n_c[n]}{g[n]}}_{\text{rescaled distortion}}. \tag{1}$$

This reveals that the adjustable gain $g[n]$ has the ability to influence the impact of the quantization distortion. In fact, this strategy is of central importance in the proposed method. Nevertheless, it should be noted that the $n_c[n]$ component itself is directly proportional to $g[n]$, since clippings are more likely to occur when the quantizer's input is scaled up. Thus, (1) should not lead to the conclusion that higher values of $g[n]$ yield lower overall distortion levels. Instead, it should be interpreted that there is a trade-off between quantization and clipping noise that is controlled by $g[n]$.

A useful metric to evaluate the referred trade-off is the loading factor [6], defined as $\gamma = V/\sigma_{es}$, where V is the maximum quantizer's output amplitude and σ_{es} is the root mean square (RMS) of the quantizer's input signal ($e_s[n]$). For the scheme of Fig. 1, this is equivalent to:

$$\gamma = \frac{V}{\sigma_e g[n]}, \tag{2}$$

where σ_e is the RMS value of the original prediction error $e[n]$. Since V is fixed in the method (set during training) and σ_e is approximately constant within an OFDM symbol, γ is varied solely by $g[n]$ and vice-versa.

In particular, since the value of γ is increased by decreasing $g[n]$ and the latter scales the prediction error that is input to the quantizer, high values of γ tend to reduce the occurrence of clipping noise. This can also be derived following that the variance of the clipping noise n_c is given by [7]: $\sigma_c^2 = 2V^2 \left[\left(1 + \frac{1}{\gamma^2} \right) Q(\gamma) - \frac{1}{\gamma\sqrt{2\pi}} e^{-\frac{\gamma^2}{2}} \right]$, where $Q(u) = (1/\sqrt{2\pi}) \int_u^\infty \exp(-v^2/2) dv$ is the Q-function.

In contrast, γ has no effect on the quantization noise power, given the latter depends solely on the quantization step ($\Delta \approx V/2^{b_Q - 1}$), i. e., the quantization noise power is given by $\sigma_q^2 = \Delta^2/12$ [6], [7]. Ultimately, the quantization noise power and clipping noise power, when expressed as a function of γ, consist respectively of a constant and a monotonically decreasing curve. Nevertheless, after the rescaling that follows the quantizer, the quantization noise power becomes monotonically increasing with γ (or decreasing with $g[n]$), as the $1/g[n]$ factor in (1) indicates. Meanwhile, the clipping noise power remains monotonically decreasing with γ, since the decrease of the clipping noise power σ_c^2 is faster than the increase in quantization noise power as $g[n]$ is reduced (γ is increased) in (1). For reference, Fig. 2 shows numerical results of σ_q^2, σ_c^2, σ_d^2 and their rescaled versions varying with γ. The curves were obtained with the block diagram in Fig. 1, where $e[n]$ follows a standard Gaussian distribution $\mathcal{N}(0,1)$, γ is changed with $g[n]$ as in (2), the quantizer has $b_Q = 7$ bits and $V = 5$.

As illustrated in Fig. 2, when γ is low ($g[n]$ is high) the clipping noise dominates among the two distortion components and determines most of the total rescaled distortion. Meanwhile, as γ is increased, the clipping noise decreases and the increasing rescaled quantization noise tends to dominate. Naturally, there is a crossing point in the total rescaled distortion curve where the prevailing noise component switches ($\gamma \approx 3.9$ in Fig. 2). Furthermore, there is also a point associated with the minimum total distortion. The minimum rescaled distortion in the case of Fig. 2, at $\gamma = 3.6$, is roughly 3 dB lower than the minimum total distortion before rescaling. Finally, and more importantly, note from Fig. 2 that the overall distortion can be adjusted in fine steps with γ, which is advantageous in comparison to the coarse adjustments achieved with b_Q, known to be of approximately 6 dB/bit [6].

B. Proposed Adjustable Scaling Scheme and Training Phase

Based on (2), the proposed scheme adopts:

$$g[n] = \frac{V}{\hat{\sigma}_e[n]\gamma}, \tag{3}$$

where γ is a target loading factor and $\hat{\sigma}_e[n] = \left(\sum_{m=1}^{N_g} \hat{e}[n-m]^2/N_g\right)^{1/2}$ is the estimated standard deviation of the predictor error, based on the last N_g samples of $\hat{e}[n]$. Hence, the gain $g[n]$ changes for each new $\hat{e}[n]$. Since a *backward* adaptation is adopted for $g[n]$, the encoder and decoder can both use the past $\hat{e}[n]$ samples to find the same $g[n]$, without exchanging side-information.

During run-time, when it becomes desirable to change γ to tune the compression factor, a new target γ can be sent to both the encoder and decoder and they both can find $g[n]$ from (3). The *forward* adaptation of γ needs to be done at most once per OFDM symbol such that its overhead is small.

The history of $\hat{e}[n]$ samples are reset at the end of every OFDM symbol to overcome an eventual abrupt change in power between adjacent symbols, and to avoid eventual error propagation among OFDM symbols. Thus, the first N_g values of $g[n]$ are forced to unit at both encoder and decoder sides in the beginning of each OFDM symbol.

In the proposed method, the predictor coefficients $\mathbf{a} = -[a_1, \ldots, a_P]$, quantization levels e_i (correspondingly V) and Huffman codewords are all found during the training phase that can be done in initialization or, for simplicity, even offline. After this phase, these parameters are kept constant.

Assuming off-line training, the adopted training signal is a distortionless sequence, i.e. one that has not passed through a channel. First, the predictor design is conducted as proposed in [5], where the predictor coefficients are calculated according to the Levinson-Durbin algorithm [6] and the autocorrelation function of the training signal. Then, the open loop prediction errors are computed by $e_o[m] = x[m] - \sum_{i=1}^{P} a_i x[m-i]$, so that the quantizer's dynamic range $[-V, V]$ can be determined using $V = \max_{m \notin \mathcal{C}} |e_o[m]|$, where $\mathcal{C}$ is the set of samples that correspond to the P initial samples of each OFDM symbol. These initial samples are skipped to avoid prediction errors with relatively high amplitude that can occur in the OFDM symbol borders, as detailed in [5]. The value of V is chosen to avoid quantizer clipping during the initial estimation of $\hat{\sigma}_e$, given high clipping noise would otherwise lead to poor initial estimation of $\hat{\sigma}_e$ [6]. Thirdly, once the predictor and quantizer are ready, the probability distribution at the output of the quantizer is found by applying the training signal into the closed-loop LPC of Fig. 1 and, finally, the Huffman codewords are computed. During this process, the gain is kept at $g[n] = 1$.

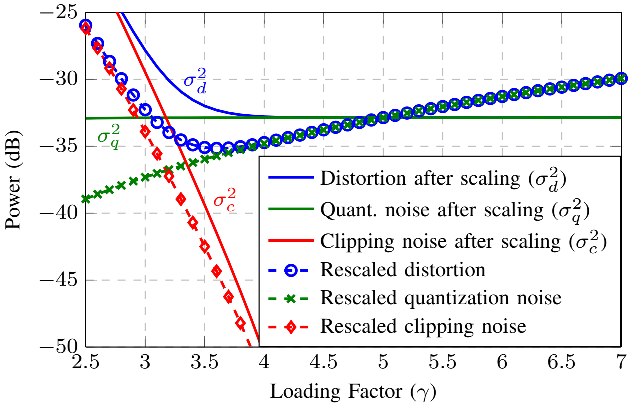

Fig. 2. Quantization and clipping noise power levels evaluated over varying γ before and after the rescaling gain, using the block diagram in Fig. 1.

Regarding the computational cost, there are three main operations that dominate the added complexity in relation to [5]: the divisions in (1) and (3), and the square root used to estimate $\hat{\sigma}_e[n]$. The operations inside the square root can consume less resources than the mentioned three operations because the summation $E_e[n] = \sum_{m=1}^{N_g} \hat{e}[n-m]^2$ can be found iteratively with $E_e[n] = \hat{e}[n-1]^2 + E_e[n-1] - \hat{e}[n-N_g-1]^2$, and the division $E_e[n]/N_g$ can be implemented with arithmetic shifts by choosing N_g as a power of 2.

III. Simulation Results

The proposed method was simulated with DL LTE signals and the performance was evaluated in terms of compression factor (F), EVM and *compression signal-to-noise ratio* (SNR). The latter measures how much distortion the FSC method adds to the signal and it is calculated as $\text{SNR} = \mathbf{E}\left[x^2\right]/\mathbf{E}\left[(x-\hat{x})^2\right]$, where x is the compressor input and $\hat{x}$ is the decompressor output, as illustrated in Fig. 1.

The simulations evaluate the compression and decompression of 100 DL 20 MHz LTE frames that uses 64-QAM. More specifically, the signal follows the fixed reference channel R.9 FDD defined in [8, Annex A]. The configurations of the encoder and decoder are $P = 5$, $N_g = 32$, $b = 15$ bits, and b_Q was varied from 5 to 8 bits.

At first, Fig. 3 aims to illustrate how the loading factor γ regulates the trade-off between quantization and clipping noise and consequently, how it impacts the SNR (left vertical axis). Note that all SNR curves have an optimal point, which comes from the best combination between clipping and quantization noise. Secondly, Fig. 3 shows how the compression factor (right vertical axis) varies with the loading factor, more specifically that F increases with γ. This happens because, from (3), a higher γ leads to lower $g[n]$ values (for $\hat{\sigma}_e$ and V fixed), so that the scaled prediction error is diminished and more likely quantized to levels near zero. These levels, in turn, have a higher probability of occurrence under a Gaussian distribution, so that their corresponding Huffman codewords are shorter (recall the Huffman dictionary is designed during training and kept fixed). The opposite happens for lower γ values: namely the scaled prediction error becomes more often quantized to levels that are distant from zero, which are associated to longer Huffman codewords, so that the compression factor decreases.

Furthermore, in Fig. 3 the relevant values of γ are the ones to the right of the SNR peaks, where the compression factor can be increased and traded by corresponding reductions in compression SNR. In Fig. 3, these regions correspond to $\gamma \geq 3.3$, 3.7, 4.1 and 4.3 for b_Q= 5, 6, 7, and 8, respectively. In contrast, the values of γ to the left of the SNR peaks are not interesting for rate adaptation as they result in lower SNR and lower compression. Lastly, Fig. 3 reinforces that only coarse adjustments of SNR and F are achieved by altering b_Q. For instance, with $\gamma = 5$, the SNR changes about 6 dB per bit in b_Q. In contrast, the proposed method allows fine adjustments in the SNR and F by changing γ with the resolution of interest.

As mentioned, the main advantage of the new method is the improved resolution. By changing b_Q, previous methods [3]–[5] allow only coarse resolution, as indicated by their discrete and relatively dispersed markers in Fig. 4. In contrast, the curves corresponding to the proposed method indicate its finer resolution. The curves with colored dots show results obtained with the proposed method for $b_Q = 5, \ldots, 8$ bits, where γ was varied from 2.6 to 10, in steps of 0.2. The actual performance of the new method corresponds to the smooth black dashed curve, which is the minimum EVM achieved for a given F, for relevant values of γ as explained in the previous paragraph. Note that some of the results of the proposed method in Fig. 4 are dispersed, but these regions at the left of the curves are not used, given they present a higher EVM and lower F.

Besides the compression resolution itself, the proposed method also presents competitive performance with others methods, as shown in Fig. 4. For instance, the proposed method has better performance than the method shown in [3], i. e., the new method achieves a lower EVM for the same compression factor, especially for $F > 3$. When compared to the method in [5] with $P = 5$, the proposed method has almost identical performance, except for $F = 5.27$, where the previous method achieves a lower EVM. Finally, as expected due to the VQ, the method shown in [4] outperforms the proposed method with respect to rate-distortion, but at a higher computational cost in the encoding stage.

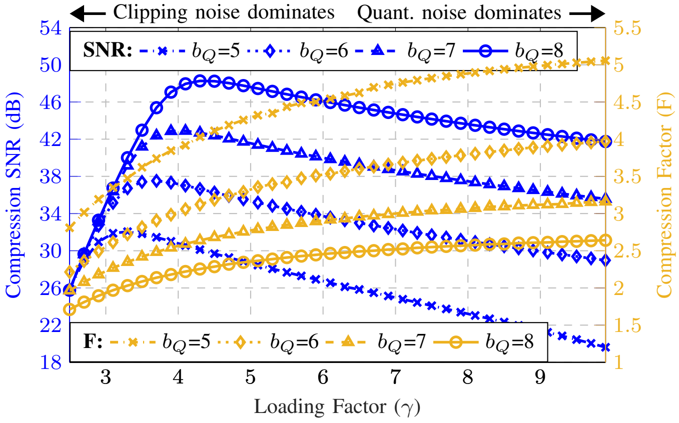

Fig. 3. Compression SNR and compression factor evaluated with DL LTE for varying loading factors and for $b_Q = 5$ to 8 bits in the quantizer.

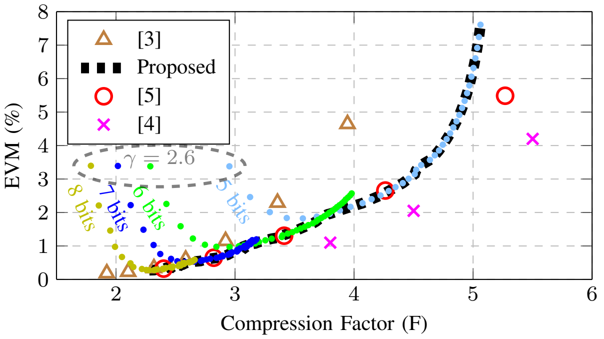

Fig. 4. Performance of the FSC methods, in terms of EVM and F.

IV. Conclusion

This paper proposed the introduction of adjustable gains into the LPC-based FSC scheme previously presented in [5]. The new mechanism combines adjustable gains to a fixed quantizer and a fixed Huffman dictionary. The proposed method allows fine regulations between the achievable compression factor and the corresponding compression SNR or EVM. Ultimately, the proposed FSC method could be used in FH networks to adapt the compression and distortion to specific values, in accordance to the FH capacity.

References

[1] "Common Public Radio Interface: (eCPRI) Specification v1.0," 2017.

[2] J. Bartelt *et al.*, "Fronthaul and Backhaul Requirements of Flexibly Centralized Radio Access Networks," *IEEE Wireless Communications*, vol. 22, no. 5, pp. 105–111, October 2015.

[3] B. Guo *et al.*, "LTE/LTE-A Signal Compression on the CPRI Interface," *Bell Labs Technical Journal*, vol. 18, no. 2, pp. 117–133, Sept 2013.

[4] H. Si *et al.*, "A Novel and Efficient Vector Quantization Based CPRI Compression Algorithm," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 8, pp. 7061–7071, Aug 2017.

[5] L. Ramalho *et al.*, "An LPC-Based Fronthaul Compression Scheme," *IEEE Communications Letters*, vol. 21, no. 2, pp. 318–321, Feb 2017.

[6] N. S. Jayant and P. Noll, *Digital Coding of Waveforms: Principles and Applications to Speech and Video*. Prentice-Hall, 1990.

[7] M. Bernhard *et al.*, "Analytical and Numerical Studies of Quantization Effects in Coherent Optical OFDM Transmission with 100 Gbit/s and Beyond," *ITG Symposium on Photonic Networks*, pp. 34–40, May 2012.

[8] 3GPP TS 36.101, "Evolved Universal Terrestrial Radio Access (E-UTRA); User Equipment (UE) radio transmission and reception," 2014. [Online]. Available: http://www.3gpp.org/dynareport/36101.htm