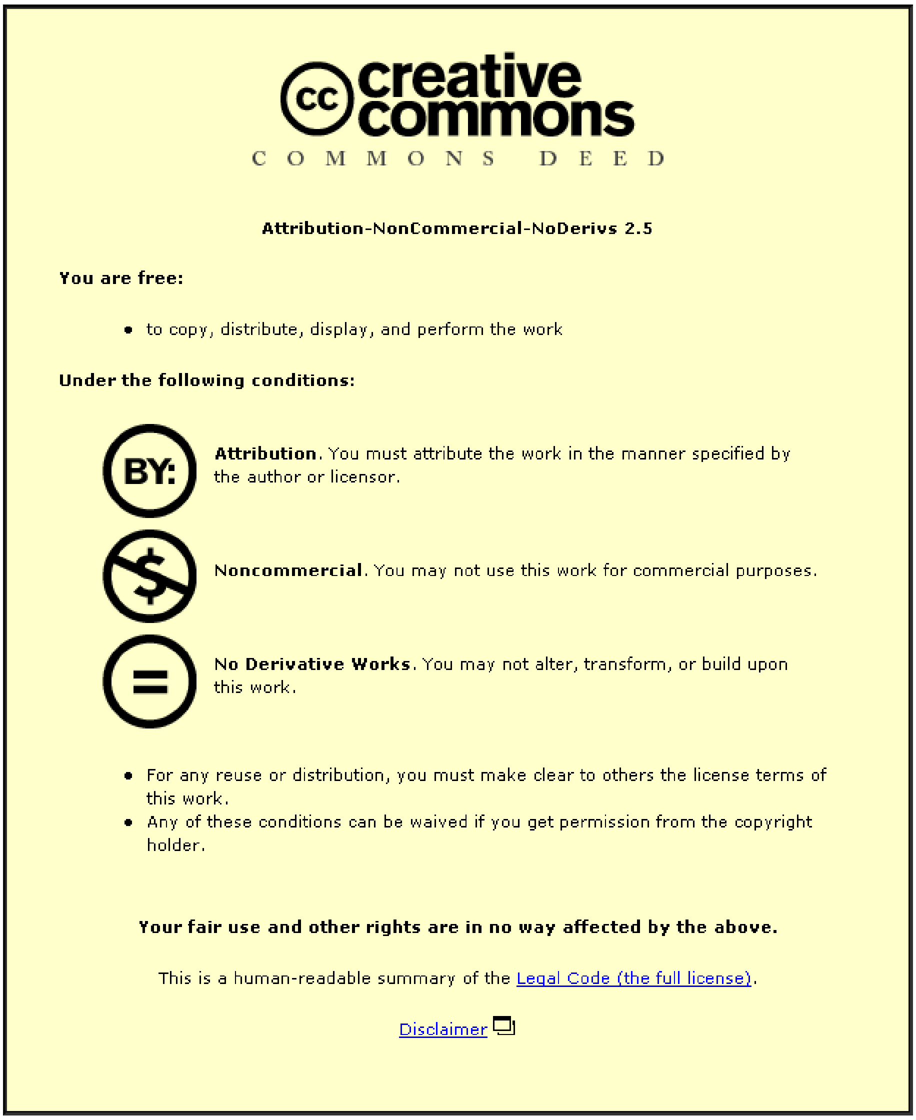


A system for on-line compression of high speed network measurements

Konstantinos G. Kyriakopoulos*
Electronic and Electrical Engineering,
Loughborough University
E-mail: elkk@lboro.ac.uk
*Corresponding author

David J. Parish
Electronic and Electrical Engineering,
Loughborough University
E-mail: d.j.parish@lboro.ac.uk

Abstract: Measuring various metrics of high speed and high capacity networks produces a vast amount of information over a long period of time, making the conventional storage of the data practically inefficient. Such metrics are derived from packet level information and can be represented as time series signals. Thus, they can be analyzed using signal analysis techniques. This paper looks at the Wavelet transform as a method of analyzing and compressing measurement signals (such as delay, utilization, data rate etc.) produced from high-speed networks. A live system can calculate these measurements and then perform wavelet techniques to keep the significant information and discard the small variations. An investigation into the choice of an appropriate wavelet is presented along with results both from off-line and on-line experiments. The quality of the decompressed signal is measured by the PSNR and a comparison of compression performance is presented against the lossless tool bzip2.

Keywords: wavelets; compression; computer networks; measurements.

1 INTRODUCTION

This paper is motivated by the need to measure the performance of high-speed communication networks of the future and particularly of the UKLight experimental network. The UKLight initiative is a 10 Gbps, high capacity research network facility that interconnects JANET, the UKs research and educational network, with NetherLight and StarLight research networks in Amsterdam and Chicago respectively.

The purpose of the UKLight venture is to create an international experimental testbed, giving researchers the opportunity to access facilities located on different continents and participate in Grid-nature projects where huge data sets are transfered between distant sites. Additionally, researchers can deploy in the testbed altered and enhanced versions of transport and network layer protocols and examine their impact on the network.

The MASTS (Measurement and Analysis in all Scales of Time and Space) project has been initiated to provide a traffic monitoring system for the UKLight network capable of recording data at various time scales and replying to real-time queries. At the heart of this project is the need to develop an efficient method of on-line analysis and reduction of information, spanning the early stages of the network's operation and onwards.

Monitored network data allows network managers and operators to gain valuable insight into the health and status of a network, and if interpreted correctly, can assist in planning upgrades and remedial action to keep the network operating in a near optimum manner. Whilst such data is useful for real-time analysis, there is often a need to post-process historical network performance data. Such analysis is useful if ongoing long-term problems or if more detailed analysis of a previous situation needs investigation.

Monitoring and measuring various metrics of high speed and high capacity networks produces a vast amount of information over a long period of time. These metrics describe the status and performance of the network in terms of utilization, congestion, packets lost, etc. The storage of the produced measurements is practically inefficient, as it requires a large number of storage devices. This itself significantly increases the cost for measuring high-speed networks. For this reason, there is a need to derive an efficient method of data reduction in order to reduce and store the enormous amount of measurements describing the monitored traffic.

Compression of monitored performance data is an attractive option to reduce long-term storage requirements. However, selection of a suitable compression mechanism is a non-trivial activity. Conventionally, lossless compression algorithms have been utilised for this purpose, however it is generally accepted that higher compression ratios are achievable using lossy algorithms. These of course cannot support perfect regeneration of the original data. However, if the important and significant elements of the original data are preserved, lossy compression becomes attractive.

The work described in this paper considers this issue and proposes the use of the Wavelet Transform as a first step in the compression of a time-series of delay or utilization measurements. Using the Wavelet Transform in the manner described in the paper allows useful further compression to be obtained over competing lossless algorithms, whilst providing controlled degradation of the signal. The degradation ensures that the important characteristics of the source data are retained along with the signal energy in the regenerated signal. Furthermore, the use of the Wavelet Transform allows later analysis of the measurement data to be implemented efficiently.

The rest of the paper is structured as follows. In section 2, the advantages of wavelet analysis over other transformation techniques are discussed along with information on wavelet analysis implementation. In section 3, the methodology of the wavelet coefficient thresholding and compression algorithm are discussed. In section 4, eight different mother wavelets are compared against each other in order to choose an appropriate wavelet in terms of efficiency, quality of results and energy preservation. Section 5 presents results from off-line experiments and the proposed compression algorithm is compared against the lossless compression tool bzip2. Section 6 discuses a monitoring platform for live calculation and compression of measurements and practical results are presented by compressing sampled traffic recorded from a live network. Finally, conclusions and ideas for future work are given in section 7.

2 INTRODUCTION TO WAVELET ANALYSIS

2.1 Related Work

So far, wavelets have been generally used to detect network performance problems. They have been applied to traffic rate signals in order to infer the time scale associated with the dominant RTT through the examination of the energy function of the detail coefficients (Huang et al. (2001)). They have also been used for de-noising one-way delay signals in order to detect shared congestion between different flows by Kim et al. (2004a). Barford et al. (2002) show that wavelet filters are quite effective at exposing the details and characteristics of ambient and anomalous traffic. Kim et al. (2004b,c) analyze the correlation of destination IP addresses of outgoing traffic at an egress router. Based on statistical historical margins, estimated after using wavelet analysis, sudden changes are detected.

2.2 Wavelet Analysis Advantages

The Heisenberg uncertainty principle suggests that it is impossible to know the exact frequency and the exact time of occurrence of this frequency in a signal but it is possible to obtain the frequency bands that exist in a time interval. In other words, there is a trade off between the resolution of the time and the frequency domain.

In contrast with other techniques that use a constant window size to analyze a section of a signal (for example DCT, STFT), wavelet analysis has the benefit of varying the window size. This means that wavelets can efficiently trade frequency resolution for time resolution or vice versa. For this reason, wavelets can adapt to various time-scales and perform local analysis. In essence, wavelets can reveal both the forest and the trees (Misiti et al. (2004) Agbinya (1996)).

The local analysis feature of the wavelets provides the additional benefit of approximating an examined signal compactly, i.e. with few coefficients. There has been a lot of research conducted in image and speech signal processing, describing the suitability of wavelets in compression applications. Shapiro (1993); Chang et al. (2000); Agbinya (1996) are a few examples.

The finite nature of the wavelet can describe local features of the signal better than the infinite length of a sinusoid. Thus, another attribute of wavelet analysis is the ability to detect characteristics of non-stationary signals, i.e. stochastic (random) signals whose statistical properties change with time. Most interesting signals are non-stationary signals.

2.3 Multi-resolution Signal Decomposition

The wavelet analysis transforms a given signal s[n] of n samples into n/2 approximation (scaling) and n/2 detail (wavelet) coefficients. The approximation coefficients represent the smoothed version of the signal (low frequency bands), while the detail coefficients represent the detailed version (high frequency bands).

The basic idea of wavelet analysis is that an average of two samples of signal s at scale j produces an approximation coefficient at the next higher scale j+1. Whereas, the difference between those samples produces a detail coefficient at scale j+1.

Thus, for a specific scale, approximation coefficients are associated with the averages, whereas detail coefficients represent the change of averages. In the case of the simplest wavelet function, Haar, this can be expressed as:

$$\alpha_{j+1} = \frac{S_j[1] + S_j[2]}{2} \qquad d_{j+1} = \frac{S_j[1] - S_j[2]}{2} \qquad (1)$$

Wavelet Analysis can be used as a Multi-resolution Signal Decomposition (MSD) tool, decomposing a signal into scales of varying time and frequency resolution.

Initially, the first level (or scale) of decomposition of the multi-resolution analysis takes place. The same process can be applied again on the resulting (n/2) approximation coefficients yielding n/4 detail and approximation coefficients and so on for higher scales. The total group of detail coefficients from scale 1 up to J and the approximation coefficients at scale J compose the wavelet decomposition tree at scale J.

3 METHODOLOGY

Wavelet analysis is not a compression tool but a transformation to a domain that provides a different view of the data that is more eligible to compression than the original data itself.

Fig. 1 shows the flow chart of the following algorithm for analysis and compression of each examined signal. First the signal is transformed into the wavelet domain (see section 3.1) where the threshold is calculated (see section 3.2) and applied (see section 3.3) on the wavelet coefficients. Then, the thresholded coefficients are normalized (see section 3.4) so that each coefficient can be stored in 1 byte and finally run length encoding is applied (see section 3.5).

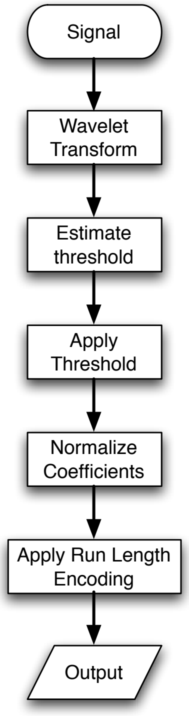


Figure 1: Methodology flow chart

3.1 Quadrature Mirror Filter

In 1988, Mallat (1998) developed a Fast Wavelet Transform (FWT) algorithm that became well known in the signal processing community as a two channel subband coder using conjugate filters or quadrature mirror filters (QMF) (see Fig. 2).

For the decomposition phase, two finite impulse filters are used. The high-pass filter (HPF) produces the detail coefficients and the low-pass filter (LPF) the approximation of the signal. The output of the LPF becomes the input of the next pair of filters for further decomposition at higher scales.

The QMF pair divides the input signal into low-frequency and high-frequency components. The dividing frequency is between 0 Hz and the maximum frequency of the analyzed signal, which according to the Nyquist theorem is half of the data sampling frequency.

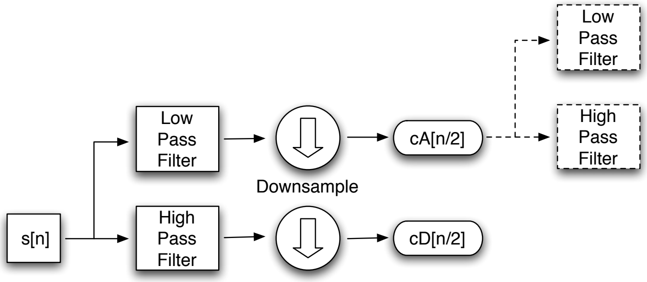


Figure 2: Quadrature Mirror Filter Bank

3.2 Thresholding

As was mentioned in section 2.3, detail coefficients actually reveal the changes of the average. In other words, detail coefficients with large magnitude represent significant changes in the original signal. These large magnitude detail coefficients need to be preserved because they represent important characteristics of the signal and they should be kept in order to preserve the quality of the signal (see Chang et al. (2000) Kaur et al. (2002))

On the other hand, many of the wavelet coefficients produced from the wavelet analysis have an absolute value close to zero. These small coefficients are likely attributable to small variations of the signal and contain a small percentage of the signal's total energy. These small coefficients can be discarded without a significant loss in the quality of the signal and more importantly of the interesting features. Thus, a threshold is required below which all coefficients will be discarded.

Gupta and Kaur (2002) proposed an adaptive thresholding technique that is calculated from the absolute value of the wavelet coefficients. This scheme is not based on signal denoising but rather tries to statistically identify significant coefficients. Specifically, the standard deviation (σ) and mean (μ) of the absolute value of non-zero detail coefficients are first calculated. If the standard deviation is larger than the mean, then the threshold is set to two times the mean ($2 * \mu$), otherwise it is equal to the mean minus the standard deviation ($\mu - \sigma$). Thus, the threshold, T, can be expressed mathematically as:

$$T = \begin{cases} 2 * \mu, & if \sigma > \mu \\ \mu - \sigma, & if \sigma \leq \mu \end{cases} \tag{2}$$

Previous work by the authors (Kyriakopoulos and Parish (2007)) has compared the above thresholding technique against another widely used technique proposed by Birge and Massart (see Misiti et al. (2004)). The results of the comparison showed that the Birge-Massart threshold is not well-suited for our needs as it can not restrain the mean square error (MSE) at acceptable levels while the scale of analysis increases. Indeed, the compression ratio is very high for large analysis scales but this makes the reconstructed signals very distorted.

3.3 Applying the Threshold

The hard and soft thresholding techniques are two of the most common ways of applying a threshold. In both cases coefficients with absolute value smaller than the threshold are set to zero. With the hard threshold all coefficients with absolute value larger than the chosen threshold are kept intact, while with the soft threshold, positive coefficients are reduced and negative increased by the value of the threshold. In other words coefficients that have an absolute value larger than the threshold are shrunk towards zero, thus naming this method shrinkage as described by Donoho (1995). The hard and soft threshold functions are given below and are illustrated in Fig. 3 for threshold T=0.5.

$$f_{hard}(x) = \begin{cases} x, & if \quad |x| \geq T \\ 0, & otherwise \end{cases} \tag{3}$$

$$f_{soft}(x) = \begin{cases} x - T, & if \quad x > 0 \quad and\, |x| \geq T \\ 0, & if \quad |x| < T \\ x + T, & if \quad x < 0 \quad and\, |x| \geq T \end{cases} \tag{4}$$

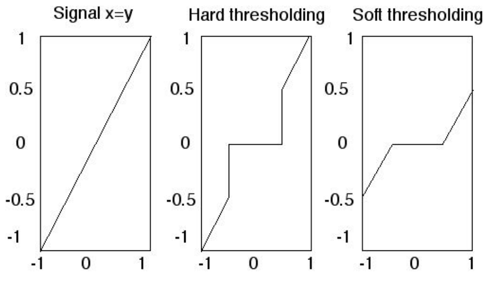


Figure 3: Hard and soft thresholding functions for threshold T=0.5

Depending on the examined signal and the application, the threshold technique has to be chosen carefully. Kaur et al. (2002); Yoon and Vaidyanathan (2004) use soft thresholding to avoid abrupt artefacts produced from the discontinuity of hard thresholding (see Fig. 3) and to provide smoothness to the reconstructed signal.

Downie and Silverman (1998) recommend using the hard thresholding technique rather than the soft technique as it gives less MSE. The hard thresholding method performs better as it offers a more precise reconstruction of signals that contain abrupt changes but also retains a large amount of the original signal energy. Previous work by the authors (Kyriakopoulos and Parish (2006)) has shown that the hard threshold is more appropriate for compressing computer network measurements as it gives considerably better quality results. The hard threshold was applied in the experiments described in this work.

3.4 Normalization

In order to improve the way that data are stored, normalization of the coefficients takes place. The aim is to use just 8 bits to store each coefficient. However, with an 8 bit variable only 256 values can be stored (values 0...255) or 127 values (i.e. 2^7 bits) saving one bit for the sign of the wavelet coefficient. Thus, first the coefficient values have to be normalised using the following formula:

$$norm(x) = round\Big(\frac{x - min}{max - min} * scalingfactor\Big) \tag{5}$$

where x is a coefficient, min is the minimum value that appears in the array of coefficients and max the maximum. The scaling factor in this case is 127, which is the maximum value of a signed number that can be stored in 1 byte. In order to avoid the detail coefficients being skewed by the larger values of the approximation coefficients, the normalization process is applied separately for the detail and for the approximation coefficients.

3.5 Run Length Encoding

The simplest version of the run length encoding (RLE) algorithm replaces a sequentially repetitive symbol with the symbol itself followed by a number that indicates how many times the symbol should be repeated. However, this simple version of RLE expands single symbols into a pair of symbol-run length.

In order to avoid this shortcoming, a more sophisticated RLE implementation utilises a run length that is used only for symbols that appear more than 2 times. This method has beneficial effect only for symbols that appear 3 or more times. However, the RLE limitation persists for symbols that appear sequentially for just two instances and expand into symbol-symbol-run length triples. This version of RLE is preferred to the previous one, because the probability of a symbol appearing twice is significantly lower than appearing once.

3.6 Examined Data

For the wavelet comparison investigation and the off-line experiments (Sections 4 and 5 respectively) thirty delay and thirty data rate signals of 1024 points were used.

The delay signals were measured on a research test bed. Traffic generators were used to emulate a time of day profile similar to that of a commercial network. Delay signals are usually smooth with sudden discrete bursts spread over the data.

The data rate signals are from a real commercial network that generates around 25 TB of data per day and has data rates between 300 Mbps and 1.4 Gbps. In a window size of 2 seconds 30000 unique IP addresses may be observed in that network.

4 WAVELET COMPARISON

The mother-wavelet chosen to analyze the network measurement signals is of prime importance as some wavelets offer better quality than others. However, there is no wavelet that gives the best results for all kinds of signals.

Regarding speech and audio signals, Agbinya (1996); Najih et al. (2003), among others, have found that for better reconstruction quality wavelets with many vanishing moments should be utilized, as they introduce less distortion and concentrate more signal energy in the approximation coefficients. Wavelets with many vanishing moments are described with many coefficients in the scaling and wavelet functions, thus increasing the computation overhead of the wavelet transform, the complexity of the algorithm and most importantly the output file size.

Eight wavelets were chosen and compared against each other in order to find out, firstly which one keeps more energy in the wavelet coefficients and secondly which one offers better reconstruction results after thresholding with the technique described in section 3.2.

The following wavelets were compared: Haar, Meyer, Biorthogonal 3.9 and Daubechies D4, D6, D8, D10, D12. The index of Daubechies wavelets indicates the number of coefficients. The number of vanishing moments each Daubechies wavelet has is half of the number of coefficients, i.e. D4 has 2 vanishing moments.

4.1 Wavelet Energy Attributes

The first task of estimating the percentage of kept energy was split into two parts. The signals described above were used as input and all eight wavelets were examined one by one. In the first part, the flow chart in Fig. 4 was followed in order to estimate the percentage of preserved energy in the approximation coefficients. The threshold was such that only the approximation coefficients were kept and all detail coefficients filtered out. The percentage of kept energy for each signal was calculated by the L^2-norm recovery criterion (Misiti et al. (2004)) given in percentage terms by:

$$PERFL2 = 100 * \left(\frac{l^2\text{-}norm(CXC)}{l^2\text{-}norm(C)} \right) \quad (6)$$

where CXC are the coefficients of the thresholded decomposition and C the coefficients of the original decomposition. The l^2-$norm$, $|x|$, of a vector $x = (x_1, x_2, x_3)$ is defined as:

$$|x| = \sqrt{{x_1}^2 + {x_2}^2 + {x_3}^2} \quad (7)$$

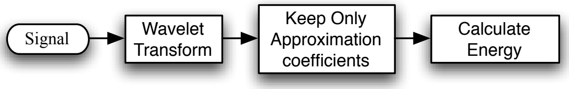


Figure 4: Flow chart for Energy calculation in approximation coefficients.

Tables 1 and 2 show the percentage of kept energy in the approximation coefficients at level 6 of the decomposition for delay and data rate signals respectively.

For delay signals there is a significant difference in the percentage of preserved energy among wavelets. For data rate signals the percentage of kept energy is very similar among wavelets. However, in both cases, the Haar wavelet has the worst score in keeping the energy.

Even though the reconstruction quality depends largely on the approximation coefficients, it also depends to some extent on the detail coefficients preserved from the suggested thresholding algorithm.

In the second part of the energy preservation investigation, the flow chart in Fig. 5 was followed in order to estimate the percentage of preserved energy after applying the proposed threshold. In this case both approximation and detail coefficients are preserved. Again, the percentage of kept energy for each signal was calculated by equation (6).

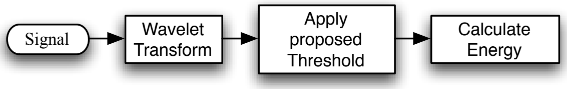


Figure 5: Flow chart for Energy calculation in thresholded coefficients

Tables 3 and 4 show the percentage of kept energy in the coefficients after applying the threshold described in section 3.2 at level 6 of decomposition for delay and data rate signals respectively.

For delay signals there is a significant difference in the percentage of preserved energy among wavelets. For data rate signals the percentage of kept energy is very similar among wavelets. However, in both cases, the Haar wavelet has the worst score in keeping the energy.

Table 3 shows that the Haar wavelet ranks third in keeping the energy of delay signals after applying the proposed threshold. This is a significant improvement in comparison to the results of Table 1. For data rate signals (see Table 4) the inclusion of detail coefficients adds a very small percentage to the preserved energy as most of the energy is included in the approximation coefficients. In both cases, the average preserved energy is very high. It should be noted that including detail coefficients is essential for preserving the signal characteristics.

Table 1: Percentage of Preserved Energy in Approximation Coefs. for Delay Signals at Level 6

Wavelet	Haar	D4	D6	D8	D10	D12	Meyer	Bio 3.9
Min. %	31.0	34.6	35.9	38.4	41.1	42.2	61.0	35.9
Max. %	95.7	97.2	97.6	97.6	97.7	97.8	98.6	95.3
Avg. %	63.34	69.68	71.58	73.59	74.61	76.08	86.22	73.38
Rank	8	7	6	4	3	2	1	5

Table 2: Percentage of Preserved Energy in Approximation Coefs. for Data Rate Signals at Level 6

Wavelet	Haar	D4	D6	D8	D10	D12	Meyer	Bio 3.9
Min. %	99.1	99.4	99.5	99.5	99.5	99.6	99.8	99.1
Max. %	99.8	99.9	99.9	99.9	99.9	99.9	100.0	99.9
Avg. %	99.73	99.82	99.84	99.86	99.87	99.88	99.93	99.79
Rank	8	6	5	4	3	2	1	7

Table 3: Percentage of Preserved Energy in Thresholded Coefs. for Delay Signals at Level 6

Wavelet	Haar	D4	D6	D8	D10	D12	Meyer	Bio 3.9
Min. %	97.06	96.88	96.57	95.95	96.70	96.22	96.60	96.91
Max. %	99.97	99.92	99.93	99.93	99.82	99.94	99.91	99.97
Avg. %	98.44	98.37	98.29	98.33	98.34	98.32	98.99	98.50
Rank	3	4	8	6	5	7	1	2

Table 4: Percentage of Preserved Energy in Thresholded Coefs. for Data Rate Signals at Level 6

Wavelet	Haar	D4	D6	D8	D10	D12	Meyer	Bio 3.9
Min. %	99.88	99.92	99.92	99.93	99.93	99.94	99.96	99.89
Max. %	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00
Avg. %	99.99	99.99	99.99	100.00	99.99	100.00	100.00	99.99
Rank	2	2	2	1	2	1	1	2

4.2 Wavelet Quality Attributes

For the second task of examining which wavelet offers best reconstruction results, two phases were followed; the decomposition and the reconstruction phases (see Fig. 6). In order to measure the quality of the reconstructed signals with different wavelets, the inverse wavelet transform was applied on the produced coefficients after thresholding.

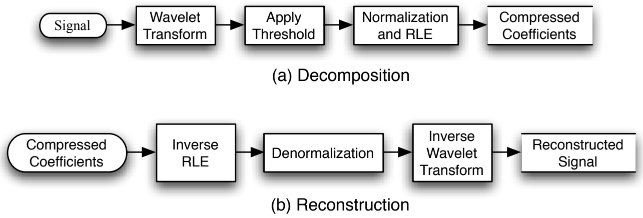
The quality of the reconstruction signal was compared with the original by using the PSNR value calculated from

$$PSNR = 10 * log\left(\frac{MAX^2}{MSE}\right) \tag{8}$$

where MAX is the maximum value of the original signal and MSE is the mean square error calculated from

$$MSE = \frac{1}{N}\sum_{i=0}^{N-1}\left|x_i - \bar{x}_i\right|^2 \tag{9}$$

where x_i is the i^{th} sample from the original signal, $\bar{x}_i$ is the i^{th} sample of the reconstructed signal and N is the total number of samples.



(a) Decomposition

(b) Reconstruction

Figure 6: Flow chart for wavelet quality comparison: (a) Decomposition, (b) Reconstruction

Tables 5 and 6 show the average PSNR value after reconstruction at level 6 for the thirty delay and data rate signals respectively.

As can be inferred from the tables, wavelets with more vanishing moments do not provide higher PSNR values for the reconstructed signals. Even though the Haar wavelet

Table 5: PSNR Values for Delay Signals after Reconstruction with Different Wavelets

Wavelet	Haar	D4	D6	D8	D10	D12	Meyer	Bio 3.9
Min. PSNR	32.8	32.1	31.6	31.5	31.3	31.7	31.0	31.4
Max. PSNR	50.1	46.8	45.4	45.5	45.2	44.8	46.0	45.8
Avg. PSNR	39.6	38.2	37.6	37.5	37	37	37.1	37.3
Rank	1	2	3	4	7	8	6	5

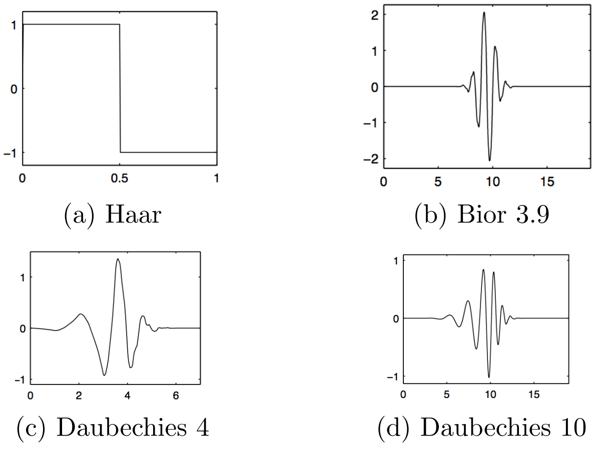
Table 6: PSNR Values for Data Rate Signals after Reconstruction with Different Wavelets

Wavelet	Haar	D4	D6	D8	D10	D12	Meyer	Bio 3.9
Min. PSNR	34.0	34.4	34.2	34.2	33.6	34.0	34.5	34.0
Max. PSNR	57.2	59.9	59.0	59.1	58.0	58.7	57.9	60.4
Avg. PSNR	53.4	51.6	51.7	49.7	50.8	51.5	53.0	50.4
Rank	1	3	4	8	6	5	2	7

misses some detail and does not always represent detail at all resolution scales, it ranks first on average in both delay and data rate signals in terms of quality of results.

Specifically, for delay signals, the Haar wavelet gives the best PSNR for all signals and for data rate signals, it is the best option on average. The better performance of the Haar wavelet can be explained if we consider the fact that the value of the produced coefficients simply indicates how similar the selected wavelet is to the corresponding part of the examined signal (Misiti et al. (2004)). The closer the similarity the greater the value of the coefficient. Wavelets have different characteristics and their forms can be seen in Fig. 7.

For most cases, the wavelets are not as similar to the examined signals as is the Haar wavelet. For this reason, the produced coefficients have smaller values which means that more coefficients are thresholded during the compression process. This does not happen as frequently when using the Haar wavelet. As a result, the reconstruction quality in those cases is better when using the Haar wavelet.



(a) Haar

(b) Bior 3.9

(c) Daubechies 4

(d) Daubechies 10

Figure 7: Examples of different wavelet forms.

The Haar wavelet is the simplest wavelet algorithm that can give perfect reconstruction and has the following additional advantages Nievergelt (1999):

- It is conceptually simple
- Fast
- Memory efficient
- and is exactly reversible without producing edge effects.

Because of the above advantages and the best reconstruction results offered by the Haar wavelet, it was chosen for the off-line and practical experiments discussed in Sections 5 and 6.

5 OFF-LINE EXPERIMENTS

Off-line experiments were run with the signals, described in section 3.6, fed as input to the wavelet thresholding algorithm (Fig. 6a). Experiments were run on an Intel Core Duo machine at 1.83Ghz and were aimed at producing results regarding the compression ratio (C.R.) and PSNR of the de-compressed signal. For a level-6 wavelet analysis and compression procedure, each signal of 1024 measurements required on average 33 ms to complete.

From experimental experience, for both types of signals, PSNR values less than 35 dB loose some of the important signal characteristics while PSNR values less than 30 dB are not acceptable. Along with numerical scores, figures are provided to demonstrate how the decompressed signal compares to the original. Additionally, the error is given in the graphs separately for easier judgement.

5.1 Delay Signals

For delay signals, the C.R. between the original and the compressed signal begins at 7.5 for level 1 and stabilizes around 17 from level 6 and above. PSNR stabilizes after level 7. The PSNR and C.R. average values of delay signals for all analysis levels are given in Table 7.

Fig. 8 shows delay signal 30, which is a good representative of most delay signals, before and after the compression. Because the two signals are very similar, the error between them is also provided for better judgment (lower line). The signal is decomposed at level 10 and the reconstruction quality is 37.85 dB while the C.R. is 13.7.

Fig. 9 shows signal 10, which differs from most delay signals because it is very bursty. Due to this fact, it includes many high frequency components and it is the hardest delay signal to compress. The algorithm is able to adjust to this burstiness and produce a reconstructed signal with very low error. PSNR is 44.3 dB and C.R. is 5.65.

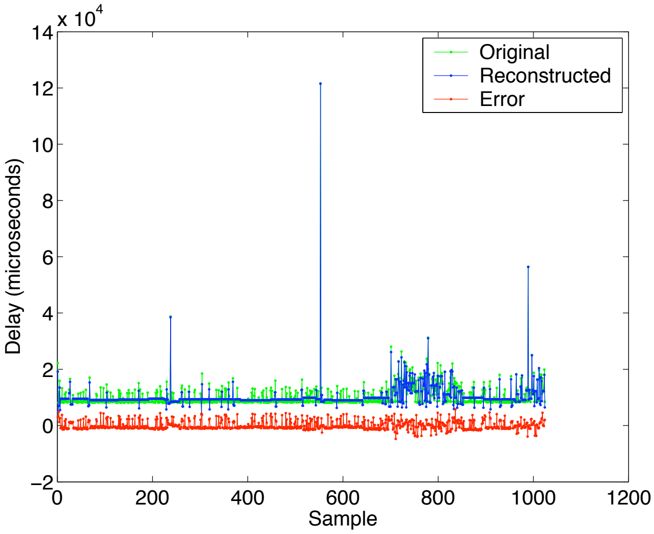


Figure 8: Delay signal 30 decomposed at level 10, PSNR= 37.85 dB and C.R.= 13.7.

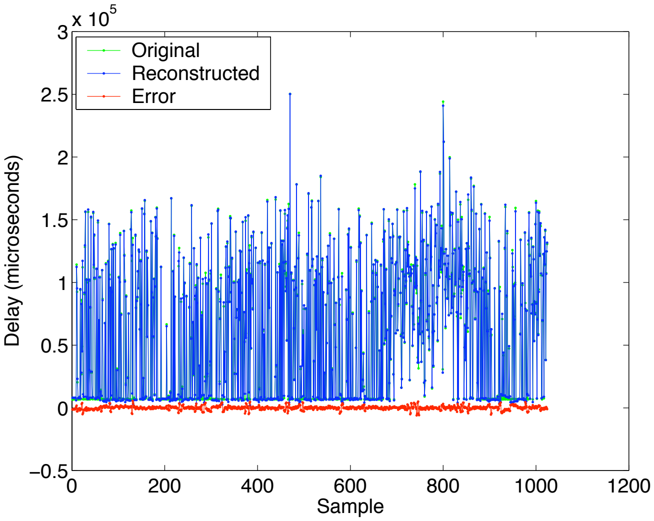


Figure 9: Delay signal 10 decomposed at level 10, PSNR= 44.3 dB and C.R.= 5.65.

5.2 Data Rate Signals

For the data rate signals there is no significant increase in the C.R. as the level of decomposition increases. The C.R. range is between 10.5 and 11.2 (see Table 8) in contrast with the wider range (7.5 to 17.4) for delay signals. This happens because data rate signals have a lot of high frequency components and the proposed algorithm tries to keep a lot of detail coefficients in order to preserve the signal quality. The PSNR and C.R. average values for all levels of analysis for data rate signals are given in Table 8.

Fig. 10 shows an example of a data rate signal analyzed at level 5. The reconstructed signal has very good quality, giving PSNR=56.9 dB and C.R.=10.53.

Fig. 11 shows a more interesting case of a data rate signal. This signal includes a spike, which is kept intact after the compression. A characteristic of the proposed algorithm is that it detects the spike as a more interesting feature than the rest of the signal. As a result, the algorithm's first priority is to preserve this characteristic and after this, the rest of the signal.

This actually means that the large coefficients representing the big spike will shift the threshold to higher values, thus, filtering out more small magnitude coefficients. This is the reason why PSNR is around 35 dB and there is higher error for the rest of the signal in comparison to signal 20 (Fig. 10), which is similarly bursty but with no spike. The high reconstruction error is accompanied by a high C.R.= 26.57.

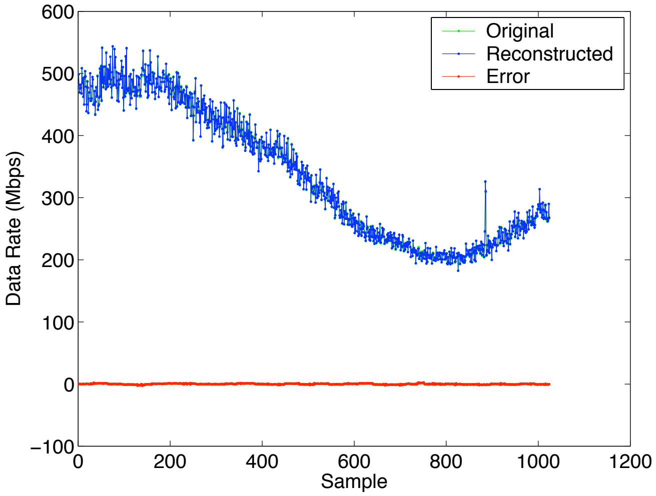


Figure 10: Data rate signal 20 analyzed at level 5, PSNR = 56.9 and C.R.= 10.53.

Fig. 12 and Fig. 13 compare the C.R. results of the suggested wavelet technique against bzip2. It is interesting to examine the results from the wavelet transform against a non-transform compression technique. For that reason bzip2 was chosen, as it is an excellent lossless compression tool and a natural option when no other obvious alternatives exist for compressing network measurements. In comparison to other transform based compression techniques, wavelets are preferred due to the advantages described in

Table 7: PSNR and C.R. Values for Delay Signals After Reconstruction

Level	L1	L2	L3	L4	L5	L6	L7	L8	L9	L10
Min. PSNR	37.0	34.3	33.5	33.1	33.1	32.8	32.8	32.8	32.8	32.8
Max. PSNR	54.6	53.1	52.6	51.8	50.9	50.1	49.7	48.5	48.5	48.5
Avg. PSNR	44.3	42	41	40.5	40	39.6	39.2	39.2	39.2	39.2
C.R.	7.5	10.7	13.6	15.5	16.5	17	17	17	17.2	17.4

Table 8: PSNR and C.R. for Data Rate Signals After Reconstruction

Level	L1	L2	L3	L4	L5	L6	L7	L8	L9	L10
Min. PSNR	39.0	37.2	36.4	36.4	35.4	34.0	34.0	33.2	32.7	32.7
Max. PSNR	58.6	58.1	57.6	58.0	57.5	57.2	56.7	55.6	55.0	52.8
Avg. PSNR	56.3	55.6	55.4	55.2	54.9	53.4	49	43.9	42.1	39.4
C.R.	10.5	10.8	11	11.1	11.2	11.2	11.1	10.9	10.7	11

Section 2.1.

Each examined signal is located on the x-axis. The y-axis shows the file size in bytes. On average, for delay signals (Fig. 12) the suggested method (WT) achieves compression 6.5 times more than bzip2 with the best score being 11 times and the worst score 2.3 times. For data rate signals (Fig. 13) the average compression is 4.7 times more than bzip2 with the best score being 12 times and the worst 4 times.

6 ON-LINE EXPERIMENTS

6.1 Practical Implementation

The full algorithm has been implemented in CoMo (Iannaccone et al. (2004)). CoMo is a passive monitoring platform developed for the purpose of monitoring network links at high speeds and replying to real-time queries regarding network statistics.

CoMo has various modules that calculate one or more network measurements. There are some built-in modules that come with CoMo but it can also accept third-party

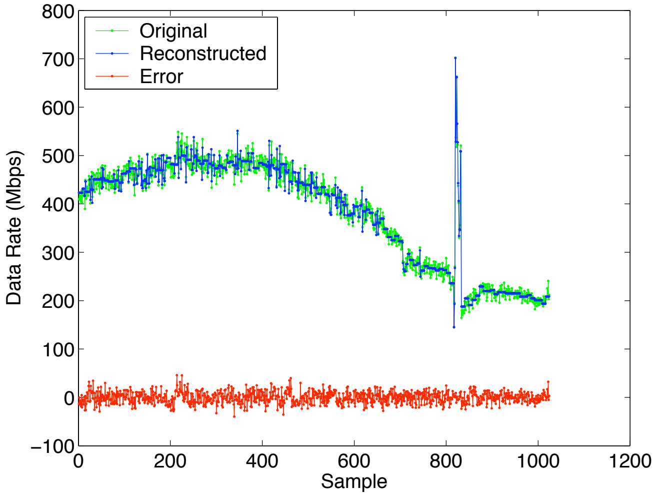


Figure 11: Data rate signal 16 analyzed at level 5, PSNR = 35.4 dB and C.R.= 26.57.

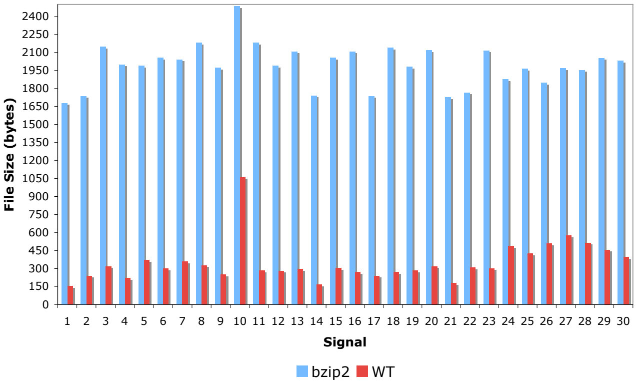


Figure 12: Compression performance of the wavelet algorithm against bzip2 for delay signals

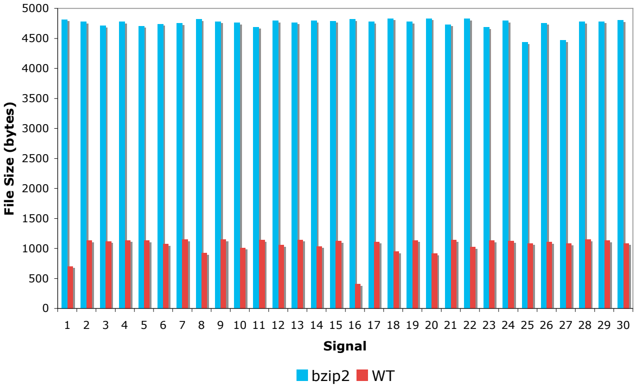


Figure 13: Compression performance of the wavelet algorithm against bzip2 for data rate signals

modules. The proposed algorithm can be imbedded in the modules and compress the calculated measurements. When CoMo receives a query, the information is first decompressed and then shown to the end user.

CoMo's first aim is to capture the packets that arrive at the monitoring end. Then, the module-dependent statistics of the packets (for example packet length) are binned per second (or in any other specified time interval) thus producing measurements per second (for example data rate).

In order to examine the wavelet compression scheme in practice, two CoMo modules are compared, one with, and the other without compression. Both modules count bytes per second (data rate) and store a timestamp for each captured block of 1024 samples. More specifically, packet information is aggregated per second and the resulting value represents one element of the 1024 signal samples. Thus, each block of samples spans a duration of 17.06 minutes. However, only one module performs compression and passes the block of samples through the compression algorithm while the next block of 1024 samples is being captured and calculated.

As each data rate sample is generated in 1 second (an aggregated measurement per second), the analysis and compression processes are actually *independent of the bandwidth* of the link being monitored. There is a window of around 17 minutes for the analysis and compression phases to complete before the next block of 1024 measurements takes its turn. So, calculation time of the algorithm is not an issue for the on-line implementation of the compression algorithm, as a few milliseconds of processing time are adequate for each block of measurements.

For the same reasons, the described system is scalable to high data rates as long as the measurements are aggregated. So, no matter what the bandwidth of the monitored link is, there will always be 1024 samples in each data block and each block will always span 17.06 minutes. The compression and quality results of the reconstructed signal only depend on the original signal characteristics and attributes (for example if it is bursty or not) and not on the bandwidth of the monitored link.

The window size of 1024 was chosen as it is a power of two and wavelet algorithms are more efficient and simple with signal lengths of power of two. There is no window size better than others for all situations as the window size is dependent on the characteristics and the type of each examined signal. However, as a general rule, the bigger the window-size the more the compression and the reconstructed error are.

6.2 Practical Results

The experiment lasted for 8 days and CoMo was monitoring traffic recorded on a research group's live network. This network supported up to 20 users using a mixture of machines running Windows, Linux and Mac operating systems. Activities ranged from standard office and email applications to real time media and games sessions and dedicated research related operations including remote log retrieval. The overall traffic mix would be expected to show considerable variation.

Times of very high network usage could occur and should lead to spikes in the data traces. On a public network, such spikes would be of considerable importance as they indicate periods of high load and their further investigation may indicate if changes to the network are required. On a research network such as UKLight, analysis of the spikes may indicate what impact a particular experiment has had on a given network path. Preservation of these spikes for later analysis is therefore of great importance.

The overall achieved compression for the whole signal is 34.5 times. The wavelet analysis, thresholding and compression takes place up to level 6 of decomposition. Fig. 14 presents a segment of 34 minutes from the 8 day duration experiment. This signal is characterized by discrete bursts of data rate. Some bursts have amplitude of 70 kB/s while others are half that size or less. The reconstruction keeps intact the peaks and smoothes out the relatively small variation of the signal. PSNR for that segment is 55.9 dB.

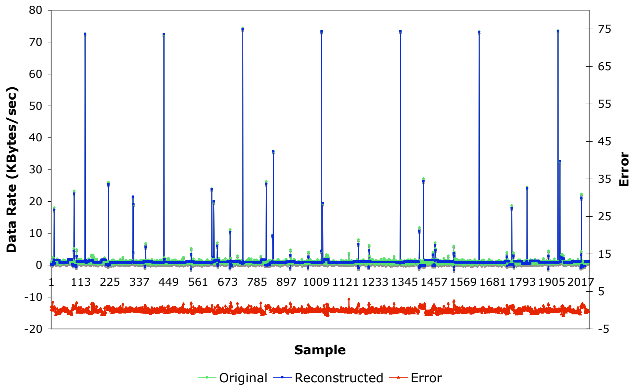


Figure 14: Data rate signal of 34 minutes compressed live by CoMo. Error is given on the secondary y-axis on the right (lower line). PSNR = 55.9 dB.

The 34 minute signal is actually analyzed as two blocks of 1024 samples, i.e. two blocks of 17 minute duration. Thus, it's statistics and energy scaling behavior are presented in two different tables and two different figures. Table 9 shows the statistics for the first part and Table 10 for the second part. Respectively Fig. 15 shows the energy scaling behavior for the first part and Fig. 16 for the second.

For the 34 minute signal, as can be inferred from the tables, the relative errors for mean, standard deviation and energy are very small. Fig. 15 shows a maximum energy divergence of 0.2 dB at scale 5 of decomposition. In the graph the difference looks significant but this is due the small scale in this particular figure. Fig. 16 shows a maximum energy divergence of 0.12 dB at scale 5 of decomposition. For all other scales, for both Fig. 15 and 16 the energy of the reconstructed signal is almost identical to the energy of the original signal.

Table 9: Percentage Relative Error of Three Statistics for the First Part of 34 Minute Signal

Statistic	Original	Reconstructed	Percentage Error
Mean	1364.51	1365.52	0.074%
Standard deviation	4511.83	4482.70	-0.64%
Energy	22218574	21959308	-1.16%

Table 10: Percentage Relative Error of Three Statistics for the Second Part of 34 Minute Signal

Statistic	Original	Reconstructed	Percentage Error
Mean	1392.78	1392.86	0.005%
Standard deviation	4913.17	4879.3	-0.689%
Energy	26079122.39	25747639	-1.27%

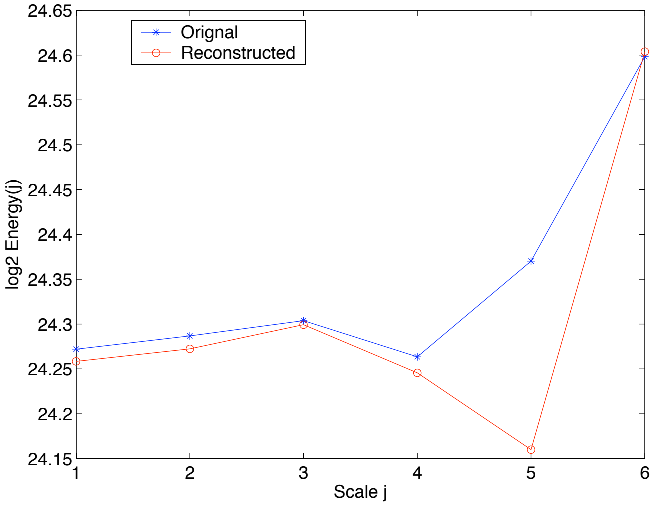


Figure 15: Energy behavior across scales for first half of 34 minute on-line signal.

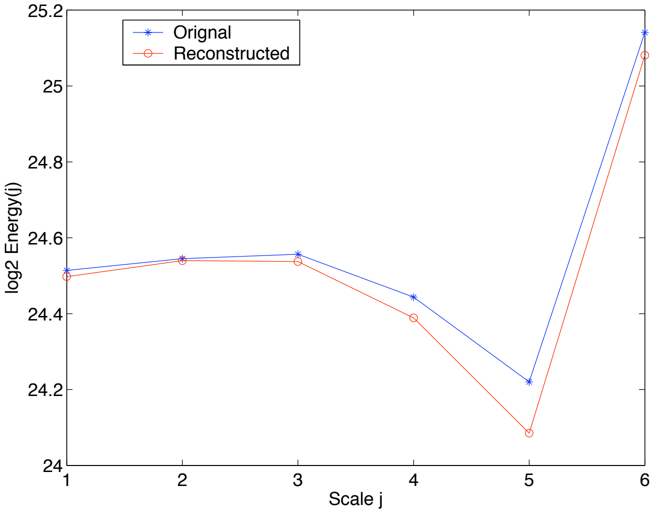


Figure 16: Energy behavior across scales for first half of 34 minute on-line signal.

Another segment of duration 17 minutes is presented in Fig. 17. The two high peaks at the beginning and end of the signal are preserved and the small variation from sample 150 up to 400 is smoothed. The medium amplitude peaks in the region of sample 400 up to 950 are also preserved but with more distortion in comparison to the highest peaks. PSNR for this signal is 34.4 dB.

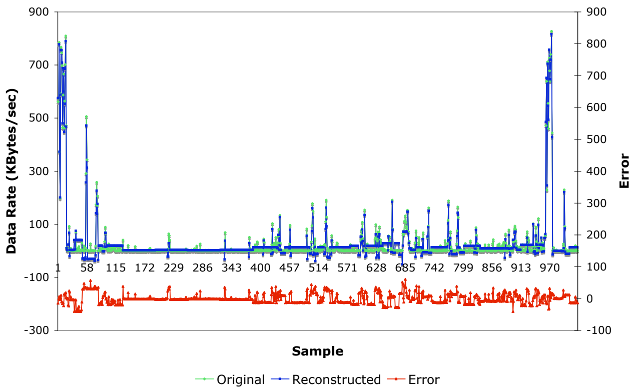


Figure 17: Data rate signal of 17 minutes compressed live by CoMo. Error is given on the secondary y-axis on the right (lower line). PSNR = 34.4 dB.

Table 11 shows the percentage relative error for mean, standard deviation and the energy between the reconstructed and the original signal. The first two order statistics are preserved with very high accuracy. Fig. 18 shows the energy scaling behavior of the original and reconstructed signals. The behavior is almost identical.

7 CONCLUSIONS

This paper proposes the use of wavelet analysis techniques along with a wavelet coefficient thresholding method for compressing computer network measurements such as data rate and delay. Even though the compression is lossy, the important characteristics of the examined signal are preserved along with its mean, standard deviation and energy.

Table 11: Percentage Relative Error of Three Statistics for 17 Minute Signal

Statistic	Original	Reconstructed	Percentage Error
Mean	30712.799	30556.62	-0.5%
Standard deviation	107992.10	106930.84	-0.98%
Energy	12605569898	12367912005	-1.88%

In order to increase the compression, the detail characteristics are smoothed out by discarding the corresponding detail coefficients.

An evaluation of various wavelets with increasing vanishing moments was presented in order to determine which wavelet is more appropriate for performing the analysis. From experimental results, the Haar wavelet is found to be the best option as it offers the best results in terms of quality and compression ratio. Wavelets with many vanishing moments increase the computation overhead, the complexity of the algorithm and the output file size.

The thresholding technique proposed by Gupta and Kaur (2002) restricts the C.R. from exceeding a specific range as the level of decomposition increases. As a result, the reconstructed signals have a limited distortion that does not increase no matter what the depth of decomposition.

For delay signals, the results show an average PSNR of 39.6 dB with a compression ratio of 17 at level 6 of decomposition. In comparison to the compression performance of bzip2 the proposed technique compresses on average 6.5 times more. For data rate signals, the average PSNR is 54.9 and the compression ratio is 11.2 at level 5. On average, the achieved compression is 4.7 times more than bzip2.

Additionally, the algorithm can adapt and perform well in the case of a very bursty signal, such as the case of Fig. 9. No parameter setting is required prior to the signal analysis and compression.

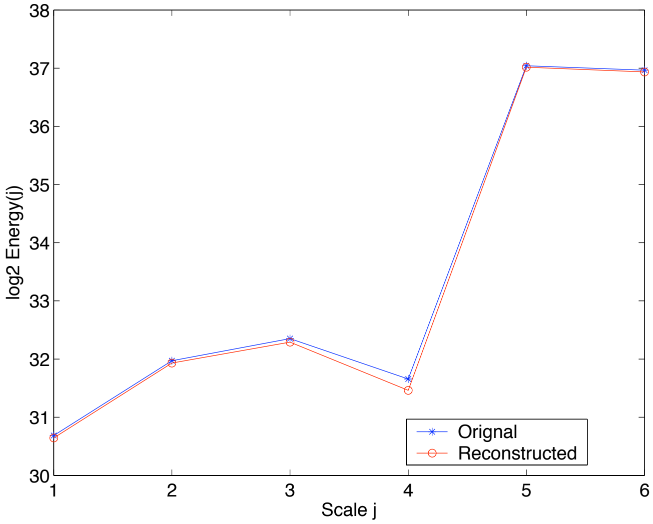


Figure 18: Energy behavior across scales for 17 minute on-line signal.

However, some improvements could be made in how the algorithm deals with the threshold in cases that spikes occur in an *already bursty signal* such as that in signal 16 (Fig. 11). This would lead to more control over the quality of the reconstructed signal and more consistent PSNR values. A possible solution would be the embedded zerotree wavelet (EZW) transform proposed by Shapiro (1993). The EZW can achieve high compression ratios by predicting insignificant wavelet coefficients across different scales.

For capturing, compressing and storing packet characteristics, such as packet length, the algorithm could be enhanced to perform faster due to the small inter-arrival rate of the packets in a 10 Gbps network. This could be done by implementing in a dedicated hardware (FPGA) a faster algorithm of wavelet analysis, such as the lifting scheme described by Sweldens and Schröder (1996).

REFERENCES

Agbinya, J. I. (1996). Discrete wavelet transform techniques in speech processing. In *Proceedings of the 1996 IEEE Region 10 TENCON - Digital Signal Processing Applications Conference*, volume 2, pages 514–519, Perth, Aust. IEEE. 2.2, 4

Barford, P., Kline, J., Plonka, D., and Ron, A. (2002). A signal analysis of network traffic anomalies. In *Proceedings of the 2nd Internet Measurement Workshop (IMW 2002)*, pages 71–82. 2.1

Chang, S. G., Yu, B., and Vetterli, M. (2000). Adaptive wavelet thresholding for image denoising and compression. *IEEE Transactions on Image Processing*, 9(9):1532 – 1546. 2.2, 3.2

Donoho, D. L. (1995). De-noising by soft-thresholding. *IEEE Transactions on Information Theory*, 41(3):613 – 627. 3.3

Downie, T. R. and Silverman, B. W. (1998). The discrete multiple wavelet transform and thresholding methods. *IEEE Transactions on Signal Processing*, 46(9):2558–2562. 3.3

Gupta, S. and Kaur, L. (2002). Wavelet based image compression using daubechies filters. In *8th National conference on communications, I.I.T.* 3.2, 7

Huang, P., Feldmann, A., and Willinger, W. (2001). A non-intrusive, wavelet-based approach to detecting network performance problems. In *Proceedings of the ACM SIGCOMM Internet Measurement Workshop*, pages 213–227. 2.1

Iannaccone, G., Diot, C., McAulley, D., Moore, A., Pratt, I., and Rizzo, L. (2004). The CoMo white paper. Technical report, INTEL. Page last visited 22/08/05. 6.1

Kaur, L., Gupta, S., and Chauhan, R. C. (2002). Image denoising using wavelet thresholding. In Chaudhuri, S., Zisserman, A., Jain, A. K., and Majumder, K. L., editors, *ICVGIP*. Allied Publishers Private Limited. 3.2, 3.3

Kim, M. S., Kim, T., Shin, Y., Lam, S. S., and Powers, E. J. (2004a). A wavelet-based approach to detect shared congestion. In *ACM SIGCOMM Computer Communication Review*, volume 34, pages 293–305. 2.1

Kim, S. S., Reddy, A. L. N., and Vannucci, M. (2004b). Detecting traffic anomalies through aggregate analysis of packet header data. In Mitrou, N., Kontovasilis, K. P., Rouskas, G. N., Iliadis, I., and Merakos, L. F., editors, *NETWORKING*, volume 3042 of *Lecture Notes in Computer Science*, pages 1047–1059. Springer. 2.1

Kim, S. S., Reddy, A. L. N., and Vannucci, M. (2004c). Detecting traffic anomalies using discrete wavelet transform. In Kahng, H.-K., editor, *ICOIN*, volume 3090 of *Lecture Notes in Computer Science*, pages 951–961. Springer. 2.1

Kyriakopoulos, K. G. and Parish, D. J. (2006). Wavelet compression of network delay measurements. In *Procceedings of the 3rd International Confernce on Telecommunications and Computer Networks*, pages 115–119. IADAT. 3.3

Kyriakopoulos, K. G. and Parish, D. J. (2007). Wavelet compression techniques for computer network measurements. In *Proceedings of the Fourth IASTED International Conference*, pages 109–115. IASTED. 3.2

Mallat, S. (1998). *A wavelet tour of signal processing*. Academic Press. 3.1

Misiti, M., Misiti, Y., Oppenheim, G., and Poggi, J. (1997-2004). Matlab wavelet toolbox. Technical report, The MathWorks, Inc. 2.2, 3.2, 4.1, 4.2

Najih, A. M. M., Ramli, A. R., Ibrahim, A., and A, R, S. (2003). Comparing speech compression using wavelets with other speech compression schemes. In *Proceedings of Student Conference on Research and Development, SCORED*, pages 55–58. 4

Nievergelt, Y. (1999). *Wavelets Made Easy*. Birkhauser. 4.2

Shapiro, J. M. (1993). Embedded image coding using zerotrees of wavelet coefficients. *IEEE Transactions on Signal Processing*, 41(1):3445–3462. 2.2, 7

Sweldens, W. and Schröder, P. (1996). Building your own wavelets at home. In *Wavelets in Computer Graphics*, pages 15–87. ACM SIGGRAPH Course notes. 7

Yoon, B.-J. and Vaidyanathan, P. P. (2004). Wavelet-based denoising by customized thresholding. In *International Conference on Acoustics, Speech and Signal Processing*, volume 2, pages 925–928. 3.3