EXPERIMENTAL ANALYSIS OF WAVELET TRANSFORMS FOR ESTIMATING PSK SYMBOL RATE

Kenneth L. Holladay
Southwest Research Institute®
San Antonio, TX 78238
kholladay@swri.edu

Kay A. Robbins
University of Texas at San Antonio
San Antonio, TX 78249
krobbins@cs.utsa.edu

Abstract

For automated surveillance applications, estimating the symbol rate of an unknown digital communication signal is an important step in the analysis process. Several papers have investigated using the wavelet transform in symbol rate estimation algorithms. Due to its complexity, closed form analysis of performance is often limited, and simulations may not include practical factors such as carrier frequency offset or symbol pulse shaping. This paper uses an automated statistically based test framework to investigate the performance of the wavelet transform against PSK signals with parameters that span a realistic portion of the High Frequency (HF) signal space. The analysis identifies signal and algorithm parameters that affect performance. We also demonstrate that accurate metrics for estimating the probability of failure/success under realistic operating conditions are available for the db6 wavelet.

Keywords: Symbol Rate, Wavelet, PSK

Introduction

Estimation of the symbol rate of an unknown digital communication signal is an important early analysis step for most automated surveillance applications. Since later analysis stages often rely on the availability of an accurate symbol rate, a practical symbol rate analyzer should provide a reliable estimate of accuracy as well as the symbol rate value. The performance of symbol rate algorithms is heavily dependent on the method of signal encoding, so many automated applications determine signal type upstream of symbol rate detection. This paper assumes upstream signal type detection and focuses on Phase Shift Key (PSK) signals.

Many symbol rate estimation algorithms have been developed, analyzed, and published [1][2][3][4][5]. Most of these algorithms have a similar structure consisting of two computational stages. The first stage develops a feature space by applying a transform to the digitized signal. The goal of the transform is to locate and emphasize symbol transitions. The second stage analyzes this feature space to calculate a periodic value that is the estimate of the symbol rate.

Traditional first stage transforms applied to PSK signals include the derivative of the phase angle, the magnitude of the signal squared, and autocorrelation. In recent years, there has been significant interest in using the wavelet transform, and several authors have explored this potential [6][7][8].

In practice, it is difficult to quantitatively compare the relative performance of these algorithms based solely on the published information. An author may concentrate on a specific aspect of an algorithm and, due to space limitations, omit key implementation details required to independently reproduce the results. In some cases, simplifying assumptions required to perform a closed form mathematical analysis may not reflect realistic operating environments. In addition, few publications characterize the operation of an algorithm with respect to all of the parameters that may affect its performance. Typical simulations fix some parameters and omit boundary values for others.

Symbol rate detection is representative of a large class of practical signal processing problems in which analytical evaluation is an inadequate predictor of how these algorithms perform in practice. The complexity of the interacting factors that determine efficacy suggests that techniques pioneered in experimental algorithmics [9] could yield useful insight into performance. Experimental algorithmic methods emphasize several key concepts including clearly specifying the testing goal, articulating parameter variations and any hidden algorithmic factors, carefully constructing large test sets that span the problem domain, and systematically evaluating the performance results using statistical techniques.

This paper describes the application of experimental algorithmics methodology to the evaluation of a published wavelet transform based symbol rate estimation algorithm. The goal was to determine whether this algorithm could accurately estimate the symbol rate of an unknown signal in the context of an existing automated surveillance system. We begin by reviewing the algorithm to establish its adjustable parameters. Next, we describe the limitations of the original test domain and the extensions in both factor types and ranges necessary to more closely match the expected application domain. Finally, using these goals and data sets, we present three experiments designed to validate the published behavior, select the appropriate wavelet and scale factors, and evaluate performance within the target environment.

To manage the large data sets required in the algorithmics setting, we implemented a test framework [10] in MATLAB [11] with functions for generating test signals, automatically cycling them through the algorithm under test, and analyzing the results.

Designing and Implementing the Algorithm

Chan et al. [6] present a detailed mathematical description of the wavelet algorithm. Their paper also includes a brief description of the implementation used in their simulations, which follows the two-stage format. The first stage applies the continuous wavelet transform using multiple scale values. The squared magnitudes of the resulting coefficients are summed across the scales to produce a vector that remains time coherent.

A MATLAB code snippet to implement the algorithm of Chan et al. is:

```
c = cwt( x, scales, wavelet );
y = c.*conj(c);
if numel(scales) > 1
    y = sum(y);
end
```

Here `x` is the analytical (complex) signal vector, `c` is a matrix containing the complex coefficients of the resulting transform, and `y` is the first stage output vector.

There are two algorithm specific parameters in the first stage of the wavelet algorithm: the wavelet and the scales. The original algorithm specifies a Haar wavelet with scales of 4, 6, and 8. The Haar wavelet is a popular choice because its simple structure allows closed form analysis of the wavelet transform across a symbol change and it is computationally efficient. However, since the intended test set includes signals with pulse-shaped symbols [12], it was neither analytically nor intuitively obvious that the Haar wavelet was sufficiently matched with realistic input to adequately emphasize the symbol transitions. Therefore, one experiment included a large number of available mother wavelet families [13], as well as an expanded number and combination of scales.

The wavelet algorithm also has two hidden parameters in the first stage: the sampling rate and length of the signal vector. Many published studies assume a sampling rate that is an integral multiple of the symbol rate. More importantly, these studies often assume a fixed number of symbols or worse, a fixed number of symbol changes. Based on expected limitations and desired characteristics of the target automated surveillance system, we selected a fixed sampling rate of 8000 complex samples per second and a maximum signal duration of approximately 2 seconds.

The description of the algorithm's second stage in [6] states only that the frequency bin associated with the first peak in the Fast Fourier Transform (FFT) (of the first stage output) is an estimate of the symbol rate. However, with no definition of what constitutes a peak, and without published code, reproducing this behavior was a challenge.

The hidden peak search algorithm present in many published studies of symbol rate algorithms turns out to be an important practical factor in performance. The process of selecting a peak varies from simply choosing the value with the largest magnitude to sophisticated application of thresholds, filtering, and weighting prior to value selection [14]. The peak search step is critical since it not only affects the result of the overall algorithm, but it may introduce additional parameters that must be characterized.

Attempting to avoid the peak search ambiguities, our first implementation simply selected the peak with the largest magnitude. Visualization of failed test cases revealed circumstances where the correct peak was obvious, but was not the maximum. In addition, at low symbol rates significant harmonic peaks appeared in the output of the Fourier transform, and since the harmonics occurred at multiples of the symbol rate, they could be exploited to improve performance.

The final design of the second stage for the current wavelet study included several adjustable parameters. Since the sample rate was fixed, we used a fixed FFT size that resulted in a minimum bin resolution of 2 Hz. To reduce spectral leakage, we applied a Hamming window prior to each FFT. To enhance the desired peaks, we used a standard overlap and add technique. This also necessitated a low pass filter on the final FFT accumulation to remove any trend at the low frequencies. Table 1 details the complete set of algorithm parameters and values used for the experiments described in this paper. The last parameter in the list is a peak threshold value used to detect the presence of harmonics.

Initial testing indicated that while this implementation enhanced the second stage performance over simply selecting the largest peak, the enhancement was not very sensitive to specific parameter values, and the enhancement scaled uniformly across variations in the first stage parameters. This was an important observation since it implied that the performance of the second stage would not mask the performance of the first stage.

Table 1 – Algorithm Parameters

Parameter	Stage	Values used
Wavelet	1	Haar, db6, sym6, coif3, bior1.3, bior5.5, mexh
Scales	1	2, 4, 6, 8, & combinations
Sample rate	1	8000 Hz
Signal duration	1	1, 2 second
Minimum cutoff	2	20 symbols per second (baud)
FFT Size	2	4096 points
Overlap in FFT	2	50 %
Windowing	2	Hamming
Low pass filter	2	16 point moving average
Harmonic peak threshold	2	40 % of maximum peak value

Bounding the Input Signal Domain

The research work that started this investigation [15] focused on non-cooperative reception of digitally modulated signals in the High Frequency (HF) communication band. These signals are typically limited to channels with a bandwidth of 5 kHz or less. For this evaluation, we excluded signals that contain multiple data streams within a single channel or that change modulation mode as a function of time. We further restricted the channel parameters to additive white Gaussian noise (AWGN). Even with these restrictions, the problem space is quite large.

Table 2 lists some of the common signal, channel, and receiver properties that can affect the performance of a blind symbol rate estimation algorithm. The values used represent expected realistic ranges for the target surveillance system. The test sets also include complexities that arise in practice but that are often not considered in published studies, such as pulse shaping and symbol rates that are not divisors of the sample rate.

Table 2 - HF Digital Signal Properties

Property	Valid Values in HF	Values Used
Modulation type	FSK, MSK, PSK, DPSK, OQPSK, QAM, ASK	PSK
Pulse shape	None, raised cosine, root raised cosine, Gaussian, and others	None, Raised cosine (RC), root raised cosine (RRC)
Excess bandwidth (rolloff)	Limit: 0.00 to <1.00. Typical: 0.10 to 0.35	0.1, 0.2, 0.35
Symbol rate	Typical: 10 to 2400 symbols per second	50, 100, 300, 1280, 2400
Symbol states	2, 4, 8, 16	2, 4, 8
Signal to noise ratio (SNR)	Practical range: 0 to 60 dB	9, 12, 16, 20, 40
Frequency offset from baseband	Possible range: 0 to 500 Hz	0, 50, 100

After fixing the modulation type to PSK (assuming the existence of upstream modulation identification), we have an experimental test set with six multiple-valued test signal factors. For each combination of these factors, we generated 10 random variations of noise and symbol content, yielding an ensemble of 15,750 test signals. The test framework manages variable signal duration by generating a test signal file that exceeds the maximum length of interest and then extracting sections as needed.

Running the Experiments

A test framework organizes the signal parameters, algorithm parameters, and test results in a structured environment that fosters rapid data manipulation. The framework includes functions to evaluate performance metrics, create lists of test files matching specified signal or performance criteria, and to produce visualizations such as group plots, probability density curves, and receiver operating characteristic (ROC) plots [16].

All experiments used MATLAB scripts to direct the framework functions. As it processes each test signal through the algorithm, the framework collects not only the symbol rate estimate, but also standard statistical measures of dispersion and central tendency for the first and second stage outputs, including the mean, and standard deviation. For our analysis, a symbol rate estimate was deemed correct when the error was less than 2% of the actual value.

Validating Original Behavior

The intent of the first experiment was to create a reasonable reproduction of the original simulation conditions of [6] to ensure that our implementation performed comparably. However, since the published conditions included constraints that did not correlate directly with our expected deployment environment, some changes and compromises were required.

Limiting the test signal set satisfied two of the original constraints: no pulse shaped symbols and a baseband carrier offset of zero. A third constraint, related to the sampling frequency, was intentionally altered. Originally, the carrier frequency was fixed to be an integral multiple of the symbol rate and the sampling frequency was four times the carrier frequency. We used a fixed sample rate of 8000 Hz and included multiple noise levels.

The final constraints imposed on the original test signals were that every symbol transition contained a phase change, and each test signal contained exactly 100 symbols. A more realistic simulation of the target environment uses a fixed signal duration and randomly generated symbols.

Theoretical analyses of symbol rate algorithms often rely on the symbols being independently and identically distributed (IID). For slow symbol rates, this assumption may be difficult to satisfy in practice. By viewing the symbol distribution of a signal as a sampling problem (IID relative to the number of symbol states), we can apply statistical methods to estimate the number of symbols needed to ensure compliance with the assumed normal distribution. Using a proportional sampling model [17] we have:

$$n = z^2_{\left(1-\alpha/2\right)} \frac{p\left(1-p\right)}{r^2}$$

Here n is the number of symbols, p is the reciprocal of the number of symbol states, r is the desired accuracy, and z is the unit normal distribution with confidence factor α. For a binary (2 symbol states) signal, we would need about 380 symbol samples to achieve 5% accuracy with a 95% confidence level. Satisfying this condition requires a reasonable 0.16 seconds at 2400 symbols per second, but a lengthy 7.6 seconds at 50 symbols per second. Therefore, for the first experiment, we limited the symbol rate to values above 300, and we fixed the input duration to 1 second.

The results of this experiment were encouraging. The algorithm achieved 100% correct responses for the 450 test signals with an average symbol rate error of 0.04%

Selecting Wavelets and Scales

The goal of the second experiment was to develop some insight about how the wavelet type and combination of scales might affect the algorithm performance against a broader range of signals. We expanded the test set to include pulse shaped symbols, lowered the symbol rate range to 50 baud, but only included the highest signal to noise ratio (40 dB) to yield a test set of 1050 signals. These were run against 49 different wavelets with 9 scale combinations each.

Table 3 – Percent Correct Symbol Rate for Experiment 2

Scale	Haar	sym6	db6	bior5.5	mexh
[2]	61.2	58.4	55.8	52.5	72.6
[4]	63.7	65.2	66.6	66.4	51.0
[6]	61.7	69.3	70.5	65.8	40.7
[8]	52.0	57.2	56.0	57.4	42.3
[2 4]	63.6	66.6	69.2	67.2	62.3
[4 6]	63.0	73.1	72.6	71.3	44.4
[6 8]	57.6	64.6	62.9	64.1	42.5
[2 4 6]	62.9	73.0	72.7	69.0	50.6
[4 6 8]	59.9	69.0	67.8	68.9	40.6

Table 3 summarizes the overall performance of four of the better-performing wavelets plus the original Haar wavelet. The shaded areas mark the highest percent correct for each scale. Note that combining scales does not always improve performance, and in many cases, it degrades performance.

Fig. 1 displays the data from Table 3 in a radar plot. The radial position indicates the performance (% correct) for each of the scale factors in the nine angular positions.

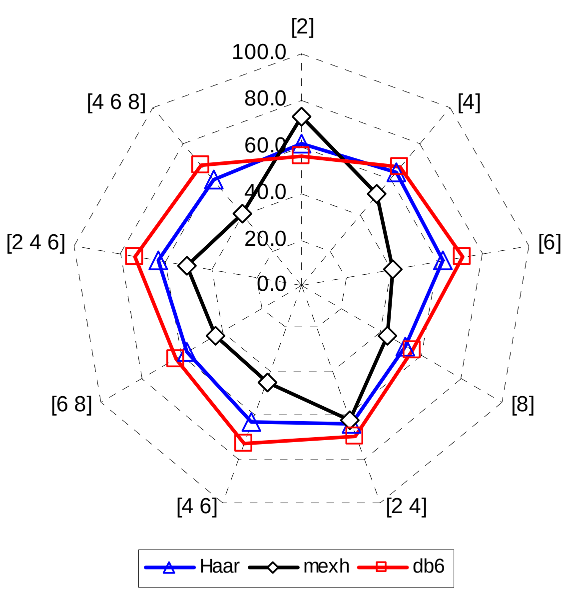

Fig. 1 – Percent correct symbol rate estimation for three wavelets

Fig. 1 clearly shows that the db6 wavelet had the best overall performance at all scales except the smallest. Like Haar, db6 is a member of the Daubechies family. Fig. 2 compares the wavelet function for the Haar and db6 families. Notice that db6 more closely resembles the raised cosine symbol pulse often used in pulse shaping as shown in Fig. 3.

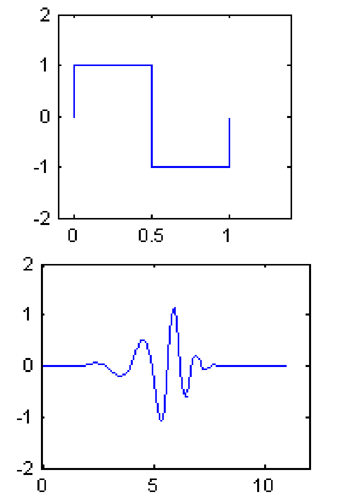

Fig. 2 –Wavelet function ψ for Haar (left) and db6 (right)

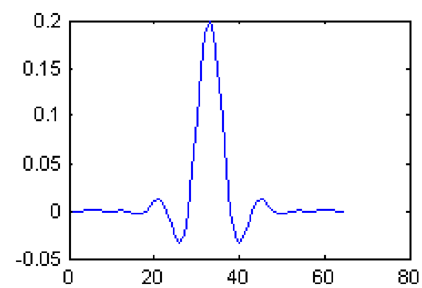

Fig. 3 –Raised cosine symbol pulse

Space does not permit publishing the complete set of statistical analysis data. Table 3 summarizes the performance for db6 and Haar using the scale for each with the best overall performance. It is readily apparent that while pulse shaping affects both, the db6 wavelet does better. Considering that both do poorly with raised cosine (RC) pulse shaping at low symbol rates, increasing the signal length may improve performance.

Table 4 – Percent Correct by Symbol Rate for Experiment 2

	Wavelet/Scale	Haar [4]			db6 [2 4 6]		
	Pulse shape	None	RC	RRC	None	RC	RRC
Symbol Rate	50	100.0	0.0	24.4	100.0	1.1	45.6
	100	100.0	12.2	42.2	100.0	1.1	100.0
	300	100.0	37.8	78.9	100.0	41.1	100.0
	1280	100.0	88.9	100.0	100.0	100.0	100.0
	2400	100.0	92.2	100.0	100.0	92.2	100.0

Evaluating Target Performance

The final experiment applies both the original Haar wavelet and the db6 wavelet to the full 15,750 test signals and extends the signal duration to two seconds. This test set also introduces a carrier offset factor, which simulates the target system where the process of converting a detected signal to baseband may not precisely center tune the signal.

As seen in Table 5, both wavelets achieved 100% correct response for non pulse shaped symbols, demonstrating immunity to both noise and carrier offset. However, the performance against pulse shaped symbols deteriorated, especially at the lower symbol rates.

Table 5 – Percent Correct by Symbol Rate for Experiment 3

Wavelet/Scale		Haar [4]			db6 [2 4 6]		
Pulse shape		None	RC	RRC	None	RC	RRC
Symbol Rate	50	100.0	0.2	5.4	100.0	2.2	18.9
	100	100.0	2.0	11.0	100.0	2.4	21.3
	300	100.0	20.2	51.8	100.0	13.7	21.3
	1280	100.0	95.8	100.0	100.0	99.8	100.0
	2400	100.0	99.3	100.0	100.0	98.9	100.0

Developing a Quality Metric

The performance achieved in experiment 3 would not qualify either wavelet algorithm for deployment. However, if there were an easily measured quality metric that could accurately indicate when the calculated symbol rate is correct, then these wavelet algorithms might be useful in conjunction with other algorithms.

To explore this possibility, we analyzed histograms of the magnitudes of the FFT bins used to select the symbol rate peak. We examined statistical measures of central tendency and dispersion.

Fig. 4 shows the derived probability density curves of the normalized mean of the FFT magnitude bins for signals correctly analyzed (passed) and incorrectly analyzed (failed). Note the small overlap area between the probability of the calculated symbol rate being correct (solid line) and the probability of it being incorrect (dashed line). For the db6 wavelet, the normalized mean appears to be an excellent quality measure.

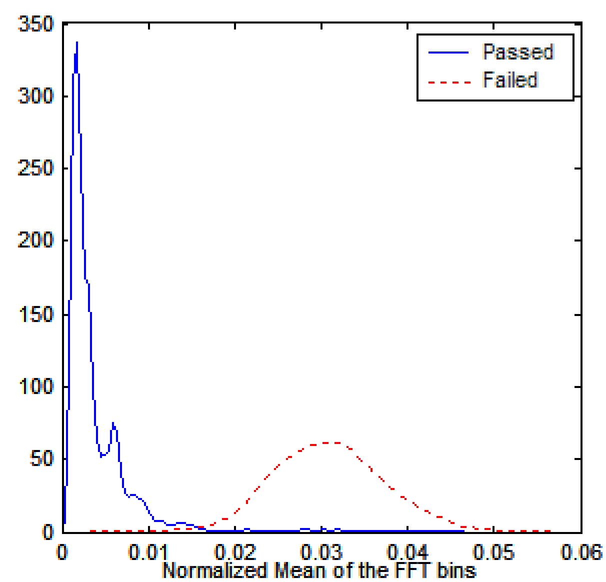

Fig. 4 – Probability density for db6 wavelet using normalized mean

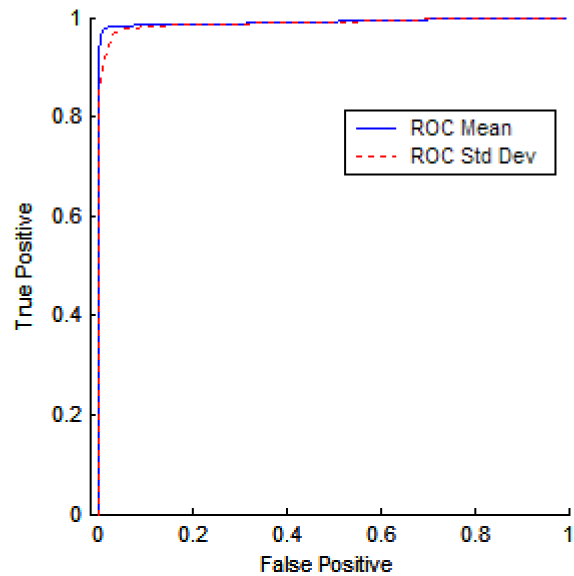

Fig. 5 – ROC curves for db6 wavelet using the normalized mean of FFT bins and the standard deviation of the FFT bins.

Fig. 5 shows the ROC curve for the distributions of Fig. 4. An example operating point might be a false positive rate below 1%. To achieve this, we select a threshold value for the normalized mean of 0.016, resulting in a true positive rate greater than 98%. For comparison, Fig. 5 also displays the ROC curve obtained using the standard deviation. While this also exhibits the desired rapid rise from the origin, its false positive rate increases slightly faster than the ROC curve obtained using the mean.

By contrast, the Haar wavelet performed dismally on realistic signal test sets. Fig. 6 shows corresponding ROC curves for the Haar. Using the same desired false positive rate of 1%, the means had a true positive rate of 4%, while standard deviation had a true positive rate of less than 6%.

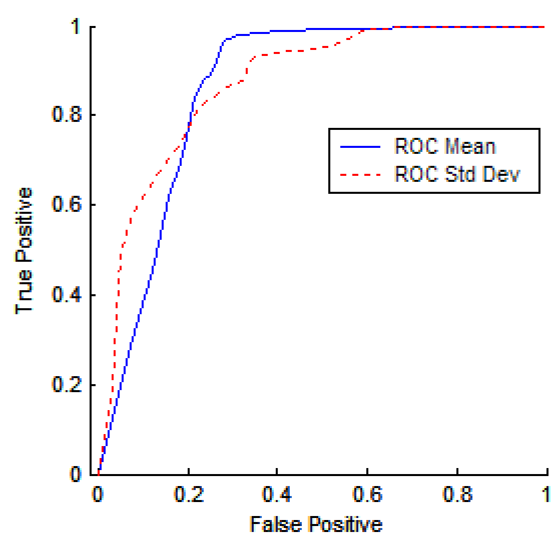

Fig. 6 – ROC for Haar wavelet using the normalized mean of FFT bins and the standard deviation of the FFT bins.

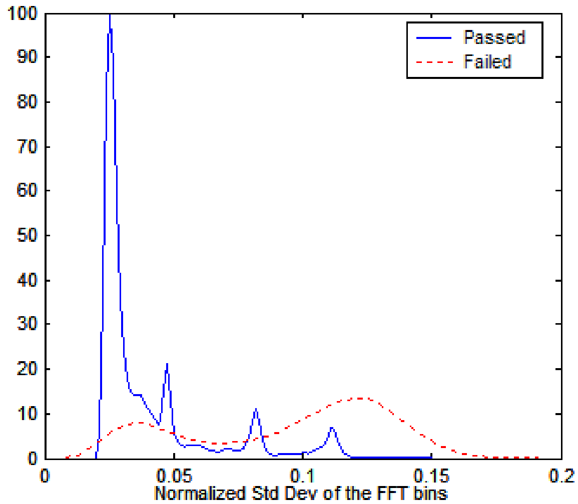

Fig. 7 – Probability density for Haar using normalized mean.

Fig. 7 shows the probability density curves for the Haar wavelet calculated from the normalized standard deviation of the magnitudes of the FFT bins. Note the large overlap area between the probability curves for pass and failure. While the distribution for signals whose symbol rate was correctly estimated by Haar is low and sharply peaked, the broad distribution of the failures results in a low-quality algorithm.

Intuitively, a low value for the mean of the FFT bin magnitudes implies that the FFT magnitude is small for most frequencies and that a peak associated with the symbol rate would have good separation from the background. Similarly, low values of the standard deviation imply a flat spectrum, again suggesting good peak separation. The failed tests in the Haar algorithm show a bi-modal probability density distribution. The first concentration coincides with the highest probability area of tests that passed. This indicates that the Haar often produces spectrums with the desired peak separation characteristics, but where the major peak is not at the correct symbol rate.

Discussion

This paper used an empirical testing framework to evaluate the effectiveness of wavelets for symbol rate estimation in PSK signals. For signal parameters typically reported in the literature, the Haar wavelet performed with nearly a 100% success rate. However, when pulse shaping, noise, as well as realistic constraints on sampling rate and signal length are imposed, the results change dramatically. The Haar wavelet performed consistently worse than other wavelets as shown in Table 1. While there is some improvement gain when multiple wavelet scales are considered, the improvement is not enough to improve success rates to an acceptable level.

Practical automated surveillance systems use many different algorithms during analysis. An algorithm that provides a high level of confidence in the predictions it classifies as correct can be very useful, even if it is unable to make a prediction for every signal. For example, while both the Haar and db6 wavelet algorithms are able to correctly estimate the symbol rate for only 50% - 75% of the signals in our test set, they have significant differences in quality. The db6 has a simple classifier that allows it to mark correct predictions with almost 100% accuracy. In contrast, we have been unable to find a classifier for accurately separating correct and incorrect predictions using the Haar wavelet. ROC curves such as those shown in Fig. 5 and Fig. 6 give a simple visual representation of the success or failure of a classifier.

The study shows that an accurate representation of the real problem domain in the test set is essential for an analysis to provide results that are useful in a practical setting. In addition, hidden parameters and commonly applied simplifying assumptions can invalidate performance results.

References

[1] A.W. Wegener, Practical techniques for baud rate estimation, *International Conference on Acoustics, Speech, and Signal Processing (ICASSP), vol. 4*, 1992, pp. 681-684.

[2] W.A. Gardner, Signal interception: a unifying theoretical framework for feature detection, *IEEE Transactions on Communications, vol. 36*, issue 8, Aug. 1988, pp. 897-906.

[3] R.J. Mammone, R.J. Rothaker, C.I. Podilchuk, Estimation of carrier frequency, modulation type, and bit rate of an unknown modulated signal, *IEEE International Conference on Communications (ICC '87), vol. 2*, June 1987, pp. 1006-1012.

[4] J.A. Sills, J.F. Wood, Application of the Euclidean algorithm to optimal baud-rate estimation, *Military Communications Conference Proceedings (MILCOM '96), vol. 3*, 1996, pp. 1015-1019.

[5] W. Su, J. Kosinski, A survey of digital modulation recognition methods, *International Signal Processing Conference*, March 2003.

[6] Y.T. Chan, J.W. Plews, K.C. Ho, Symbol rate estimation by the wavelet transform, *IEEE International Symposium on Circuits and Systems (ISCAS 97), vol. 1*, June 1997, pp. 177-180.

[7] R. Sawai, et. al., General-purpose symbol rate and symbol timing estimation method by using multi-resolution analysis based on wavelet transform for multimode software radio, *IEEE Vehicular Technology Conference (VTC), vol. 4*, 1999, pp. 2128-2132.

[8] K.C. Ho, W. Prokopiw, Y.T. Chan, Modulation identification of digital signals by the wavelet transform, *IEEE Proceedings Radar, Sonar and Navigation, vol. 147*, Aug 2000, pp. 169-176.

[9] B.M. Moret, Towards a discipline of experimental algorithms, *5^{th} Proceedings of the 5^{th} DIMACS Challenge Workshop*, 1996.

[10] K. Holladay, K. Robbins, A framework for automatic large-scale testing and characterization of signal processing algorithms, *UTSA CS Technical Report*, 2004.

[11] Mathworks, http://www.mathworks.com/. 2003.

[12] B. Sklar, *Digital Communications* (Upper Saddle River, NJ: Prentice Hall, 2001) pp. 136-148.

[13] Mathworks, *Wavelet Toolbox User's Guide*, (Natick, MA: Mathworks, Revised Version 2.2, July 2002) pp. 33-37.

[14] M. Kueckenwaitz, F. Quint, J. Reichert, A robust baud rate estimator for noncooperative demodulation, *Military Communications Conference Proceedings* (MILCOM 2000), vol. 2, pp. 971-975, 2000.

[15] K. Holladay, M. Koets, A. Burmeister, R. Dollarhide, A configurable signal analyzer for embedded systems, *Military Communications Conference Proceedings* (MILCOM) 2003.

[16] H. Van Trees, *Detection, estimation, and modulation theory* (New York, NY: John Wiley & Sons, 2001), pp. 19-52.

[17] R. Jain, *The art of computer systems performance analysis* (New York, NY: John Wiley & Sons, 1990), pp. 217.