A Low-Complexity High-Performance Bluetooth Receiver

Charles Tibenderana and Stephan Weiss
Communications Research Group
Department of Electronics & Computer Science
University of Southampton, UK

Email: {ct02r,sw1}@ecs.soton.ac.uk

Abstract

This paper presents an implementation of a GFSK receiver based on matched filtering of a sequence of K successive bits. This enables improved detection and superior BER performance but requires 2^K matched filters of considerable complexity. Exploiting redundancy, and performing phase propagation of successive single-bit stages, we propose a new receiver structure of low complexity. Simulation results presented highlight the benefits of proposed method in terms of computational cost and performance compared to standard methods.

1. Introduction

Software Defined Radios (SDR) perform signal processing tasks by running software algorithms on multi-purpose Digital Signal Processors (DSP). Flexibility offered by DSPs facilitate efficient integration of multiple standards, such as Bluetooth and WLAN, on a single radio system. For example, standard integration has been reported for Bluetooth with HiperLAN/2 [1, 2] and with IEEE 802.11b [3, 4]. As both HiperLAN/2 and 802.11b are considerably more complex than Bluetooth, a common hardware platform can provide extra computational power for the latter. This motivates to deviate from simplistic Bluetooth receivers to more sophisticated algorithms in order to achieve an improved system performance, whereby battery power consumption will benefit from an efficient realisation.

Therefore, this paper begins with a brief review of the Bluetooth modulation standard in Sec. 2, after which two possible realisations of a Bluetooth modulator are highlighted in Sec. 3. A low-complexity high-performance algorithm for Bluetooth demodulation is presented in Sec. 4, and evaluated in Sec. 5. Conclusions will be drawn in Sec. 6.

2. Gaussian Frequency Shift Keying

The modulation scheme specified for Bluetooth is Gaussian Frequency Shift Keying (GFSK). GFSK is a form of Continuous Phase Frequency Shift Keying (CPFSK), which in turn is a variation of Frequency Shift Keying (FSK). In binary FSK, equal amplitude sinusoidal waveforms at frequencies $f_c - \Delta f$ and $f_c + \Delta f$ are transmitted to signal bit 0 and 1 respectively, by switching between two oscillators. f_c is the carrier frequency and Δf the frequency deviation.

Discontinuities at symbol boundaries cause spurious transmissions, and poor bandwidth utilisation of FSK [5]. Spectral efficiency is improved in CPFSK, which constrains the phase of the transmitted signal to be continuous. This is achieved through FM modulation of a single carrier by a stream of binary data pulses. CPFSK is *full response* if the influence of a data pulse on the modulated signal does not exceed one bit period, otherwise it is *partial response* [6].

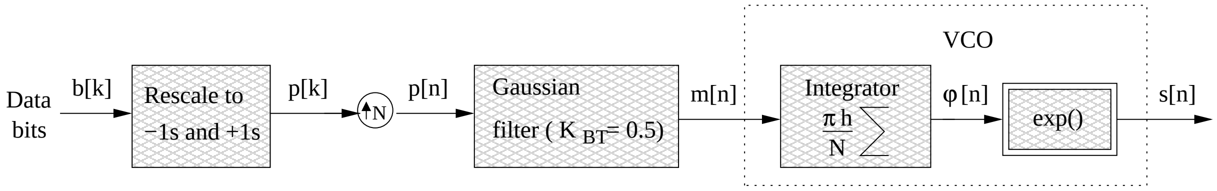


Fig. 1: GFSK modulation. K_{BT}, h and N, are the bandwidth-time product, modulation index, and number of samples per symbol respectively.

A GFSK modulator as portrayed in Figure 1, deploys a Gaussian filter prior to FM modulation to reduce the speed of frequency transitions and avoid discontinuities in the transmitted signal. The coefficients of the Gaussian pre-modulation filter are given by [7]

$$g[n] = \frac{1}{4\eta}\left[\mathrm{erf}\left(\pi K_{BT}\sqrt{\frac{2}{\ln 2}}\left(\frac{n}{\eta}+\frac{1}{2}\right)\right) - \mathrm{erf}\left(\pi K_{BT}\sqrt{\frac{2}{\ln 2}}\left(\frac{n}{\eta}-\frac{1}{2}\right)\right)\right], \qquad (1)$$

where n is an integer, η is the ratio of the bit period to the sample period, $\mathrm{erf}(\cdot)$ is the error function, and K_{BT} is the bandwidth-time product which is specified as 0.5 in the Standard [8]. $g[n]$ for different values of K_{BT} are plotted in Figure 2.

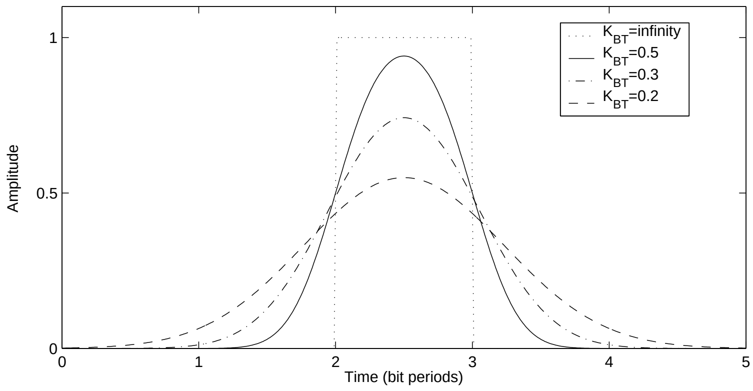


Fig. 2: Impulse response of a Gaussian filter, $g[n]$.

Figure 3 shows how the Gaussian filter improves bandwidth utilisation, while Figures 4 and 2 illustrate that K_{BT}=0.5 results in a partial response signal with each bit affecting adjacent bit periods.

The modulation index, h, defined as

$$h = 2\Delta f T, \qquad (2)$$

in which Δf and T are the frequency deviation and bit period respectively [9], may vary in a Bluetooth system between $0.28 \leq h \leq 0.35$ [8]

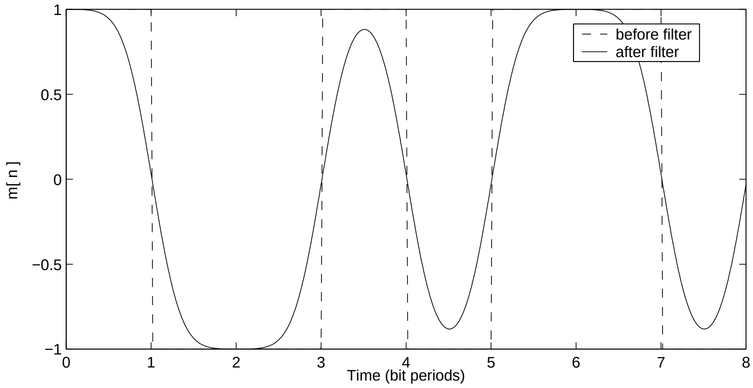


Fig. 3: Effect of the Gaussian pre-modulation filter on the baseband pulse stream ($K_{BT} = 0.5$).

It follows that a GFSK modulated signal is given by

$$s[n] = \exp\left[j\frac{\pi h}{N}\sum_{\mu=-\infty}^{n} m[\mu]\right], \tag{3}$$

where

$$m[n] = \sum_{k=-\infty}^{\infty} p_k g[n-kN], \qquad \text{and} \quad p_k \in \pm 1. \tag{4}$$

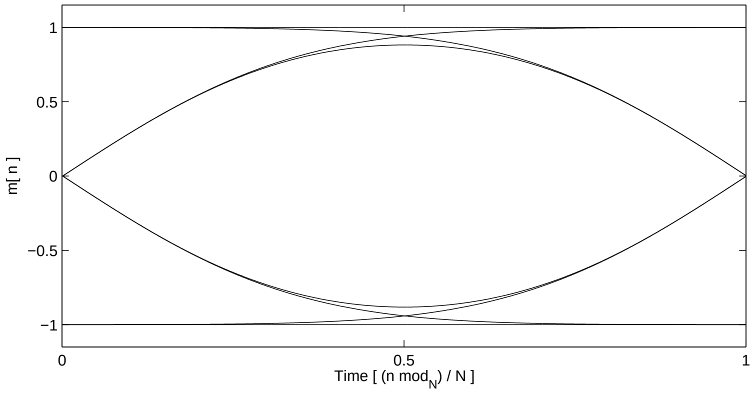


Fig. 4: GFSK eye diagram of the instantaneous frequency signal $m[n]$ with $K_{BT} = 0.5$.

3. Modulator

Figure 5 portrays a possible realisation of a Bluetooth modulator on a DSP. Incoming data is up-sampled by a factor of N, and then passed to an FIR filter with a Gaussian impulse response, $g[n]$. The output of the Gaussian filter is multiplied by $\frac{\pi h}{N}$, and then an exponential function is performed before each sample is multiplied by its predecessor to increment the phase of the transmitted signal.

In the above structure, convolution of N pulses with the $g[n]$ entails N^2 multiply-accumulates in a bit period. The phase accumulator, and multiplication by $\frac{\pi h}{N}$, will each consume N multiplications per bit. The exponential function can be realised on a DSP by performing an interpolation between points on a 256-valued lookup table. Interpolation requires 4 additions, 2

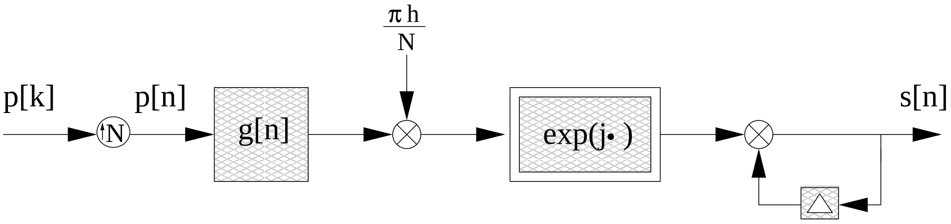


Fig. 5: Possible baseband implementation of a conventional GFSK modulator for a DSP.

multiplications, and 1 division per sample [10], or $22N$ operations per bit. The total computational cost is $2N^2$+$24N$ operations/bit.

An equivalent but more efficient modulator is portrayed in Figure 6. The lookup table contains 8 sequences of N samples each,

$$s_i[n] = \exp\left(j\frac{\pi h}{N}\sum_{\mu=1}^{n} m_i[\mu]\right), \qquad \text{for} \quad \{1 \le n \le N\} \quad \text{and} \quad i = \{1, 2, \ldots, 8\} \tag{5}$$

where the sequences m_i represent the 8 different waveforms in Figure 4. Due to being partial response, the output $s_i[n]$ depends not only on the current bit, but also the preceding and following one. Therefore, the serial-to-parallel converter in Figure 6 feeds 3 successive bit values into the look-up table, from which the correct instantaneous frequency signal segment $s_i[n]$ is determined. Multiplication of each of the lookup table output samples by the previous one accumulates the phase of the transmitted signal.

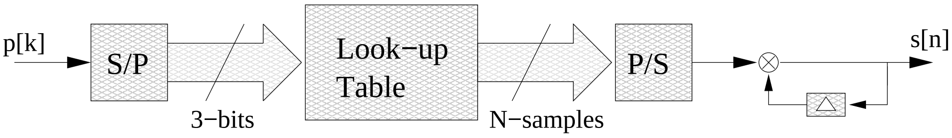


Fig. 6: Efficient baseband GFSK modulator for a DSP.

In this model, all operations are achieved through indexing, except for the phase accumulator, which requires N operations per bit.

4. Demodulator

A variety of demodulation and detection algorithms exist for Bluetooth. Relatively simple ones include FM-AM conversion, phase-shift discrimination, zero-crossing detection, and frequency feedback [11]. However, our project aims to integrate Bluetooth with IEEE 802.11b WLAN in an SDR. 802.11b is considerably more complex than Bluetooth, and so a common hardware platform can provide extra computational power for execution of Bluetooth. This is the motivation to deviate from simplistic Bluetooth receivers to more sophisticated algorithms. In order to achieve improved system performance, but to limit battery consumption by efficient implementation.

Hence, we adopt a high-performance demodulator structure, shown in Figure 7, which achieves the best possible performance in AWGN by utilising a bank of 2^K FIR filters, matched to the expected waveforms over a K bit observation interval [12, 13, 14, 15]. The filter outputs are processed, and a decision is made on the central bit. System performance improves with increase in K. In the diagram $\Psi = 2^{K-1}$, and the filters matched to waveforms $s_\alpha[n, P_1]$ to $s_\alpha[n, P_\Psi]$ signal receipt of a bit α.

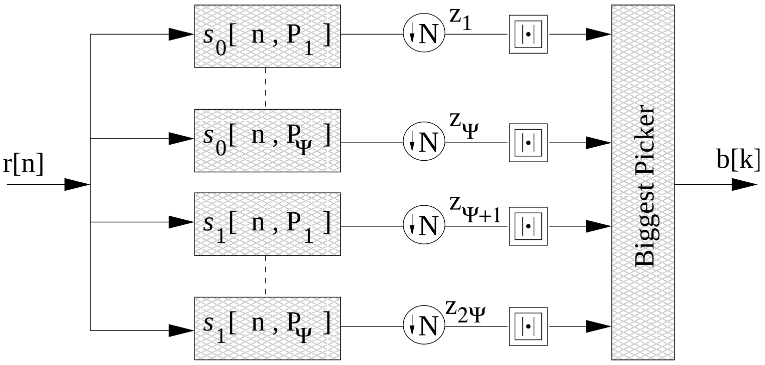


Fig. 7: High performance CPFSK demodulator.

The high-performance CPFSK receiver is attractive for use in Bluetooth because of its enhanced performance, and its ability to improve BER by increasing the observation interval, K. It is also suitable for integration with 802.11b, which also utilises a filter bank for demodulation [3, 4]. However, it is not popular because of its enormous complexity. For example, an observation interval of K bits will require 2^K filters, each KN taps long. Thus, $2^K KN$ multiply-accumulates are required for filtering a bit worth of the received signal. Also, 2^{K+1} multiplications and 2^K additions are needed to compute the magnitude. This brings the number of operations per bit to $2^{K+1}(KN + \frac{3}{2})$.

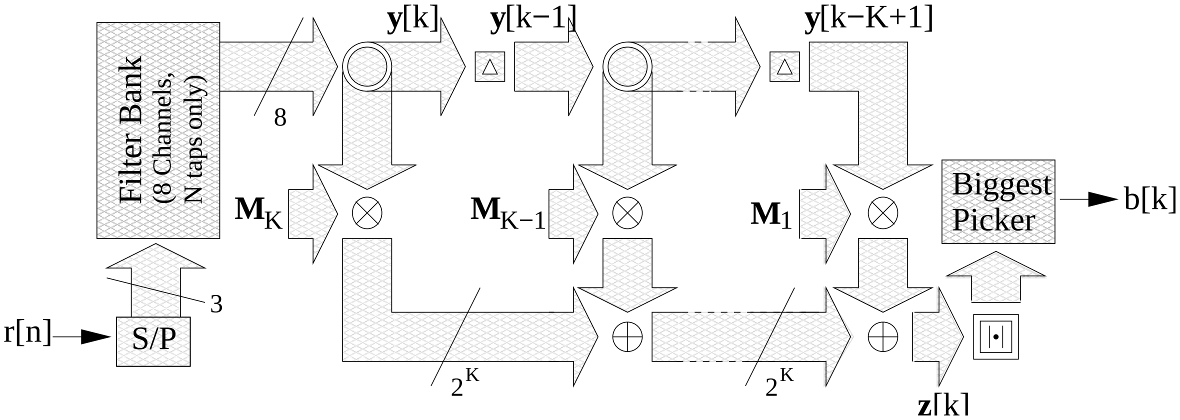


Fig. 8: Conceptual diagram of a low-complexity implementation of the high-performance GFSK demodulator for DSP.

The scheme in Figure 8 depicts the proposed low-complexity algorithm. In brief, the algorithm involves the use of a bank of 8 matched FIR filters with N coefficients spanning a single bit period, to filter each bit worth of received signal, and propagating the intermediate results, $\mathbf{y}[k]$, to finally obtain the 2^K matched filter outputs, $\mathbf{z}[k]$. The filter result with the largest magnitude determines the received bit.

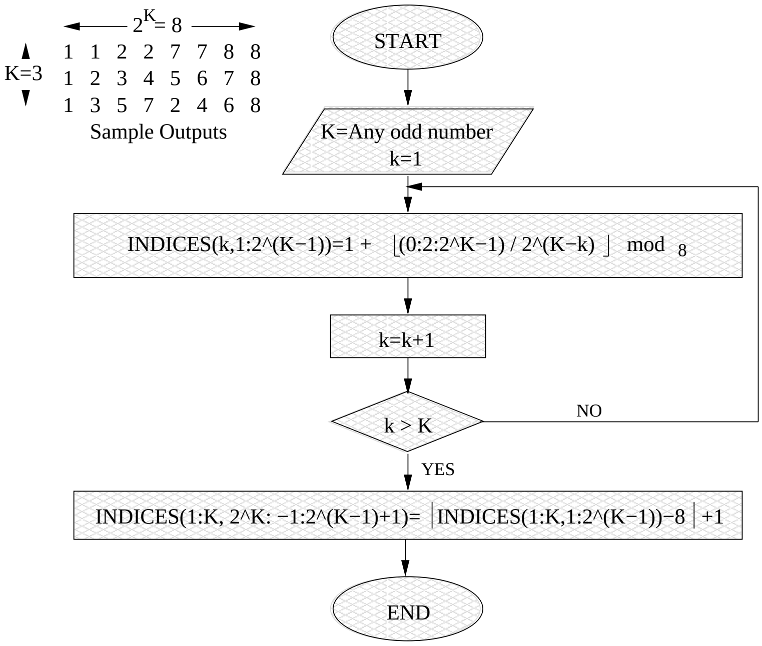
4.1 Filter Bank

The filter bank comprises of 8 matched FIR filters with N coefficients spanning one bit. Only 8 filters are required because only 8 legitimate waveforms that can occur in one bit time-space when $K_{BT} = 0.5$ The instantaneous frequencies of these waveforms were portrayed in Figure 4, and their modulated symbols were given in Equation 5. In the absence of carrier frequency mismatch, 4 of the filter bank entries are complex conjugate copies of the remaining ones, leading to a reduction in complexity by a factor of 2.

4.2 Matrices $\mathbf{M}_k$

The last K sets of outputs from the filter bank are multiplied by matrices $\mathbf{M}_k$ ($k = \{1, 2, \ldots, K\}$). Through these multiplications, the results for 2^K filters with KN coefficients covering K-bit span, are constructed. M_k are 2^K by 8 matrices with 2^K non-zero elements, which effect a permutation and phase shift on the intermediate outputs from the filter bank, $y_i[k - \gamma]$, (where i=$\{1, 2, \ldots, 8\}$, and γ=$\{0, 2, \ldots, K - 1\}$).

Figure 9 contains a flow graph for an algorithm that can be used to compute the positions of the non-zero elements of $\mathbf{M}_k$, off-line, for arbitrary K, while Figure 10 computes their values. In both cases, coefficients of filter 1 to 8, emanate from modulation of the middle pulse in sequences $\{-1 - 1 - 1\}$ to $\{+1 + 1 + 1\}$ respectively.



2^K = 8
K=3
1 1 2 2 7 7 8 8
1 2 3 4 5 6 7 8
1 3 5 7 2 4 6 8
Sample Outputs
START
K=Any odd number
k=1
INDICES(k,1:2^(K−1))=1 + ⌊(0:2:2^K−1) / 2^(K−k)⌋ mod 8
k=k+1
k > K
NO
YES
INDICES(1:K, 2^K: −1:2^(K−1)+1)= |INDICES(1:K,1:2^(K−1))−8| +1
END

Fig. 9: Algorithm to determine the position of the non-zero elements of matrices $\mathbf{M}_k$.

4.3 Example

$$\mathbf{Z}[k] = \mathbf{M}_3 \cdot \mathbf{y}[k] + \mathbf{M}_2 \cdot \mathbf{y}[k-1] + \mathbf{M}_1 \cdot \mathbf{y}[k-2] \tag{6}$$

An example in which $K = 3$ is considered here. In this case, the algorithm is summarised by Equation 6 in which $\mathbf{y}[k]$, $\mathbf{y}[k-1]$, and $\mathbf{y}[k-2]$ are vectors whose elements are the intermediate results depicted in Equation 7, $\mathbf{z}[k]$ is a vector containing the final outputs given in Equation 8, and $\mathbf{M}_1$, $\mathbf{M}_2$, and $\mathbf{M}_3$ are the matrices in Equations 9, 10 and 11 respectively.

$$\mathbf{y}[k] \quad = \quad \begin{pmatrix} y_1[k] \\ \vdots \\ y_8[k] \end{pmatrix} \tag{7}$$

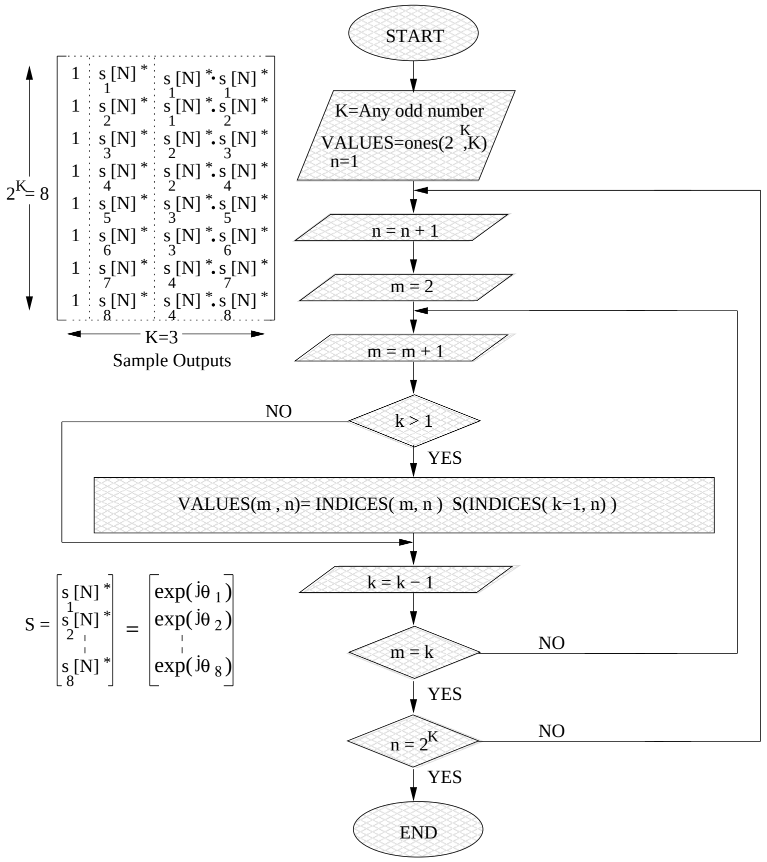


Fig. 10: Algorithm to determine the values of the non-zero elements of matrices $\mathbf{M}_k$.

$$\mathbf{z}[\mathbf{k}] \;=\; \begin{pmatrix} z_1[k] \\ \vdots \\ z_8[k] \end{pmatrix} \tag{8}$$

$$\mathbf{M}_1 \;=\; \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \tag{9}$$

$$\mathbf{M}_2 \;=\; \begin{pmatrix} e^{j\theta_1} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & e^{j\theta_1} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & e^{j\theta_2} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & e^{j\theta_2} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & e^{j\theta_7} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & e^{j\theta_7} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & e^{j\theta_8} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & e^{j\theta_8} \end{pmatrix} \tag{10}$$

$$\mathbf{M}_3 = \begin{pmatrix} e^{j2\theta_1} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & e^{j(\theta_1+\theta_2)} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & e^{j(\theta_2+\theta_3)} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & e^{j(\theta_2+\theta_4)} & 0 \\ 0 & e^{j(\theta_7+\theta_5)} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & e^{j(\theta_7+\theta_6)} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & e^{j(\theta_8+\theta_7)} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & e^{j2\theta_8} \end{pmatrix} \quad (11)$$

For example, $z_3[k]$ is given by

$$z_3 = e^{j(\theta_2+\theta_3)} \cdot y_5[k] + e^{j\theta_2} \cdot y_3[k-1] + y_2[k-2] \quad (12)$$

The computational complexity per bit, of the efficient demodulator is as follows: $4N$ multiply-accumulates are needed to calculate $\mathbf{y}_i[k]$ once per bit, $8(K-1)$ multiplications are required to impose the phase shifts, after which $8(K-1)$ additions are performed to obtain the final outputs, $\mathbf{z}[k]$. Magnitude calculations consume 2^{K+1} multiplications and 2^K additions. This totals up to $\frac{3}{2}2^{K+1} + 8N + 16(K-1)$ operations/bit.

5. Simulation Results

The computational complexities of a conventional and efficient implementation of a Bluetooth modem as a function of the observation interval K, and the number of samples per bit N, are given in Table 1, and plotted in Figure 11. Complexity is evaluated in operations/bit, whereby multiplication or addition involve one operation, while division requires 16 operations. These illustrations show that complexity rises less rapidly with increase in K in the efficient case.

method	parameters	complexity (operations/bit)		SNR at BER=10^{-3}
conventional	$K=1, N=2$	$2^{K+1}(\frac{3}{2}+KN)$	14	20dB
[12, 13, 14]	$K=9, N=2$		19968	11dB
proposed	$K=9, N=2$	$\frac{3}{2}2^{K+1}+8N+16(K-1)$	1680	11dB

Tab. 1: Comparison of complexity and SNR performance of conventional and proposed GFSK detectors.

The bit error ratio (BER) performances of the conventional and efficient implementations are the same, and their performance improves with a larger K [12]. Hence, BER improvements offered by the low-complexity realisation in an SDR will be attained by increasing K. For example, if the computing power available on the DSP was limited to 2000 operations per bit period, then the conventional method would only support a receiver based on $K = 5$ successive bit periods, while the efficient implementation can afford $K = 9$, as see in Figure 11). The resulting BER performance of the two systems are displayed in Figure 12; the efficient method has 1.2 dB improvement at a BER of 10^{-3}.

In both cases, the performance of the high-performance receiver is better than the demodulation algorithms in [2], the best of which attained a BER of 10^{-3} at 14.8 dB.

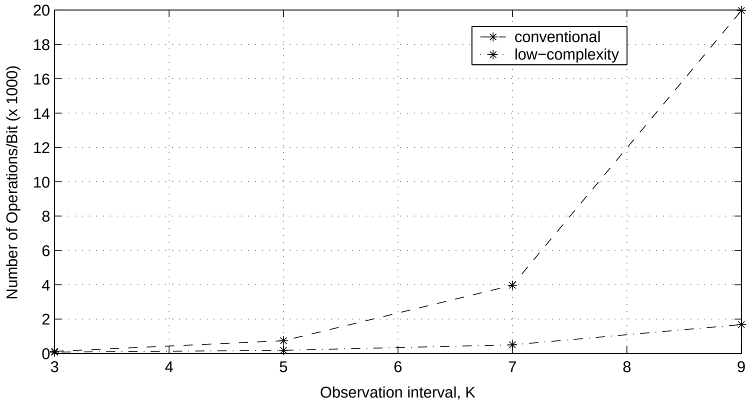


Fig. 11: Computation complexity: Conventional vs. low-complexity with N=2.

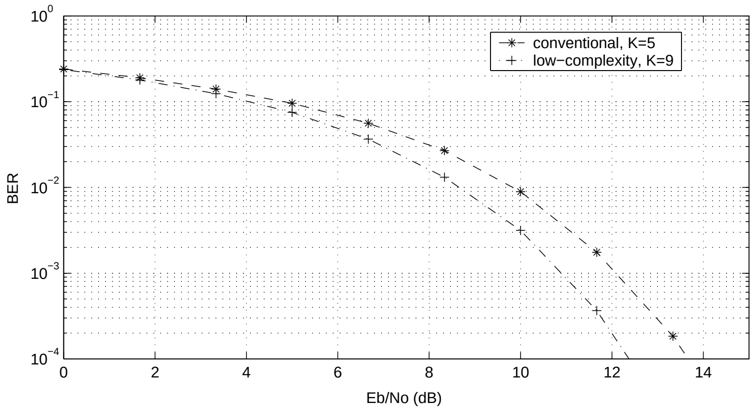


Fig. 12: BER performance curves of the modems with a limit of 2000 operations per bit for parameters $N = 2, K_{BT} = 0.5$, and $h = 0.3$.

6. Conclusion

The maximum permissible BER for a Bluetooth system is 10^{-3} [8]. Schiphorst *et al.* demonstrated that the best of the comparatively simple algorithms mentioned above achieve acceptable performance above a channel SNR of 14.8 dB, while some practitioners assume a channel SNR of 21 dB [16] to attain the minimum Bluetooth performance requirement. By using a high-performance demodulator described in [12, 13, 14, 15], where decisions are based on an observation interval spanning $K = 9$ bit periods, a BER of 10^{-3} is reached at a channel SNR of 11 dB (as demonstrated in Figure 12). The 3.8 dB gain achieved with this technique is substantial, and yet the high-performance demodulator is not popular because of enormous computational requirements for large K

A common hardware platform for IEEE 802.11b WLAN and Bluetooth will have extra capacity to improve Bluetooth performance. Since it might not necessarily have the computational capacity to handle a high-performance demodulator with sufficiently large observation intervals, the efficient high-performance algorithm described in this paper helps to considerably reduce the computational requirements. As demonstrated in an example for $K = 9$, the proposed method can achieve a 90% reduction in complexity . If compared on the basis of identical complexity, our approach can outperform standard implementations in terms of BER.

7. References

[1] Roel Schiphorst, Fokke Hoeksema, and Kees Slump, "Channel Selection Requirements for Bluetooth Receivers using a Simple Demodulation Algorithm ," in *Proceedings of ProRISC 2001*, Veldhoven, Netherlands, November 2001, pp. 584–591.

[2] Roel Schiphorst, Fokke Hoeksema, and Kees Slump, "Bluetooth Demodulation Algorithms and their Performance," in *Proc. 2nd Karlsruhe Workshop on Software Radios*, Karlsruhe, March 2002, pp. 99–105.

[3] Charles Tibenderana, Terence E. Dodgson, Stephan Weiss, and Derek Babb, "Towards Software Defined Radio (SDR) Bluetooth and IEEE 802.11b Modem Integration," in *9th Wireless World Research Forum Meeting*, Zurich, July 2003.

[4] Charles Tibenderana and Terence E. Dodgson, "Integrated modulators and demodulators," UK Patent Application 0219740.8, Samsung Electronics Research Institute, Stains, August 2002.

[5] Theodore S. Rappaport, *Wireless Communications: Principles and Practice*, Prentice Hall, New Jersey, July 1999.

[6] John B. Anderson, Tor Aulin, and Carl-Erik Sundberg, *Digital Phase Modulation*, Plenum Press, New York and London, 1986.

[7] Raymond Steele and Lajos Hanzo, *Mobile Communications*, John Wiley and Sons, West Sussex, 2nd edition, 1999.

[8] Bluetooth Special Interest Group, *Specification of the Bluetooth System*, February 2002, Core.

[9] Rudi De Buda, "Coherent Demodulation of Frequency-Shift Keying With Low Deviation Ratio," in *Proc. International Conference on Communications*, Montreal, June 1971, pp. 429–435.

[10] Erwin Kreyszig, *Advanced Engineering Mathematics*, John Wiley and Sons, New York, 6th edition, 1988.

[11] Bruce A. Carlson, *Communication Systems*, McGraw-Hill, Singapore, 3rd edition, 1986.

[12] William P. Osborne and Micheal B. Luntz, "Coherent and Noncoherent Detection of CPFSK," in *IEEE Transactions on Communications*, August 1974, vol. COM-22, pp. 1023–1036.

[13] Thomas A. Schonhoff, "Symbol Error Probabilities for M-ary CPFSK: Coherent and Noncoherent Detection," in *IEEE Transactions on Communications*, June 1976, vol. COM-24, pp. 644–652.

[14] W. Hirt and S. Pasupathy, "Suboptimal Reception of Binary CPSK Signals," in *Proc. IEE Communications, Radar and Signal Processing*, United Kingdom, June 1981, vol. 128, pp. 125–134.

[15] John G. Proakis, *Digital Communications*, McGraw-Hill, New York, 3rd edition, 1995.

[16] Ericsson, *Ericsson Bluetooth Development Kit documentation*, October 1999.