On Error Correction for Asynchronous Communication Based on Iterative Processing

Chen Yi and Jörg Kliewer
Helen and John C. Hartmann Department of Electrical and Computer Engineering
New Jersey Institute of Technology
Email: cy235@njit.edu, jkliewer@njit.edu

***Abstract*—A forward error correction scheme for asynchronous sensor communication is proposed, where a continuous-time sparse waveform signal is asynchronously sampled and communicated over a noisy channel via Q-ary frequency-shift keying. The presented concatenated code employs outer systematic convolutional codes and inner embedded marker codes, which effectively preserve the timing information along with a protection against symbol insertions and deletions. We show that by iteratively decoding marker and convolutional codes along with interleaving a short block of parity bits a significant reduction in terms of the expected end-to-end distortion between original and reconstructed signals can be obtained compared to non-iterative processing.**

I. Introduction

Short range wireless sensing with low-power sensors has been an emerging application during the last couple of years (see, e.g., [1]–[3]) in many areas, for example in the areas of environmental observation, biomedical, and health care monitoring. These applications require integrated sensors being capable of providing reliable wireless links under low circuit complexity and long battery life.

In order to address the challenges of such a power efficient sensing operation, we proposed to replace the classical Nyquist-based synchronous signal processing by an asynchronous sensing architecture. This is motivated by the observation that the Nyquist "sampling and quantization" approach is not always the optimal solution for recording correlated sparse continuous-time signals [4]. For instance, if the input signal is inactive or no changes are detected, sampling and transmission do not need to be carried out in an asynchronous sensing scheme [5], which has the potential of significant power savings. Another advantage is that the sensor hardware is not required to have a clock circuit and thus allows for a very energy efficient hardware design.

Although there are many works related to theory and implementation of asynchronous sampling, reliably communicating asynchronous samples over noisy channels has only been sparsely addressed in the existing open literature. In previous work [6] we have proposed a VLSI implementation of an uncoded asynchronous communication system, which however leads to synchronization losses due to symbol insertions and deletions at the receiver.

This work was supported in part by National Science Foundation grant ECCS-1407910.

On the other hand, a large amount of work has addressed insertion and/or deletion error correcting schemes for synchronous communication systems (see, e.g., [7]). Concatenated codes have been considered to maintain synchronization of the bitstream (see, e.g., [8]–[11]), where an outer forward error correction code and an inner resynchronization code are combined. Also, convolutional codes for the insertion/deletion channel have been considered by either extending the state space of the code [12] or by modifying the path metric of the Viterbi decoder [13]. Note that these synchronous insertion/deletion error correcting schemes from above cannot be directly applied to the asynchronous setting. For example, in asynchronous communication the timing information of the samples is contained in the transmitted signal pulses. Thus, any code redundancy such as parity bits can only be added via extending the modulation alphabet instead of by adding extra pulses. For the same reason systematic codes are required.

In the following we extend our previous error correction results from [14] to general Q-ary frequency-shift keying (FSK) modulated asynchronous communication systems and introduce iterative processing at the decoder. We propose a causal mapping of information and code bits to the symbols of a Q-ary modulation alphabet based on an outer systematic convolutional code and an embedded inner marker code, which protects against symbol insertions and deletions. We also show that by solely interleaving a short block of parity bits in combination with an iterative decoder a significant iterative gain can be obtained both in terms of symbol error probability and the expected end-to-end distortion between original and reconstructed signals.

II. Asynchronous Sampling

Fig. 1 illustrates how a waveform is uniformly quantized by level-crossing sampling [15], where $S(t_k)$ denotes the amplitude of the waveform signal at time t_k. At each time the waveform is compared with M equally-spaced quantization decision thresholds defined as $\Delta \triangleq \frac{2\max_t(|S(t)|)}{M}$. Whenever the waveform exceeds a decision threshold in the direction of increasing amplitude a "+1" sample is recorded at that specific time, otherwise a "-1" sample is placed; "+1" and "-1" are mapped to bits 1 and 0, respectively. $\tau_{\min}$ denotes the smallest possible sampling interval, which is a function of the source waveform signal $S(t)$ and M.

III. System Model

The system model for the proposed asynchronous communication system is shown in Fig. 2 [14]. The analog signal $S(t)$ is asynchronously sampled into the sample bits u_{t_k}, $k = 1, \ldots, T$, where T denotes the number of transmitted bits. These bits are assumed to be known at the receiver. If no coding scheme is employed, u_{t_k} becomes the length Q vector $\mathbf{x}_{t_k}$. This vector is modulated by FSK where the carrier pulses are orthogonal sinc waveforms. The length of the pulses is fixed to $\tau_{\min}$ in order to avoid intersymbol interference due to overlapping FSK pulses. Encoding generates the corresponding parity bits p_{t_k}, which are subsequently permuted by an interleaver (π) whose interleaving pattern is known to the receiver. Then both interleaved parity and marker bits, used to ensure synchronization, form the redundant bits. These bits are combined with the information bits to generate $\mathbf{x}_{t_k}$, which is subsequently modulated via Q-ary FSK. Note that not only the value of the sampled bits need to be recovered but also the timing information of the samples u_{t_k} must be preserved. This means that systematic codes are required.

Fig. 3 shows an example how asynchronously sampled information bits u_{t_k}, encoded and interleaved parity bits $p_{t'_k}$, and periodic marker bits $m_{t''_k}$ can be arranged to form Q-ary modulation symbols. The first row always contains the information symbols, as we need to preserve the timing information. K denotes the block length of uncoded symbols, and N_c represents the number of parity bits in each block.

For 4-FSK pulse forming (see Fig. 4), the second row is a combination of parity and marker bits, where a pair of marker bits $m_{t_{\ell K}}$ and $m_{t_{\ell K+1}}$ appear at times $t_{\ell K}$ and $t_{\ell K+1}$, $\ell \in \mathbb{N}$, periodically. These marker bits are assumed to be known at the decoder and are used to resynchronize the received symbol sequence from deletion and insertion errors. For 8-FSK pulse forming an additional row full of markers is added as it is shown in Fig. 3. Higher order Q-ary FSK pulses can be obtained by adding a larger number of marker and parity bits. Due to causality, the parity bits during the first block of K symbols are set to zeros. Also, note that the parity and information bits in the same symbol $\mathbf{x}_{t_k}$ can be regarded as independent as they are associated with different codewords, which simplifies the computation of the joint probabilities for iterative decoding. Since K is directly related to the memory requirements and the chip area of the sensor, it is required to be not too large. We employ systematic convolutional codes whose code rates are $R_c = K/(K+N_c)$. Also, let R_m denotes the code rate associated with the marker symbols. Then, the trade off between the two code rates is constrained as

$$R_c R_m = \frac{1}{\log_2(Q)}. \tag{1}$$

After encoding, the modulated signal $x(t)$ is transmitted over an AWGN channel. The observation at the channel output $y(t)$ is applied to a matched filter (MF) receiver, it is then passed to the iterative channel decoder (see Section V) which ensures resynchronization (i.e., insertion and deletion error correction) and the correction of substitution errors on the channel, followed by a reconstruction stage. In the MF receiver, the output of the 2^Q matched filters is compared with a threshold on a very fine grid (by running a high frequency local clock at the base station), and whenever the threshold is exceeded, the corresponding Q-ary symbol is reconstructed. A deletion error occurs when the energy waveform after the MF lies below a certain threshold γ. In contrast, an insertion takes place when the channel noise triggers the given threshold during a silent segment between two adjacent pulses (see Fig. 5). Since any position during this segment may trigger the threshold when random noise exists, it is possible that there will be more than one insertion between the adjacent pulses, thus creating burst insertion errors. For a 10 ms of segment of continuous-time rat cortex signal Fig. 6 shows the length distribution of consecutive insertion errors at an SNR of 2dB, where approximately 35% of the insertion errors are burst errors.

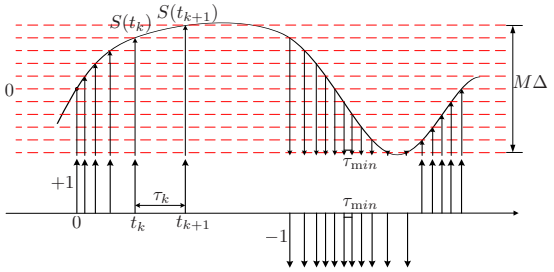

Fig. 1. Asynchronous sampling via level-crossing sampling [15].

IV. Error Correction

A. Symbol based forward backward algorithm

In [10] a forward backward algorithm (FBA) for correcting insertion, deletion, and substitution errors for *synchronous* transmission is introduced. We adapt this algorithm for our purpose as follows, where for simplicity the time t_k is replaced with the integer k. The transmitted bit sequence of length T is defined as $X_1^T = (\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_T)$, the received bit sequence of length R after channel is $Y_1^R = (\mathbf{y}_1, \mathbf{y}_2, ..., \mathbf{y}_R)$, where $\mathbf{x}$ and $\mathbf{y}$ represent transmitted and received length-Q binary vectors, resp., corresponding to a specific column in Fig. 3. Both R and T are assumed to be known at the receiver. As in the classical BCJR algorithm [16] we define $\alpha_{k,n} \triangleq P(D_{k,n}, Y_1^n)$ and $\beta_{k,n} \triangleq P(Y_{n+1}^R | D_{k,n})$, where $D_{k,n}$ is defined as the event that when the transmitted number of symbols is $k \in \{1, 2, ..., T\}$, the received number of symbols is $n \in \{1, 2, ..., R\}$. The symbol based forward recursion [10] is given as

$$\alpha_{k,n} = \frac{P_i}{Q^2}\alpha_{k-1,n-2} + P_d\alpha_{k-1,n} + P_t\alpha_{k-1,n-1}\sum_{\mathbf{x}_k} P(\mathbf{x}_k)(1-P_s)^{\delta_{\mathbf{x}_k,\mathbf{y}_n}} P_s^{1-\delta_{\mathbf{x}_k,\mathbf{y}_n}}, \tag{2}$$

where $\delta_{\mathbf{x},\mathbf{y}}$ denotes the Kronecker delta. P_i, P_d and P_s are insertion, deletion and substitution probabilities, resp., and $P_t \triangleq 1 - P_i - P_d$. $P(\mathbf{x}_k)$ is the a priori probability of the transmitted symbols. At time slot k, $\mathbf{x}_k$ contains j uniformly distributed binary non-marker bits $\mathbf{c}_k^j = [c_k^j(1), c_k^j(2), ..., c_k^j(j)]$, $j \in$

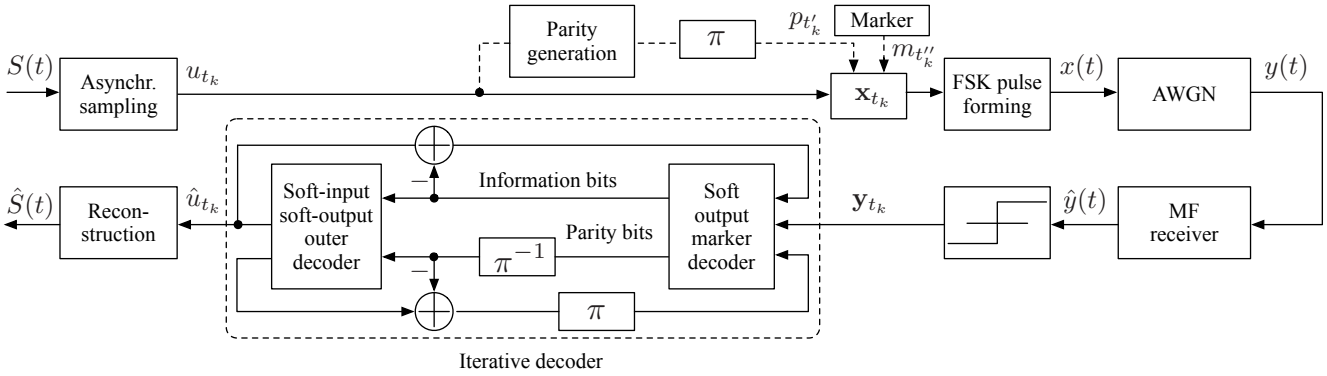

Fig. 2. System model

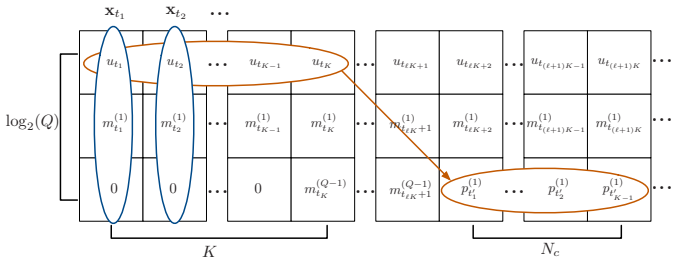

Fig. 3. Combination of information and redundant bits for Q-FSK pulse forming.

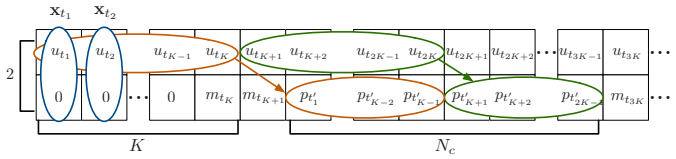

Fig. 4. Combination of information and redundant bits for 4-FSK pulse forming.

$\{1, 2, ..., \log_2(Q)\}$, and $\bar{j}$ binary marker bits with fixed values known to the receiver, $\mathbf{m}_k^{\bar{j}} = [m_k^{\bar{j}}(1), m_k^{\bar{j}}(2), ..., m_k^{\bar{j}}(\bar{j})]$, $\bar{j} = \log_2(Q) - j$, respectively. If $\mathbf{x}_k$ contains the bits $\mathbf{m}_k^{\bar{j}}$, they are determined with probability 1 since the marker bits are assumed to be known at the receiver. Therefore, in (2) we have

$$P(\mathbf{x}_k = [\mathbf{c}_k^j, \mathbf{m}_k^{\bar{j}}]^T) = P(\mathbf{c}_k^j) = \prod_{i=1}^{j} P(c_k^j(i)) = \frac{1}{2^j}. \quad (3)$$

For example, in 4-FSK pulse forming, if $\mathbf{x}_k$ contains a marker bit 0 and $j = 1$, we have $P(\mathbf{x}_k = [00]^T) = P(\mathbf{x}_k = [10]^T) = 0.5$. If $\mathbf{x}_k$ contains a parity bit and $j = 2$, $P(\mathbf{x}_k) = P(c_k^2(1)) \cdot P(c_k^2(2)) = 0.25$.

The recursion for $\beta_{k,n}$ is defined similarly as in (2). Finally, by combining $\alpha_{k,n}$ and $\beta_{k,n}$ (see [14]) we obtain the soft-output of the FBA algorithm $P(Y_1^R|\mathbf{x}_k)$, which is used as soft input for the outer SISO decoder.

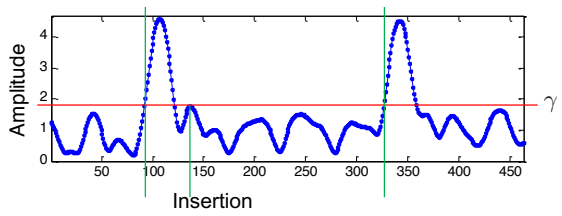

Fig. 5. Waveform $\hat{y}(t)$ at the input of the MF in Fig. 2

The obtained results from the inner FBA synchronization decoding are further employed to localize the position of deleted and inserted FSK pulses. The inserted pulses can be removed directly. Any deleted pulse is placed at the midpoint between two adjacent pulses, which only incurs a small additional average end-to-end distortion (see [14]).

V. Iterative Decoding

In order to combat burst insertion errors, a random interleaver is employed in the proposed scheme. The length of interleaver is kept moderate due to the limited size of the sensor chip storage and given transmission delay constraints. Since the order of the information bits and therefore the timing information must be preserved, and since the markers bits appear periodically and are known to the receiver, only the parity bits are interleaved.

In iterative decoding for serially concatenated codes [17], the extrinsic information of both information bits and parity bits are computed and exchanged between inner codes and outer codes. Here, we exchange the extrinsic information of both information and parity bits between inner marker codes and outer convolutional codes. The iterative decoder setup employed in this work is shown in Fig. 7. The output L-values of the information bits from the FBA are fed directly to the input of the soft-input soft-output (SISO) BCJR decoder for the outer convolutional code. The output L-values of the parity bits from the FBA are deinterleaved before being processed by the BCJR. The extrinsic information of information bits and parity bits from the BCJR decoder are obtained as

$$L_{\text{extr}}^{(2)}(u_k) = L(u_k|\hat{Y}_1^T) - L_e^{(1)}(u_k), \quad (4)$$

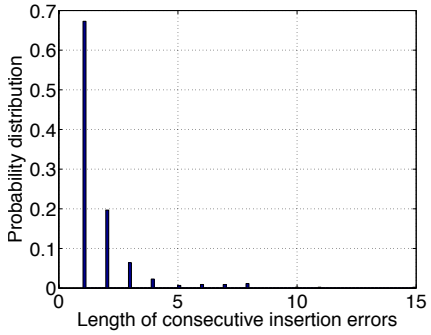

Fig. 6. Length distribution of consecutive insertion errors

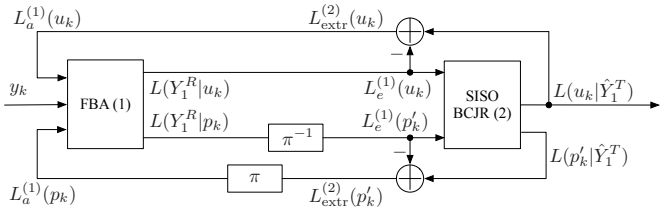

Fig. 7. Iterative decoding

$$L_{\text{extr}}^{(2)}(p'_k) = L(p'_k|\hat{Y}_1^T) - L_e^{(1)}(p'_k), \tag{5}$$

where

$$L_e^{(1)}(u_k = 0/1) = \log \frac{P(Y_1^R|u_k = 0/1)}{P(Y_1^R|u_k = 1/0)}, \tag{6}$$

$$L_e^{(1)}(p'_k = 0/1) = \log \frac{P(Y_1^R|p'_k = 0/1)}{P(Y_1^R|p'_k = 1/0)}, \tag{7}$$

which feeds back as a priori information to the input of the FBA to update $P(\mathbf{x}_k)$ in (2). The non-marker symbol is $\mathbf{c}_k^j = [u_k, \mathbf{p}_k'^{j-1}]$, where $\mathbf{p}_k'^{j-1} = [p_k'^{j-1}(1), p_k'^{j-1}(2), ..., p_k'^{j-1}(j-1)]$ is the interleaved parity bit vector. If $\mathbf{x}_k$ contains $\mathbf{m}_k^{\bar{j}}$,

$$P(\mathbf{x}_k = [\mathbf{c}_k^j, \mathbf{m}_k^{\bar{j}}]^T) = P(u_k) \cdot \prod_{i=1}^{j-1} P(p_k'^{j-1}(i)). \tag{8}$$

For example, in 8-FSK pulse forming, if $j = 2$, e.g., the second and third row in Fig. 3 completely consist of marker bits, we have $P(\mathbf{x}_k = [010]^T) = P(u_k = 0)$. If the second row contains parity bits, but the third row contains a marker bit 1, i.e., $j = 2$, we have $P(\mathbf{x}_k = [001]^T) = P(u_k = 0) \cdot P(p_k'^1(1) = 0)$. The update of $P(\mathbf{x}_k)$ for general Q-ary FSK can be obtained in a similar way.

VI. Simulation Results

In this section, a 10 ms excerpt of the above mentioned recorded rat cortex signal is employed to evaluate the proposed error correction strategy. This signal is asynchronously quantized with $M = 63$ threshold levels. The periodic pair of marker bits is fixed as $[m_{t_{\ell K}}, m_{t_{\ell K+1}}] = [0, 1]$. Table I shows the employed outer convolutional codes for different values of N_c along with the corresponding mother code rates R, punctured rates R_c, and code rates with the markers R_m. In order to preserve the timing information of the pulses, only parity bits are punctured. Also, the puncturing pattens are selected such that the combined deletion, insertion, and substitution error probabilities are minimized. In this paper, only the performances for 2-FSK, 4-FSK, and 8-FSK modulation are compared.

In Fig. 8, we compare the symbol error probability of different systematic convolutional codes displayed in Table I under 4-FSK and 8-FSK modulation. Fig. 8 shows that a smaller R_m, i.e., a larger number of marker bits under 4-FSK modulation, leads to a smaller error probability. The reason is that even though reducing R_m will lead to an increase of the code rate of the employed convolutional code according to (1) which cleans up a smaller number of residual errors

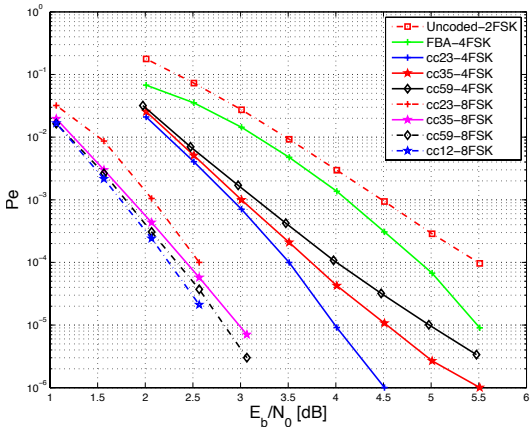

Fig. 8. Error probability for different codes displayed in Table I under different FSK modulation

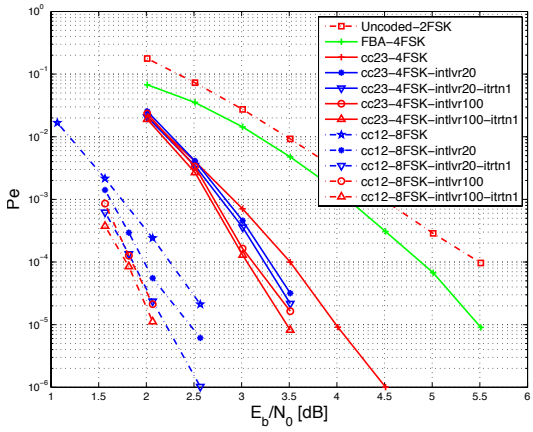

Fig. 9. Iterative decoding performance for different interleaver sizes under different FSK modulation

after insertion/deletion decoding, it provides a better protection for information bits from insertion/deletion errors. However, for 8-FSK modulation, a smaller R_c provides a better overall error correcting performance. This is due to the fact that the third row of the pulse forming scheme in Fig. 3 only consist of markers, which already provides sufficient protection for the information bits from synchronization errors. Therefore, the overall error correcting performance gain for 8-FSK is dominated by convolutional codes with lower code rates.

Fig. 9 compares the iterative decoding performance for different outer convolutional codes and interleaver sizes of 20 and 100, resp., for both 4-FSK and 8-FSK modulation. Convolutional codes with the best overall error correcting performance under 4-FSK and 8-FSK in Fig. 8 are investigated, respectively. The overall performance for 8-FSK improves on 4-FSK because a larger amount of redundancy is employed. We can also see that a moderate increase of the interleaver size from 20 to 100 improves the error correcting performance as burst insertions are dispersed to a larger extent. Due to the small interleaver size, only one decoding iteration is employed; we have observed diminishing gains in further iterations.

Fig. 10 shows the expected end-to-end MSE distortion versus the SNR, where similar observations as in Figs. 8 and 9

TABLE I
CODE RATES AND PUNCTURING SCHEMES FOR THE EMPLOYED SYSTEMATIC CONVOLUTIONAL CODES

R	Generators	Puncturing matrix	R_c	Label	N_c	$R_m(Q=4)$	$R_m(Q=8)$
$\frac{2}{3}$	$\mathbf{G}(D) = \begin{bmatrix} 1 & 0 & (D^2+D+1)/(D^3+D^2+D+1) \\ 0 & 1 & (D^3+D+1)/(D^3+D^2+D+1) \end{bmatrix}$	—	$\frac{2}{3}$	cc23	2	$\frac{3}{4}$	$\frac{3}{6}$
$\frac{1}{2}$	$\mathbf{G}(D) = \begin{bmatrix} 1 & (D^3+D+1) \end{bmatrix}$	—	$\frac{1}{2}$	cc12	0	—	$\frac{2}{3}$
$\frac{1}{2}$	$\mathbf{G}(D) = \begin{bmatrix} 1 & (D^3+D+1) \end{bmatrix}$	$\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 0 \end{bmatrix}$	$\frac{3}{5}$	cc35	4	$\frac{5}{6}$	$\frac{5}{9}$
$\frac{1}{2}$	$\mathbf{G}(D) = \begin{bmatrix} 1 & (D^3+D+1) \end{bmatrix}$	$\begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 1 & 1 \end{bmatrix}$	$\frac{5}{9}$	cc59	8	$\frac{9}{10}$	$\frac{9}{15}$

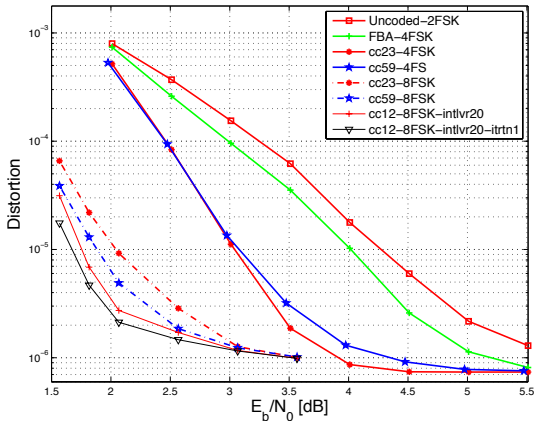

Fig. 10. MSE distortion for the codes from Table I .

can be made. In contrast to Figs. 8 and 9 the distortion includes errors due to incorrect localization of the position of inserted and deleted pulses. Again, we can see that employing iterative decoding along with interleaving of parity bits also reduces the distortion.

VII. CONCLUSION

We extend our previous work from 4-FSK to Q-FSK modulation under a concatenated error correction scheme for asynchronous communication. The proposed scheme comprises a combination of a systematic outer convolutional code and an embedded inner marker code at the sensor node. Simulation results have shown that using 8-FSK modulation symbols provides a performance gain of over 1 dB in SNR compared to 4-FSK. If the number of marker bits is not large enough to protect every information bit (e.g., not all transmitted FSK symbols contain marker bits for 4-FSK), increasing the number of marker bits provides a better error correcting performance and a smaller end-to-end distortion. In contrast, if a sufficient number of marker bits is already employed (e.g., every transmitted FSK symbol contains a marker bit as in the proposed 8-FSK scheme), the overall performance is dominated by convolutional codes with lower code rates. Also, iterative decoding with a small number of iterations in combination with interleaving the parity bits further reduces both the number of residual substitution errors and the end-to-end distortion.

REFERENCES

[1] J. Foerster, E. Green, S. Somayazulu, and D. Leeper, "Ultra-wideband technology for short- or medium-range wireless communications," *Intel Technology Journal*, vol. 5, no. 2, pp. 1–11, May 2001.

[2] A. Willig, "Recent and emerging topics in wireless industrial communications: A selection," *IEEE Transactions on Industrial Informatics*, vol. 4, no. 2, pp. 102 –124, May 2008.

[3] V. C. Gungor and G. P. Hancke, "Industrial wireless sensor networks: Challenges, design principles, and technical approaches," *IEEE Trans. on Industrial Electronics*, vol. 56, no. 10, pp. 4258 –4265, Oct. 2009.

[4] J. A. Tropp, J. N. Laska, M. F. Duarte, J. K. Romberg, and R. G. Baraniuk, "Beyond Nyquist: Efficient sampling of sparse bandlimited signals," *IEEE Trans. Inf. Theory*, vol. 56, no. 1, pp. 520–544, 2010.

[5] B. Schell and Y. Tsividis, "A continuous-time ADC/DSP/DAC system with no clock and with activity-dependent power dissipation," *IEEE Journ. of Solid-State Circ.*, vol. 43, no. 11, pp. 2472–2481, Nov. 2008.

[6] Q. Hu, C. Yi, J. Kliewer, and W. Tang, "Asynchonous communication for wireless sensors using ultra wideband impulse radio," in *IEEE 58th International Midwest Symposium on Circuits and Systems*, Fort Collins, CO, Aug. 2015, pp. 1–4.

[7] N. J. A. Sloane, "On single-deletion-correction codes," in *Codes and Designs*. Berlin: Walter de Gruyter, May 2000, pp. 273–291.

[8] M. C. Davey and D. J. C. MacKay, "Reliable communication over channels with insertions, deletions, and substitutions," *IEEE Trans. Inf. Theory*, vol. 47, no. 2, pp. 687–698, Feb. 2001.

[9] J. Chen, M. Mitzenmacher, C. Ng, and N. Varnica, "Concatenated codes for deletion channels," in *Proc. IEEE Int. Sympos. on Inform. Theory*, Yokohama, Japan, Jun. 2003, pp. 218–218.

[10] F. Wang, D. Fertonani, and T. M. Duman, "Symbol-level synchronization and LDPC code design for insertion/deletion channels," *IEEE Trans. Comm.*, vol. 59, no. 5, pp. 1287–1297, May 2011.

[11] F. Wang, D. Aktas, and T. M. Duman, "On capacity and coding for segmented deletion channels," in *Proc. 49th Annual Allerton Conference on Communication, Control, and Computing*, Monticello, IL, Sep. 2011, pp. 1408–1413.

[12] M. F. Mansour and A. H. Tewfik, "Convolutional decoding in the presence of synchronization errors," *IEEE J. Sel. Areas in Commun.*, vol. 28, no. 3, pp. 218–227, Feb. 2010.

[13] H. Mercier and V. K. Bhargava, "Convolutional codes for channels with deletion errors," in *Proc. 11th Canadian Workshop on Information Theory, 2009*, May 2009, pp. 136–139.

[14] C. Yi and J. Kliewer, "Error correction for asynchronous communication," in *IEEE 9th International Symposium on Turbo Codes and Iterative Information Processing (ISTC)*, Brest, France, September, 2016, pp. 310–314.

[15] L. F. Chaparro, E. Sejdic, A. Can, O. A. Alkishriwo, S. Senay, and A. Akan, "Asynchronous representation and processing of nonstationary signals: A time-frequency framework," *IEEE Signal Processing Magazine*, vol. 30, no. 6, pp. 42–52, 2013.

[16] L. R. Bahl, J. Cocke, F. Jelinek, and J. Raviv, "Optimal decoding of linear codes for minimizing symbol error rate," *IEEE Trans. Inf. Theory*, vol. 20, no. 2, pp. 284–287, Mar. 1974.

[17] S. Benedetto, D. Divsalar, G. Montorsi, and F. Pollara, "Serial concatenation of interleaved codes: Performance analysis, design, and iterative decoding," *IEEE Transactions on information Theory*, vol. 44, no. 3, pp. 909–926, 1998.