Constrained Codes for Passive RFID Communication

Ángela I. Barbero[†], Eirik Rosnes[‡], Guang Yang[‡], and Øyvind Ytrehus[‡]
[†]Department of Applied Mathematics, University of Valladolid, 47011 Valladolid, Spain
(e-mail: angbar@wmatem.eis.uva.es)
[‡]Department of Informatics, University of Bergen, N-5020 Bergen, Norway
(e-mail:{eirik,guang.yang,oyvind}@ii.uib.no)

***Abstract*—This paper[1] addresses coding for power transfer, modulation, and error control for the *reader-to-tag* channel in near-field passive radio frequency identification (RFID) systems using inductive coupling as a power transfer mechanism. Different assumptions on channel noise (including two different models for bit-shifts, insertions and deletions, and additive white Gaussian noise) are discussed. In particular, we propose a discretized Gaussian shift channel for the reader-to-tag channel in passive RFID systems, and design some new simple codes for *error avoidance* on this channel model. Finally, some simulation results are presented to compare the proposed codes to the Manchester code and two previously proposed codes for the bit-shift channel model.**

I. Introduction

Inductive coupling is a technique by which energy from one circuit is transferred to another without wires. Simultaneously, the energy transfer can be used as a vehicle for information transmission. This is a fundamental technology for near-field passive radio frequency identification (RFID) applications as well as lightweight sensor applications.

In the passive RFID application, a *reader*, containing or attached to a power source, controls and powers a communication session with a *tag*; a device without a separate power source. The purpose of the communication session may be, for examples, object identification, access control, or acquisition of sensor data.

Several standards exist that specify lower layer coding for RFID protocols. However, it seems that most standards employ codes that have been shown to be useful in general-purpose communication settings. Although this is justifiable from a pragmatic point of view, we observe that a thorough *information-theoretic approach* may reveal alternate coding schemes that, in general, can provide benefits in terms of reliability, efficiency, synchronization, simplicity, or security.

Operating range of a reader-tag pair is determined by communications requirements as well as power transfer requirements. To meet the communications requirements, the reader-to-tag and the tag-to-reader communication channels satisfy specified demands on communication transfer rate and reliability. To meet the power transfer requirements, the received power at the tag must be sufficiently large as to provide operating power at the tag.

According to [1, 2], with present day technology it is the power transfer requirements that present the bottleneck with respect to operating range for a two-way reader-tag communication session. Nevertheless, there is a value in determining the information-theoretic aspects, such as tradeoffs between reliability and transmission rate, of this communication: First, because future technologies may shift the relation between communication and power transfer requirements, and second, because present cheap tag technologies impose challenges on communication which are not directly related merely to received signal power.

In this paper, we address issues related to lower layer coding of information on inductively coupled channels, with emphasis on *coding for error control* for the *reader-to-tag* channel. The remainder of the paper is organized as follows. In Section II, we describe the characteristics of the reader-to-tag channel and discuss power issues and processing capabilities. Different relevant channel models are described in Section III. The section focuses on an idealized channel where the noise-free received signal replicates the transmitted signal, but where additive noise and timing inaccuracies may affect system performance. In particular, we propose a discretized Gaussian shift channel as a model for the reader-to-tag channel for passive near-field RFID. In Section IV, we present several new and very simple codes for the discretized Gaussian shift channel, as well as their encoding/decoding techniques. Simulation results are presented in Section V, and we draw some conclusions and present a discussion of future work in Section VI.

II. Characteristics of the Reader-to-Tag Channel

In this paper, we will be concerned with data transfer from a reader to a tag. An *information source* generates an information *frame* of k bits $\mathbf{u} = (u_1, \dots, u_k)$. The information frame is passed through an encoder to produce an encoded frame $\mathbf{c} = (c_1, \dots, c_n)$. The encoded frame is interpreted as a waveform that modulates a carrier wave, as shown in Fig. 1, [3, 4].

Please observe that the concept of a frame in this context refers to a collection of bits that belong together, for some semantic reason related to the application layer. The actual

[1]This work was supported by NFR through the ICC:RASC project, and by the NILS ABEL program.

encoder may work at a different length. Due to the strictly limited computing power of the tag, the actual encoder may work on a bit-by-bit basis, as in most of the examples later in this paper. The encoded frame length n may be fixed, depending only on k, or variable, depending on k and also on the information frame, but in general $n \geq k$.

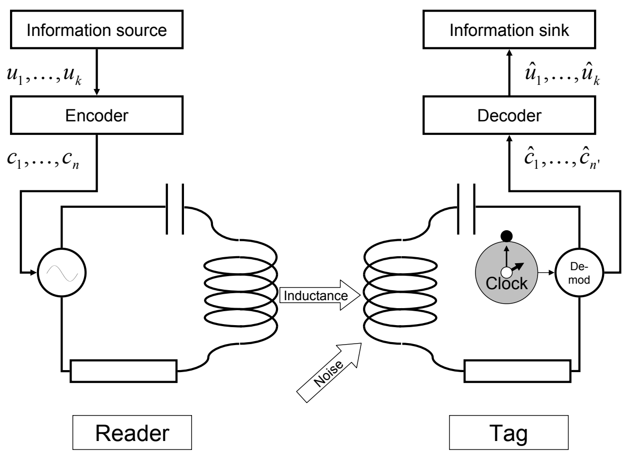

Fig. 1. A simplified view of the reader-to-tag channel.

Meanwhile, back at Fig. 1, the demodulator in the tag samples the physical waveform at time intervals determined by the tag's timing device, and converts it into an estimate $\hat{\mathbf{c}} = (\hat{c}_1, \ldots, \hat{c}_{n'})$ of the transmitted frame, where in general $n' \neq n$. Ideally, $\hat{\mathbf{c}}$ should be identical to $\mathbf{c}$, but additive noise, interference, timing inaccuracies, and waveform degradation due to limited bandwidth may contribute to corrupt the received frame $\hat{\mathbf{c}}$. We will discuss some of these signal corruptions later in this paper. A decoder at the tag subsequently attempts to recover an information frame $\hat{\mathbf{u}} = (\hat{u}_1, \ldots, \hat{u}_k)$ from $\hat{\mathbf{c}}$. Correct decoding is achieved if $\hat{\mathbf{u}} = \mathbf{u}$.

A. Power Issues

The tag in Fig. 1 has no internal power source. Rather, it collects the power derived from the carrier. After some initial transient delay, the tag's power circuitry will be charged sufficiently to provide operating power for the tag. Commonly, amplitude modulation, or more precisely *on-off keying (OOK)* is employed. In OOK, a "1" (resp. "0") is transmitted by the presence (resp. absence, or alternatively a low amplitude) of the carrier for the duration specified for transmitting that particular bit.

The transmitted power is limited by regulation [2]. However, the amount of transferred power can still be influenced by the encoding scheme used. Although the tag has no traditional battery or other means of accumulating energy over an extended period, it is possible to "ephemerally" store energy over a short time (say, a few bit periods) in the power circuitry. Thus, it makes sense to impose constraints on power content in the transmitted signal [5–7], for example, by demanding that m_P out of every n_P consecutive transmitted bits are 1's. Thus, a high power content (i.e., the ratio m_P/n_P is large) is an advantage. The precise manifestation of this advantage depends on technology and is difficult to measure. Therefore, we will consider different measures of power (to be defined below) as a figure of merit for a given coding scheme.

Formally, we will define the *power content* of a binary vector $\mathbf{a} \in \mathrm{GF}(2)^n$, denoted by $P(\mathbf{a})$, as the rational number $w(\mathbf{a})/n$, where $w(\cdot)$ denotes the Hamming weight of its binary argument.

Let $\mathcal{C}$ denote a block code or a variable-length code, i.e., a collection or set of codewords. Furthermore, let $\mathcal{C}^{[N]}$ be the set of sequences of length $N \geq 1$ over $\mathcal{C}$, i.e., the set of N consecutive codewords. The *average power* of $\mathcal{C}$ is defined as the average power content of the sequences in $\mathcal{C}^{[N]}$ as $N \to \infty$. For block codes, this average does not depend on N, and the average power of a block code $\mathcal{C}$ is $P_{\mathrm{avg}}(\mathcal{C}) = \frac{1}{|\mathcal{C}|}\sum_{\mathbf{a}\in\mathcal{C}} P(\mathbf{a})$. However, for variable-length codes, the average depends on N, and we need to consider the limit as $N \to \infty$. In general, the average power of a code $\mathcal{C}$ can be computed from [5]

$$P_{\mathrm{avg}}(\mathcal{C}) = \frac{\sum_{j=1}^{|\mathcal{C}|} w_j}{\sum_{j=1}^{|\mathcal{C}|} n_j}$$

where w_j and n_j denote the Hamming weight and length of the jth codeword in $\mathcal{C}$, respectively.

The *minimum sustainable power* of a block or variable-length code $\mathcal{C}$ is defined as $P_{\min}(\mathcal{C}) = \min_{\mathbf{a}\in\mathcal{C}} P(\mathbf{a})$. We remark that for codes defined by a state diagram, the various notions of power can refer to any cycle in the state diagram. Thus, $P_{\min}$ refers to the minimum average cycle weight of a cycle in the state diagram and can, for instance, be computed using Karp's algorithm [8]. The average power content P_{avg} can be computed from the stationary probabilities of the states in the state diagram.

As a final figure of merit, we will consider the *local minimum power* of a code $\mathcal{C}$ as the minimum positive value of the ratio m_P/n_P over all possible sequences in $\mathcal{C}^{[N]}$, for any finite value of N, where $n_P \geq m_P$ are arbitrary positive integers.

B. Processing Capability

Due to the limited tag power, processing capability is severely limited in a tag. This applies to any processing involved in whatever service the tag is supposed to provide, but also signal processing involved in receiving information.

1) Error Avoidance Versus Error Correction: For many communication channels studied in the literature, approaching channel capacity (or even achieving a significant coding gain over naive implementations) relies on error correction codes. However, although classes of codes are known for which the decoder can be efficiently encoded, the decoding process may still require a significant amount of processing. We will argue below that for channels for which the error probabilities depend on the transmitted data, reliability can be increased by using a code enforcing an appropriate set of modulation constraints. Such *error avoiding* codes can typically be decoded by a simple table, mapping received sequences into information estimates.

C. Applications of Murphy's Law: What Could Possibly Go Wrong?

For the information theorist, it is at this point interesting to consider whether the information transmitted from the reader to the tag is received correctly. (In fact, for the information theorist the channel is more interesting if, at first, it is *not* received correctly.) The received signal may be corrupted by additive noise or by incorrect timing. We will study these issues below, after some initial general remarks.

The frame error rate (FER), i.e., the probability of a frame error (FE), is given by

$$P(\text{FE}) = \sum_{\forall \mathbf{u}} P(\mathbf{u}) P(\text{FE}|\mathbf{u}) \tag{1}$$

where $P(\text{FE}|\mathbf{u})$ is the conditional probability of a frame error given that the information vector $\mathbf{u}$ has been transmitted. Now, we will use the general term *error pattern* to represent any random corruption of the transmitted frame. For convenience, we assume that there is a finite number of error patterns. The conditional probability $P(\text{FE}|\mathbf{u})$ can be expressed as a summation over error patterns e as follows

$$\begin{aligned} P(\text{FE}|\mathbf{u}) &= \sum_{\forall e} P(e|\mathbf{u}) P(\text{FE}|e, \mathbf{u}) \\ &= \sum_{\forall e} P(e|\mathbf{c}) P(\text{FE}|e, \mathbf{c}) \end{aligned} \tag{2}$$

where $P(e|\mathbf{c})$ is the conditional probability of a given error pattern e given a transmitted frame $\mathbf{c}$. The probability $P(e|\mathbf{c})$ depends only on the channel, but it determines which types of error we should be concerned about in the code design process. If $\mathbf{c}$ is transmitted and then corrupted by e so that $\hat{\mathbf{c}}$ is received, the decoder will make a frame error unless the decoder produces $\mathbf{c}$ when $\hat{\mathbf{c}}$ is the decoder input. Frame errors occur because there may exist different frames $\mathbf{c}$, $\mathbf{c}'$ and different error patterns e, e' so that $\mathbf{c}$ corrupted by e and $\mathbf{c}'$ corrupted by e' both yield the same received vector $\hat{\mathbf{c}}$.

Thus, the probability $P(\text{FE}|e, \mathbf{c}) = P(\text{FE}|\hat{\mathbf{c}})$ for some received vector $\hat{\mathbf{c}}$ represents the probability of frame error when vector $\hat{\mathbf{c}}$ is received. This quantity is a decoder property. The goal of the code design is to minimize the FER, given prior knowledge of the nature and behavior of signal corruptions. In Section III below, we will discuss some of these signal corruptions in greater detail.

III. Channel models

In this section, we will discuss different channel models for the reader-to-tag channel.

A. The Additive White Gaussian Noise (AWGN) Channel

A common assumption in communication engineering is that the channel is affected by AWGN. It is well-known that in this case the channel capacity is given by

$$C = 1/2 \log(1 + \text{SNR})$$

where SNR is the signal-to-noise ratio. We observe that on the reader-to-tag channel, the SNR is usually high, since readers need to pass a fair amount of power through the inductive channel in order to provide operating power for the tag.

Connecting the AWGN channel (resp. binary symmetric channel (BSC)) to the general error description in (1)-(2), it is well-known that the "likely error patterns" e will not depend on $\mathbf{u}$ and represent additive errors of low Euclidean weight or low Hamming weight. For the BSC, (2) simplifies to

$$P(\text{FE}|\mathbf{u}) \approx A_w p^w \tag{3}$$

for p small, where p is the BSC crossover probability, w is the minimum Hamming weight of an additive error pattern that will cause a frame error, and A_w is the number of such minimum-weight error patterns. Assuming an optimum decoder, A_w and w are parameters associated with the specific code.

B. The Traditional Bit-Shift Channel

In [5, 6], we considered a bit-shift channel with binary input $\mathbf{x} = (x_1, \ldots, x_L)$, where by assumption the value of x_1 is known to the receiver, and binary output $\mathbf{y} = (y_1, \ldots, y_{L'})$, where L' is equal to $L-1$, L, or $L+1$, and $L \geq 2$ is a positive integer.

The binary input sequence $\mathbf{x}$ is parsed into a sequence of phrases, where each phrase is a consecutive sequence of equal bits. Please observe that this parsing of $\mathbf{x}$ is done by the channel (and not by an encoder). Then, the integer sequence of phrase lengths $\tilde{\mathbf{x}}$ is transmitted over a channel. For instance, $\mathbf{x} = (0, 0, 1, 1, 1, 0, 0, 0, 1, 1, 1, 1)$ is transformed into the integer sequence $\tilde{\mathbf{x}} = (2, 3, 3, 4)$ of phrase lengths. The bit-shift channel is described by the model

$$\tilde{y}_i = \tilde{x}_i + \omega_i - \omega_{i-1} \tag{4}$$

where $i \geq 1$ and $(\omega_0, \omega_1, \ldots)$ is a data-dependent Markov chain with transition probabilities to be discussed below. Note that a positive (negative) value of ω_i corresponds to a right (left) bit-shift in the transition from time i to time $i+1$, $i \geq 1$. The value of ω_0 is always zero and acts like an initialization value.

In the setting of magnetic recording, the bit-shift channel has, for convenience, been used with (d, k)-constrained input sequences with $d \geq 2$, precoded with an accumulator [9, 10], which implies that the received sequence $\tilde{\mathbf{y}}$ will only contain positive integer values. In [5, 6], this traditional model was modified to accommodate the cases of length-1 and length-2 runlengths. Thus,

$$P(\omega_i = \omega | \tilde{x}_i = \tilde{x}, \omega_{i-1} = \omega')$$
$$= \begin{cases} \epsilon, & \text{if } \omega = -1 \text{ and } (\tilde{x} \neq 1 \text{ or } \omega' \neq 1) \\ 1-2\epsilon, & \text{if } \omega = 0 \text{ and } (\tilde{x} \neq 1 \text{ or } \omega' \neq 1) \\ \epsilon, & \text{if } \omega = 1 \text{ and } (\tilde{x} \neq 1 \text{ or } \omega' \neq 1) \\ \frac{1-2\epsilon}{1-\epsilon}, & \text{if } \omega = 0,\ \tilde{x} = 1, \text{ and } \omega' = 1 \\ \frac{\epsilon}{1-\epsilon}, & \text{if } \omega = 1,\ \tilde{x} = 1, \text{ and } \omega' = 1 \\ 0, & \text{otherwise} \end{cases}$$

where ϵ is the *bit-shift probability*, $0 \leq \epsilon \leq 1/2$.

Finally, if the received integer sequence $\tilde{\mathbf{y}}$ does contain a zero at position i, i.e., $\tilde{y}_i = 0$, then this coordinate is removed and $\tilde{y}_{i-1}$ and $\tilde{y}_{i+1}$ are added. The resulting sequence is denoted by $\hat{\mathbf{y}}$. For instance, if $\tilde{\mathbf{x}} = (1,3,2,1,1,1,2,1)$ and $\boldsymbol{\omega} = (0,-1,0,1,1,0,0,1,0)$, then $\tilde{\mathbf{y}} = (0,4,3,1,0,1,3,0)$ and $\hat{\mathbf{y}} = (4,3,2,3)$. Thus, the received integer sequence $\hat{\mathbf{y}}$ contains only positive integer values. The received integer sequence from the channel $\hat{\mathbf{y}}$ can be uniquely transformed back into a binary sequence $\mathbf{y}$ of phrases if the value x_1 is known, which it is by assumption.

C. The Discretized Gaussian Shift Channel

If the receiver resynchronizes its internal clock each time a bit is detected, the model in Section III-B needs to be modified. We will first introduce the *Gaussian shift channel*.

Suppose the reader transmits a run of $\tilde{x}$ consecutive equal symbols. This corresponds to an amplitude modulated signal of duration $\tilde{x}$. At the tag, we will assume that this is detected (according to the tag's internal clock) as having duration

$$\tilde{y} = \tilde{x} \cdot K \tag{5}$$

where K is a random variable with, in general, a Gaussian distribution $\mathbf{N}(\alpha, \varepsilon^2)$ with mean α and variance ε^2. Consecutive samplings of K are assumed to be independent. If $\alpha \neq 1$, it means that the tag has a systematic drift, which may affect the tag's ability to function at all. Thus, we will focus on the case $\alpha = 1$. With this definition, the input to the demodulator will be a sequence of alternating runs of high and low amplitude values; the detected duration $\tilde{y}$ of each run being a *real-valued* number.

We might attempt decoding directly at the Gaussian shift channel, but the computationally complexity will probably be high for the tag receiver. As a simplification, and to deal with the fact that $\tilde{y}$ may become negative (K has a normal distribution), which of course does not have any physical interpretation, we propose to discretize the timing and truncate K. The optimal choice for the quantization thresholds, i.e., the thresholds when mapping the real-valued numbers $\tilde{y}$ to *positive* integers, will depend on the code under consideration. For instance, suppose a and b are the only two possible runlengths, where $b > a$. Then, there is a single threshold and its optimum value from a *local* perspective[2] to determine if a or b was transmitted is

$$t = t(a,b) = \frac{2ab}{a+b}.$$

The corresponding decision error with one such decision is

$$Q\left(\frac{t-a}{a\varepsilon}\right) = Q\left(\frac{b-a}{(a+b)\varepsilon}\right) > Q\left(\frac{1}{\varepsilon}\right)$$

where $Q(x)$ is the probability that a sample of the standard normal distribution has value larger than x, i.e.,

$$Q(x) = \int_x^\infty \frac{1}{\sqrt{2\pi}} e^{-y^2/2} dy$$

or

$$Q(x) \approx \frac{e^{-x^2/2}}{x\sqrt{2\pi}}$$

for large x, and where the inequality follows from the fact that $(b-a)/(a+b)$ is smaller than 1. Furthermore, when $a = b-1$, $t = 2ab/(a+b) = 2b(b-1)/(2b-1)$ will approach $(a+b)/2 = b - 1/2$ as b goes to infinity.

We remark that we do not allow the mapping of a real-valued number (from the output of the Gaussian shift channel) to zero (or a negative integer), which means that the channel can not make a runlength disappear. This appears to be consistent with properties practical inductively coupled channels.

In general, let $\mathcal{Q}(\mathcal{A}, \mathcal{T})$ denote a quantization scheme with quantization values $\mathcal{A} = \{a_1, \ldots, a_{|\mathcal{A}|}\}$, where $1 \leq a_1 < \cdots < a_{|\mathcal{A}|} \leq L$, and L is some positive integer, and quantization thresholds $\mathcal{T} = \{t_2, \ldots, t_{|\mathcal{A}|}\}$, where $a_l < t_{l+1} < a_{l+1}$, $l = 1, \ldots, |\mathcal{A}|-1$. The quantization scheme works in the following way. Map a received real-valued number to an integer in $\mathcal{A}$ using quantization thresholds in $\mathcal{T}$, i.e., if the received real-valued number is in the range $[t_l, t_{l+1})$, $l = 2, \ldots, |\mathcal{A}|-1$, map it to a_l, if it is in the range $[t_{|\mathcal{A}|}, \infty)$, map it to $a_{|\mathcal{A}|}$, and, otherwise, map it to a_1.

Now, we define the discretized Gaussian shift channel with quantization scheme $\mathcal{Q}(\mathcal{A}, \mathcal{T})$ as the cascade of the Gaussian shift channel and the quantization scheme $\mathcal{Q}(\mathcal{A}, \mathcal{T})$, where the quantization scheme $\mathcal{Q}(\mathcal{A}, \mathcal{T})$ is applied to the real-valued sequence at the output of the Gaussian shift channel.

As an example, we can define a discretized Gaussian shift channel, where the quantization thresholds are chosen such that the integer sequence is obtained from the real-valued sequence by rounding its values to the nearest positive integer value. This particular quantization scheme will be denoted by $\mathcal{Q}_{\text{rounding}}$. As a further modification, we may introduce a parameter Γ into the quantization scheme $\mathcal{Q}_{\text{rounding}}$, and in this way get a family of discretized Gaussian shift channel. The resulting quantization scheme works in the following way. If the reader has transmitted a run of L symbols, the tag will

[2]We can do better with a maximum-likelihood detector which considers the whole transmitted sequence.

detect it as having length

$$\begin{cases} L-l, & \text{if } K \in \left[-\frac{2l+1}{2L}, -\frac{2l-1}{2L}\right) \text{ and} \\ & l = 1, \dots, \Gamma'-1 \\ L-\Gamma', & \text{if } K \in \left(-\infty, -\frac{2\Gamma'-1}{2L}\right) \\ L, & \text{if } K \in \left[-\frac{1}{2L}, \frac{1}{2L}\right) \\ L+l, & \text{if } K \in \left[\frac{2l-1}{2L}, \frac{2l+1}{2L}\right) \text{ and} \\ & l = 1, \dots, \Gamma-1 \\ L+\Gamma, & \text{if } K \in \left[\frac{2\Gamma-1}{2L}, \infty\right) \end{cases}$$

where $\Gamma \geq 1$ is a truncation integer parameter and $\Gamma' = \min(\Gamma, L-1)$. With $\Gamma = 1$, we denote the channel as the discretized Gaussian single-shift channel. With $\Gamma = 2$, the channel is called the discretized Gaussian double-shift channel, and so on. Now, if we want to express the discretized Gaussian single-shift channel in terms of runlengths with additive error terms ω (as in (4)), (4) is modified by (5) and discretization to

$$\tilde{y}_i = \tilde{x}_i + \omega_i$$

where

$$P(\omega_i = \omega | \tilde{x}_i = \tilde{x}) = \begin{cases} p(\tilde{x}), & \text{if } \omega = -1 \text{ and } \tilde{x} > 1 \\ 0, & \text{if } \omega = -1 \text{ and } \tilde{x} = 1 \\ 1 - 2p(\tilde{x}), & \text{if } \omega = 0 \text{ and } \tilde{x} > 1 \\ 1 - p(\tilde{x}), & \text{if } \omega = 0 \text{ and } \tilde{x} = 1 \\ p(\tilde{x}), & \text{if } \omega = 1 \text{ and } \tilde{x} \geq 1 \\ 0, & \text{otherwise} \end{cases}$$

and $p(L) = Q\left(\frac{1}{2L\varepsilon}\right)$.

As another example, we can define a quantization scheme $\mathcal{Q}(\mathcal{A}) = \mathcal{Q}(\mathcal{A}, \mathcal{T})$, where the quantization threshold $t_l = 2a_{l-1}a_l/(a_{l-1} + a_l)$, $l = 2, \dots, |\mathcal{A}|$. As will become clear later, this quantization scheme outperforms the general rounding scheme defined above. However, note that when $a_{|\mathcal{A}|-1} = a_{|\mathcal{A}|} - 1$ and $a_{|\mathcal{A}|}$ is large, the performance approaches the performance of the discretized Gaussian shift channel with quantization scheme $\mathcal{Q}_{\text{rounding}}$ for low values of ε.

We can make the following remarks in connection with the Gaussian shift channel.

(i) As can be seen from Fig. 2, when considering the "likely error patterns" e in (2), we need to be concerned mainly about the longest runs of equal symbols. The exception to this pragmatic rule occurs when, for some codes, it is possible to correct all shifts (up to some order, where a single shift is a shift of order one, a double shift is a shift of order two, and so on) corresponding to maximum-length runs.

(ii) In analogy with (3), we can *in principle* simplify (1) for *many simple codes* used on the discretized Gaussian shift channel to, respectively,

$$P(\text{FE}) \approx S_L \cdot p(L)$$

and

$$P(\text{FE}) \approx S_L \cdot p(L - 1/2)$$

with quantization schemes $\mathcal{Q}_{\text{rounding}}$ and $\mathcal{Q}(\mathcal{A})$, where $\mathcal{A} = \{a_1, \dots, a_{|\mathcal{A}|-2}, L-1, L\}$, and S_L is some constant depending on L, assuming that the most likely error event when using the code is connected with the confusion of runlengths of length L with some other run of length $L-1$. We omit the details, but will show examples later.

(iii) Error avoidance versus error control: Suppose we can design an error correction encoder that admits runlengths of length at most 2; that has a decoder that can correct all error events involving a single shift of a *single* run of length 2, but that will make a mistake if two or more such event occurs. Such a decoder should have a FER on the order of $p(2)^2$ (with quantization scheme $\mathcal{Q}_{\text{rounding}}$) for small ε. Observe from Fig. 2 that $p(2)^2 > p(1)$. Can we design a code with a simple decoder that behaves as $p(1)$? Yes, we can; see Sections IV-G, IV-H, and IV-J.

(iv) Observe that the discretized Gaussian single-shift channel is a *special form of an insertion-deletion channel*, which randomly may extend or shorten the runs of transmitted identical symbols, but where the statistics of this random process depend on the length of the runs. Codes for insertion-deletion channels have been studied, but to a moderate extent, and some of the best known codes, such as the Varshamov-Tenengolz codes [11] and the codes in [12], are apparently too complex for the application in question and also do not possess the appropriate modulation constraints, to be discussed below.

(v) An intelligent receiver tag should realize that any received run longer than the maximum run must be the result of an insertion. Thus, such insertions can trivially be corrected. In consequence, *for some codes*, the discretized Gaussian shift channel is approximately simply a special deletion channel that applies only to runs of

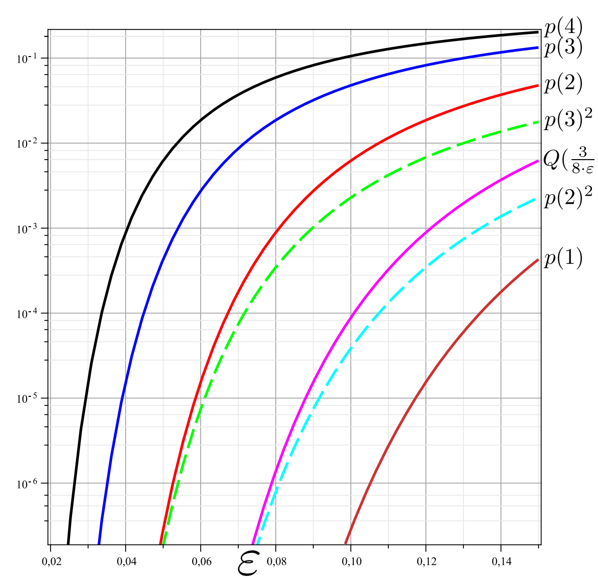

Fig. 2. Comparison of shift probabilities (with quantization scheme $\mathcal{Q}_{\text{rounding}}$) versus ε for runlengths 1, 2, 3, and 4.

maximum length.

(vi) In general, for any code and channel, a receiver may use a forward error correction scheme (FEC), or an automatic-repeat-request (ARQ) scheme asking for retransmissions if an error is detected. Obviously, error detection is computationally simpler than error correction. Indeed, ARQ is typically used in standard protocols for passive RFID, utilizing a standard embedded cyclic redundancy check code.

For the BSC it is further well-known that the FER associated with FEC is typically much higher than the probability of undetected error corresponding to ARQ. Counter-intuitively, this property does not necessarily apply with the discretized Gaussian shift channel.

IV. Coding schemes for the reader-to-tag channel

Among the encoding schemes in use in communication standards for RFID applications, one can find several codes that are popular in general communication protocols, such as NRZ, Manchester, Unipolar RZ, and Miller coding [2]. Here we will study the effect of some new encoding schemes for the reader-to-tag channel, considering power constraints (see Section II-A) and the communication channel described in Section III-C, i.e., the Gaussian shift channel. As a reference, we will provide the Manchester code (described in Section IV-B), and two variable-length codes presented in [5] (and described in Sections IV-C and IV-D, respectively) and designed for the bit-shift channel in Section III-B.

Before describing the specific code constructions, we will briefly explain the concept of constrained coding.

A. Runlength Limitations and Other Coding Constraints

We may desire and enforce that an encoded sequence satisfies certain constraints specified by a *constraint graph* [13–15]. These constraints may, for example, be the power constraints described in Section II-A, or runlength limitations, or a combination of these constraints. For the purpose of this paper, we shall denote a particular binary runlength limitation as $\mathrm{RLL}(\mathcal{L}_0, \mathcal{L}_1)$, where $\mathcal{L}_b$ is the set of admissible runlengths of binary symbol b. We can prove the following theorems.

Theorem 1: If a code satisfying the $\mathrm{RLL}([1, L], [1, L])$ limitation, where $[1, L] = \{1, 2, \ldots, L\}$, is used on the discretized Gaussian shift channel with quantization scheme $\mathcal{Q}_{\mathrm{rounding}}$ and with a maximum-likelihood decoder, then the FER behaves as $\mathcal{O}(p(L))$ for small ε.

Theorem 2: If a code satisfying the $\mathrm{RLL}([1, L], [1, L])$ limitation is used on the discretized Gaussian shift channel with quantization scheme $\mathcal{Q}([1, L])$ and with a maximum-likelihood decoder, then the FER behaves as $\mathcal{O}(p(L-1/2))$ for small ε.

The maximum rate of a constrained code is determined by the *capacity* of the constraint, which can readily be calculated from the constraint graph [13–15]. There exist several techniques [13–15] for designing an encoder (of code rate upper-bounded by the capacity), and we refer the interested reader to these works for further details.

B. The Manchester Code

The Manchester code is a very simple block code that maps 0 into 01, and 1 into 10. The code is popular in many communication protocols, but can observe that it also satisfies several of the criteria we can derive for a coding scheme to be used on a reader-to-tag discretized Gaussian shift channel: The maximum runlength is two; the average power is $1/2$; the minimum sustainable power is $1/2$; the local minimum power is $1/4$; the minimum Hamming distance is two, and the code is simple to decode. The performance of this code on the discretized Gaussian shift channel will be presented in Section V.

C. The Code $\{10, 011\}$ [5, 6]

The variable-length code $\{10, 011\}$ is single bit-shift error correcting, i.e., it corrects any single bit-shift on the traditional bit-shift channel from Section III-B, and has minimum sustainable power $1/2$, local minimum power $1/3$, and average power $3/5$. The rate of the code is $2/5$, the minimum runlength is 1, and the maximum runlength is 3. The performance of this code on the discretized Gaussian shift channel will be presented in Section V.

D. The Code $\{101, 01101\}$ [5]

The variable-length code $\{101, 01101\}$ is single bit-shift error detecting, i.e., it detects any single bit-shift on the traditional bit-shift channel from Section III-B, and has minimum sustainable power $3/5$, local minimum power $1/3$, and average power $5/8$. The rate of the code is $1/4$, the minimum runlength is 1, and the maximum runlength is 2. The performance of this code on the discretized Gaussian shift channel will be presented in Section V.

E. $\mathrm{RLL}(\{1, 2\}, \{1, 2\})$-Limited Codes

The capacity of the constraint $\mathrm{RLL}(\{1, 2\}, \{1, 2\})$ is 0.694. Furthermore, it follows from Theorems 1 and 2 that, similar to the Manchester code, any code with this runlength limitation has a FER on the order of $\mathcal{O}(p(2))$ and $\mathcal{O}(p(3/2))$, for small ε, on the discretized Gaussian shift channel with quantization scheme $\mathcal{Q}_{\mathrm{rounding}}$ and $\mathcal{Q}([1, 2])$, respectively.

Example 1: A two-state, rate-$2/3$ encoder for a $\mathrm{RLL}(\{1\}, \{1, \ldots, \infty\})$-constrained code is given in [14]. The encoder can be transformed into a four-state encoder for a $\mathrm{RLL}(\{1, 2\}, \{1, 2\})$-constrained code by a simple differential mapping. The encoder is shown in Fig. 3, while a very simple decoder/demapper is provided in Table I. The code has minimum sustainable power $1/3$, local minimum power $1/5$, and average power $1/2$.

Example 2: A code with a very simple encoding and decoding can be obtained by using bit-stuffing. The resulting code is a variable-length code. The encoder keeps the information symbols u_t, $t = 1, \ldots, k$, unmodified, but inserts an extra inverted symbol $1 - u_t$ if $u_t \equiv t \pmod 2$. The decoder destuffs the extra inserted symbols in a similar way. The encoder has rate $2/3$, minimum sustainable power $1/3$, local

TABLE I

LOOK-UP TABLE DECODING OF THE RLL($\{1,2\},\{1,2\}$)-CONSTRAINED CODE FROM EXAMPLE 1 AND WITH THE ENCODER GIVEN IN FIG. 3. BEFORE DECODING, IF A RUN OF AT LEAST THREE ZEROS OR ONES IS OBSERVED, CHANGE IT TO TWO, SINCE IT MOST LIKELY COMES FROM INSERTIONS.

Current word	Next bits	Decode to
000	Not possible	Detect insertion
001	010, 001 1, 011	00 01
010	0, 100 110, 101	11 10
011	0 (1 means insertion)	00
100	1 (0 means insertion)	01
101	010, 001 1, 011	11 10
110	0, 100 110, 101	00 01
111	Not possible	Detect insertion

minimum power $1/5$, average power $1/2$, and maximum runlength 2.

F. RLL($\{1\},\{1,2\}$)-Limited Codes

The capacity of the constraint RLL($\{1\},\{1,2\}$) is 0.406. Thus, a practical rate is no higher than $2/5$. However, the FER on the discretized Gaussian shift channel behaves (for small ε) as $\mathcal{O}(p(2))$ and $\mathcal{O}(p(3/2))$ with quantization scheme $\mathcal{Q}_{\text{rounding}}$ and $\mathcal{Q}([1,2])$, respectively. The only advantage over the RLL($\{1,2\},\{1,2\}$) limitation is a higher power content.

G. RLL($\{1,3\},\{1,3\}$)-Limited Codes

The capacity of the constraint RLL($\{1,3\},\{1,3\}$) is 0.552.

Theorem 3: The FER on the discretized Gaussian shift channel with quantization scheme $\mathcal{Q}_{\text{rounding}}$ for RLL($\{1,3\},\{1,3\}$)-constrained codes is on the order of $\mathcal{O}(p(1))$ for small ε.

Proof: The decoder works in the following way. Every received run of length 1 (on the discretized Gaussian shift channel with quantization scheme $\mathcal{Q}_{\text{rounding}}$) is kept as is, and every received run of length ≥ 2 is assumed to be a run of length 3. This decoder makes an error if a run of length 1 is extended by the Gaussian shift channel to length more than $3/2$ (this happens with probability $p(1)$), or if a run of length 3 is shortened to less than $3/2$ (this happens with probability $Q\left(\frac{3}{6\varepsilon}\right) = p(1)$). ■

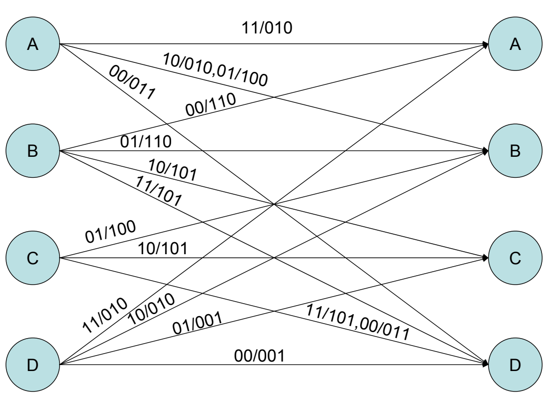

Fig. 3. An encoder for a RLL($\{1,2\},\{1,2\}$)-constrained code.

TABLE II

LOOK-UP TABLE DECODING OF THE RLL($\{1,3\},\{1,3\}$)-CONSTRAINED CODE FROM EXAMPLE 3 AND WITH THE ENCODER GIVEN IN FIG. 4. BEFORE DECODING, IF A RUN OF TWO ZEROS OR TWO ONES IS OBSERVED, CHANGE IT TO THREE, SINCE IT MOST LIKELY COMES FROM A DELETION OF A LENGTH THREE RUN. SIMILARLY, IF A RUN OF FOUR ZEROS OR ONES IS OBSERVED, CHANGE IT TO THREE.

Current word	Next bit pair	Decode to
00	Whatever	0
01	Not possible	Detect error
10	00 or 11 10	0 1
11	Whatever	1

We remark that on the discretized Gaussian shift channel with quantization scheme $\mathcal{Q}(\{1,3\})$, the error probability is of the same order for small ε, i.e., it behaves as $\mathcal{O}(p(1))$.

Example 3: A three-state, rate-$1/2$ encoder for a RLL($\{1,3\},\{1,3\}$)-constrained code is depicted in Fig. 4, while a very simple decoder/demapper is provided in Table II. The code has minimum sustainable power $1/4$, local minimum power $1/7$, and average power $13/24$.

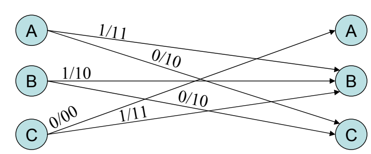

Fig. 4. An encoder for a RLL($\{1,3\},\{1,3\}$)-constrained code.

Example 4: A code with a very simple encoding and decoding can be obtained by using bit-stuffing. The resulting code is a variable-length code. The encoder keeps the information symbols u_t, $t = 1,\ldots,k$, unmodified, but inserts a pair of extra symbols $(u_t, 1-u_t)$ if $u_t \equiv t \pmod 2$. The decoder destuffs the extra inserted symbols in a similar way. The encoder has rate $1/2$, minimum sustainable power $1/4$, local minimum power $1/7$, average power $1/2$, and allowed runlengths 1 and 3.

H. RLL($\{1\},\{1,3\}$)-Limited Codes

The capacity of the constraint RLL($\{1\},\{1,3\}$) is 0.347. Furthermore, there is no difference in the asymptotic FER (i.e., the FER for small values of ε) with respect to RLL($\{1,3\},\{1,3\}$)-limited codes. Thus, the only advantage over the RLL($\{1,3\},\{1,3\}$) limitation is a higher power content.

Example 5: The variable-length RLL($\{1\},\{1,3\}$)-constrained code with codewords $\{01, 0111\}$ has rate $1/3$, minimum sustainable power $1/2$, local minimum power $1/3$, and average power $2/3$.

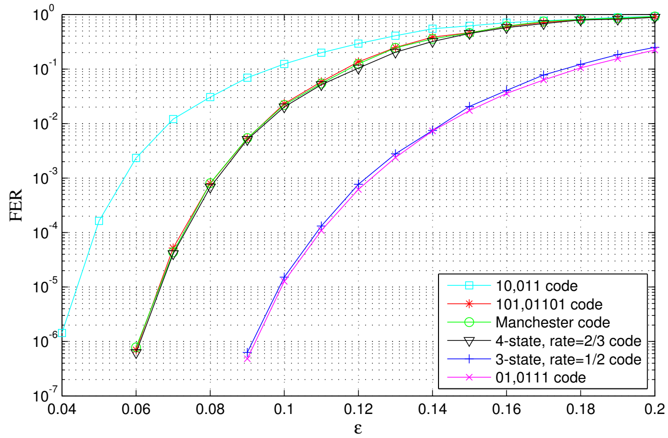

Fig. 5. FER on the discretized Gaussian shift channel as a function of ε for different codes.

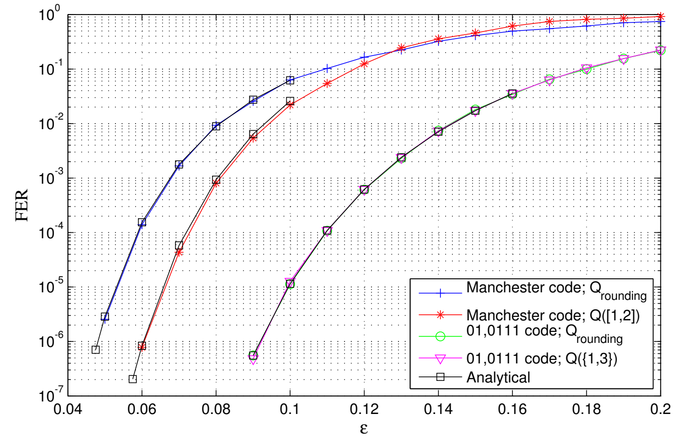

Fig. 6. A comparison of the FER (as a function of ε) on the discretized Gaussian shift channel with two different quantization schemes and with analytical expressions for the asymptotic performance, for the Manchester code and the variable-length code $\{01, 0111\}$.

I. RLL$(\{1,2,4\},\{1,2,4\})$*-Limited Codes*

Codes satisfying the constrains

$$\text{RLL}(\{1,2,4\},\{1,2,4\}),\ \text{RLL}(\{1,2\},\{1,2,4\}),\ \text{and}\ \text{RLL}(\{1\},\{1,2,4\})$$

have capacities 0.811, 0.758, and 0.515, respectively. The latter constraint may be attractive from a power transfer point of view; the two former constraints admit code rates of $4/5$ and $3/4$, respectively, but may be hard to implement. For the RLL$(\{1\},\{1,2,4\})$ constraint, a rate-1/2, 6-state encoder can be designed using the state-splitting algorithm from [16]. Finally, we remark that the FER on the discretized Gaussian shift channel is on the order of $\mathcal{O}(p(2))$ and $\mathcal{O}(p(3/2))$ with the quantization scheme $\mathcal{Q}_{\text{rounding}}$ and $\mathcal{Q}(\{1,2,4\})$, respectively, for small values of ε for these codes.

J. Related Constraints

Any RLL$(\{3^i : i = 0,\dots,L\},\{3^i : i = 0,\dots,L\})$-limited code, for any positive integer L, has a FER of the order of $\mathcal{O}(p(1))$ (with both quantization schemes $\mathcal{Q}_{\text{rounding}}$ and $\mathcal{Q}(\{3^i : i = 0,\dots,L\})$) for small ε. This can be shown with a similar argument to that used to prove Theorem 3. We remark here that the $\mathcal{O}(p(1))$ performance guarantee under the quantization scheme $\mathcal{Q}_{\text{rounding}}$ assumes that the decoder deals with non-admissible (with respect to the code) observed runlengths in the appropriate way. Notice that the capacity seems to approach a limit at about 0.58 as L increases. Thus, there seems to be no immediate practical advantage on extending these ideas further.

V. Simulation Results

In this section, we provide some simulation results of some of the above-mentioned codes on the discretized Gaussian shift channel. In particular, we consider the Manchester code from Section IV-B, the $\{10, 011\}$ code from Section IV-C, and the $\{101, 01101\}$ code from Section IV-D, in addition to the newly designed codes from Examples 1, 3, and 5. The information block length k is chosen to be 40 bits.

In Fig. 5, we have plotted the FER performance of these codes as function of ε on the discretized Gaussian shift channel with quantization scheme $\mathcal{Q}([1,2])$ for the Manchester code, for the $\{101, 01101\}$ code from Section IV-D, and for the code from Example 1, with quantization scheme $\mathcal{Q}([1,3])$ for the $\{10, 011\}$ code from Section IV-C, and with quantization scheme $\mathcal{Q}(\{1,3\})$ for the remaining codes. As can be observed from the figure, the RLL$(\{1,3\},\{1,3\})$-constrained code from Example 3 and the RLL$(\{1\},\{1,3\})$-constrained code from Example 5 have the best error rate performance, while the variable-length code $\{10, 011\}$ designed in [5, 6] for the traditional bit-shift channel has the worst performance among the simulated codes. However, this is not surprising, since this code has not been designed for the discretized Gaussian shift channel.

In Fig. 6, we have compared the performance of two different codes, namely the RLL$(\{1,2\},\{1,2\})$-constrained Manchester code and the RLL$(\{1\},\{1,3\})$-constrained code $\{01, 0111\}$ from Example 5 with two different quantization schemes. We have used the quantization schemes simulated in Fig. 5 and the quantization scheme $\mathcal{Q}_{\text{rounding}}$. Notice that there is no performance difference between the two quantization schemes for the RLL$(\{1\},\{1,3\})$-constrained code, while there is a significant performance difference for the other code. This is consistent with our earlier discussion in Section IV. In the figure, we also show analytical expressions for the asymptotic performance which depend on both the quantization scheme used and the particular decoding algorithm. These expressions match perfectly with the simulation results. For instance, for the RLL$(\{1\},\{1,3\})$-constrained code from Example 5, a detailed analysis shows that the FER (with both quantization schemes $\mathcal{Q}(\{1,3\})$ and $\mathcal{Q}_{\text{rounding}}$) is approximately

$$k \cdot Q\left(\frac{1}{2\varepsilon}\right) = k \cdot p(1)$$

as ε becomes smaller. For the Manchester code, the corresponding expressions are

$$(3k/2 + 1/2) \cdot Q\left(\frac{1}{3\varepsilon}\right) = (3k/2 + 1/2) \cdot p(3/2)$$

and

$$k/4 \cdot Q\left(\frac{1}{4\varepsilon}\right) = k/4 \cdot p(2)$$

with quantization schemes $\mathcal{Q}([1,2])$ and $\mathcal{Q}_{\text{rounding}}$, respectively. Finally, we remark that we have used look-up table decoding in all simulations. For instance, for the codes from Examples 1 and 3, we have used Tables I and II, respectively, in the decoding.

VI. Conclusion and Future Work

In this work, we have considered coding for power transfer, modulation, and error control for the reader-to-tag channel in near-field passive RFID systems. We have discussed power issues and proposed the (discretized) Gaussian shift channel as a channel model for the reader-to-tag channel in near-field passive RFID systems. Furthermore, some new simple codes for error avoidance on this channel model were presented and their performance were compared to the Manchester code and two previously proposed codes for the bit-shift channel model.

As future work, we will consider other important aspects of the inductively coupled channel and incorporate these into a more sophisticated channel model. For instance, how will the designed codes perform on a hybrid channel which also incorporates AWGN? Another interesting topic for future work is the computation of the capacity (or upper and lower bounds on it) of the different proposed channel models. We intend to address these issues in future papers.

References

[1] P. V. Nikitin and K. V. S. Rao, "Performance limitations of passive UHF RFID systems," in *Proc. IEEE Antennas and Propagation Society International Symposium*, Albuquerque, NM, Jul. 2006, pp. 1011–1014.

[2] M. Bolić, D. Simplot-Ryl, and I. Stojmenović, Eds., *RFID Systems: Research Trends and Challenges*. Wiley, 2010.

[3] G. Horler, "Inductively coupled telemetry," in *Sensors for Environment, Health and Security*, ser. NATO Science for Peace and Security Series C: Environmental Security, M.-I. Baraton, Ed. Springer Netherlands, 2009, pp. 239–252.

[4] P. V. Nikitin, K. V. S. Rao, and S. Lazar, "An overview of near field UHF RFID," in *Proc. IEEE Int. Conf. RFID*, Grapevine, TX, Mar. 2007, pp. 167–174.

[5] E. Rosnes, Á. I. Barbero, and Ø. Ytrehus, "Coding for inductively coupled channels," 2010, *IEEE Trans. Inf. Theory*, submitted for publication.

[6] ——, "Coding for a bit-shift channel with applications to inductively coupled channels," in *Proc. IEEE Global Telecommun. Conf. (GLOBECOM)*, Honolulu, HI, Nov./Dec. 2009.

[7] Á. I. Barbero, G. D. Horler, E. Rosnes, and Ø. Ytrehus, "Modulation codes for reader-tag communication on inductively coupled channels," in *Proc. Int. Symp. Inf. Theory and its Appl. (ISITA)*, Auckland, New Zealand, Dec. 2008, pp. 578–583.

[8] R. M. Karp, "A characterization of the minimum cycle mean in a digraph," *SIAM J. Discr. Math.*, vol. 23, pp. 309–311, 1978.

[9] S. Shamai and E. Zehavi, "Bounds on the capacity of the bit-shift magnetic recording channel," *IEEE Trans. Inf. Theory*, vol. 37, no. 3, pp. 863–872, May 1991.

[10] S. Baggen, V. Balakirsky, D. Denteneer, S. Egner, H. Hollmann, L. Tolhuizen, and E. Verbitskiy, "Entropy of a bit-shift channel," *IMS Lectures Notes-Monograph Series Dynamics & Stochastics*, vol. 48, pp. 274–285, 2006.

[11] T. Kløve, "Codes correcting a single insertion/deletion of a zero or a single peak-shift," *IEEE Trans. Inf. Theory*, vol. 41, no. 1, pp. 279–283, Jan. 1995.

[12] Z. Liu and M. Mitzenmacher, "Codes for deletion and insertion channels with segmented errors," *IEEE Trans. Inf. Theory*, vol. 56, no. 1, pp. 224–232, Jan. 2010.

[13] K. A. S. Immink, *Coding Techniques for Digital Recorders*. Prentice Hall, 1991.

[14] B. H. Marcus, P. H. Siegel, and J. K. Wolf, "Finite-state modulation codes for data storage," *IEEE J. Sel. Areas Commun.*, vol. 10, no. 1, pp. 5–37, Jan. 1992.

[15] B. H. Marcus, R. M. Roth, and P. H. Siegel, *Constrained Systems and Coding for Recording Channels*, ser. Handbook of Coding Theory. Elsevier, North-Holland, 1998, pp. 1635–1764.

[16] R. L. Adler, D. Coppersmith, and M. Hassner, "Algorithms for sliding block codes–An application of symbolic dynamics to information theory," *IEEE Trans. Inf. Theory*, vol. IT-29, no. 1, pp. 5–22, Jan. 1983.