On Universal Decoding over Memoryless Channels with the Krichevsky-Trofimov Estimator

Henrique K. Miyamoto, Sheng Yang

On Universal Decoding over Memoryless Channels with the Krichevsky–Trofimov Estimator

Henrique K. Miyamoto and Sheng Yang
Laboratoire des Signaux et Systèmes (L2S)
Université Paris-Saclay, CNRS, CentraleSupélec
Gif-sur-Yvette, France
Email: {henrique.miyamoto, sheng.yang}@centralesupelec.fr

***Abstract*—We study the problem of universal decoding over memoryless channels with a decoder based on the Krichevsky–Trofimov estimator. We show that this decoder is random-coding universal for codebooks of any size, i.e., despite being ignorant of the channel in use, it has asymptotically the same random-coding error exponent as the optimal maximum-likelihood decoder for that channel. Then, we incorporate this decoding rule in schemes to decode practical linear block codes and convolutional codes when the channel is unknown to the receiver. Numerical results show that efficient performance can be achieved even for moderate blocklength or constraint length.**

I. Introduction

When communications take place in practice, the channel is often unknown, which precludes the use of the optimal maximum likelihood (ML) decoder. Instead, it is usually estimated by sending a training sequence known to the receiver, which can then study the statistics of the corresponding received sequence to get an estimate of the channel law. This strategy comes, nonetheless, with a trade-off: a sequence too short results in an inaccurate estimation, while a sequence too long harms the communication rate. A possible way to overcome this is to use universal decoders for the family of channels in consideration. These are decoders that, despite having no prior knowledge on the specific channel in use, can asymptotically achieve the same random-coding error exponent as the optimal ML decoder tuned for that channel.

Among known universal decoders for finite-alphabet channels, we mention the maximum mutual information (MMI) [1], [2] and maximum conditional entropy [3] rules for discrete memoryless channels; the decoder based on Lempel–Ziv (LZ) parsing for finite-state channels [3], [4]; and merged decoders for general families of channels [5]. Universal decoding for Gaussian intersymbol interference channels has been studied in [6], [7], and other recent developments for more sophisticated setups include [8]–[11].

Universal decoding has mostly remained a theoretical topic so far. Indeed, a number of potential issues appear when trying to implement them in practice. First, the results are asymptotic with blocklength, and performance with finite blocklength may be far from the promised one. Then, they are stated for random codes, and might not hold for practical codes that have strong algebraic structure. Finally, the computational complexity for implementing universal decoding rules is usually very high: they typically require comparing the universal metric of each codeword (this has the same complexity as that of an ML decoder that compares each codeword, but becomes unpractical for large codebooks).

An exception to that are the modified stack algorithms proposed in [3], [12], [13] to decode convolutional codes. There, the usual Fano metric, which is consistent with ML decoding, is replaced by universal metrics that do not depend on the channel in use. These metrics lend themselves to asymptotic error probabilities similar to those of stack decoding with the Fano metric and ML decoding with Viterbi algorithm, both of which depend on the channel law.

In this work, we consider a universal decoder based on the (conditional) Krichevsky–Trofimov (KT) estimator [14], which enjoys a simple sequential update expression. It corresponds to the weighting of posterior distributions with Dirichlet priors, and is asymptotically optimal in the minimax sense [15]. A decoding rule based on the KT estimator has been proposed to decode convolutional codes over memoryless channels, and shown to be random-coding universal for codebooks of two codewords in [12]. A Bayesian motivation for its use, in the particular case of binary symmetric channels, was given in [13]. Here, we first extend these results, by giving a more general justification for this rule, and showing that it is in fact random-coding universal with codebooks of any size, for the family of discrete memoryless channels.

Then, we incorporate the KT-based decoder in decoding schemes for practical codes over unknown memoryless channels. For linear block codes of moderate blocklength, our decoder performs exhaustive search on a modified version of the original code. In numerical results with Golay codes, the KT decoder is able to track the block error rate of the ML decoder. For convolutional codes, we implement a modified version of the sequential stack decoding proposed in [3], [12], exploiting the structure of the KT decoder to apply further practical simplifications. Numerical results reveal a performance that is not too far from that of a stack decoder that knows the channel.

This paper is organised as follows. Section II introduces preliminary results and notation on universal decoding and the method of types. Random-coding universality of the KT decoder is derived in Section III. In Section IV, we study decoding schemes for practical codes. Section V concludes the paper with some future perspectives.

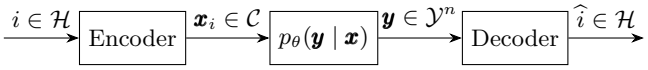

Fig. 1. Communication model.

II. Preliminaries

A. Universal Decoding

Consider the communication model depicted in Fig. 1: a code $\mathcal{C} \coloneqq \{\boldsymbol{x}_1, \dots, \boldsymbol{x}_M\} \subseteq \mathcal{B}_n$, of blocklength n and rate $R = (\log M)/n$, has its codewords selected from the input set $\mathcal{B}_n \subseteq \mathcal{X}^n$. A message $i \in \mathcal{H} \coloneqq \{1, 2, \dots, M\}$ is uniformly selected, encoded as codeword $\boldsymbol{x}_i \in \mathcal{C}$, and transmitted through the channel p_θ, producing the received sequence $\boldsymbol{y} \in \mathcal{Y}^n$. The channel belongs to a parametric family $\mathcal{F} \coloneqq \{p_\theta(\boldsymbol{y} \mid \boldsymbol{x}) : \theta \in \Theta\}$ with input and output alphabets, respectively, $\mathcal{X} \coloneqq \{1, \dots, |\mathcal{X}|\}$ and $\mathcal{Y} \coloneqq \{1, \dots, |\mathcal{Y}|\}$.

The decoder implements a rule $\phi\colon \mathcal{Y}^n \to \mathcal{H}$ that maps the channel output $\boldsymbol{y} \in \mathcal{Y}^n$ to a guess $\widehat{i} = \phi(\boldsymbol{y})$ of the original message. The maximum *a posteriori* (MAP) rule $\phi_{\mathrm{MAP}}(\boldsymbol{y}) = \arg\max_{i \in \mathcal{H}} p_\theta(\boldsymbol{x}_i \mid \boldsymbol{y})$ minimises the probability of error, and coincides with the ML rule $\phi_{\mathrm{ML}}(\boldsymbol{y}) = \arg\max_{i \in \mathcal{H}} p_\theta(\boldsymbol{y} \mid \boldsymbol{x}_i)$ when messages are equiprobable. Both require knowing the channel distribution p_θ. A *universal decoder*, on the other hand, is a decoder that, despite having no prior information on the channel p_θ (other than the family $\mathcal{F}$ to which it belongs), can asymptotically achieve the same random-coding error exponent as the optimal ML decoder tuned for p_θ.

When considering random codes, we suppose that the codewords in $\mathcal{C}$ are chosen independently and uniformly among the sequences in $\mathcal{B}_n$. Let $P_{\theta,\phi}(\mathrm{error})$ denote the average error probability (over messages and codes) when decoder ϕ is used in channel p_θ. We adopt the following definitions from [5].

Definition 1: A sequence of decoders $(u_n)_{n\in\mathbb{N}}$ is said to be *random-coding (weakly) universal* for the family of channels $\mathcal{F} = \{p_\theta(\boldsymbol{y} \mid \boldsymbol{x}) : \theta \in \Theta\}$ and the sequence of input sequences $(\mathcal{B}_n)_{n\in\mathbb{N}}$, if $\lim_{n\to\infty} \frac{1}{n} \log\left(\frac{P_{\theta,u_n}(\mathrm{error})}{P_{\theta,\mathrm{ML}}(\mathrm{error})}\right) = 0$, for all $\theta \in \Theta$; and *random-coding strongly universal*, if $\lim_{n\to\infty} \sup_{\theta\in\Theta} \frac{1}{n} \log\left(\frac{P_{\theta,u_n}(\mathrm{error})}{P_{\theta,\mathrm{ML}}(\mathrm{error})}\right) = 0$.

We borrow the following lemma from [5, Eq. (25)] (see also [3, Corollary 1]), which will be helpful in proving universality.

Lemma 1 ([5, Eq. (25)]): Consider two decoders of the form $\phi_j(\boldsymbol{y}) = \arg\max_{i\in\mathcal{H}} f_j(\boldsymbol{x}_i, \boldsymbol{y})$, for functions $f_j\colon \mathcal{X}\times\mathcal{Y} \to \mathbb{R}$, $j \in \{1, 2\}$. Their average error probabilities are related by

$$\frac{P_{\theta,\phi_2}(\mathrm{error})}{P_{\theta,\phi_1}(\mathrm{error})} \le \max_{(\boldsymbol{x},\boldsymbol{y})\in\mathcal{B}_n\times\mathcal{Y}^n} \frac{\left|\mathcal{E}_{\phi_2}(\boldsymbol{x},\boldsymbol{y})\right|}{\left|\mathcal{E}_{\phi_1}(\boldsymbol{x},\boldsymbol{y})\right|}, \tag{1}$$

where $\mathcal{E}_{\phi_j}(\boldsymbol{x},\boldsymbol{y}) \coloneqq \left\{\boldsymbol{x}' \in \mathcal{B}_n : f_j(\boldsymbol{x}',\boldsymbol{y}) \ge f_j(\boldsymbol{x},\boldsymbol{y})\right\}$.

The next lemma, inspired by [12], gives an upper bound on $|\mathcal{E}_\phi(\boldsymbol{x},\boldsymbol{y})|$ when the decoder metric is a probability $\widehat{p}(\cdot \mid \boldsymbol{y})$ on $\mathcal{B}_n$. It formalises the fact that there can only be so many sequences $\boldsymbol{x}' \in \mathcal{B}_n$ with probability larger than $\widehat{p}(\boldsymbol{x} \mid \boldsymbol{y})$.

Lemma 2: Let $\boldsymbol{x} \in \mathcal{B}_n$ and $\boldsymbol{y} \in \mathcal{Y}^n$. Consider a decoder of the form $\phi(\boldsymbol{y}) = \arg\max_{i\in\mathcal{H}} \frac{1}{n} \log \widehat{p}(\boldsymbol{x}_i \mid \boldsymbol{y})$, with $\widehat{p}(\cdot \mid \boldsymbol{y})$ a probability distribution over $\mathcal{B}_n \subseteq \mathcal{X}^n$. Then,

$$\left|\mathcal{E}_\phi(\boldsymbol{x},\boldsymbol{y})\right| \le \frac{1}{\widehat{p}(\boldsymbol{x} \mid \boldsymbol{y})}. \tag{2}$$

Proof: Let $\boldsymbol{X}'$ be a uniform random variable over $\mathcal{B}_n$. We have, by Markov's inequality,

$$\begin{aligned}
|\mathcal{E}_\phi(\boldsymbol{x},\boldsymbol{y})| &= \left|\left\{\boldsymbol{x}' : \frac{1}{n}\log\widehat{p}(\boldsymbol{x}' \mid \boldsymbol{y}) \ge \frac{1}{n}\log\widehat{p}(\boldsymbol{x} \mid \boldsymbol{y})\right\}\right| \\
&= |\mathcal{B}_n| \cdot \mathbb{P}\left[\widehat{p}(\boldsymbol{X}' \mid \boldsymbol{y}) \ge \widehat{p}(\boldsymbol{x} \mid \boldsymbol{y})\right] \\
&\le |\mathcal{B}_n| \frac{\sum_{\boldsymbol{x}'\in\mathcal{B}_n} \frac{1}{|\mathcal{B}_n|}\widehat{p}(\boldsymbol{x}' \mid \boldsymbol{y})}{\widehat{p}(\boldsymbol{x} \mid \boldsymbol{y})} = \frac{1}{\widehat{p}(\boldsymbol{x} \mid \boldsymbol{y})}.
\end{aligned}$$

■

B. Method of Types

We briefly recall definitions from the method of types in order to introduce our notation. The *type* of a sequence $\boldsymbol{x} \in \mathcal{X}^n$ is the probability distribution $\pi_{\boldsymbol{x}}$ given by the relative frequency of symbols, i.e., $\pi_{\boldsymbol{x}}(x) = a_{\boldsymbol{x}}(x)/n$, where $a_{\boldsymbol{x}}(x)$ is the number of times that symbol $x \in \mathcal{X}$ appears in sequence $\boldsymbol{x}$. We denote $\boldsymbol{a}_{\boldsymbol{x}} \coloneqq \big(a_{\boldsymbol{x}}(x)\big)_{x\in\mathcal{X}}$ the vector of counts. The *type class* $\mathcal{T}^{(n)}(\pi_{\boldsymbol{x}})$ is the set of all sequences in $\mathcal{X}^n$ that have type $\pi_{\boldsymbol{x}}$. We denote $H(\pi_{\boldsymbol{x}}) \coloneqq -\sum_{x\in\mathcal{X}} \pi_{\boldsymbol{x}}(x) \log \pi_{\boldsymbol{x}}(x)$.

Given sequences $\boldsymbol{x} \in \mathcal{X}^n$ and $\boldsymbol{y} \in \mathcal{Y}^n$, denote $\boldsymbol{a}_{\boldsymbol{x},\boldsymbol{y}} \coloneqq \big(a_{\boldsymbol{x},\boldsymbol{y}}(x,y)\big)_{(x,y)\in\mathcal{X}\times\mathcal{Y}}$, where $a_{\boldsymbol{x},\boldsymbol{y}}(x,y)$ is the number of times that $(x,y) \in \mathcal{X}\times\mathcal{Y}$ appears in the joint sequence $\boldsymbol{x}\otimes\boldsymbol{y} \coloneqq \big((x_1,y_1),\dots,(x_n,y_n)\big) \in (\mathcal{X}\times\mathcal{Y})^n$. Note that $a_{\boldsymbol{y}}(y) = \sum_{x\in\mathcal{X}} a_{\boldsymbol{x},\boldsymbol{y}}(x,y)$. The *joint type* is $\pi_{\boldsymbol{x},\boldsymbol{y}}(x,y) = a_{\boldsymbol{x},\boldsymbol{y}}(x,y)/n$, and the *marginal type*, $\pi_{\boldsymbol{y}}(y) = a_{\boldsymbol{y}}(y)/n$. The *conditional type* $\pi_{\boldsymbol{x}|\boldsymbol{y}}$ is given by $\pi_{\boldsymbol{x}|\boldsymbol{y}}(x \mid y) \coloneqq \pi_{\boldsymbol{x},\boldsymbol{y}}(x,y)/\pi_{\boldsymbol{y}}(y)$, if $\pi_{\boldsymbol{y}}(y) \ne 0$, and 0 otherwise. We then have

$$\pi_{\boldsymbol{x}|\boldsymbol{y}}(\boldsymbol{x} \mid \boldsymbol{y}) \coloneqq \prod_{x\in\mathcal{X}} \prod_{y\in\mathcal{Y}} \pi_{\boldsymbol{x}|\boldsymbol{y}}(x \mid y)^{a_{\boldsymbol{x},\boldsymbol{y}}(x,y)}.$$

We adopt the notations

$$H(\pi_{\boldsymbol{x}|\boldsymbol{y}} \mid \pi_{\boldsymbol{y}}) \coloneqq -\sum_{y\in\mathcal{Y}} \pi_{\boldsymbol{y}}(y) \sum_{x\in\mathcal{X}} \pi_{\boldsymbol{x}|\boldsymbol{y}}(x \mid y) \log \pi_{\boldsymbol{x}|\boldsymbol{y}}(x \mid y), \tag{3}$$

and $I(\pi_{\boldsymbol{x}}\colon \pi_{\boldsymbol{y}}) \coloneqq H(\pi_{\boldsymbol{x}}) + H(\pi_{\boldsymbol{y}}) - H(\pi_{\boldsymbol{x},\boldsymbol{y}})$.

III. Random-Coding Universality

Let us consider the class of discrete memoryless channels (DMCs). Denoting $\boldsymbol{x} \coloneqq x_1^n \coloneqq x_1 x_2 \cdots x_n \in \mathcal{X}^n$ and, analogously, $\boldsymbol{y} \coloneqq y_1^n \in \mathcal{Y}^n$, the channel law can be written as

$$p_\theta(\boldsymbol{y} \mid \boldsymbol{x}) = \prod_{i=1}^n p_\theta(y_i \mid x_i) = \prod_{x\in\mathcal{X}} \prod_{y\in\mathcal{Y}} p_\theta(y \mid x)^{a_{\boldsymbol{x},\boldsymbol{y}}(x,y)}. \tag{4}$$

This family can be parametrised by vectors $\theta \in \left(\Delta^{(|\mathcal{Y}|-1)}\right)^{|\mathcal{X}|}$, where $\Delta^k \subseteq \mathbb{R}^{k+1}$ denotes the k-dimensional simplex. The next lemma, borrowed from [3, Lemma 1, Eq. (23a)], states a property of the family of channels (4).

Lemma 3 ([3, Lemma 1]): Let $\boldsymbol{y} \in \mathcal{Y}^n$ and $\boldsymbol{x} \in \mathcal{B}_n$, with $\mathcal{B}_n = \mathcal{X}^n$ or $\mathcal{B}_n = \mathcal{T}^{(n)}(\pi)$ for some type π on $\mathcal{X}^n$. Then,

$$\begin{aligned}\left|\mathcal{E}_{\mathrm{ML}}(\boldsymbol{x},\boldsymbol{y})\right| &= \left|\{\boldsymbol{x}' \in \mathcal{B}_n : \ p_\theta(\boldsymbol{y} \mid \boldsymbol{x}') \geq p_\theta(\boldsymbol{y} \mid \boldsymbol{x})\}\right| \\ &\geq 2^{nH(\pi_{\boldsymbol{x}|\boldsymbol{y}}|\pi_{\boldsymbol{y}})}(n+1)^{-|\mathcal{X}||\mathcal{Y}|}. \end{aligned} \tag{5}$$

We are interested in studying a decoding rule based on the KT estimator [14]. The KT estimator for a sequence $\boldsymbol{x} \in \mathcal{X}^n$ is the probability distribution

$$p_{\mathrm{KT}}(\boldsymbol{x}) = \frac{\Gamma\left(\frac{|\mathcal{X}|}{2}\right)}{\Gamma\left(\frac{1}{2}\right)^{|\mathcal{X}|}} \frac{\prod_{x\in\mathcal{X}} \Gamma\left(a_{\boldsymbol{x}}(x) + \frac{1}{2}\right)}{\Gamma\left(n + \frac{|\mathcal{X}|}{2}\right)}, \tag{6}$$

where $\Gamma(z) = \int_0^\infty t^{z-1}e^{-t}\,\mathrm{d}t$ is the Gamma function.

To derive a conditional version of that, note that the posterior distribution can be written as

$$\begin{aligned} p_\theta(\boldsymbol{x} \mid \boldsymbol{y}) &= \frac{p_\theta(\boldsymbol{y} \mid \boldsymbol{x})p(\boldsymbol{x})}{p_\theta(\boldsymbol{y})} \\ &= \frac{|\mathcal{B}_n|^{-1}}{p_\theta(\boldsymbol{y})} \prod_{x\in\mathcal{X}}\prod_{y\in\mathcal{Y}} p_\theta(y \mid x)^{a_{\boldsymbol{x},\boldsymbol{y}}(x,y)}, \end{aligned}$$

with $p_\theta(\boldsymbol{y}) \coloneqq \sum_{\boldsymbol{x}\in\mathcal{B}_n} p_\theta(\boldsymbol{y} \mid \boldsymbol{x})p(\boldsymbol{x})$. We denote $f_\theta(x,y) \coloneqq p_\theta(y \mid x)$, so that the normalised version

$$q_\xi(x \mid y) \coloneqq \frac{f_\theta(x,y)}{C_\theta(y)}, \quad \text{with} \quad C_\theta(y) \coloneqq \sum_{x'\in\mathcal{X}} f_\theta(x',y)$$

can be seen as a posterior distribution on $\mathcal{X}$ parametrised by $\xi = \xi(\theta) \in \left(\Delta^{|\mathcal{X}|-1}\right)^{|\mathcal{Y}|}$. Then, we have

$$\begin{aligned} p_\theta(\boldsymbol{x} \mid \boldsymbol{y}) &= \frac{|\mathcal{B}_n|^{-1}}{p_\theta(\boldsymbol{y})} \prod_{x\in\mathcal{X}}\prod_{y\in\mathcal{Y}} C_\theta(y)^{a_{\boldsymbol{x},\boldsymbol{y}}(x,y)} q_\xi(x \mid y)^{a_{\boldsymbol{x},\boldsymbol{y}}(x,y)} \\ &= \left(\frac{\prod_{y\in\mathcal{Y}} C_\theta(y)^{a_{\boldsymbol{y}}(y)}}{|\mathcal{B}_n|\ p_\theta(\boldsymbol{y})}\right) \underbrace{\prod_{x\in\mathcal{X}}\prod_{y\in\mathcal{Y}} q_\xi(x \mid y)^{a_{\boldsymbol{x},\boldsymbol{y}}(x,y)}}_{(\star)}. \end{aligned}$$

For a fixed $\boldsymbol{y} \in \mathcal{Y}^n$, maximising $p_\theta(\boldsymbol{x} \mid \boldsymbol{y})$ over $\boldsymbol{x}$ is thus equivalent to maximising $(\star)$, where q_ξ can be seen as a memoryless law and some sort of 'backward channel'. If the channel parameters θ, and therefore ξ, are unknown, we can marginalise it with a choice of prior on $\pi(\xi)$, that is, compute

$$\widehat{p}(\boldsymbol{x} \mid \boldsymbol{y}) \propto \int \left(\prod_{x\in\mathcal{X}}\prod_{y\in\mathcal{Y}} q_\xi(x \mid y)^{a_{\boldsymbol{x},\boldsymbol{y}}(x,y)}\right) \mathrm{d}\pi(\xi).$$

Choosing independent Dirichlet priors with parameters $\left(\frac{1}{2},\dots,\frac{1}{2}\right)$ for each $q_\xi(\cdot \mid y)$ corresponds to Jeffreys prior and is asymptotically optimal in the minimax sense [15]. In this case, the integral can be computed in much the same way as in [16, Prop. 2.15], resulting in the conditional version of the KT estimator (also in [12, Eq. (49)]):

$$p_{\mathrm{KT}}(\boldsymbol{x} \mid \boldsymbol{y}) = \frac{\Gamma\left(\frac{|\mathcal{X}|}{2}\right)^{|\mathcal{Y}|}}{\Gamma\left(\frac{1}{2}\right)^{|\mathcal{X}||\mathcal{Y}|}} \frac{\prod_{x\in\mathcal{X}}\prod_{y\in\mathcal{Y}} \Gamma\left(a_{\boldsymbol{x},\boldsymbol{y}}(x,y) + \frac{1}{2}\right)}{\prod_{y\in\mathcal{Y}} \Gamma\left(a_{\boldsymbol{y}}(y) + \frac{|\mathcal{X}|}{2}\right)}. \tag{7}$$

Note that this quantity only depends on the counts $\boldsymbol{a}_{\boldsymbol{x},\boldsymbol{y}}$ and $\boldsymbol{a}_{\boldsymbol{y}}$, or, equivalently, on the joint and marginal types $\pi_{\boldsymbol{x},\boldsymbol{y}}$ and $\pi_{\boldsymbol{y}}$. We will be interested in studying the universal decoder

$$\phi_{\mathrm{KT}}(\boldsymbol{y}) = \arg\max_{i\in\mathcal{H}} \frac{1}{n} \log p_{\mathrm{KT}}(\boldsymbol{x}_i \mid \boldsymbol{y}). \tag{8}$$

First, we provide some results on the conditional KT estimator.

Lemma 4: For $\boldsymbol{x} \in \mathcal{X}^n$ and $\boldsymbol{y} \in \mathcal{Y}^n$,

$$\begin{aligned}\log\left(\frac{\prod_{x\in\mathcal{X}}\prod_{y\in\mathcal{Y}} \pi_{\boldsymbol{x}|\boldsymbol{y}}(x \mid y)^{a_{\boldsymbol{x},\boldsymbol{y}}(x,y)}}{p_{\mathrm{KT}}(\boldsymbol{x} \mid \boldsymbol{y})}\right) \\ \leq \frac{(|\mathcal{X}|-1)|\mathcal{Y}|}{2}\log n + 2|\mathcal{Y}|. \end{aligned} \tag{9}$$

Proof: Using the expression in (7), we get

$$\begin{aligned}&\log\left(\frac{\prod_{x\in\mathcal{X}}\prod_{y\in\mathcal{Y}} \pi_{\boldsymbol{x}|\boldsymbol{y}}(x \mid y)^{a_{\boldsymbol{x},\boldsymbol{y}}(x,y)}}{p_{\mathrm{KT}}(\boldsymbol{x} \mid \boldsymbol{y})}\right) \\ &\quad= \sum_{y\in\mathcal{Y}} \log\left(\frac{\prod_{x\in\mathcal{X}} \pi_{\boldsymbol{x}|\boldsymbol{y}}(x \mid y)^{a_{\boldsymbol{x},\boldsymbol{y}}(x,y)}}{\frac{\Gamma(\frac{|\mathcal{X}|}{2})}{\Gamma(\frac{1}{2})^{|\mathcal{X}|}} \frac{\prod_{x\in\mathcal{X}}\Gamma\left(a_{\boldsymbol{x},\boldsymbol{y}}(x,y)+\frac{1}{2}\right)}{\Gamma\left(a_{\boldsymbol{y}}(y)+\frac{|\mathcal{X}|}{2}\right)}}\right) \\ &\quad\leq \sum_{y\in\mathcal{Y}} \left(\frac{|\mathcal{X}|-1}{2}\log a_{\boldsymbol{y}}(y) + 2\right) \\ &\quad\leq \frac{(|\mathcal{X}|-1)|\mathcal{Y}|}{2}\log n + 2|\mathcal{Y}|, \end{aligned}$$

where, for the first inequality, we apply [16, Thm. 2.16] on each subsequence of $\boldsymbol{x}$ of length $a_{\boldsymbol{y}}(y)$ whose symbols jointly appear with the same $y \in \mathcal{Y}$ in $\boldsymbol{x} \otimes \boldsymbol{y}$. ∎

Lemma 5: Given sequences $x_1^{n+1} = x_1\cdots x_n x_{n+1} \in \mathcal{X}^{n+1}$ and $y_1^{n+1} = y_1\cdots y_n y_{n+1} \in \mathcal{Y}^{n+1}$, we have

$$\begin{aligned}&p_{\mathrm{KT}}(x_1^{n+1} \mid y_1^{n+1}) = \\ &\qquad\left(\frac{a_{x_1^n,y_1^n}(x_{n+1},y_{n+1}) + \frac{1}{2}}{a_{y_1^n}(y_{n+1}) + \frac{|\mathcal{X}|}{2}}\right) p_{\mathrm{KT}}(x_1^n \mid y_1^n). \end{aligned} \tag{10}$$

Proof: Write (7) for sequences of length n and $n+1$, and use the fact that $\Gamma(z+1) = z\Gamma(z)$. ∎

This nice sequential behaviour is analogous to the one for the simple estimator (6). It means that $p_{\mathrm{KT}}(\boldsymbol{x} \mid \boldsymbol{y})$ can be sequentially computed by keeping track only of the counts $\boldsymbol{a}_{\boldsymbol{x},\boldsymbol{y}}$. In particular, we can write

$$\begin{aligned} p_{\mathrm{KT}}\left(x_{n+1}^{n+k}, y_{n+1}^{n+k} \mid \boldsymbol{a}_{x_1^n,y_1^n}\right) &\coloneqq \frac{p_{\mathrm{KT}}(x_1^{n+k} \mid y_1^{n+k})}{p_{\mathrm{KT}}(x_1^n \mid y_1^n)} \\ &= \prod_{i=n}^{n+k-1} \left(\frac{a_{x_1^i,y_1^i}(x_{i+1},y_{i+1}) + \frac{1}{2}}{a_{y_1^i}(y_{i+1}) + \frac{|\mathcal{X}|}{2}}\right). \end{aligned} \tag{11}$$

Remark 1: In [3], the minimum conditional entropy (MCE) decoding rule $\phi_{\mathrm{MCE}}(\boldsymbol{y}) = \arg\min_{i\in\mathcal{H}} H(\pi_{\boldsymbol{x}_i|\boldsymbol{y}} \mid \pi_{\boldsymbol{y}})$ was proposed for the family of memoryless channels, and shown

to be strongly random-coding universal, when either $\mathcal{B}_n = \mathcal{X}^n$ or $\mathcal{B}_n = \mathcal{T}^{(n)}(\pi)$. This rule is equivalent to

$$\begin{aligned}\phi_{\mathrm{MCE}}(\boldsymbol{y}) &= \arg\max_{i\in\mathcal{H}} \pi_{\boldsymbol{x}_i|\boldsymbol{y}}(\boldsymbol{x}_i \mid \boldsymbol{y}) \\ &= \arg\max_{i\in\mathcal{H}} \sup_{\xi} q_\xi(\boldsymbol{x}_i \mid \boldsymbol{y}), \end{aligned} \tag{12}$$

where the supremum is taken over all memoryless laws $q_\xi(x \mid y)$, making it somewhat similar to the generalised likelihood test [17, p. 2166], with the likelihood replaced by a posterior law. When $\mathcal{B}_n = \mathcal{T}^{(n)}(\pi)$, this rule coincides with the MMI decoder $\phi_{\mathrm{MMI}}(\boldsymbol{y}) = \arg\max_{i\in\mathcal{H}} I(\pi_{\boldsymbol{x}_i} \colon \pi_{\boldsymbol{y}})$ [2], since $H(\pi_{\boldsymbol{x}_i|\boldsymbol{y}} \mid \pi_{\boldsymbol{y}}) = H(\pi_{\boldsymbol{x}_i}) - I(\pi_{\boldsymbol{x}_i} \colon \pi_{\boldsymbol{y}})$. Finally, we note that a sequential expression analogous to (10) holds for the estimator in (12). The updating term, given in the following, has a slightly less simple expression:

$$\begin{aligned}&\pi_{x_1^{n+1}|y_1^{n+1}}(x_1^{n+1} \mid y_1^{n+1}) = \pi_{x_1^n|y_1^n}(x_1^n \mid y_1^n) \\ &\quad\times \left(\frac{a_{x_1^n,y_1^n}(x_{n+1},y_{n+1})+1}{a_{y_1^n}(y_{n+1})+1}\right)^{a_{x_1^n,y_1^n}(x_{n+1},y_{n+1})+1} \\ &\quad\times \left(\frac{a_{y_1^n}(y_{n+1})}{a_{x_1^n,y_1^n}(x_{n+1},y_{n+1})}\right)^{a_{x_1^n,y_1^n}(x_{n+1},y_{n+1})}.\end{aligned}$$

We are now ready to state our main result concerning random-coding universality of the KT decoder:

Theorem 1: Let $\mathcal{B}_n = \mathcal{X}^n$ or $\mathcal{B}_n = \mathcal{T}^{(n)}(\pi)$ the type class of some type π on $\mathcal{X}^n$. The decoder (8) is strongly universal for the family of memoryless channels (4).

Proof: Using (2) with $\widehat{p}(\cdot \mid \boldsymbol{y}) = p_{\mathrm{KT}}(\cdot \mid \boldsymbol{y})$ gives $\left|\mathcal{E}_{\mathrm{KT}}(\boldsymbol{x},\boldsymbol{y})\right| \le \frac{1}{p_{\mathrm{KT}}(\boldsymbol{x}|\boldsymbol{y})}$. Together with (3) and (5), we get

$$\frac{\left|\mathcal{E}_{\mathrm{KT}}(\boldsymbol{x},\boldsymbol{y})\right|}{\left|\mathcal{E}_{\mathrm{ML}}(\boldsymbol{x},\boldsymbol{y})\right|} \le (n+1)^{|\mathcal{X}||\mathcal{Y}|} \frac{\prod_{x\in\mathcal{X}}\prod_{y\in\mathcal{Y}} \pi_{\boldsymbol{x}|\boldsymbol{y}}(x \mid y)^{a_{\boldsymbol{x},\boldsymbol{y}}(x,y)}}{p_{\mathrm{KT}}(\boldsymbol{x} \mid \boldsymbol{y})}.$$

Taking the logarithm on both sides and applying (9) yields

$$\begin{aligned}&\log \frac{\left|\mathcal{E}_{\mathrm{KT}}(\boldsymbol{x},\boldsymbol{y})\right|}{\left|\mathcal{E}_{\mathrm{ML}}(\boldsymbol{x},\boldsymbol{y})\right|} \\ &\qquad \le |\mathcal{X}||\mathcal{Y}| \log(n+1) + \frac{(|\mathcal{X}|-1)|\mathcal{Y}|}{2}\log n + 2|\mathcal{Y}|.\end{aligned}$$

Finally, using (1), we conclude that

$$\begin{aligned}&\frac{1}{n}\log \frac{P_{\theta,\mathrm{KT}}(\mathrm{error})}{P_{\theta,\mathrm{ML}}(\mathrm{error})} \\ &\qquad \le \frac{|\mathcal{X}||\mathcal{Y}|}{n}\log(n+1) + \frac{(|\mathcal{X}|-1)|\mathcal{Y}|}{2n}\log n + \frac{2|\mathcal{Y}|}{n},\end{aligned}$$

which goes to 0 as $n \to \infty$, and does not depend on θ. ∎

This means that, in the ensemble of random codes, and asymptotically with blocklength n, the error exponent obtained with the KT decoder is the same as that of the optimal ML decoder. In particular, as far as asymptotic performance is concerned, the KT decoder (8) is on par with the MCE decoder (12). The particular case of this result for codebooks of size $M = 2$ appeared in [12].

IV. Decoding Schemes for Practical Codes

A. Linear Block Codes

When applying universal decoders to binary (k, n)-linear block codes, two issues appear due to the code structure. First, it contains the all-zero codeword, which is always chosen by the KT decoder, since it maximises (7), regardless of $\boldsymbol{y}$. To avoid this, all the codewords are shifted by a constant sequence known to the receiver. Second, since the code contains pairs of antipodal codewords, the decoder assigns the same metric for each pair, as it does not know whether or not bits are flipped by the channel. To deal with this, we identify antipodal codewords as representing the same message. This reduces the code rate from $\frac{k}{n}$ to $\frac{k-1}{n}$. Another solution could be to send a short training sequence to initialise the decoder.

We simulate binary channels ($|\mathcal{X}| = |\mathcal{Y}| = 2$) and encode our messages using a Golay code ($k = 12$, $n = 24$) [18, Sec. 1.9], with codewords shifted and antipodal pairs identified. For every block of $B = 10$ messages, a DMC with random cross-over probabilities $p := P_{Y|X}(1 \mid 0)$ and $q := P_{Y|X}(0 \mid 1)$ is uniformly drawn with $p, q \sim \mathcal{U}\left([0,\alpha]\right)$, $\alpha \in]0,1]$. Decoding is done by comparing the metrics of each codeword, as in (8), which is still feasible for moderate codebook sizes. Instead of computing the KT estimator $p_{\mathrm{KT}}(\boldsymbol{x}_i \mid \boldsymbol{y})$ in (7) using only the counts $\boldsymbol{a}_{\boldsymbol{x}_i,\boldsymbol{y}}$ obtained from the received $\boldsymbol{y} \in \mathcal{Y}^n$ and codeword $\boldsymbol{x}_i \in \mathcal{C}$, we use a version of the counts stored in the receiver memory. For each codeword, if decoding is done above a confidence threshold (i.e., if the ratio between first and second highest metrics associated to codewords is greater than a certain threshold), then the counts obtained from the decoded codeword and corresponding received sequence are used to update the counts in the memory.

We compare the performance of the KT decoder with that of the MCE decoder (12), MMI decoder [1], [2], LZ decoder from [3], [4], and an omniscient ML decoder. All decoders (except the ML) are allowed to use a similar confidence threshold to update its rule according to previously decoded messages. In Fig. 2, we compare the block error rate (BLER) as function of the channel meta-parameter α for these schemes. We observe that KT, MCE and MMI essentially track the performance of the ML decoder, despite having no prior information on the channel. The more distant performance of LZ is explained by the slow convergence of this algorithm, which is actually universal for the broader family of finite-state channels.

B. Convolutional Codes

We consider feed-forward convolutional codes of constraint length K: at time i, a binary b-tuple u_i is input to the encoder, and the corresponding output, a binary ν-tuple, is added to $\boldsymbol{v}_{0,i} \in \mathbb{F}_2^\nu$, resulting in the ν-tuple $\boldsymbol{v}_i$; this is then mapped to $\boldsymbol{x}_i := (x_{i,1}, \dots x_{i,l}) = \mathcal{L}(\boldsymbol{v}_i)$, a sequence in $\mathcal{X}^l$, see [19, Fig. 5.1]. To simplify, we assume that $|\mathcal{X}|^l = 2^\nu$. Denote $\boldsymbol{x}_i^j \in \mathcal{X}^{l(j-i+1)}$ the sequence formed by the concatenation of the symbols in $\boldsymbol{x}_i, \dots, \boldsymbol{x}_j$. The rate of the encoder is $R = b/l$ bits per channel use. To apply universal decoding to convolutional codes, we follow [3], [12] and use a modified stack decoder.

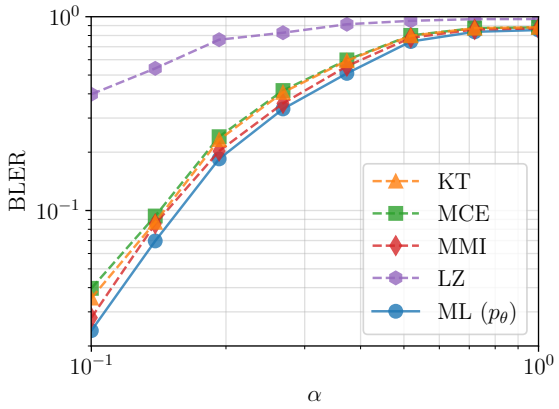

Fig. 2. BLER of different decoders for modified Golay code ($R = 11/24$) in random binary DMCs with cross-over probabilities $p, q \sim \mathcal{U}([0, \alpha])$.

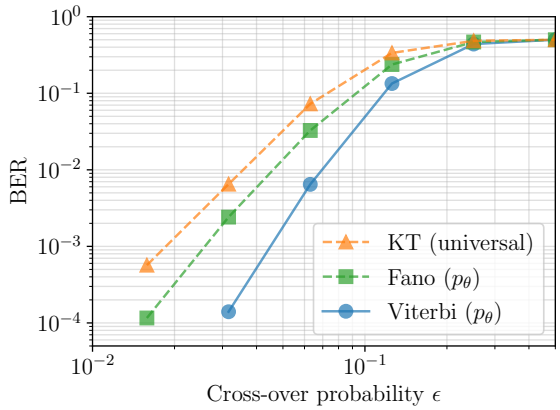

Fig. 3. BER of different decoders for convolutional code ($K = 7$, $R = 1/2$) in a BSC with cross-over probability ϵ.

Sequential decoding of convolutional codes achieves reduced complexity by avoiding to compute the metrics of all paths in the trellis, and concentrating instead on the paths with higher metrics. The basic *stack sequential decoding* [19, Ch. 6] (see also [20], [21]) works as follows[1]: the algorithm keeps a stack of searched paths (of different lengths), ordered according to their metric. At each step, the path with higher metric is extended by one branch, and replaced by its 2^b successors; then, the stack is sorted by metric. The algorithm ends when the path on top of the stack (higher metric) reaches the end of the trellis. For memoryless channels of known parameters, the usual choice of metric, which is consistent with ML decoding, is the Fano metric [19, Eq. (6.1.2)].

Following [3], [12], when the channel is unknown, we replace the Fano metric by a universal metric. Let $\boldsymbol{x}_i^j := \boldsymbol{x}_i \cdots \boldsymbol{x}_j \in \mathcal{X}^{l(j-i+1)}$ denote an input sequence (path in the trellis) and $\boldsymbol{y}_i^j := \boldsymbol{y}_i \cdots \boldsymbol{y}_j \in \mathcal{Y}^{l(j-i+1)}$ an output sequence of the channel. For the KT estimator, we have the path metric

$$M_{\mathrm{KT}}(\boldsymbol{x}_i^j, \boldsymbol{y}_i^j) := \log p_{\mathrm{KT}}(\boldsymbol{x}_i^j \mid \boldsymbol{y}_i^j) + l(j-i+1)\left[\log|\mathcal{X}| - (R+\Delta)\right],$$

for $\Delta > 0$. Despite not being additive in the sense that, in general, $M_{\mathrm{KT}}(\boldsymbol{x}_i^j, \boldsymbol{y}_i^j) \neq M_{\mathrm{KT}}(\boldsymbol{x}_i^k, \boldsymbol{y}_i^k) + M_{\mathrm{KT}}(\boldsymbol{x}_{k+1}^j, \boldsymbol{y}_{k+1}^j)$, this metric can be *sequentially computed*, in the sense that

$$M_{\mathrm{KT}}(\boldsymbol{x}_i^{j+1}, \boldsymbol{y}_i^{j+1}) = M_{\mathrm{KT}}(\boldsymbol{x}_i^j, \boldsymbol{y}_i^j) + \left[\log|\mathcal{X}| - (R+\Delta)\right] + \log p_{\mathrm{KT}}\left(\boldsymbol{x}_{j+1}, \boldsymbol{y}_{j+1} \mid \boldsymbol{a}_{\boldsymbol{x}_i^j, \boldsymbol{y}_i^j}\right),$$

so long as one keeps track of the counts $\boldsymbol{a}_{\boldsymbol{x}_i^j, \boldsymbol{y}_i^j}$. Storing the counts needs no more than $|\mathcal{X}||\mathcal{Y}|\log(n+1)$ bits, since each count is a value from 0 to n, which is much less than the $n\log(|\mathcal{X}||\mathcal{Y}|)$ bits used to store the whole sequence $\boldsymbol{x} \otimes \boldsymbol{y}$. Note that this simplifies implementation when using the KT estimator, and cannot be done, for instance, with the LZ metric [3], due to the changes in the incremental parsing every time a path is extended.

[1]We ignore the merging step proposed in [19], as doing so reduces complexity, and does not significantly increase error probability [19, p. 371].

Therefore, we can use the basic stack sequential decoding algorithm in much the same way as described previously, except that, now, along with each path $\boldsymbol{x}_i^j$ in the stack, we store not only its metric $M_{\mathrm{KT}}(\boldsymbol{x}_i^j, \boldsymbol{y}_i^j)$, but also the counts $\boldsymbol{a}_{\boldsymbol{x}_i^j, \boldsymbol{y}_i^j}$. As in [12], we limit the number of computations in a given branch of the trellis: when decoding u_i, if more than $L_{\max}$ nodes are visited in the branch corresponding to $u_i = u \in \mathbb{F}_2^b$, we declare $\widehat{u}_i = u$. Additionally, we propose that, when this occurs, instead of reinitialising the counts to restart the process for u_{i+1}, we keep the previous values in the counts. We remark that [12, Thm. 1] assures an average bit error probability of order $O(K2^{-bK})$, over the ensemble of random convolutional codes, for rates below the cut-off rate.

In our simulations, we set $|\mathcal{X}| = |\mathcal{Y}| = 2$, $L_{\max} = 2^{bK}$, $\Delta = 10^{-5}$, and use the rate-$1/2$ convolutional code with $K = 7$ determined by the coefficients 133_8 and 171_8 in octal notation. The channel is a binary symmetric channel (BSC) with cross-over probability $\epsilon \in [0, 1]$. In Fig. 3, we plot the bit error rate (BER) as a function of ϵ for stack decoders using the KT and Fano metrics, and for the Viterbi decoder implementing the ML rule. We see that, despite having no prior information about the channel (not even that it is symmetric), the stack decoder with KT metric has performance not too far from that of the stack decoder that knows the channel and uses the Fano metric.

V. Conclusion

We have studied universal decoding with the KT estimator over DMCs. We showed that this decoder is random-coding strongly universal for the family of DMCs, and integrated it in decoding schemes for linear block codes and convolutional codes. Numerical results revealed that efficient decoding can be done even for moderate blocklength or constraint length. Future perspectives include investigating analogous decoders for more general channels. If the same proof techniques are to be followed, the main difficulty so far seems to reside in deriving an analogue to Lemma 3 for those channels, which could require new combinatorial results.

REFERENCES

[1] V. D. Goppa, "Nonprobabilistic mutual information with memory," *Problems Control Inf. Theory*, vol. 4, pp. 97–102, 1975.

[2] I. Csiszár and J. Körner, *Information Theory: Coding Theorems for Discrete Memoryless Systems*, 2nd ed. Cambridge, UK: Cambridge Univ. Press, 2011.

[3] J. Ziv, "Universal decoding for finite-state channels," *IEEE Trans. Inf. Theory*, vol. 31, no. 4, pp. 453–460, 1985.

[4] A. Lapidoth and J. Ziv, "On the universality of the LZ-based decoding algorithm," *IEEE Trans. Inf. Theory*, vol. 44, no. 5, pp. 1746–1755, 1998.

[5] M. Feder and A. Lapidoth, "Universal decoding for channels with memory," *IEEE Trans. Inf. Theory*, vol. 44, no. 5, pp. 1726–1745, 1998.

[6] N. Merhav, "Universal decoding for memoryless Gaussian channels with a deterministic interference," *IEEE Trans. Inf. Theory*, vol. 39, no. 4, pp. 1261–1269, 1993.

[7] W. Huleihel and N. Merhav, "Universal decoding for Gaussian intersymbol interference channels," *IEEE Trans. Inf. Theory*, vol. 61, no. 4, pp. 1606–1618, 2015.

[8] O. Shayevitz and M. Feder, "Universal decoding for frequency-selective fading channels," *IEEE Trans. Inf. Theory*, vol. 51, no. 8, pp. 2770–2790, 2005.

[9] R. Averbuch and N. Merhav, "Exact random coding exponents and universal decoders for the asymmetric broadcast channel," *IEEE Trans. Inf. Theory*, vol. 64, no. 7, pp. 5070–5086, 2018.

[10] N. Merhav, "Universal decoding using a noisy codebook," *IEEE Trans. Inf. Theory*, vol. 64, no. 4, pp. 2231–2239, 2018.

[11] R. Tamir and N. Merhav, "Universal decoding for the typical random code and for the expurgated code," *IEEE Trans. Inf. Theory*, vol. 68, no. 4, pp. 2156–2168, 2022.

[12] A. Lapidoth and J. Ziv, "On the decoding of convolutional codes on an unknown channel," *IEEE Trans. Inf. Theory*, vol. 45, no. 7, pp. 2321–2332, 1999.

[13] J. K. Nelson and A. C. Singer, "Bayesian sequential detection for the BSC with unknown crossover probability," in *Proc. 2006 IEEE Int. Symp. Inf. Theory (ISIT)*, 2006, pp. 640–644.

[14] R. Krichevsky and V. Trofimov, "The performance of universal encoding," *IEEE Trans. Inf. Theory*, vol. 27, no. 2, pp. 199–207, 1981.

[15] B. S. Clarke and A. R. Barron, "Jeffreys' prior is asymptotically least favorable under entropy risk," *J. Statist. Planning Inference*, vol. 41, no. 1, pp. 37–60, 1994.

[16] E. Gassiat, *Universal Coding and Order Identification by Model Selection Methods*. Cham, Switzerland: Springer, 2018.

[17] A. Lapidoth and P. Narayan, "Reliable communication under channel uncertainty," *IEEE Trans. Inf. Theory*, vol. 44, no. 6, pp. 2148–2177, 1998.

[18] W. C. Huffman and V. Pless, *Fundamentals of Error-Correcting Codes*. Cambridge, UK: Cambridge Univ. Press, 2003.

[19] A. J. Viterbi and J. K. Omura, *Principles of Digital Communication and Coding*. Mineola, NY, USA: Dover, 2009.

[20] F. Jelinek, "Fast sequential decoding algorithm using a stack," *IBM J. Res. Develop.*, vol. 13, no. 6, pp. 675–685, 1969.

[21] K. S. Zigangirov, "Procedures of sequential decoding," in *Coding and Complexity*, G. Longo, Ed. Vienna, Austria: Springer, 1975, pp. 109–130.