Optimal Parametric Backward-Adaptive Lossy Compression [*]

Yuval Kochman and Ram Zamir

Dpt. of EE-Systems, Tel Aviv University, Israel
yuvalko & *zamir@eng.tau.ac.il*

June 1, 2005

Abstract

We present a new generic mechanism for "on-line" construction of a vector quantizer codebook, based on blockwise backward-adaptive parametric encoding. The workings of the proposed scheme is explained by the principle of "natural type selection": In the limit of large vector dimension, the type of the first distortion-matching codeword within a random codebook coincides with an iteration of the Blahut-Arimoto algorithm for computation of the rate-distortion function. We extend this observation to parametric codebooks, and demonstrate that the parameter sequence converges to an optimum solution within the reproduction class. In comparison to other methods, adaptation is simple due to the parametric model, yet it is optimal even in the low coding rate regime.

Keywords: parametric encoding, natural type selection, Arimoto-Blahut algorithm, vector quantization, alternating minimization, universal coding, approximate string matching.

[*] This work was presented in part at the Data Compression Conference, 2002, and at the Allerton Conference on Communication, 2003.

I Introduction

In most scenarios of source coding, source statistics are unknown, or changing over time. A non-adaptive system may be robust for a family of sources [20], but it usually suffers a large loss of performance. Thus most compression algorithms use some sort of adaptation mechanism, e.g., dynamic Huffman coding and Lempel-Ziv coding for universal lossless compression [26], or adaptive pulse code modulation (ADPCM) and code excited linear prediction (CELP) for speech coding [17, 13]. For analysis purposes, we often assume that the source statistics are constant but a priori unknown [25].

Adaptive compression schemes divide into two main categories: *forward* adaptation and *backward* adaptation [17]. The former approach is based on a two-stage code: looking ahead at the source sequence, the encoder learns the source statistics, computes the optimal codebook parameters and encodes them as a header to the compressed data; examples include dynamic Huffman coding and CELP. Backward adaptation does not waste rate on sending header information; instead, both the encoder and the decoder learn the statistics "on the fly" by looking backwards at the past code sequence (which is available to both), and sequentially adapt the codebook accordingly; examples include Lempel-Ziv-like algorithms and ADPCM.

Can backward-adaptive lossy compression achieve the rate-distortion function of a source whose statistics is unknown a priori? This question is related to the feasibility of *sequential* universal lossy compression.

To put this question on more concrete grounds, consider a state-machine which sequentially encodes vectors $\mathbf{x}_1, \mathbf{x}_2, \ldots$ (say of ℓ letters each) emitted by a stationary and ergodic source:

$$\begin{aligned} i_n &= f(\mathbf{x}_n, S_n) \\ \mathbf{y}_n &= g(i_n, S_n) \\ S_{n+1} &= h(\mathbf{y}_n, S_n) \end{aligned} \tag{1}$$

for $n = 1, 2, \ldots$, where $f(\cdot,\cdot)$ is the encoding function, i_n is the codeword index sent to the decoder at step n, $g(\cdot,\cdot)$ is the decoding function, $\mathbf{y}_n$ is the reconstruction vector, S_n is the system state, $h(\cdot,\cdot)$ is the next-state function, and where the initial state S_1 is given. We may think of $f(\cdot, S)$ and $g(\cdot, S)$ as an encoder-decoder pair depending on some "estimated statistics" S, and of h as a learning or an adaptation mechanism for S.

An essential difference between lossless and lossy compression in the sequential model (1), is that system adaptation in the lossy case must be done using the *distorted* version of the source $\mathbf{y}_n$; the raw source vector $\mathbf{x}_n$ is not available to the decoder. This difference obviously becomes more significant at low coding rates, when the distortion between $\mathbf{y}_n$ and $\mathbf{x}_n$ becomes higher. Ideally, we like that the state sequence

S_n would converge to S^*, the optimal "state" for encoding the current source. Can backward adaptation achieve this goal even when the distortion between $\mathbf{y}_n$ and $\mathbf{x}_n$ is significant?

Quite surprisingly, as recent work shows [24, 23], the answer - at least in the limit of large codebook dimension ($\ell \to \infty$) - is that it can. The key to this observation, is that we do not seek to estimate the statistics of the source P, but rather these of the optimal reproduction distribution Q^* that achieves the rate-distortion function $R(P, d)$ [6]. It turns out that the frequency the encoding algorithm uses codewords provides a natural law for optimum backward codebook adaptation that approaches this optimal Q^*. This phenomena was termed "gold washing" in [24] and "natural type selection" (NTS) in [23].

In the lossless case the idea of backward-adaptive compression is clear and well known: The LZ78 algorithm, for example, grows a tree of code words as the coding progresses, in a way that asymptotically the proportion of typical source sequences within the code words approaches one. We see that, while the source statistics P may be unknown, the source reveals them gradually through the sequence it produces [26]. Although less obvious, the NTS principle tells us that also in lossy compression, when $Q^* \neq P$, the matching process supplies a mechanism for learning the statistics Q^* of the optimal codebook. Note that the deviation of Q^* from P usually increases with distortion; e.g., in the quadratic Gaussian case Q^* is given by the reverse water-filling solution which dictates that, at high distortion, at some frequencies where source energy exists, codebook energy would equal zero.

In [23], the NTS is described as a bootstrap-like procedure, in which one starts with an initial guess Q_1 for the reproduction codebook statistics, and then "measures" the type (i.e., empirical distribution) of the first word $\mathbf{y}$ in the codebook that matches a source word $\mathbf{x}$ under the fidelity criterion $\rho(\mathbf{x}, \mathbf{y}) \leq d$, where ρ is the distortion measure. By regenerating the codebook, using the type of the matching codeword as the random code generating probability, and repeating this process over and over again, a sequence of codebook distributions $Q_1, Q_2, \ldots$ is generated. It turns out that for a memoryless source with distribution P, as the word length ℓ goes to infinity, this sequence of distributions obeys a deterministic recursion rule

$$Q_{n+1} = Q^*(P, Q_n, d) \quad , n = 1, 2, \ldots \tag{2}$$

(where the function $Q^*(\cdot, \cdot, \cdot)$ will be defined in the sequel), that converges to the optimum reproduction distribution Q^*. Moreover, this recursion corresponds to a fixed-distortion version of the Blahut-Arimoto iterative algorithm for computation of the rate-distortion function [2].

In this paper we aim to incorporate the NTS principle into a practical, low complexity coding scheme. Doing so, however, reveals some drawbacks:

1. The procedure involves regenerating the whole codebook after each match is made.

2. For small values of the word length ℓ the matching codeword does not capture the full multi-dimensional distribution of $Q^*(P, Q, d)$ (in particular if it has long memory).

3. Large block dimension is needed for the empirical distribution of the matching codeword to converge to $Q^*(P, Q, d)$, causing high complexity of the matching process.

We address the first two problems by restricting the evolving codebooks to ones that can be generated from a fixed (random) codebook, using some parametric transformation characterized by a parameter θ in a set Θ. Our target rate thus becomes $R(P, \Theta, d)$, the best rate achievable within the resulting family of parametric reproductions, rather than the rate-distortion function $R(P, d)$. Maximum-likelihood estimation of θ from the matching codeword $\mathbf{y}$ turns out to be the analogy in the parametric case of the codeword type in (2). Furthermore, the resulting iteration corresponds to a parametric constrained version of the Blahut algorithm [18]. To deal with the third problem, we use a the least mean square (LMS)-like algorithm, as widely used in practical adaptive filters [16], to smooth out the parameter sequence. We call this new mechanism "Natural Parameter Adaptation" (NPA).

In Section II we present the NPA scheme and lay the foundations to the analysis of its performance, and in Section III we prove its convergence to $R(P, \Theta, d)$ for discrete memoryless sources and reproductions. In section IV we discuss the extension of our results to sources and reproductions with memory. In section V we show special cases that shed light on possible applications, together with simulation results. We conclude in section VI by discussing the advantages and the limitations of the NPA system, and pointing out directions for further research.

II Natural Parameter Adaptation

II-1 System Description

The encoder we present is shown in Figure 1. It is composed of a parametric encoder and a parameter adaptation feedback loop. Let $\mathcal{X}$ and $\mathcal{Y}$ denote the source and reconstruction alphabets. The system uses a *base codebook*, which is a fixed (non-adaptive) codebook over the base alphabet $\mathcal{Z}$, $C = \{\mathbf{c}^i \in \mathcal{Z}^\ell, i = 0 \ldots M-1\}$. Each word of length ℓ is randomly and independently generated by some *universal distribution* Q_U. The distribution Q_U would typically be simple, e.g. an i.i.d. uniform or Gaussian distribution[1]. The number of codewords M will be large enough to ensure that a matching codeword will be found, in a manner that will be explained in the sequel. The codebook is an ordered list, where each codeword has an index, and the search will be always carried out in the same order, according to that index. These

[1] Alternatively, one may design a structured algebraic code, e.g. a lattice code for the continuous case or a parity check code for the discrete case

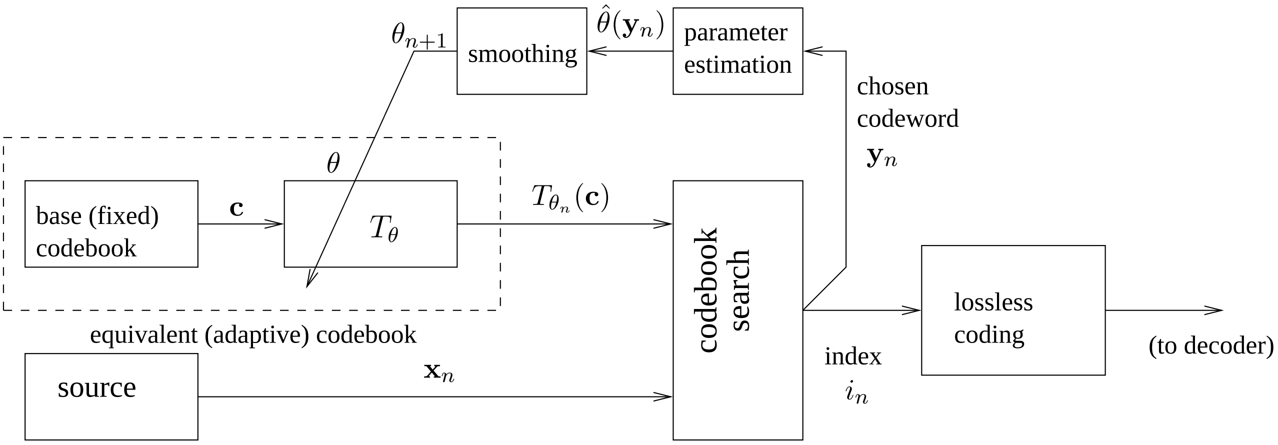

Figure 1: Encoder Structure

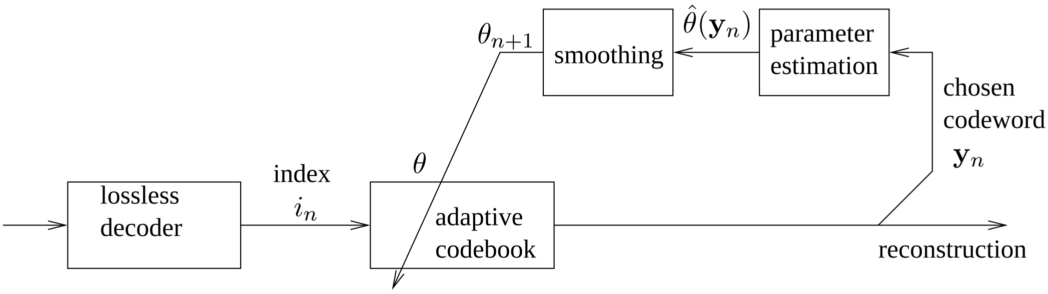

Figure 2: Decoder Structure

base codewords are fed into the transformation T. The transformation, for some set of parameter values $\Theta = \{\theta\}$, is a parameter dependent function $T_\theta : \mathcal{Z}^\ell \to \mathcal{Y}^\ell$ operating on each codeword, and producing the *adaptive codebook*

$$C_\theta = \{T_\theta(\mathbf{c}) : \mathbf{c} \in C\}. \tag{3}$$

The adaptive codebook plays the role of an effective codebook to the rest of the system. It is equivalent to a codebook where each codeword is randomly and independently generated by the *adaptive reproduction distribution* Q_θ which is the distribution of $T_\theta(\mathbf{U})$ when $\mathbf{U} \sim Q_U$, i.e.

$$Q_\theta(\mathbf{y}) = \sum_{\mathbf{u}:T_\theta(\mathbf{u})=\mathbf{y}} Q_U(\mathbf{u}) \ , \tag{4}$$

with the necessary adjustments in the continuous case. Using all possible parameter values, the adaptive reproduction distribution is restricted to the set:

$$\mathcal{Q}_\Theta \triangleq \{Q_\theta, \theta \in \Theta\} \quad . \tag{5}$$

This general definition allows a diverse choice of sets, such as memoryless finite-alphabet distributions (using a memoryless continuous base codebook and a per-letter step function transformation) or Gaussian auto-regressive (AR) distributions (using a memoryless gaussian codebook and an AR filter transformation).

The system starts with some initial parameter θ_1, and at each coding step it uses the current parameter and generates a new parameter value for the next step. At step n, θ_{n+1} is a function of all past reproduction vectors until the current one. One may think of the parameter θ as representing the system state S of (1). The encoding and the parameter adaptation are done as follows.

The quantizer selects the first codeword in the codebook that fits the source sequence with respect to a distortion constraint d, and sends its index to the decoder. We denote the d-matching codeword chosen by the quantizer at iteration n as $\mathbf{y}_n$, and the corresponding index by i_n:

$$\begin{aligned} \mathbf{y}_n &= T_{\theta_n}(\mathbf{c}^{i_n}) \quad , \\ i_n &= \min\Big\{i : \rho\Big(\mathbf{x}_n, T_{\theta_n}(\mathbf{c}^i)\Big) \le d\Big\} \quad . \end{aligned} \tag{6}$$

The index i_n is losslessly encoded, to create a variable-length representation of the source. We can use a simple encoder, which asymptotically achieves the logarithm of the index, such as the Elias coding of the integers [12]. The encoder passes this index to the decoder.

The backward adaptation loop consists of a maximum likelihood (ML) parameter estimator and a "smoothing" block. The estimator finds the parameter which maximizes the likelihood of the "measurement" $\mathbf{y}_n$ w.r.t. the parametric family $\mathcal{Q}_\Theta$:

$$\hat{\theta}(\mathbf{y}) = \theta_{ML}(\mathbf{y}) = \arg\max_{\theta\in\Theta} Q_\theta(\mathbf{y}) \quad . \tag{7}$$

The smoothing block $s\Big(\hat{\theta}(\mathbf{y}_n), \theta_1^n\Big)$ produces the parameter for the next iteration, weighting the estimate with the parameter history in order to average the stochastic nature of the estimate. One can think of many such functions, but for simplicity we will restrict ourselves to:

$$\theta_{n+1} = s\Big(\hat{\theta}(\mathbf{y}_n), \theta_1^n\Big) = \alpha\hat{\theta}(\mathbf{y}_n) + (1-\alpha)\theta_n \ , \tag{8}$$

where $0 < \alpha \le 1$ is the smoothing coefficient.

The decoder structure is depicted in Figure 2. It has a copy of the base codebook. It receives the index and decodes it to obtain a copy of the respective base codeword. Recall that each parameter is only a function of the initial parameter θ_1 and of the past reconstructions. Thus the decoder is able to reconstruct the parameter sequence, using an adaptation loop identical to the one in the encoder. Using the parameter θ_n, the decoder can pass $\mathbf{c}^{i_n}$ through the transformation T_{θ_n}, to obtain the reconstruction $\mathbf{y}_n$.

II-2 Some Definitions and Observations

The system described so far did not assume any restriction upon the source and reproduction models, e.g. the source may have memory or T_θ may be a general vector transformation. We now restrict our attention to memoryless sources and base codebooks, and to scalar (per letter) transformations T_θ, so that also the adaptive reproduction distribution Q_θ (4) is memoryless.

1. **String Matching and Favorite Type:** We look at the d-matching process between a source string and the codebook words. Assume that the codebook is generated memoryless i.i.d. $\sim Q$, and the source is governed by an i.i.d. distribution P. Define [22, 23]:

$$R(P,Q,d) = \inf_{W:\rho(P,W)\leq d} \Big\{ D(P \circ W \| P \times Q) \Big\} \quad , \tag{9}$$

where $D(\cdot\|\cdot)$ is the divergence, or Kullback-Liebler distance between two measures, defined by:

$$D(B\|A) = \left\{ \begin{array}{ll} \int \log(\frac{dB}{dA})dB, & \text{if } B \ll A \\ \infty, & \text{otherwise} \end{array} \right\}$$

which reduces in the discrete memoryless case to [6]:

$$D(B\|A) = \sum_i B_i \log \frac{B_i}{A_i}$$

where we assume $0\log(0) = 0$. On the left hand side of the divergence in (9) is the joint input-output distribution of a channel W with an input distribution P, while the right hand side is the product distribution of the source distribution P with the codebook distribution Q; $\rho(P,W)$ is the input-output distortion induced by the aforementioned channel:

$$\rho(P,W) = E_{P,W}\{\rho(\mathbf{x},\mathbf{y})\}. \tag{10}$$

The function $R(P,Q,d)$ amounts to the minimum rate needed to encode a source $\sim P$ using a random i.i.d. codebook $\sim Q$ with distortion d [22]. This holds, even though the size of codebook needed is, in principle, not bounded. Denoting the index of the first d-matching codeword of length ℓ by I_ℓ (recall (6)), the following holds in probability:

$$\lim_{\ell\to\infty} \frac{1}{\ell}\log(I_\ell) = R(P,Q,d) \quad .$$

Lossless coding of that index can, thus, achieve $R(P,Q,d)$ asymptotically.

Assume that (9) has a minimizer:

$$W^*(P,Q,d) = \arg \min_{W:\rho(P,W)\leq d} \Big\{ D(P \circ W \| P \times Q) \Big\} \tag{11}$$

and let $Q^*(P,Q,d)$ be the output distribution induced by the source distribution P and that minimizing transition distribution:

$$Q^*(P,Q,d) = [P \circ W^*(P,Q,d)]_y \quad . \tag{12}$$

Then the type of the first d-matching codeword, denoted by $Q_{\mathbf{y}}$, converges in probability to $Q^*(P,Q,d)$ as the word length $\ell \to \infty$ (proven for the finite-alphabet case in [23, Theorem 4], extended to general alphabets in [19]):

$$\lim_{\ell\to\infty} Q_{\mathbf{y}} = Q^*(P,Q,d) \text{ in prob.} \tag{13}$$

We call $Q^*(P,Q,d)$ the "favorite type". Clearly, these results hold for our equivalent codebook as well, with the adaptive reproduction distribution Q_θ playing the role of Q. We will denote the rate and the favorite type induced by a parameter θ by:

$$\begin{aligned} R(P,\theta,d) &\triangleq R(P,Q_\theta,d) \\ Q^*(P,\theta,d) &\triangleq Q^*(P,Q_\theta,d) \quad . \end{aligned} \tag{14}$$

2. **Convergence to the optimal reproduction in the non-constrained case:** The rate-distortion function for a random source governed by P is:

$$R(P,d) = \inf_Q R(P,Q,D) = \inf_{W:\rho(P,W)\le d} I(P,W).$$

The minimizer of $R(P,Q,d)$ (when exists, for instance when P has finite alphabet) is the optimal reproduction distribution $Q^* = Q^*(P,d)$.

Recall the iterative procedure $Q_{n+1} = Q^*(P,Q_n,d)$ (2). It describes a process of regenerating the codebook according to the favorite type (12) after each source string is encoded. The main result of [23] was that, for finite-alphabet sources, as the number of iterations $n \to \infty$

$$Q_n \to Q^*(P,d) \quad \text{and} \quad R(P,Q_n,d) \to R(P,d) \tag{15}$$

i.e., the sequence Q_n asymptotically approaches the optimal reproduction, and the coding rate approaches the rate-distortion function of the source.

Moreover, the recursion (2) corresponds to a fixed-distortion version of the Arimoto-Blahut algorithm as it computes $R(P,d)$ starting from Q_1 as an initial distribution [2, 8, 6].

3. **Best achievable performance within a parametric reproduction class:** Recall that the minimum rate for random coding with a given codebook distribution Q is given by $R(P,Q,d)$ defined in (9). For some set $\mathcal{Q}$ of reproduction distributions, define the *set-constrained rate-distortion function* to be

$$R(P,\mathcal{Q},d) = \inf_{Q\in\mathcal{Q}} R(P,Q,d). \tag{16}$$

If we take $\mathcal{Q}$ to be the parametric set $\mathcal{Q}_\Theta$ of (5), then the minimum random coding rate within the parametric class is given by:

$$R(P,\Theta,d) \triangleq R(P,\mathcal{Q}_\Theta,d) = \inf_{\theta\in\Theta} R(P,\theta,d) \quad . \tag{17}$$

We call $R(P, \Theta, d)$ the *(parameter-)constrained rate-distortion function* [18]. We denote the parameter that achieves this function, whenever it exists and is unique, as θ^*, namely:

$$\theta^* \triangleq \theta^*(P, \Theta, d) \triangleq \arg\min_{\theta \in \Theta} R(P, \theta, d) \quad . \tag{18}$$

If the optimum is not unique, then a similar definition can be made for the set of optimal parameters. For example, see the Gaussian case: If $Q_\theta = \{\mathcal{N}(0, \theta), \theta \geq 0\}$ and we use mean square distortion, then

$$R(P, \Theta, d) = \frac{1}{2} \log(\frac{\sigma^2}{d})$$

for any source P with variance $\sigma^2 \geq d$, and it is achieved by the parameter $\theta = \sigma^2 - d$ [9, example 2]. See further examples in [23] and in appendix C below.

4. **Maximum Likelihood and divergence projection:** In the case of a memoryless reproduction[2], the ML estimator depends on the vector $\mathbf{y}$ through its *type* or empirical distribution $Q_{\mathbf{y}}$ only, thus we define: $\theta_{ML}(Q_{\mathbf{y}}) \triangleq \theta_{ML}(\mathbf{y})$, where $\theta_{ML}(\mathbf{y})$ is as in (7). This is equivalent to finding the parameter θ inducing the distribution that matches the empirical distribution best in the divergence sense:

$$\theta_{ML}(Q_{\mathbf{y}}) = \arg\max_{\theta} E_{Q_{\mathbf{y}}} \log Q_\theta(\mathbf{y}) = \arg\min_{\theta} D(Q_{\mathbf{y}} \| Q_\theta) \quad . \tag{19}$$

The minimizer in (19) is called the *reverse I-projection* of $Q_{\mathbf{y}}$ to $\mathcal{Q}_\Theta$ [7].

5. **Role of the Smoothing Block:** The smoothing block is not actually necessary for convergence of the system to $R(P, \Theta, d)$ in the limit of large word length ($\ell \to \infty$). In fact, we will prove convergence regardless of the choice of the smoothing coefficient α. It is rather a practical way to strike a balance between the speed of convergence and the steady-state error for finite word length ℓ. The way we replace the average type by a measured one and compensate for randomness with a small step size is reminiscent of the way that the least mean square (LMS) algorithm replaces gradient with its estimation by one measurement with a small step size (see e.g. [16]). One may suggest ways to change step size over time, as is done with step sizes in adaptive filtering.

Doing the parameter update by averaging of parameter rather than by averaging of distributions requires, however, the parameterization to satisfy some technical conditions, that we will specify in the sequel.

II-3 A Simple Example

As discussed above, the base distribution Q_U and the parametric transformation $T_\theta(\cdot)$ generate a parametric set of distributions $\{Q_\theta, \theta \in \Theta\}$. For a finite-alphabet memoryless codebook $|\mathcal{Y}| < \infty$, we can choose the parameters as the distributions themselves:

[2] A generalization of this relation to sources with memory is discussed in Section IV.

$Q_\theta = \theta$, which in this case form a finite-dimensional set. To generate a desired Q_θ, the base codebook is uniform i.i.d. over the unit interval, and the transformation $T_\theta(\cdot)$ is a step function which generates Q_θ by appropriate thresholding. In general, the set $\Theta = \{\theta\}$ can be any subset of the simplex.

The maximum likelihood estimation is easily computed as follows:

1. Obtain the type of the chosen codeword, $Q_{\mathbf{y}}$.

2. Compute $\theta_{ML}(\mathbf{y}) = \arg\min_{\theta\in\Theta} D(Q_{\mathbf{y}}\|\theta)$. Note that if Θ is the whole simplex, then always $\theta_{ML}(Q_{\mathbf{y}}) = Q_{\mathbf{y}}$.

For the case where $\mathcal{Q}_\Theta$ is the whole simplex, this system coincides with the one suggested in [23], except for the practical advantages of adaptive codebook and smoothing. Examples to constrained sets Q_Θ include:

1. Minimum or maximum letter probability in the reproduction (All components of Q_Θ are bounded from above or from below).

2. Maximum number of non-zero probability letters from $\mathcal{Y}$ used in the reproduction.

As we shall see in Section III, convergence to the optimum is guaranteed when $\mathcal{Q}_\Theta$ is convex, thus we can prove convergence for the first case mentioned above, but not for the second one. In Section V we will see other examples for parametric families of reproduction distributions, such as Gaussian mixtures and auto-regressive process.

III System Performance for Finite-Alphabet Memoryless Sources and Codebooks

Our understanding of the behavior of the feedback loop at each iteration is based upon the "favorite type" property (13). This property gives direct insight into the behavior of an idealized system: As $\ell \to \infty$, the type of the chosen codeword $Q_{\mathbf{y}}$ approaches the favorite type $Q^*(P, Q, d)$ by (13), the parameter estimate (7) is no longer random, the smoothing block is no longer needed (i.e., $\alpha = 1$ in (8)), so we obtain the deterministic recursion:

$$\theta_{n+1} = \theta_{ML}\Big(Q^*(P, \theta_n, d)\Big) \quad , n = 1, 2, \ldots \quad , \tag{20}$$

where $\theta_{ML}(\cdot)$ was defined in (19). For this idealized model, Theorem 1 bellow shows convergence to the constrained rate-distortion function $R(P, \Theta, d)$ as the number of adaptation steps $n \to \infty$ under a convexity condition.

While this theorem demonstrates convergence of the idealized NPA system, it is not sufficient for showing convergence for finite ℓ. For finite word length ℓ the type of the selected codeword $\mathbf{y}_n$ is random and the recursion (20) becomes a stochastic one:

$$\theta_{n+1,\ell} = s\Big(\theta_{1,\ell} \ldots \theta_{n,\ell}, \theta_{ML}(\mathbf{y}_{n,\ell})\Big) \quad , n = 1, 2, \ldots \quad , \tag{21}$$

where we use $\theta_{n,\ell}$ for the parameter vector θ_n when the word length is ℓ, in order to emphasize the dependance upon the finite word length. In the stochastic setting, a single realization of the string matching process (one measurement of a vector y_1^ℓ) might produce a type which is very far from the favorite type $Q^*(P, \theta_n, d)$, thus cause the type sequence to deviate considerably from its idealized course. Theorem 2 in the sequel will nevertheless prove stochastic convergence of the type sequence defined by (21) and (8), under a few more technical conditions, in the limit of large word length ℓ.

The basic ingredients used to prove the theorems, namely the favorite type property, convexity and alternating minimization arguments, all hold for sources and reproductions over general alphabets. We will restrict our theorems, however, to the finite-alphabet case, for the sake of simplicity.

Theorem 1 (Natural Parameter Adaptation for $\ell = \infty$) *Let $\mathcal{Q}_\Theta$ be a parametric set of discrete, memoryless distributions. For any initial guess θ_1 inducing an initial distribution Q_{θ_1} with no zero elements, the deterministic recursion (20) generates a monotonically non-increasing sequence $R(P, \theta_n, d)$. Moreover, if the set of reproduction distributions $\mathcal{Q}_\Theta$ is convex, then as $n \to \infty$*

$$\begin{aligned} R(P, \theta_n, d) &\to R(P, \Theta, d) \\ \theta_n &\to \theta^* \end{aligned} \tag{22}$$

where $R(P, \Theta, d)$ is the best achievable rate in the reproduction class Θ defined in (17), and $\theta^* = \theta^*(P, \Theta, d)$ is an optimum parameter defined in (18) (if not unique, then it may depend upon θ_1).

The proof is given below. Note that it follows from the above, that θ^* is a fixed point of the recursion (20), thus it is a solution of the equation

$$\theta = \theta_{ML}\Big(Q^*(P, \theta, d)\Big) \quad . \tag{23}$$

Unlike the fixed point of (2), $Q^*(P, \theta^*, d)$ is in general not equal to Q_{θ^*} and is not even in Q_Θ. In other words, the matching codewords are not typical with any parametric distribution, even in the steady state.

The proof is based upon showing that the recursion (20) is an instant of alternating minimization of divergence between convex sets [8]: If $\mathcal{B}$ and $\mathcal{A}$ are convex sets of non-negative measures (e.g. distributions), then the following recursion:

$$\begin{aligned} B_{i+1} &= \arg\min_{B \in \mathcal{B}} D(B \| A_i) \\ A_{i+1} &= \arg\min_{A \in \mathcal{A}} D(B_{i+1} \| A) \quad , i = 1, 2, 3 \cdots \end{aligned} \tag{24}$$

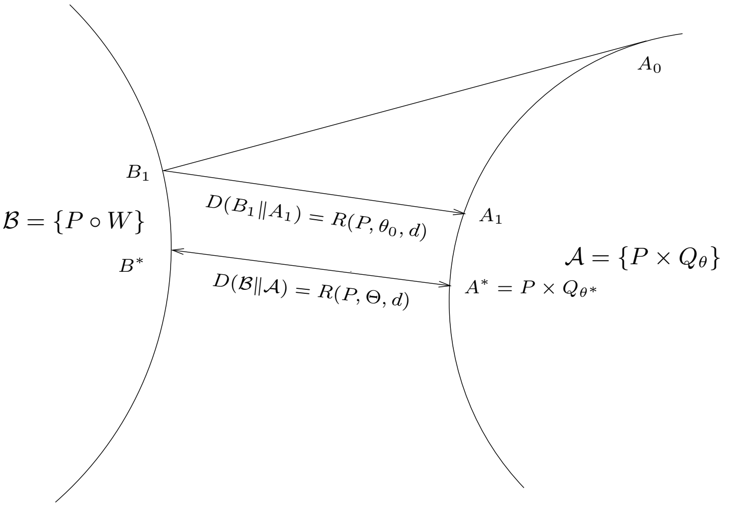
A_0
B_1
$\mathcal{B} = \{P \circ W\}$
$D(B_1\|A_1) = R(P, \theta_0, d)$
A_1
$\mathcal{A} = \{P \times Q_\theta\}$
B^*
$D(\mathcal{B}\|\mathcal{A}) = R(P, \Theta, d)$
$A^* = P \times Q_{\theta^*}$

Figure 3: Alternating Minimization

converges to the minimum divergence between the sets, for any A_0 that has finite divergence from some $B \in \mathcal{B}$ [8, Theorem 3]. Furthermore, if the measures are on a finite alphabet and the sets are closed, exist distributions $B^* \in \mathcal{B}$ and $A^* \in \mathcal{A}$ that achieve this minimum. This alternating minimization process is depicted in Figure 3.

In our system, we identify these sets as:

$$\begin{aligned} \mathcal{B} = \mathcal{B}(P,d) &= \{P \circ W : \rho(P,W) \leq d\} \\ \mathcal{A} = \mathcal{A}(P,\Theta) &= \{P \times Q_\theta : \theta \in \Theta\} \quad , \end{aligned} \tag{25}$$

where the optimal point in the set $\mathcal{A}$ is connected with the optimal parameter θ^* of (18) via $A^* = P \times Q_{\theta^*}$. Substituting (9) in (16), $R(P,\Theta,d)$ can be written as a double minimization:

$$R(P,\Theta,d) = \min_{\theta\in\Theta} \min_{W:\rho(P,W)\leq d} \Big\{ D(P \circ W \| P \times Q_\theta) \Big\} \tag{26}$$

which, using our set definitions, can be rewritten as:

$$R(P,\Theta,d) = \min_{B\in\mathcal{B}} \min_{A\in\mathcal{A}} \Big\{ D(B\|A) \Big\} \quad . \tag{27}$$

Let us also define the set $\tilde{\mathcal{A}} \supseteq \mathcal{A}$:

$$\tilde{\mathcal{A}}(P) = \{P \times Q : \textsf{any } Q\} \quad . \tag{28}$$

Observe from (25) and (12), that a single iteration of (20) can be broken into 3 steps:

1. Minimization of $D(B\|A)$ w.r.t. $B \in \mathcal{B}$ - finding $W^*(P, \theta_n, d)$.

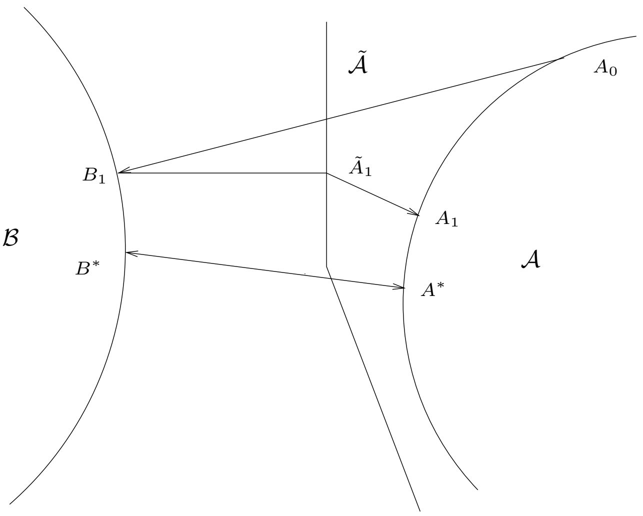

Figure 4: Indirect projection through $\tilde{\mathcal{A}}$

2. Minimization of $D(B\|\tilde{A})$ w.r.t. $\tilde{A} \in \tilde{\mathcal{A}}$ - finding $Q^*(P, \theta_n, d)$.
3. Minimization of $D(\tilde{A}\|A)$ w.r.t. $A \in \mathcal{A}$ - finding $\theta_{ML}\Big(Q^*(P, \theta_n, d)\Big)$.

The key to the proof of Theorem 1 is, that the combination of steps 2 and 3 above can be viewed as a direct minimization of divergence between $\mathcal{B}$ and $\mathcal{A}$, as illustrated in Figure 4[3]. For this we need the following lemma:

Lemma 1 *For any transition distribution $W(y|x)$,*

$$\arg\min_{\theta\in\Theta} D(P\circ W\|P\times Q_\theta) = \theta_{ML}\Big([P\circ W]_y\Big)$$

where $[P\circ W]_y$ denotes the y-marginal of the joint distribution $P\circ W$.

Proof: By an identity of Topsoe[21] (see also [6, in Lemma 13.8.1]), for any Q,

$$D(P\circ W\|P\times Q) = D\Big(P\circ W\|P\times [P\circ W]_y\Big) + D\Big([P\circ W]_y\|Q\Big). \qquad (29)$$

Since the left term of the right hand side does not depend on the choice of Q, and since Q_θ at $\theta = \theta_{ML}([P\circ W]_y)$ minimizes the right term by (19), the lemma follows. □

[3] In Euclidian geometry, $\tilde{A}_1$ and B_1 would both have to be on the normal to the surface of $\mathcal{A}$ at the same point A_1. This is not necessarily the case for minimum divergence projection, as demonstrated in the figure.

Proof of Theorem 1: Recall the double minimization representation of $R(P,\Theta,d)$ (27), using the sets $\mathcal{B}$ and $\mathcal{A}$ of (25). Incorporating (11) and (25), we see that:

$$\arg\min_{B\in\mathcal{B}} D(B\|P\times Q) = P\circ W^*(P,Q,d) \quad .$$

Now we use (28) and Lemma 1 with $W(y|x) = W^*(P,Q,d)$, to see that:

$$\arg\min_{A\in\mathcal{A}} D\Big(P\circ W^*(P,Q,d)\|A\Big) = P\times Q_{\theta_{ML}\Big(Q^*(P,Q,d)\Big)} \quad .$$

This shows, that (20) realizes alternating minimization between the sets $\mathcal{B}$ and $\mathcal{A}$. The first part of the theorem now follows since each minimization can only reduce the divergence. The second part follows because when $\mathcal{Q}_\Theta$ is convex, so is the set $\mathcal{A}_\Theta$. Since $\mathcal{B}$ is always convex, and by the theorem conditions the initial divergence is finite - convergence to the global optimum is assured by [8, Theorem 3]. □

We now turn to our main result, regarding the convergence of the stochastic recursion (21) for a finite word length. Here we need to define the following technical conditions: A parameterization Θ is said to be convex if:

1. The set Θ is convex.
2. $D(Q_{\theta'}\|Q_\theta)$ is convex in θ for all $\theta'\in\Theta$ (a sufficient condition is that the log-likelihood function $\log Q_\theta(\mathbf{y})$ is concave in θ for all $\mathbf{y}$).

A parameterization Θ is *q-bounded* if $Q_\theta(y)\geq q$, where $q>0$, for all $\theta\in\Theta$.

Theorem 2 (Natural Parameter Adaptation for finite ℓ) *Let P be the probability distribution of some discrete memoryless source. Let Θ be some convex, q-bounded parameterization of finite-alphabet memoryless distributions $\{\mathcal{Q}_\Theta\}$. Suppose that $Q^*(P,\Theta,d)$ is unique. If the set $\mathcal{Q}_\Theta$ is convex, then:*

1. *With high probability, for sufficiently large word length ℓ, the sequence of codebooks generated by the NPA system (21) arbitrarily approaches the optimal reproduction within the family Θ and the coding rate arbitrarily approaches the optimum rate, i.e.,*

$$\begin{aligned}\lim_{\ell\to\infty}\lim_{n\to\infty}\Pr\{\|Q_{n,\ell}-Q^*(P,\Theta,d)\|_1>\epsilon\} &= 0 \quad \forall\epsilon>0\\ \lim_{\ell\to\infty}\lim_{n\to\infty}\Pr\{R(P,\theta_{n,\ell},d)-R(P,\Theta,d)>\epsilon\} &= 0 \quad \forall\epsilon>0 \quad ,\end{aligned}\tag{30}$$

   where $Q_{n,\ell}=Q_{\theta_{n,\ell}}$ is the adaptive reproduction distribution at the n-th iteration and $\|\cdot\|_1$ denotes the L_1 norm of the difference between the two distributions viewed as vectors.

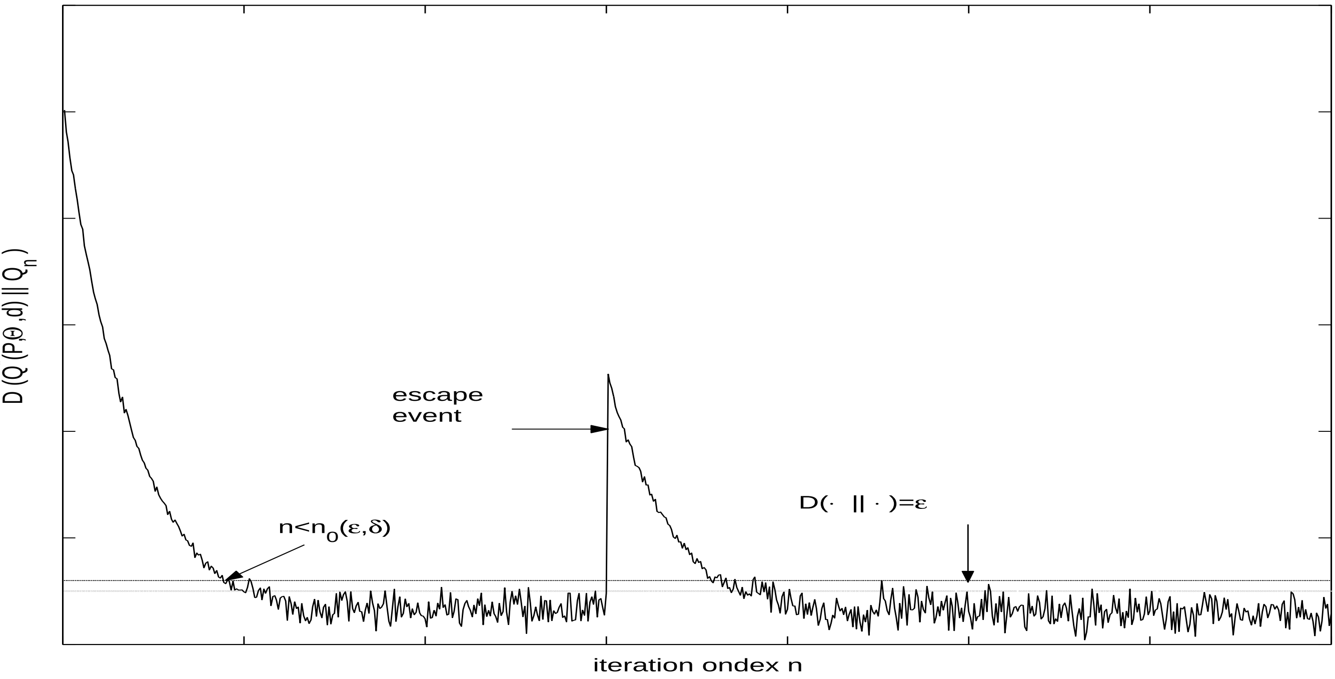

Figure 5: Evolution of a typical type sequence

2. The <u>*average*</u> *rate for the NPA coding session has arbitrary small redundancy with probability 1, i.e.,*

$$\lim_{\ell\to\infty} \Pr\Big\{\limsup_{n\to\infty} \bar{R}_{n,\ell} = R(P,\Theta,d)\Big\} = 1 \quad , \tag{31}$$

where $\bar{R}_{n,\ell} \triangleq \frac{1}{n}\Sigma_{i=1}^{n} R(P,\theta_{i,\ell},d)$.

The proof is given in appendix A. We sketch an outline of the proof after the remarks below.

Remarks:

1. In the first part of the theorem we prove convergence of the type and rate sequence in probability, rather than with probability one, because in fact a typical realization of the parameter sequence does not have a limit. Eventually, due to a source string that is atypical or that reveals some atypical behavior of the codebook, the type of $\mathbf{y}_{n,\ell}$ will be far from $Q^*(P,\theta_{n,\ell},d)$ and the feedback loop will produce a new parameter far from $\theta^*(P,\Theta,d)$. Figure 5 demonstrates this phenomenon.

2. Convergence in probability is with respect to both the source realizations and the randomness of the base codebook. This implicitly means, that while codebooks that do not achieve this performance may exist, the probability to draw such a codebook goes to zero as the word length $\ell \to \infty$.

3. This result strengthens the natural type selection convergence of [23, Theorem 6], since it refers to the stochastic type sequence rather than the average type sequence.

4. The second part of the theorem, showing almost sure convergence of the session-average rate to optimum, resembles classical results on universal coding (e.g. [26]). The first part, in comparison, can give an idea about the instantaneous behavior of the system, so it may be useful when the source statistics are slowly varying.

Outline of proof for Theorem 2 In Lemma 3 we show that for large enough ℓ, the type of the chosen codeword is δ-close to the favorite type. We define the δ-ball of types around the favorite type as:

$$\mathcal{Q}_n^*(P,d,\delta) \triangleq \{Q : \|Q - Q^*(P,\theta_n,d)\|_1 \leq \delta\} \quad . \tag{32}$$

We call the event where the actual chosen word type $Q_{\mathbf{y}_{n,\ell}}$ falls outside that ball an *escape event* and we show that the escape event probability $P_e(\delta,\ell)$ approaches zero as the word length $\ell \to \infty$, uniformly in θ_n, for all $\delta > 0$. Then we turn to the series of lemmas (Lemmas 4-7), all of "deterministic" nature. These Lemmas show that, assuming no escape events, the codebook distribution enters within $n_0(\delta)$ iterations into a small neighborhood, $\epsilon(\delta)$, of the optimal distribution. We call this neighborhood a "black hole" since we show that once entered, the system can not leave this neighborhood, until an escape event occurs. We will show that $\epsilon(\delta) \to 0$ as $\delta \to 0$, while $n_0(\delta)$ is finite for all $\delta > 0$. These four lemmas follow the proof of [23, Theorem 6], with slight changes necessary to accommodate for the parametric setting. For completeness, we will bring proofs of these lemmas in Appendix A. We then observe, that an escape event merely "resets" the system back to some arbitrary initial condition, thus we conclude, that if in the last $n_0(\delta)$ iterations there was no escape event, then the system is within the $\epsilon(\delta)$-neighborhood of the optimum. But since $P_e(\delta,\ell)$ can be made arbitrarily small by looking at large ℓ, the probability of the last n_0 iterations not to contain an escape event goes to 1. Figure 5 describes the behavior of a typical sequence of reproductions.

IV Extension to Sources and Codebooks with Memory

The system we presented in Subsection II-1 does not assume any specific memory model, but the discussion and theorems that follow assume memoryless sources and reproductions. In this section we discuss extension of our optimality results beyond the memoryless case. Throughout this section, we address finite alphabet sources and reproductions only, to avoid mathematical complications that are not necessary to demonstrate our points.

First, consider the case where the source is stationary with memory, but the codebook distribution is memoryless. As shown in [22], the coding rate in this case is the same as if the source were memoryless with the same marginal distribution. Also the marginal empirical distribution of the chosen codeword is asymptotically the same as in the memoryless case [19]. It's easy to see that the ML estimation within a class of memoryless distributions depends on the marginal measurement statistics only. It follows that in the limit of large ℓ the type sequence evolves in the same way as if the source were memoryless, i.e., as in (20), with P standing for the marginal of the source. We conclude that the observations and convergence theorems for the memoryless source case remain valid for general stationary source, provided that the codebook is memoryless. Specifically, $R(P, \Theta, d)$ remains unchanged and convergence of the system to the optimum holds under the same convexity conditions on the set of memoryless codebook distributions.

This result tells us, that unless we introduce memory into the reproductions, we can not gain from the memory of the source. We turn our attention, then, to adaptive codebooks with memory. While we do not pretend to prove rigorously the parallel of all the results proven or used in Section III, we will show how the concepts of the NPA system can be extended to codebooks with memory, and how, as the memory model order grows, our system approaches the rate-distortion function of a general stationary source. In the following subsections, we will present two different codebook memory models.

IV-1 Piecewise I.I.D. Codebooks

Assume that the word length ℓ is a multiplication of the model order k. Divide the base codebook into k-tuples, and draw each one independently using a k-dimensional base distribution. Let the transformation T_θ work on each k-tuple (typically, this base distribution will be i.i.d., inducing memoryless base codewords, and memory will be introduced by the transformation). Then the adaptive codebook is comprised of i.i.d. k-dimensional "super-symbols".

In order to state the performance of the system, we need to redefine the quantities associated with it. First, the rate-distortion function for a stationary, ergodic discrete-time source is defined as [1, Chap. 7]:

$$\begin{aligned} R(P,d) &= \lim_{k\to\infty} \tilde{R}_k(P,d) \quad , \\ \tilde{R}_k(P,d) &= \inf_{W_k:\rho(P_k,W_k)\le d} \frac{1}{k} I(P_k,W_k) \quad , \end{aligned} \tag{33}$$

where P_k is the k-order marginal of the stationary source distribution, W_k is some

k-dimensional transition distribution from the source to reproduction alphabet, and

$$\rho(P_k, W_k) = \frac{1}{k} E_{P_k \circ W_k} \sum_{j=0}^{k-1} \rho(X_j, Y_j) \quad .$$

Likewise, $R(P, Q, d)$ is defined as:

$$R(P, Q, d) = \lim_{k \to \infty} \tilde{R}_k(P, Q, d) \quad , \tag{34}$$

where

$$\tilde{R}_k(P, Q, d) = \inf_{W_k : \rho(P_k, W_k) \le d} \frac{1}{k} D(P_k \circ W_k \| P_k \times Q_k) \quad , \tag{35}$$

where Q_k is the k-order marginal of the stationary reproduction distribution (see [5], where also the existence of the limit in (34) is proven under some mixing conditions). If we constrain the stationary reproduction distribution Q to some parametric multidimensional set $\mathcal{Q}_\Theta$ of stationary ergodic distributions, we can still define $\tilde{R}_k(P, \Theta, d)$ and $R(P, \Theta, d)$ as the infima over the parametric set of $\tilde{R}_k(P, Q_\theta, d)$ and $R(P, Q_\theta, d)$ respectively, just as in (17). It's not hard to show that $\tilde{R}_k(P, Q_\theta, d)$ is non-increasing in k but bounded below by $R(P, d)$, thus it has a limit and we can define:

$$R(P, \Theta, d) = \lim_{k \to \infty} \tilde{R}_k(P, \Theta, d) = \lim_{k \to \infty} \inf_{\theta \in \Theta} \tilde{R}_k(P, Q_\theta, d) \quad . \tag{36}$$

It can also be shown, as in [1] for $R(P, d)$, that $R(P, \Theta, d)$ defined that way has the operational meaning of a rate-distortion function.

Analyzing the system with piecewise i.i.d. codebooks turns out to be very simple, since it is equivalent to a memoryless system working on k-dimensional super-symbols. We can apply the favorite type theorem and the i.i.d. coding rate theorem to these super-symbols. The optimal rate associated with these i.i.d. super-symbols happens to be $\tilde{R}_k(P, Q_\theta, d)$ of (35), thus in the limit of a large number of iterations the system can achieve $\tilde{R}_k(P, \Theta, d)$. In the limit of large model order k we could also approach $R(P, \Theta, d)$, but since we are interested in simple codebooks and finite parameter vectors, this limit is of little practical interest. For a finite k, this model has an obvious drawback: Though the source is stationary and has memory, the matching process may treat two adjacent letters as independent, only because they happen to fall into two different super-symbols. To overcome this, we next address the Markov codebook.

IV-2 Markov Codebooks

Let $Q_k(y_k | \mathbf{y}_0^{k-1})$ denote some conditional probability distribution. If it is strictly positive for all $\mathbf{y}_0^{k-1} \in \mathcal{Y}^k$, then it induces a unique stationary distribution $Q(\mathbf{Y}_0^{\ell-1})$. For finite alphabet, we can always create an adaptive codebook with such a distribution.

Define:

$$R_k(P,Q,d) \triangleq \inf_{W_k:\rho(P_0,W_0)\le d} D_{k|k-1}(P\circ W\|P\times Q) \tag{37}$$

where

$$D_{k|k-1}(P\circ W\|P\times Q) \triangleq \sum_{x\in\mathcal{X},y\in\mathcal{Y}} P(\mathbf{x}_0^k)W(\mathbf{y}_0^k|\mathbf{x}_0^k)\log\frac{W(y_k|\mathbf{y}_0^{k-1},x_k)}{Q(y_k|\mathbf{y}_0^{k-1})} \quad . \tag{38}$$

$R_k(P,Q,d)$ was first defined in [22], though in a different form. In Appendix B we show that the forms of [22] and (37) are equivalent, thus we can use the main result of [22] to establish the following: For a Markov-k reproduction Q, and any stationary ergodic source P, the coding rate of our system with a Markov-k equivalent codebook with distribution Q converges to $R_k(P,Q,d)$ with probability one[4]. While the definition of $R_k(P,Q,d)$ in [22] depends on the full multi-dimensional distribution of the source string, our definition depends on the source conditional distribution of order k only. This allows us to make the following proposition, stating that the encoder performance depends on that distribution only:

Proposition 1 *If the codebook distribution Q is Markov-k, then*

$$R_m(P,Q,d) = R_k(P,Q,d) \quad \forall m\ge k \quad .$$

This proposition is a natural extension to the result in [22], stating that performance is the same for all sources having the same marginal, when the codebook is memoryless - but to the best of our knowledge, it was never published. We can use $R_k(P,Q,d)$ to define:

$$\begin{aligned} R_k(P,d) &= \inf_Q R_k(P,Q,d) \\ R_k(P,\Theta,d) &= \inf_{\theta\in\Theta} R_k(P,Q_\theta,d) \end{aligned} \tag{39}$$

The following Lemma, which we prove in Appendix B, establishes the observation that for a fixed memory order k, a Markov codebook performs better than a piecewise-i.i.d. codebook:

Lemma 2

$$R_k(P,d) \le \tilde{R}_k(P,d)$$

[4] As a matter of fact, the system discussed in [22] is different than ours, as their encoder searches for matches through a *database* rather than a codebook of independent codewords, so that the "codewords" at the input of the encoder are statistically dependent. It can be shown, though, that performance of both systems is approximately the same when the word length ℓ is large.

As a direct consequence of this Lemma and of (33), we have that $\lim_{k\to\infty} R_k(P,d) = R(P,d)$.

Whenever (39) has a minimizer, we define:

$$\theta^* \triangleq \theta^*(P,\Theta,d) = \arg\min_{\theta\in\Theta} R_k(P,Q_\theta,d) \tag{40}$$

Defining:

$$Q_k^*(P,Q,d) \triangleq \arg\min_{Q} D_{k|k-1}(P\circ W \| P\times Q) \tag{41}$$

we can write a recursion (recall (20)):

$$\theta_{n+1} = \theta_{ML}\Big(Q_k^*(P,\theta_n,d)\Big). \tag{42}$$

For this recursion we have the following Theorem, which we prove in Appendix B. It can be seen as the equivalent of Theorem 1 of the memoryless case.

Theorem 3 (Natural Parameter Adaptation for Markov Codebooks, $\ell = \infty$) *Let $\mathcal{Q}_\Theta$ be a set of finite-alphabet, stationary, ergodic distributions. For any initial guess θ_1 inducing an initial distribution $Q_{k,1}$ with no zero elements, the deterministic recursion (42) generates a monotonically non-increasing sequence $R(P,\theta_n,d)$. Moreover, if the set of conditional reproduction distributions $\mathcal{Q}_{k|k-1,\Theta}$ induced by the set Θ is convex, then as $n\to\infty$*

$$\begin{array}{rcl} R_k(P,\theta_n,d) & \to & R_k(P,\Theta,d) \\ \theta_n & \to & \theta^* \end{array} \tag{43}$$

where $\theta^* = \theta_k^*(P,\Theta,d)$ is defined in (40).

We would like to give $Q_k^*(P,Q,d)$ the meaning of the "favorite type" as in the memoryless case (13), but unfortunately we are not aware of any result regarding the asymptotic statistics of the chosen codeword when the codebook has memory. However, we believe that this property does hold. For this we need to define $Q_{\mathbf{y},k|k-1}(\mathbf{y})$, the k-order conditional type of a vector $\mathbf{y}_0^{\ell-1}$, which is the the ratio between the number of occurrences of $\mathbf{y}_0^{k-1}$ and the number of occurrences of $\mathbf{y}_0^{k}$ in the vector $\mathbf{y}$.

Conjuncture 1 (Favorite Type for Markov Codebooks)

$$Q_{\mathbf{y},k|k-1} \to Q_k^*(P,Q,d) \ \textit{in prob.}$$

Ramarks:

1. **Convergence of Markov NPA systems.** If the conjuncture holds, then the system does obey recursion (42) in the limit of large codewords. For finite word

length ℓ, a result similar to to the memoryless Theorem 2 can be derived as well, but it is outside the scope of this work.

2. **ML Estimation in Markov models.** The ML estimator is, for any $\ell > k$:

$$\theta_{ML}(\mathbf{y}) \overset{\Delta}{=} \arg\max_{\theta\in\Theta} Q_\theta(\mathbf{y}_0^{\ell-1}) = \arg\max_{\theta\in\Theta} \log Q_\theta(\mathbf{y}_0^{k-1}) + \sum_{m=k}^{\ell-1} \log Q_\theta(y_m|\mathbf{y}_{m-k}^{m-1}) \quad ,$$

which, as the word length $\ell \to \infty$, is equivalent to a divergence projection:

$$\lim_{\ell\to\infty} \theta_{ML}(\mathbf{y}) = \arg\max_{\theta\in\Theta} \sum_{m=k}^{\ell-1} \log Q_\theta(y_m|\mathbf{y}_{m-k}^{m-1}) = \arg\min_{\theta\in\Theta} D\Big(Q_{\mathbf{y},k|k-1} \| Q_\theta(y_k|\mathbf{y}_0^{k-1})\Big) \quad . \tag{44}$$

Comparing with (19), we see that the divergence between memoryless distributions is replaced by the divergence between marginal distributions.

3. **Convexity condition on conditional distributions.** The Theorem requires that the allowed k-order *conditional* distributions form a convex set. Note, that this is not equivalent to the requirement that the k-order *marginals* of the stationary distributions form a convex set. This is similar to the case with memoryless distribution: a convex set of single-letter distribution does not induce a convex set of vector distributions.

4. **Dependance of performance upon the source memory.** As stated above in Proposition 1, our definitions of $R_k(P,\theta,d)$ and $Q_k^*(P,\theta,d)$ depend on the source through its k-order marginal transition distribution only. We conclude, that asymptotic performance of the NPA system with Markov-k codebooks is unaffected by source characteristics of higher order than k.

V Specific Examples

We turn now to describe the specific structure and behavior of the NPA system for two special examples of NPA systems.

V-1 Memoryless Mixture Codebook

In this example we model the reproduction as a mixture of M predefined distributions $Q_m, m = 1 \ldots M$, where the unknown parameters are $M-1$ weights of the components in this mixture.

Figure 6 illustrates the generation of a single equivalent codeword. The fixed code letters consist each of $M+1$ components: M components drawn according to Q_0^{M-1} and another component drawn uniformly between 0 and 1. All components within a letter and all code letters are drawn independently. The transformation uses the parameter vector to threshold the last component of each base code letter (as done

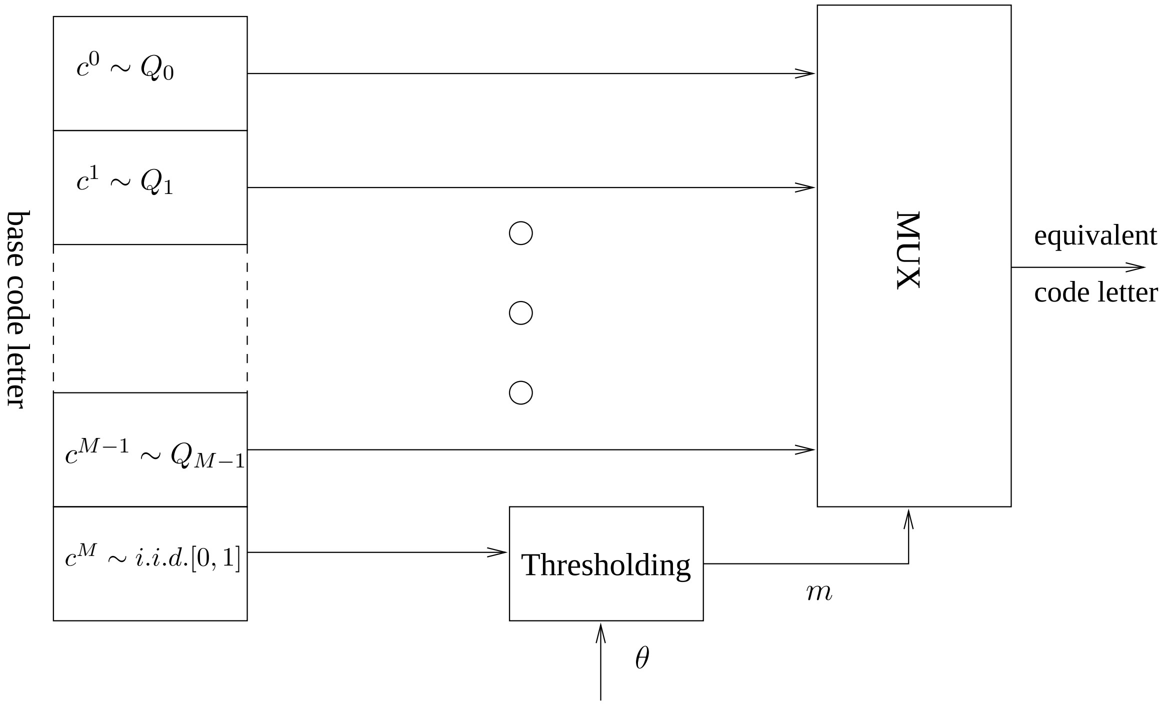

Figure 6: Formation of a Single Memoryless Mixture Equivalent Code Letter

in Subsection II-3), to produce an integer between 0 and $(M-1)$. The equivalent codebook letter is the fixed codebook letter component indexed by that integer.

ote that, in order for these theorems to hold, all distributions must be bounded away from 0. Thus, a Gaussian mixture is not a valid case, though a mixture of Gaussians truncated at any value is valid, and the Gaussian mixture case can be approached. Also note that the memoryless codebook of Subsection II-3 is a special case of the one described here with $|\mathcal{Y}| = M$, and that when $|\mathcal{Y}| < M$ the parametric representation of a distribution is not single. We will assume that $M \geq |\mathcal{Y}|$. Typically it will be large, or even infinite.

Now we turn to the maximum likelihood estimation. This specific ML problem

The parameterization and the parametric set defined above are convex, thus Theorems 1 and 2 apply. Maximum likelihood estimation of the weighting parameters $\theta_1 \ldots \theta_M$ has been studied in estimation theory. The straightforward solution is difficult, but there is an iterative procedure [15]: Start with any initial guess $\theta^{(0)}$, and iteratively compute:

$$\begin{aligned} C_i &= \sum_y \hat{Q}_n(y) \frac{\theta_i^{(n)} Q_i(y)}{\sum_j \theta_j^{(n)} Q_j(y)} \\ \theta_i^{(n+1)} &= \frac{C_i}{\sum_j C_j} \end{aligned} \tag{45}$$

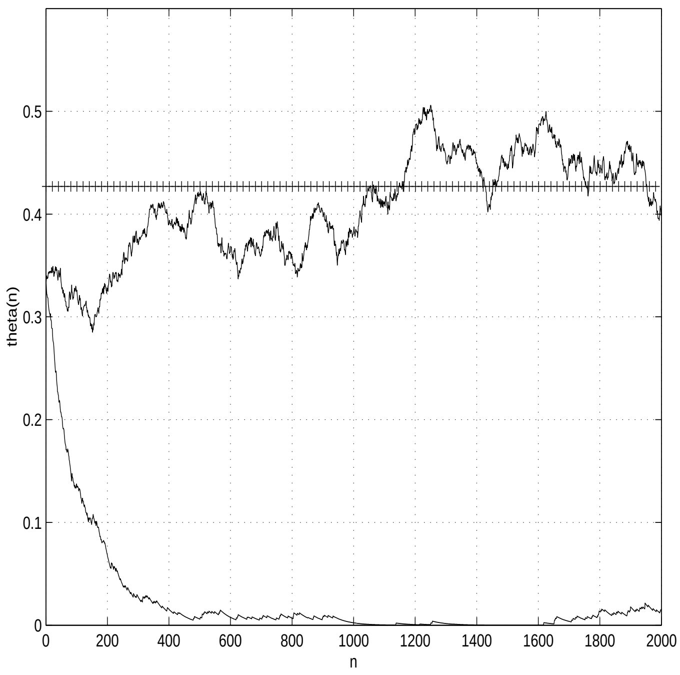

Figure 7: Adaptation of θ vs. the Optimal Value for the Gaussian Mixture Case

where $\hat{Q}_n(y)$ is the marginal of the chosen codeword. This is a special case of the estimate-maximize (EM) algorithm for ML estimation [10].

For the case where the components Q_m are Gaussian, we bring simulation results[5]. We took $Q_m \sim N(\eta_m, \sigma^2)$ with M=3, $\eta_1 = -1$, $\eta_2 = 0$, $\eta_3 = 1$, $\sigma^2 = \frac{1}{25}$. The source is a Gaussian mixture of the same components Q_m of the codebook, with weights $\{0.1, 0.3, 0.7\}$. Distortion measure is square distance.

We used the BA algorithm combined with ML estimation (using the EM algorithm), to find that the optimal parameter vector at the point of slope -1 of the constrained rate-distortion function is $\{0, 0.427, 0.533\}$ (about such computations see [18]). Then we run the system, using word length $\ell = 10$ and the smoothing of (8) with $\alpha = 0.02$. For initial condition we choose equal weights to all components. the results are shown in Figure 7 (the third parameter is omitted from the plot since it is redundant). It is evident that the system tends towards the optimal solution.

[5] Since in this case the adaptive distributions are continuous, it doesn't fall under the convergence Theorems that we formally proved. Also note that Gaussian distributions are not bounded away from zero, although a continuous-alphabet version of the theorems proven with similar technique would require that boundness property. This can be solved by truncating the densities at some finite but large value.

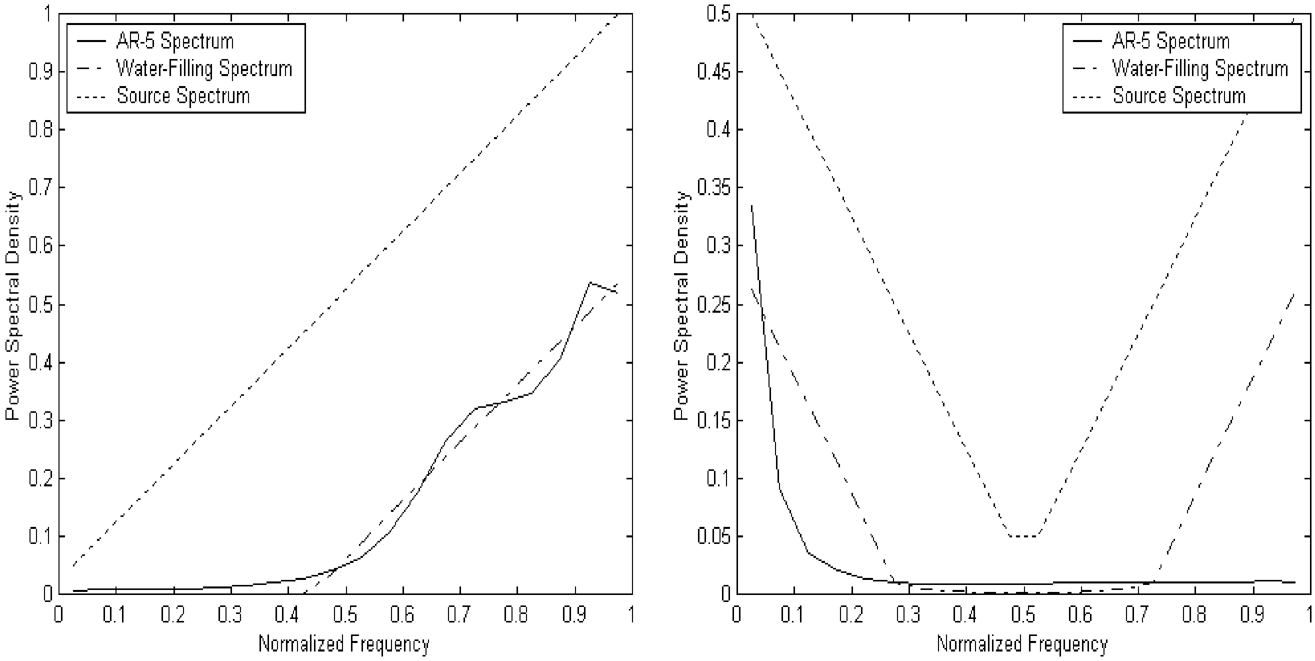

Figure 8: Source and Reconstruction Spectrum

V-2 Auto-Regressive Gaussian Codebooks

To demonstrate Markov codebooks of Subsection IV-2, we choose the example of auto-regressive (AR)-Gaussian models. The base codebook is drawn Gaussian i.i.d., and the transformation is an all-pole filter, with the parameter vector being its coefficients[6]. ML estimation of Gaussian-AR parameters is straightforward: It is the empirical value of the correlations of the chosen codeword,

$$\theta^i_{ML}(\mathbf{y}_n) = \frac{1}{\ell - i} \sum_{j=1}^{\ell-i} y^j_n \cdot y^{j+i}_n, \quad i = 0 \ldots k-1 \quad . \tag{46}$$

The optimal reproduction for this case is well known, and it is obtained through the water-filling method [6]. The next figure demonstrates the convexity problem: It shows the spectrum to which the system converges ($\ell = \infty$) against the water-filling spectrum. In the case on the left, the AR spectrum well approximates the optimal spectrum, whereas on the right, for a more difficult case where the water-filling spectrum consists of two separate bands, for some initial condition only one of the bands exists in the AR spectrum.

[6] A codebook generated this way is only asymptotically stationary. One may think of ways to set filter initial conditions in order to comply with the stationary distribution, but when the word length ℓ is much larger than the filter order k this is not substantial.

VI Discussion

We showed that a block-wise backward adaptive parametric encoder, with maximum likelihood estimation of the parameters from the *reconstructed* source, converges to the parameter-constrained rate distortion function. In particular, this system reproduces the best weights in the family of Gaussian mixture reproductions, and (ignoring the convexity issue) for quadratic distortion it reproduces the reverse water-filling spectrum within the family of stationary Gaussian reproductions [6].

In the special case of linear auto-regressive (AR) reproduction, the proposed system is similar to adaptive differential pulse code modulation (ADPCM), the parameters being the linear prediction coefficients (LPC) [17]. The main difference between the two systems is that while in ADPCM the quantizer is *scalar*, our analysis requires large block length in order to obtain the "natural parameter adaptation" (NPA) property. It is interesting to note, though, that parameter estimation in ADPCM is done over a block of past samples; so in a sense it can be viewed as a hybrid scalar-vector system, whose exact low rate behavior requires further study.

Another related coding system, which does employ *vector* quantization, is code excited linear prediction (CELP) [13]. However, regular CELP uses *forward*-adaptation scheme, hence it reproduces the source spectrum rather than the optimum reverse water-filling solution. As discussed in the Introduction, this spectral mismatch may result in large loss of performance at low coding rate. A backward adaptive variant of CELP, called Low-Delay CELP, [4], corresponds to the linear AR reproduction case of the NPA system. Hence, a possible implication of our results is that LD-CELP indeed has the potential of achieving the Gaussian rate-distortion function at any coding rate, provided that the issue of non-convexity of this family is properly resolved (see the discussion below).

An alternative popular method for source-matched non-parametric lossy compression is the generalized Lloyd algorithm, which allows to iteratively design an optimal vector quantizer [14]. However, this method suffers from very high complexity, and therefore is limited to off-line applications and small codebook dimensions.

A few remarks are in order regarding the limitations of NPA as the basis for a practical universal lossy coding scheme.

1. *Choice of the parametric family*: When the parametric family $\mathcal{Q}_\Theta$ is "too narrow" relative to the sources to be compressed, the resulting performance may be poor. For example, as discussed in Section IV, memoryless reproduction cannot provide "memory gain", thus it is not efficient for encoding sources with strong memory. Another example is that of encoding non-Gaussian sources under quadratic distortion using Gaussian reproduction. As shown in Appendix C, if Θ corresponds to the family of all Gaussian distributions, then for any source P

$$R(P, \Theta, d) = R(P_G, \Theta, d) = R(P_G, d)$$

where P_G denotes the Gaussian source having the same auto-correlation as P. This rate may be much higher than the rate-distortion function, $R(P, d)$, if the source is far from Gaussianity.

2. *Lossless encoding of the index:* We may reduce this sensitivity of the coding performance to the choice of the reproduction family Θ, by *conditional* entropy coding of the index i_n given the current codebook. Specifically, it follows from [19] that for uniform or flat Gaussian reproductions and for *any source*, $H(i_n|\theta_n)$ is asymptotically at most $C^*(\rho)$ bits away from the rate-distortion function, where for quadratic distortion $C^* = 1/2$ bit. In practice, the conditional index entropy may be achieved by re-ordering the codewords according to their probability to match the source. Thus, the increased robustness comes at the cost of lossless encoding complexity. Furthermore, entropy coding the index would cause higher variations of the output coding rate, which is un-desired for some real-time applications. Finally, to achieve exactly zero redundancy, $\mathcal{Q}_\Theta$ must contain the true Q^* corresponding to the source, even when lossless encoding of the index is applied.

3. *Codebook search complexity:* The search for the first matching codeword requires computation of distortion for $\sim 2^{\ell R}$ codewords. Normally, to avoid exponential search time, random codebooks are replaced by structured ones, e.g., lattice codes. But then also the search is for the *closest* word and not for the first match within a desired distortion. This gives rise to a fixed rate (minimum distortion) universal scheme. It remains for further study how the search for the first distortion-matching codeword, needed for the NTS mechanism, can be implemented with a structured codebook.

4. *Convexity of the parametric family:* Many parametric families of interest, such as the linear AR-Gaussian family, are not convex. As discussed in Section V, we cannot ensure convergence of NPA to Q^* in these cases. To overcome the convexity issue, we suggest to select a set of "corner" points of Θ, and approximate $\mathcal{Q}_\Theta$ by the convex hull of the distributions associated with these base parameters. Specifically, Choose some M parameter values $\theta^1, \theta^1, \ldots, \theta^M$. Define a new M-dimensional parameter vector $\tilde{\theta}$, which will play the role of the weight of each original parameter. For each codeword letter, draw (according to $\tilde{\theta}$) which θ value is to be used. All the adaptation is done on the vector $\tilde{\theta}$, and since $Q_{\tilde{\Theta}}$ is the simplex, the resulting family is convex, and convergence to the best parameters is ensured. Clearly, the set of reproduction distributions over which we optimize, is a subset of the convex hull of the original Q_Θ. Also, using a large number of base parameters M, we can approach this convex hull. However, we are interested in a system with low parameter dimension, thus the remaining open question is, how to choose a small number of points that will form a good basis.

As a concluding remark, we go back to the connection between the NPA system and the Blahut-Arimoto algorithm for computation of the rate-distortion function [2]. Each step of the deterministic recursion (20) is equivalent to one step of the

BA algorithm[7], followed by a step of maximum likelihood estimation, thus we have at hand a computational means for finding the constrained rate-distortion function. This may still be a difficult computational task since the ML estimator itself is not always readily computed. An iterative way to compute the ML estimator is the estimate-maximize (EM) algorithm [10]. In cases where this algorithm is applicable, a step of (20) may be broken into a BA step, followed by a large number of EM steps - but, in fact, an alternative recursion, which at each step performs one BA step and *one* EM step converges to θ^* under the same conditions [18]. This allows easier computation of the constrained rate-distortion function, and also allows us to replace the ML estimate block of the NPA system with an "EM-step" block.

Acknowledgements

We thank K. Rose and E. Tuncel for helpful discussions.

Appendix

A Proof of Theorem 2

Lemma 3 (Maximal deviation from favorite type) *For any $\delta > 0$ and any $P_e > 0$, there exists ℓ large enough such that for any n the escape event probability* $\Pr\{Q_{\mathbf{y}_{n,\ell}} \notin \mathcal{Q}_n^*(P,Q,d,\delta)\} \leq P_e$.

Proof: By [19, Theorem 2], $\Pr\{Q_{\mathbf{y}_{n,\ell}} \notin \mathcal{Q}_n^*(P,Q,d,\delta)\} \to 0$ exponentially fast in ℓ for all $\delta > 0$, w.p.1 w.r.t. the source string realization. Furthermore, the exponential coefficient is $R(P,Q,d)$, which is bounded from below by $R(P,d) > 0$. Thus the convergence is uniform in θ, hence holds for all θ_n □

For the following series of "deterministic" lemmas we use the notation: Divergence distance of a parameter θ from the optimum parameter θ^* of (18) is:

$$L(\theta) \triangleq D\Big(Q_{\theta^*} \| Q_\theta\Big) \quad , \tag{47}$$

while the redundancy of a parameter over the constrained $R-d$ function is:

$$\Delta(\theta) \triangleq R(P,\theta,d) - R(P,\Theta,d) \quad . \tag{48}$$

Figure 9 demonstrates these basic quantities used in analysis of the NPA iterations. We also use the abbreviation $L_n = L(\theta_n)$ and $\Delta_n = \Delta(\theta_n)$, for the true divergence from the optimum and the redundancy at step n, respectively. We will also use $e_n(\delta)$

[7] or rather, a fixed-distortion version of the BA algorithm

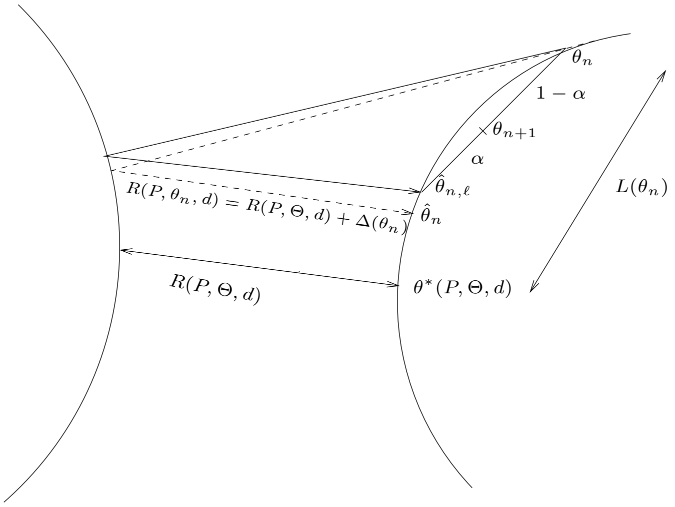

Figure 9: NPA iteration

to denote the escape event defined in (32), and $\overline{e_n(\delta)}$ to denote the complimentary of the escape event (i.e. when the selected type falls within the δ-ball of the favorite type).

Lemma 4 (Minimum decrease of divergence in NPA iteration with $\ell = \infty$: "ideal projection")

$$L(\hat{\theta}_n) \leq L_n - \Delta_n \quad ,$$

where $\hat{\theta}_n = \theta_{ML}\Big(Q^*(P,\theta_n,d)\Big)$ *was defined in (20).*

Since this Lemma deals with ideal (deterministic) projections, it follows directly from analysis in [8] (see also [23, Lemma 1]), but we will bring it here to keep this paper self-contained.

Proof: From the "three points property" and "four points property" of [8, Theorem 3] it follows that in an alternating minimization of $D(B\|A)$:

$$D(B_n\|A_n) - D(B^*\|A^*) \leq D(B^*\|A_n) - D(B^*\|A_{n+1})$$

where B^* and A^* are the points of minimum divergence. Now we turn to the iteration of NTS + ML shown in the proof of Theorem 1 to be a special case of this mechanism. The left hand side is, by definition, Δ_n, while the right hand side is:

$$\begin{aligned} & D\Big(P \circ W^*(P,\mathcal{Q},d)\|P \times Q_n\Big) - D\Big(P \circ W^*(P,\mathcal{Q},d)\|P \times Q_{\hat{\theta}_n}\Big) \\ = \; & E_{P(x)\circ W^*(y|x)} \log \frac{Q_{\hat{\theta}_n}(y)}{Q_n(y)} \\ = \; & D(Q^*\|Q_n) - D(Q^*\|Q_{\hat{\theta}_n}) = L_n - L(\hat{\theta}_n) \quad . \end{aligned}$$

□

Lemma 5 (Minimum decrease of divergence in NPA iteration with finite ℓ: "noisy projection") *If there was no escape event at step n, i.e. $\overline{e_n(\delta)}$, then*

$$L_{n+1} \leq L_n - \alpha\Big(\Delta_n - \nu(\delta)\Big) \tag{49}$$

where $0 < \alpha \leq 1$ is the smoothing coefficient of (8) and $\nu(\delta) = \nu(\delta, q) \to 0$ as $\delta \to 0$ for all $q > 0$, where q is a lower bound on the letter probability in all distributions of $\mathcal{Q}_\Theta$ (see before Theorem statement).

Proof: By definition,

$$L_{n+1} = L(\theta_{n+1}) = D(Q_{\theta^*} \| Q_{\theta_{n+1}})$$

and using the smoothing formula (8) we have that:

$$\begin{aligned} L_{n+1} &= D\Big(Q_{\theta^*} \| Q_{(1-\alpha)\theta_n + \alpha\hat{\theta}_{n,\ell}}\Big) \\ &\leq (1-\alpha)L_n + \alpha D\Big(Q_{\theta^*} \| Q_{\hat{\theta}_{n,\ell}}\Big) \\ &\leq L_n - \alpha\Delta_n + \alpha\Big[D\Big(Q_{\theta^*} \| Q_{\hat{\theta}_{n,\ell}}\Big) - D\Big(Q_{\theta^*} \| Q_{\hat{\theta}_n}\Big)\Big] \end{aligned}$$

where $\hat{\theta}_n$, the parameter associated with the ideal projection, is as in the previous lemma, and $\hat{\theta}_{n,\ell}$, is the parameter associated with the noisy projection. The first inequality is justified by the convex parameterization assumption, and the second is a consequence of Lemma 4. Then it remains to be seen that:

$$D\Big(Q_{\theta^*} \| Q_{\hat{\theta}_{n,\ell}}\Big) - D\Big(Q_{\theta^*} \| Q_{\hat{\theta}_n}\Big) \leq \nu(\delta) \quad , \tag{50}$$

where $\nu(\delta) \to 0$ as $\delta \to 0$. To see this, choose

$$\nu(\delta) = \frac{\delta \log(e)}{(|\mathcal{Y}| \cdot q)^{-\frac{1}{q}} - \delta}$$

where $|\mathcal{Y}|$ is the reproduction alphabet size and q is the minimum reproduction letter probability within the parametric class, positive by the theorem conditions. The inequality (50) then follows by the uniform bound on divergence difference, [23, Lemma 6], whenever δ is small enough to ensure that the denominator of the expression above is positive. Using this choice, $\nu(\delta) \to 0$ as $\delta \to 0$ as required. □

For the following two lemmas we will need the definition:

$$L_\nu = \sup_{\theta \in \Theta : \Delta(\theta) < \nu} L(\theta) \tag{51}$$

for the maximum distance from optimal distribution at a given redundancy. Note that when Q_{θ^*} is unique as in the conditions of the theorem, then L_ν is a monotonically non-decreasing function of ν, and $L_\nu \to 0$ as $\nu \to 0$. Using this, define:

$$\epsilon(\delta) = \alpha\nu(\delta) + L_{\nu(\delta)} \quad , \tag{52}$$

where $\nu(\delta)$ is as in Lemma 5 and α is the smoothing coefficient.

Lemma 6 (Entering an ϵ-neighborhood of Q_{θ^*} in finite time) *For any $\delta > 0$ and for $\epsilon(\delta)$ of (52), there exists a finite $n_0 = n_0(\delta)$, such that if there were no escape events for all first n_0 iterations, i.e. $\overline{e_n(\delta)}$ for $n = 1, \dots n_0$, then $L_n \leq \epsilon(\delta)$ for some $n \leq n_0$. This holds uniformly in the initial parameter θ_1.*

Proof: First, we prove that for some $n \leq n_0$,

$$\Delta_n \leq \alpha\nu(\delta) + \frac{L_1}{n_0} \quad .$$

The way we derive that is somewhat similar to the deterministic analysis of convergence for the BA algorithm in [3]. Summing the result of Lemma 5 for $n = 1, \dots, n_0$, we have that:

$$L_{n_0} \leq L_1 - \alpha[\Sigma_{n=1}^{n_0}\Delta_n - n_0\nu(\delta)] \quad .$$

Asserting $L_{n_0} \geq 0$ and rearranging, we see that:

$$\bar{\Delta}_{n_0} \leq \frac{L_1 - L_{n_0}}{n_0} + \alpha\nu(\delta) \leq \frac{L_1}{n_0} + \alpha\nu(\delta) \quad ,$$

where $\bar{\Delta}_n$ is the average of $\Delta_1 \dots \Delta_n$[8]. Since at least one term of the average must be as small as the average, exists $n \leq n_0$ s.t. Δ_n is as required. Thus, for every $\Delta > \alpha\nu(\delta)$ we can find a finite n_0 s.t. $\Delta_n \leq \Delta$. Now we turn to L_ν as defined in (51). By monotonicity and continuity of L_ν, $L_n \leq \epsilon$ for all $\epsilon > L_{\alpha\nu(\delta)}$ within a finite number iterations. Since $\epsilon(\delta)$ satisfies this condition, the Lemma follows. □

Lemma 7 ("Black Hole") *If there was no escape event at step n, i.e. $\overline{e_n(\delta)}$, and if $L_n \leq \epsilon(\delta)$, where $\epsilon(\delta)$ is defined in (52), then $L_{n+1} \leq \epsilon(\delta)$ as well.*

Proof: By Lemma 5 $L_{n+1} \leq L_n + \alpha\nu$, thus if $L_n \leq L_\nu$ then $L_{n+1} \leq L_\nu + \alpha\nu \leq \epsilon$. On the other hand, if $L_n > L_\nu$ then by definition (51) $\Delta_n > \nu$, thus $L_n \leq L_{n+1} \leq \epsilon$. □

As a direct result of these last two lemmas, we have:

[8] In the deterministic case, i.e. $\nu(\delta) = 0$, this reduces to the result of [3].

Corollary 1 (Type sequence limit without escape events) *If there were no escape events for all first n_0 iterations, i.e. $\overline{e_n(\delta)}$ for $n = 1, 2, \ldots n_0$, then $L_{n_0} \leq \epsilon(\delta)$ of (52).*

Proof of Theorem 2: For the first part, consider an escape event at some iteration n. In such case, the type sequence may take some value outside the ϵ neighborhood of Q_{θ^*} even if it was inside that neighborhood already. But since we assumed that θ_1 could be any $\theta \in \Theta$, for instance the one leading to the largest L-difference, the distance from optimum after such event can not be worse. Therefore, Corollary 1 holds, with a time shift, to the parameter evolution after an escape event. Thus, for any $\delta > 0$ and $n \geq n_0(\delta)$ we can write:

$$\Pr\{L_n > \epsilon\} \leq \Pr\{\bigcup_{k=n-n_0+1}^{n} e_n(\delta)\} \quad , n > n_0(\delta) \quad ,$$

with $\epsilon = \epsilon(\delta) = \alpha\nu(\delta) + L_{\nu(\delta)}$ as in the corollary and $n_0 = n_0(\delta)$ of Lemma 6. Now, using a union bound for these (dependent) events, we assert:

$$\Pr\{L_n > \epsilon\} \leq \Sigma_{k=n-n_0+1}^{n} \Pr\{e_n(\delta)\} \quad , n > n_0 \quad .$$

Incorporating Lemma 3, we have that for any $P_e > 0$ and large enough ℓ:

$$Pr\{L_n > \epsilon(\delta)\} \leq n_0(\delta) P_e(\delta) \quad , n > n_0(\delta) \tag{53}$$

where n_0 is finite, and $\epsilon \to 0$ whenever $\delta \to 0$. The first limit in (30) now follows since closeness in the divergence sense implies closeness in $\mathcal{L}_1$, while the second limit follows since by Lemma 4 $L_n \geq \Delta_n$.

For the second part, define:

$$\Delta_\ell \triangleq \inf \left\{\Delta : \Pr\{\limsup_{n\to\infty} \bar{\Delta}_{n,\ell} \leq \Delta\} = 1\right\} \quad ,$$

where $\bar{\Delta}_{n,\ell}$ is the average of $\Delta_1 \ldots \Delta_n$ when the word length is ℓ. To bound this asymptotic redundancy, we recall that there exists some $L_M < \infty$ s.t. $\sup\{\Delta_n\} \leq \sup\{L_n\} \triangleq L_M$, to assert:

$$\bar{\Delta}_{n,\ell} \leq \epsilon(\delta, \ell) + \frac{n_e(n, \delta, \ell)}{n} L_M \quad ,$$

where $n_e(n, \delta, \ell)$ is the count of iterations in which $\Delta_{k,\ell} > \epsilon(\delta, \ell)$ in the period $k = 1, 2 \ldots n$. So we see that:

$$\Delta_\ell \leq \inf \left\{\Delta : \Pr\{\limsup_{n\to\infty} \{\frac{n_e(n, \delta, \ell)}{n}\} \leq K(\Delta, \delta)\} = 1\right\} \quad ,$$

where $K(\Delta, \delta) \triangleq \frac{\Delta - \epsilon(\delta)}{L_M}$. By the Borell-Cantelli Lemma [11], the limit will hold with probability one as required, if:

$$\sum_{n=1}^{\infty} \Pr\{\frac{n_e(n, \delta, \ell)}{n} > K(\Delta, \delta)\} < \infty \quad . \tag{54}$$

Now, we define a new process, which is i.i.d., taking the value L_M w.p. $n_0(\delta)P_e(\delta)$, and the value $\epsilon(\delta)$ otherwise. Let $\tilde{n}_e(n,\delta,\ell)$ be the number of i.i.d. samples taking the high value (L_M) out of the first n samples, then by (53) we have that:

$$\Pr\{n_e(n,\delta,\ell) > nK(\Delta,\delta)\} \leq \Pr\{\tilde{n}_e(n,\delta,\ell) > nK(\Delta,\delta) - n_0(\delta)\} \quad .$$

But, by the central limit theorem, $\tilde{n}_e$ is asymptotically Gaussian, thus the probability drops exponentially with n and the sum is finite as required by (54). □

B Proofs for Markov Codebooks

B-1 Equivalence of Definitions for $R_k(P,Q,d)$

$R_k(P,Q,d)$ is defined in [22] as follows: Let $Y \sim Q$ be a semi-infinite stationary database sequence, defined for negative indices. Let $X \sim P$ be a semi-infinite stationary source sequence, defined for non-negative indices. Then:

$$\begin{aligned} R_k^{YK}(P,Q,d) \triangleq \inf_{\mathbf{U}_0^k}\Big\{ & \quad I(X;U_k|\mathbf{U}_0^{k-1}) \\ & + \quad D\Big(Q'(U_k|\mathbf{U}_0^{k-1})\|Q(Y_{-1}|\mathbf{Y}_{-k-1}^{-2})\Big)\Big\} \quad , \qquad (55) \end{aligned}$$

where conditional divergence is, as defined in [6]:

$$\begin{aligned} D\Big(P(X|Y)\|Q(X|Y)\Big) & \triangleq D\Big(P(X|Y)\|Q(X|Y)|P(X)\Big) \\ & \triangleq \sum_y \Pr(Y=y)D\Big(P(X|Y=y)\|Q(X|Y=y)\Big) \quad , (56) \end{aligned}$$

and the infimum is taken over all random vectors $\mathbf{U}_0^k$ jointly distributed with $\mathbf{X}$ s.t.:

1. stationarity: $(\mathbf{X},\mathbf{U}_0^{k-1})$ and $(\mathbf{X}_2^{\infty},\mathbf{U}_1^k)$ have the same joint distribution.
2. Distortion: (X_0,U_0) satisfy the distortion constraint.

Lemma 8 *$R_k^{YK}(P,Q,d) = R_k(P,Q,d)$ defined by (37), and furthermore, if exists a distribution $W(\mathbf{U}_1^{k+1}|\mathbf{X})$ minimizing (55), then it induces W_k minimizing (37).*

Proof: Consider the argument of the infimum in (55):

$$\begin{aligned} & I(\mathbf{X};U_k|\mathbf{U}_0^{k-1}) + D\Big(Q'(U_k|\mathbf{U}_0^{k-1})\|Q(Y_{-1}|\mathbf{Y}_{-k-1}^{-2})\Big) \\ \geq\ & I(X_k;U_k|\mathbf{U}_0^{k-1},\mathbf{X}_0^{k-1}) + D\Big(Q'(U_k|\mathbf{U}_0^{k-1})\|Q(Y_{-1}|\mathbf{Y}_{-k-1}^{-2})\Big) \\ =\ & I(\mathbf{X}_0^k;V_k|\mathbf{V}_0^{k-1}) + D\Big(Q'(V_k|\mathbf{U}_0^{k-1})\|Q(Y_{-1}|\mathbf{Y}_{-k-1}^{-2})\Big) \quad , \end{aligned}$$

where $\mathbf{V}_0^k$ is a random vector, such that

$$\Pr\{V_k|\mathbf{X}, \mathbf{V}_0^{k-1}\} = \Pr\{V_k|X_k, \mathbf{V}_0^{k-1}\} = E_{P(\mathbf{X}_{-\infty}^{k-1}, \mathbf{X}_{k+1}^{\infty})} \Pr\{U_k|\mathbf{X}, \mathbf{U}_0^{k-1}\} \quad .$$

Now, if $\mathbf{U}_0^k$ belongs to the set over the infimum in (55) is taken, then so is $\mathbf{V}_0^k$. The reason is, that averaging over conditions can not affect stationarity. The distortion condition is still satisfied, since the distortion only depends on $\Pr(U_0, X_0)$, and this marginal does not change by the averaging. Thus we can redefine $R_k(P, Q, d)$ as an infimum over all random vectors that satisfy the conditions 1 and 2, and also satisfy that V_k is independent of $\mathbf{X}$ for all times other than k, given V_0^{k-1}. For such $\mathbf{V}$, we have:

$$\begin{aligned}
& I(\mathbf{X}_0^k; V_k|\mathbf{V}_0^{k-1}) + D\Big(Q'(U_k|\mathbf{V}_0^{k-1})\|Q(Y_{-1}|\mathbf{Y}_{-k-1}^{-2})\Big) \\
\overset{(a)}{=} \; & D\Big(P(\mathbf{X}_0^k)W(V_k|X_k, \mathbf{Y}_0^{k-1})\|P(X_0^k)Q'(Y_k|\mathbf{Y}_0^{k-1})\Big) + D\Big(Q'(Y_k|\mathbf{Y}_0^{k-1})\|Q(Y_k|\mathbf{Y}_0^{k-1})\Big) \\
\overset{(b)}{=} \; & D\Big(P(\mathbf{X}_0^k)W(Y_k|X_k, \mathbf{Y}_0^{k-1})\|P(\mathbf{X}_0^k)Q(Y_k|\mathbf{Y}_0^{k-1})\Big) \\
\overset{(c)}{=} \; & D_{k|k-1}(P \circ W\|P \times Q)
\end{aligned} \tag{57}$$

Equality (a) involves a change of notation and substituting divergence for mutual information, in (b) we used the identity of Topsoe [21], and (c) follows from the definition (38). Remains to be seen that each vector $\mathbf{V}$ in the infimum range corresponds to a distribution W in the infimum range of (37). But this follows, since the distortion condition is identical, and $P \circ W$ is stationary since it's defined by stationary P and W. □

B-2 Markov vs. Piecewise-I.I.D. Rate (Proof of Lemma 2)

We will first show, that $R_k(P, Q, d) \leq \tilde{R}_k(P, Q, d)$ for all Q. By the chain rule for divergences, we have that:

$$D(P_k \circ W_k\|P_k \times Q_k) = D_{k|k-1}(P_k \circ W_k\|P_k \times Q_k) + D(P_{k-1} \circ W_k\|P_{k-1} \times Q_k) \quad .$$

Since this holds for all W, and the set over which infima are taken depends on the zero-order marginals only, we conclude that:

$$\begin{aligned}
(k+1)\tilde{R}_{k+1}(P, Q, D) &= \inf_{W_k} D(P_k \circ W_k\|P_k \times Q_k) \\
&= \inf_{W_k} \{D_{k|k-1}(P_k \circ W_k\|P_k \times Q_k) + D(P_{k-1} \circ W_k\|P_{k-1} \times Q_k)\} \\
&\geq \inf_{W_k} D_{k|k-1}(P_k \circ W_k\|P_k \times Q_k) + \inf_{W_k} D(P_{k-1} \circ W_k\|P_{k-1} \times Q_k) \\
&= k\tilde{R}_k(P, Q, D) + R_k(P, Q, d) \quad .
\end{aligned} \tag{58}$$

Reordering, we have that:

$$R_k(P, Q, d) \leq \tilde{R}_k(P, Q, d) - k\Big[\tilde{R}_k(P, Q, d) - \tilde{R}_{k+1}(P, Q, d)\Big] \leq \tilde{R}_k(P, Q, d) \quad .$$

Since this holds for all Q, then it also holds for the infima and Lemma 2 follows.

B-3 Proof Outline of Convergence for Markov Reproduction (Theorem 3)

Proof of Theorem 3 is based upon the concept of alternating minimization, similar to the proof of Theorem 1. However, in this case we are not able to present the problem as minimization of divergence between convex sets, thus we resort to a more generic result of [8]. To this end, we will define the following sets:

$$\begin{aligned} \mathcal{B} &= \{P(X_0^k)W(Y_0^k|X_0^k) : W \text{ stationary}, \rho(P,W) \leq d\} \\ \mathcal{A} &= \{P(X_0^k)Q_\theta(Y_k|Y_0^{k-1}) : \theta \in \Theta\} \quad . \end{aligned} \tag{59}$$

Also, define the conditional divergence as the "distance" between points in the sets:

$$d(B,A) = D_{k|k-1}(P \circ W \| P \times Q) \quad .$$

We can now redefine $R_k(P,Q,d)$ as:

$$R_k(P,Q,d) \triangleq d(\mathcal{B},\mathcal{A}) \triangleq \inf_{B\in\mathcal{B}} \inf_{A\in\mathcal{A}} d(B,A) \quad . \tag{60}$$

For the proof, we also need to define a "distance" between two points in $\mathcal{B}$:

$$\delta(B,B') = \sum_{\mathbf{x}_0^k} P(\mathbf{x}_0^k) \sum_{\mathbf{y}_0^k} W(\mathbf{y}_0^k|\mathbf{x}_0^k) \log \frac{W(y_{k+1}|\mathbf{x}, y_1^k)}{W'(y_{k+1}|\mathbf{x}, y_1^k)}$$

Next we observe, that the set B is convex, and if the set of conditional distributions $\mathcal{Q}_\Theta$ is convex then the set A is convex as well. This allows to prove two inequalities:

Lemma 9 (3-points property) *For convex $\mathcal{Q}_\Theta$, if $B' = \arg\min_{B\in\mathcal{B}} d(B,A')$ then*

$$\delta(B,B') + d(B',A') \leq d(B,A')$$

Lemma 10 (4-points property) *If $A' = \arg\min_{A\in\mathcal{A}} d(B',A)$ and $\mathcal{Q}$ is convex, then*

$$d(B,A') \leq \delta(B,B') + d(B,A)$$

Proof of these Lemmas is very similar to the proof of [8, Lemmas 2 and 3] for the case of divergence between convex sets. This assures, by [8, Theorem 2], that alternating minimization between the sets $\mathcal{B}$ and $\mathcal{A}$ as defined above does converge to the global optimum whenever $\mathcal{Q}_\Theta$ is convex. Remains to be shown, that the recursion (42) materializes this alternating minimization, and that follows from conditional divergence version of Lemma 1.

C Redundancy of R(P,Q,d) for Gaussian Codebooks under Mean Square Distortion

In this appendix, we show that if the parametric family Q_Θ is Gaussian (with any memory model), then the system can not gain from any non-Gaussianity of the source, i.e., $R(P, \Theta, d)$ is the same for all sources that have the same first two moments.

Lemma 11 *Let P be some k-dimensional distribution with expectation vector η and covariance matrix S. Let P_G be the Gaussian distribution with same η and S. Then, for a Gaussian distribution Q and mean square distortion,*

$$\tilde{R}_k(P, Q, d) = \tilde{R}_k(P_G, Q, d)$$

where $\tilde{R}_k(P, Q, d)$ was defined in (35).

Proof: For any k-dimensional transition distribution W_k. Let $W_{G,k}$ denote the Gaussian transition distribution such that $P \circ W_{G,k}$ has the same expectations and joint covariance as $P \circ W$. Now, for Gaussian Q_k:

$$\begin{aligned}
& D(P_k \circ W_k \| P_k \times Q_k) \\
= \;& \sum_{\mathbf{x},\mathbf{y}} P(\mathbf{x}) W(\mathbf{y}|\mathbf{x}) \log \frac{W(\mathbf{y}|\mathbf{x})}{Q(\mathbf{y})} \\
= \;& \sum_{\mathbf{x},\mathbf{y}} P(\mathbf{x}) W(\mathbf{y}|\mathbf{x}) [\log \frac{W(\mathbf{y}|\mathbf{x})}{W_G(\mathbf{y}|\mathbf{x})} + \log \frac{W_G(\mathbf{y}|\mathbf{x})}{Q(\mathbf{y})}] \\
\overset{(a)}{=} \;& \sum_{\mathbf{x},\mathbf{y}} P(\mathbf{x}) W(\mathbf{y}|\mathbf{x}) \log \frac{W(\mathbf{y}|\mathbf{x})}{W_G(\mathbf{y}|\mathbf{x})} + \sum_{\mathbf{x},\mathbf{y}} P_G(\mathbf{x}) W_G(\mathbf{y}|\mathbf{x}) \log \frac{W_G(\mathbf{y}|\mathbf{x})}{Q(\mathbf{y})} \\
= \;& D(W_k \| W_{G,k} | P_k) + D(P_{G,k} \circ W_{G,k} \| P_{G,k} \times Q_k) \\
\geq \;& D(P_{G,k} \circ W_{G,k} \| P_{G,k} \times Q_k) \quad ,
\end{aligned}$$

with equality if and only if $W_k = W_{G,k}$. The equality (a) follows because $\log \frac{W_G(\mathbf{y}|\mathbf{x})}{Q(\mathbf{y})}$, the logarithm of the ratio of Gaussian distributions, is a quadratic function of $(\mathbf{x}, \mathbf{y})$ and since $P \circ W$ and $P_G \circ W_G$ have the same first and second order moments. Recall the definition of $\tilde{R}_k(P, Q, d)$ in (35). For any distribution W_k satisfying the mean square distortion constraint, also $W_{G,k}$ satisfies the same constraint, thus it is sufficient to take the infimum over Gaussian transition distributions only. But for a Gaussian $W_{G,k}$ we saw above that $D(P_k \circ W_{G,k} \| P_k \times Q_k) = D(P_{G,k} \circ W_{G,k} \| P_{G,k} \times Q_k)$, thus the infimum is equal as well □

A similar result can be shown for the Markov codebooks of Subsection IV-2.

Recalling the definitions is Subsection IV-1, it follows that for a stationary source P and a stationary Gaussian Q,

$$R(P, Q, d) = R(P_G, Q, d) \quad ,$$

and consequently for a Gaussian class of reproductions Θ,

$$R(P, \Theta, d) = R(P_G, \Theta, d) \quad .$$

In the special case of the class of *all* Gaussian reproductions, we have that

$$R(P, \Theta, d) = R(P_G, d) \quad .$$

Recalling Figure 4, the set $\tilde{\mathcal{A}}$ corresponds to all stationary reproductions Q, while the set $\mathcal{A}$ corresponds to Gaussian reproductions only. Note, that the reproduction $Q^*(P, \theta^*, d)$ is the output induced by a non-Gaussian input P and a Gaussian transition distribution W_G, thus it is not Gaussian in general.

References

[1] T. Berger. *Rate Distortion Theory: A Mathematical Basis for Data Compression.* Prentice-Hall, Englewood Cliffs, NJ, 1971.

[2] R. E. Blahut. Computation of channel capacity and rate-distortion functions. *IEEE Trans. Information Theory*, IT-18:460–473, 1972.

[3] P. Boukris. An upper bound on the speed of convergence of the Blahut algorithm for computing rate-distortion functions. *IEEE Trans. Information Theory*, pages 708–709, Sept. 1973.

[4] J.H. Chen. High-quality 16 kbps speech coding with a one-way delay less than 2 ms. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Singnal Processing, Albuquerque, N.M.*, volume 1, pages 453–456, 1990.

[5] Z. Chi. The First-Order Asymptotics of Waiting Times with Distortion Between Stationary Processes. *IEEE Trans. Information Theory*, IT-47:338–347, May 2001.

[6] T. M. Cover and J. A. Thomas. *Elements of Information Theory.* Wiley, New York, 1991.

[7] I. Csiszár. I-divergence of probability distributions and minimization problems. *Ann. Probab.*, 3:146–158, 1975.

[8] I. Csiszár and G. Tusnády. Information geometry and alternating minimization procedures. *Statistics and decisions*, Supplement issue No. 1:205–237, 1984.

[9] A. Dembo and I. Kontoyiannis. Source coding, large deviations and approximate pattern matching. *IEEE Trans. Information Theory*, IT-48:1590–1615, June 2002.

[10] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society*, Ser. 39:1–38, 1977.

[11] R. Durrett. *Probability Theory and Examples.* Wadsworth and Brooks, Pacific Grove, CA, 1989.

[12] P. Elias. Universal codeword sets and representation of the integers. *IEEE Trans. Information Theory*, IT-26:194–203, 1975.

[13] G.D.Gibson, T.Berger, T.Lookabaugh, D.Lindbergh, and R.L.Baker. *Digital Compression for Multimedia: Principles and Standards'.* Morgan Kaufmann Pub., San Fansisco, 1998.

[14] A. Gersho and R. M. Gray. *Vector Quantization and Signal Compression.* Kluwer Academic Pub., Boston, 1992.

[15] H. Hartley. Maximum likelihood estimation from incomplete data. *Biometrics*, 14:174–194, 1958.

[16] S. Haykin. *Adaptive Filter Theory.* Prantice Hall, 1986.

[17] N. S. Jayant and P. Noll. *Digital Coding of Waveform.* Prentice-Hall, Englewood Cliffs, NJ, 1984.

[18] Y. Kochman and R. Zamir. Computation of the Rate-Distortion Function Relative to a Parametric Class of Distributions. In *Proceedings of the 41st Annual Allerton Conference on Communication, Control and Computing*, pages 211–220, 2003.

[19] I. Kontoyiannis and R. Zamir. Mismatched codebooks and the role of entropy-coding in lossy data compression. *IEEE Trans. Information Theory*, submitted.

[20] D.J. Sakrison. The rate of a class of random processes. *IEEE Trans. Information Theory*, IT-16:10–16, Jan, 1970.

[21] F. Topsoe. An information theoretical identity and a problem involving capacity. *Studia Sco. Math. Hungar.*, 2:291–292, 1967.

[22] E.H. Yang and J. Kieffer. On the performance of data compression algorithms based upon string matching. *IEEE Trans. Information Theory*, IT-44:47–65, Jan. 1998.

[23] R. Zamir and K. Rose. Natural type selection in addaptive lossy compression. *IEEE Trans. Information Theory*, 47:99–111, Jan. 2001.

[24] Z. Zhang and V. Wei. An on-line universal lossy data compression algorithm via continuous codebook refinement-part I: Basic results. *IEEE Trans. Information Theory*, IT-42:803–821, May 1996.

[25] J. Ziv. Coding of sources with unknown statistics - part II: Distortion relative to a fidelity criterion. *IEEE Trans. Information Theory*, IT-18:389–394, 1972.

[26] J. Ziv and A. Lempel. Compression of individual sequences via variable rate coding. *IEEE Trans. Information Theory*, IT-24:530–536, Sept. 1978.