Computing the Channel Capacity and Rate-distortion Function with Two-sided State Information

Samuel Cheng, Vladimir Stanković*, Zixiang Xiong

The authors are with the Department of Electrical Engineering, Texas A &M University, College Station, TX, USA. E-mail: phsamuel@ee.tamu.edu, stankovi@ee.tamu.edu, zx@ee.tamu.edu.

Abstract

We present iterative algorithms that numerically solve optimization problems of computing the capacity-power and rate-distortion functions for coding with two-sided state information. Numerical examples are provided to demonstrate efficiency of our algorithms.

Key words- Blahut-Arimoto algorithm, coding with side information, Gel'fand-Pinsker problem, Wyner-Ziv problem

I. Introduction

Coding with side information has gained increased research interest recently due to its great practical potentials. For example, source coding with side information at the decoder (a.k.a. Wyner-Ziv coding [1]) is recognized as an important component in emerging wireless sensor networks; on the other hand, channel coding with side information at the encoder (a.k.a. Gel'fand-Pinsker coding [2]) can be used to model the digital watermarking problem [3] and also applies to broadcast channel coding [4]. However, very often, it is necessary to use a more general setup with *two-sided state information* where both the encoder and the decoder have the access to (possibly different) side information. The capacity-power and the rate-distortion functions in this case are given by [5]:

$$C(P) = \max_{q'(x|u,s_1)q(u|s_1):E[\mathfrak{p}(S_1,S_2,X)]\le P} I(U;Y,S_2) - I(U;S_1) \tag{1}$$

and

$$R(D) = \min_{q(u|s_1,x)q'(\hat{x}|s_2,u):E[\mathfrak{d}(X,\hat{X})]\le D} I(U;X,S_1) - I(U;S_2), \tag{2}$$

respectively, where i.i.d. random variables X and Y are the channel input and output in the channel coding problem, X and $\hat{X}$ are the source input and the reconstructed output in the source coding problem, S_1 and S_2 are side information at the encoders and the decoders, respectively, and U is an auxiliary random variable. P and D are the power and distortion constraints for the respective channel coding and source coding problems with $\mathfrak{p}(\cdot,\cdot,\cdot)$ and $\mathfrak{d}(\cdot,\cdot)$ being the power and distortion measures. The expressions in both (1) and (2) are optimized over valid conditional probability mass functions (PMFs) $q(\cdot|\cdot)$ and $q'(\cdot|\cdot,\cdot)$.

Calculations of capacity-power and rate-distortion functions are difficult optimization problems. For conventional source and channel coding, Blahut-Arimoto algorithms [6], [7] provide efficient numerical solutions for memoryless channels and general i.i.d. sources with arbitrary power and distortion measures. These optimization techniques were later generalized in [8]. Extensions to channels and sources with memory were given in [9], [10].

However, when side information is present at the encoder and/or the decoder, calculation of channel capacity and rate-distortion function becomes more difficult. Recently, Blahut-Arimoto algorithms were generalized to the Gel'fand-Pinsker problem in [11], [12] and the Wyner-Ziv problem in [11]. In this paper we further extend the algorithms of [11], [12] to the general setup with two-sided state information; that is, we provide computation techniques for solving (1) and (2) numerically using iterative methods. Thus, our proposed methods can be regarded as generalizations of divide-and-conquer algorithms of Blahut and Arimoto to the most general case when side information maybe present at the encoder, at the decoder, at neither the encoder nor the decoder, or at both the encoder and the decoder (note that, in this case, side information at the encoder may differ from that at the decoder). Therefore, our unified framework subsumes the solutions of [11], [12] for the Gel'fand-Pinsker problem (channel coding with side information present at the encoder only), the solution of [11] for the Wyner-Ziv problem (source coding with side information present at the decoder only), and the solutions of Blahut and Arimoto [6], [7] (source and channel coding without side information); it also includes cases of source coding (channel coding) with side information present at the encoder (decoder) only and at both the encoder and the decoder which are not considered previously in [6], [7], [11], [12]. However, we treat only point-to-point transmission; the works of [11], [13] which consider broadcast and multiple-access channels, respectively, can be regarded as generalizations to multiuser communication settings.

Our methods follow the main idea of Blahut-Arimoto technique of alternating minimization/maximization; that is, we divide the original optimization problem into simpler (convex/concave) optimization problems in which only a subset of variables are optimized while the rest is kept fixed; then, the solution to such a partial optimization problem is fed into another sub-problem (with different variables kept fixed) and a different subset of variables is optimized. The algorithms continue to iterate until all the variables are optimized.

In a recent related work, Chiang and Boyd [14] proposed a geometric programming method for computing lower bounds for rate-distortion function with two-sided state information by exploiting Lagrangian duality. Since the method is based on converting the original problem into a convex/concave optimization form, exploiting the geometric programming approach for computing the capacity with two-sided state information is a difficult open problem [14].

II. Algorithm Derivation

In this section, we provide the detailed derivation of our iterative algorithms. For easy exposition, we first consider computation of the unconstrained capacity for the channel coding problem with two-sided state information. Then, we describe the iterative algorithms for computing the capacity-power and rate-distortion functions. Since the derivations are similar, for the latter two cases we will skip proofs and only state the results.

Notation-wise, p, q, q', Q, and Q_0 are used to express PMFs. However, we reserve p for those PMFs fixed by the problem setup. Random variables are denoted by upper-case letters, e.g., X, and for their realizations we use low-case letters, e.g., x. Script letters, e.g., $\mathcal{X}$, are reserved for the alphabets of random variables. $||\cdot||$ denotes the cardinality of a set.

A. Channel Capacity Without Power Constraint

The unconstrained channel capacity with two-sided state information is given by

$$C = \max_{q'(x|u,s_1)q(u|s_1)} I(U;Y,S_2) - I(U;S_1).$$

Define a conditional PMF

$$Q_0(u|y,s_2) \triangleq \frac{\sum_{x,s_1} p(s_1,s_2)q(u|s_1)q'(x|u,s_1)p(y|x,s_1,s_2)}{\sum_{x,s_1,u} p(s_1,s_2)q(u|s_1)q'(x|u,s_1)p(y|x,s_1,s_2)}, \tag{3}$$

then by the definition of mutual information [15], we can write

$$C = \max_{q'(x|u,s_1)q(u|s_1)} \sum_{s_1,s_2,u,x,y} p(s_1,s_2)q(u|s_1)q'(x|u,s_1)p(y|x,s_1,s_2) \log \frac{Q_0(u|y,s_2)}{q(u|s_1)},$$

where $p(s_1,s_2)$ and $p(y|x,s_1,s_2)$ are PMFs given by the channel. For PMFs $q(u|s_1)$, $q'(x|u,s_1)$, and $Q(u|y,s_2)$, define the functional

$$F(q,q',Q) = \sum_{s_1,s_2,u,x,y} p(s_1,s_2)q(u|s_1)q'(x|u,s_1)p(y|x,s_1,s_2) \log \frac{Q(u|y,s_2)}{q(u|s_1)}; \tag{4}$$

then we have the following lemma.

Lemma 1:

$$C = \max_{q'(x|u,s_1)q(u|s_1)} \max_{Q(u|y,s_2)} F(q,q',Q). \tag{5}$$

Proof: Since obviously $C = \max_{q'(x|u,s_1)q(u|s_1)} F(q,q',Q_0)$, it suffices to show $F(q,q',Q_0) = \max_{Q(u|y,s_2)} F(q,q',Q)$, and this is true because for any Q,

$$\begin{aligned} F(q,q',Q) - F(q,q',Q_0) &= \sum_{s_1,s_2,u,x,y} p(s_1,s_2)q(u|s_1)q'(x|u,s_1)p(y|x,s_1,s_2) \log \frac{Q(u|y,s_2)}{Q_0(u|y,s_2)} \\ &\overset{(a)}{\leq} \sum_{s_1,s_2,u,x,y} p(s_1,s_2)q(u|s_1)q'(x|u,s_1)p(y|x,s_1,s_2) \left(\frac{Q(u|y,s_2)}{Q_0(u|y,s_2)} - 1 \right) = 0, \end{aligned}$$

where the equality in (a) is achieved if $Q = Q_0$. ■

Lemma 1 is the key step in constructing our algorithm. By introducing the functional $F(\cdot,\cdot,\cdot)$, we can find the capacity by optimizing PMFs q, q', and Q one at a time alternatively. Note that from Lemma 1, the optimal PMF Q is always equal to the PMF Q_0. Now we have

Lemma 2 (Optimization of q) For fixed PMFs q' and Q, $F(q, q', Q)$ is maximized by

$$q^*(u|s_1) = \frac{\exp \sum_{s_2,x,y} p(s_2|s_1)q'(x|u,s_1)p(y|x,s_1,s_2)\log Q(u|y,s_2)}{\sum_u \exp \sum_{s_2,x,y} p(s_2|s_1)q'(x|u,s_1)p(y|x,s_1,s_2)\log Q(u|y,s_2)}, \tag{6}$$

and the achieved maximum is given by

$$F(q^*, q', Q) = \sum_{s_1} p(s_1)\max_u \sum_{s_2,x,y} p(s_2|s_1)q'(x|u,s_1)p(y|x,s_1,s_2)\log\frac{Q(u|y,s_2)}{q^*(u|s_1)}, \tag{7}$$

where $\exp(\cdot)$ denotes the exponential function.

Proof: For fixed q' and Q, $F(q, q', Q)$ is maximized by $q^*(u|s_1)$ if and only if the following Kuhn-Tucker conditions are satisfied:

$$\left.\frac{\partial F}{\partial q}\right|_{q^*} = \gamma_{s_1}, \qquad \text{if } q^*(u|s_1) > 0, \tag{8}$$

and

$$\left.\frac{\partial F}{\partial q}\right|_{q^*} \le \gamma_{s_1}, \qquad \text{if } q^*(u|s_1) = 0. \tag{9}$$

Since $\frac{\partial F}{\partial q} = \sum_{s_2,x,y} p(s_1,s_2)q'(x|u,s_1)p(y|x,s_1,s_2)\left(\log\frac{Q(u|y,s_2)}{q(u|s_1)} - 1\right)$, the first condition (8) becomes

$$\sum_{s_2,x,y} p(s_1,s_2)q'(x|u,s_1)p(y|x,s_1,s_2)\log\frac{Q(u|y,s_2)}{q^*(u|s_1)} = \tilde{\gamma}_{s_1}, \tag{10}$$

where $\tilde{\gamma}_{s_1}$ depends on s_1 only. Then, (6) follows from (10) after some mathematical manipulation. For the second part, note that

$$\begin{aligned} F(q, q', Q) &= \sum_{s_1,u} p(s_1)q(u|s_1)\sum_{s_2,x,y} p(s_2|s_1)q'(x|u,s_1)p(y|x,s_1,s_2)\log\frac{Q(u|y,s_2)}{q(u|s_1)} \\ &\le \sum_{s_1} p(s_1)\max_u \sum_{s_2,x,y} p(s_2|s_1)q'(x|u,s_1)p(y|x,s_1,s_2)\log\frac{Q(u|y,s_2)}{q(u|s_1)}, \end{aligned}$$

where equality holds when the Kuhn-Tucker conditions, and hence (10), are satisfied; that is, when q is equal to the optimal q^*. ■

The results of Lemmas 1 and 2 are combined and a partial optimality condition is summarized by the following corollary.

Corollary 1 (Optimality condition A_F) For fixed q', $F(q, q', Q)$ is maximized by q^* and Q^*. That is

$$F(q^*, q', Q^*) = \sum_{s_1} p(s_1) \max_u \sum_{s_2,x,y} p(s_2|s_1) q'(x|u, s_1) p(y|x, s_1, s_2) \log \frac{Q^*(u|y, s_2)}{q^*(u|s_1)} \triangleq A_F. \tag{11}$$

Now to optimize q', we need the following lemma.

Lemma 3 (Optimization of q') For fixed q and Q, $F(q, q', Q)$ is maximized by q'^* given by

$$q'^*(x|u, s_1) = \begin{cases} 1, & \text{if } f(x, u, s_1) = \max_{x'} f(x', u, s_1), \\ 0, & \text{otherwise} \end{cases} \tag{12}$$

and the maximum is equal to

$$F(q, q'^*, Q) = \sum_{s_1,u} p(s_1) q(u|s_1) \max_x f(x, u, s_1) \triangleq B_F, \tag{13}$$

where $f(x, u, s_1) \triangleq \sum_{s_2,y} p(s_2|s_1) p(y|x, s_1, s_2) \log \frac{Q(u|y,s_2)}{q(u|s_1)}$.

Proof: For fixed q and Q, note that

$$\begin{aligned} F(q, q', Q) &= \sum_{s_1,u,x} p(s_1) q(u|s_1) q'(x|u, s_1) \sum_{s_2,y} p(s_2|s_1) p(y|x, s_1, s_2) \log \frac{Q(u|y, s_2)}{q(u|s_1)} \\ &\overset{(a)}{\leq} \sum_{s_1,u} p(s_1) q(u|s_1) \max_x \underbrace{\sum_{s_2,y} p(s_2|s_1) p(y|x, s_1, s_2) \log \frac{Q(u|y, s_2)}{q(u|s_1)}}_{f(x,u,s_1)} \triangleq B_F, \end{aligned} \tag{14}$$

thus, $F(q, q', Q)$ reaches the maximum B_F, if the equality holds in (a). ■

Note that the optimum q' specified by (12) is not unique. For given PMFs q and Q, let $S_{q'}(Q, q)$ be the set of all PMFs q' that satisfy (12); then $||S_{q'}(Q, q)|| \leq ||\mathcal{X}||^{||\mathcal{U}\times\mathcal{S}_1||}$. Combining (13) and Corollary 1, we finally have

Corollary 2 (Optimality condition C_F)

$$F(q, q', Q) \leq \sum_{s_1} p(s_1) \max_u \max_x \sum_{s_2,y} p(s_2|s_1) p(y|x, s_1, s_2) \log \frac{Q(u|y, s_2)}{q(u|s_1)} \triangleq C_F,$$

and equality holds if q' maximizes $F(q, q', Q)$ with the other two variables fixed (i.e., if $q' = q'^*$), and q and Q maximize $F(q, q', Q)$ with q' fixed (i.e., if $q = q^*$, $Q = Q^*$).

Note that $F(q, q', Q) = C_F$ does not necessarily guarantee $F(q, q', Q) = C$ since there could be more than one optimal q'. However, if $F(q, q', Q) = C_F$ for all $q' \in S_{q'}(Q, q)$, then $F(q, q', Q) = C$.

The overall algorithm for computing C in (1) is summarized in Fig. 1. We initialize $q(u|s_1)$ to $\frac{1}{||\mathcal{U}||}$ and $q'(x|u, s_1)$ to random Kronecker delta function (KDF) of x for fixed u and s_1. We first optimize q and Q for fixed q'; F will then be compared with A_F to determined if q and Q are optimum. If so (i.e., if F

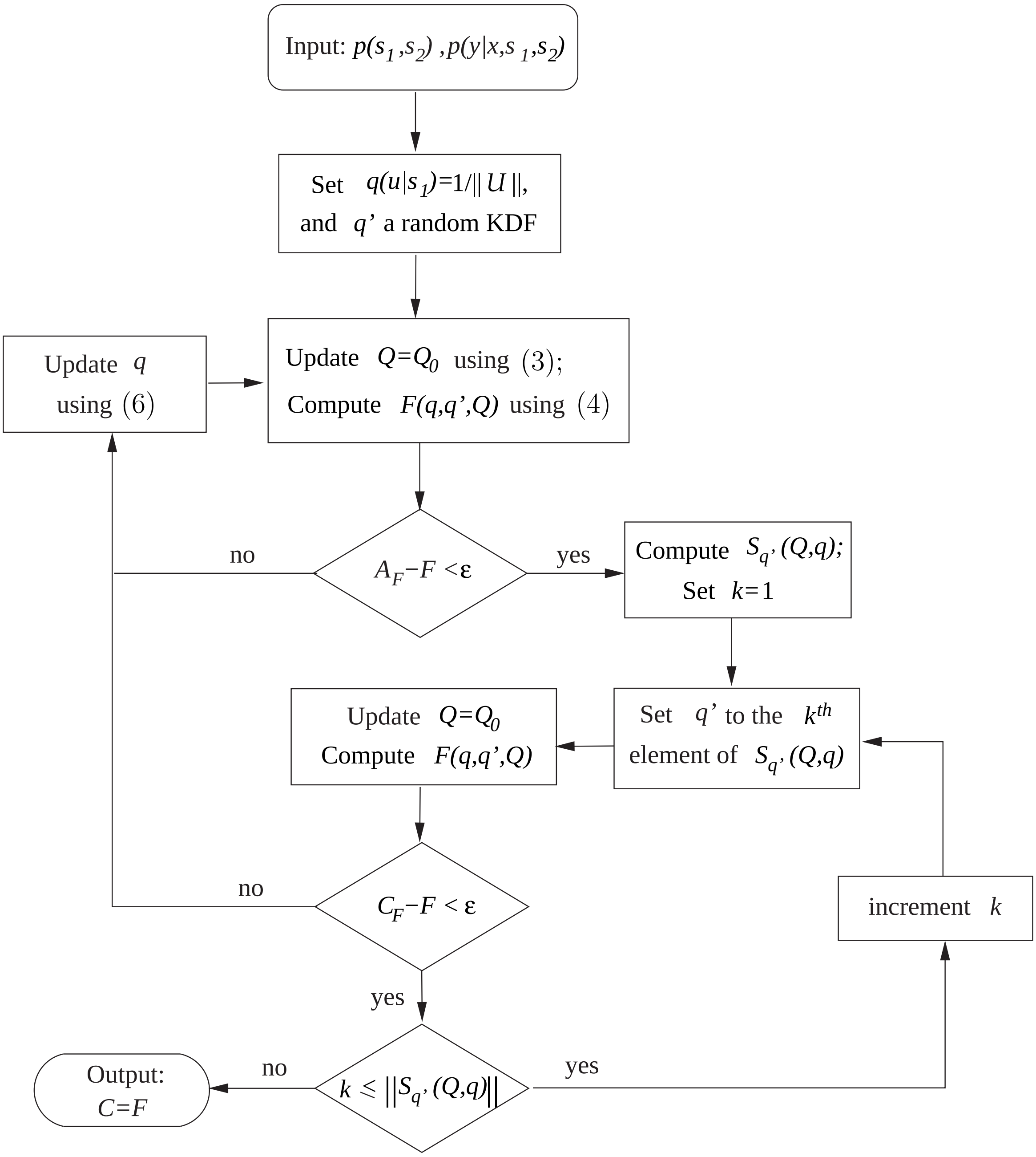

Fig. 1. Algorithm for computing capacity C of a channel with two-sided state information.

is within a threshold ϵ away from A_F), q' will be set to a previously unconsidered element of $S_{q'}(q, Q)$. The procedure repeats until all elements in $S_{q'}(Q, q)$ are exhausted.

B. Capacity-Power Function

In some cases, it is necessary to constrain the transmission power in a communication system. In conventional communication system, the transmission power is a function of channel input X only. However, to allow modelling different problems such as, e.g., digital watermarking, a more general power function $\mathfrak{p}(S_1, S_2, X)$ that also depends on S_1 and S_2 is needed. Using the standard Lagrange multiplier technique, we convert (1) into

$$C(P) = \max_{q(u|s_1)q'(x|s_1,u)} I(U; Y, S_2) - I(U; S_1) - \mu(E[\mathfrak{p}(S_1, S_2, X)] - P), \tag{15}$$

where μ, the Lagrange multiplier, rather than the power constraint P, is the actual input of computation. Both P and $C(P)$ are generated at the point where $C(P)$ curve has slope μ. After optimization, P can be computed as $P = \sum_{s_1,s_2,x,u} p(s_1, s_2)q^*(u|s_1)q'^*(x|s_1, u)\mathfrak{p}(s_1, s_2, x)$, where $q^*(u|s_1)$ and $q'^*(x|s_1, u)$ are the conditional PMFs that maximize (15). We can rewrite (15) as

$$\begin{aligned} C(P) = \max_{q'(x|u,s_1)q(u|s_1)} & \sum_{s_1,s_2,u,x,y} p(s_1, s_2)q(u|s_1)q'(x|u, s_1)p(y|x, s_1, s_2) \log \frac{Q_0(u|y, s_2)}{q(u|s_1)} \\ & - \mu \left(\sum_{s_1,s_2,x,u} p(s_1, s_2)q(u|s_1)q'(x|s_1, u)\mathfrak{p}(s_1, s_2, x) - P \right), \end{aligned}$$

where Q_0 is defined as in (3). Define the functional

$$\begin{aligned} F_c(q, q', Q) = & \sum_{s_1,s_2,u,x,y} p(s_1, s_2)q(u|s_1)q'(x|u, s_1)p(y|x, s_1, s_2) \log \frac{Q(u|y, s_2)}{q(u|s_1)} \\ & - \mu \sum_{s_1,s_2,x,u} p(s_1, s_2)q(u|s_1)q'(x|s_1, u)\mathfrak{p}(s_1, s_2, x), \end{aligned} \tag{16}$$

then we have the following lemma.

Lemma 4: $C(P) = \max_{q(u|s_1),q'(x|s_1,u)} \max_{Q(u|y,s_2)} F_c(q, q', Q) + \mu P$.

From Lemma 4, we can find $C(P)$ by maximizing F_c one variable at a time. To optimize q, we have the following lemma which extends Lemma 2 to the case with a power constraint.

Lemma 5 (Optimization of q) For fixed q' and Q, $F_c(q, q', Q)$ is maximized by

$$q^*(u|s_1, x) = \frac{\exp\left[\sum_{s_2,x,y} p(s_2|s_1)q'(x|s_1, u)p(y|x, s_1, s_2)[\log Q(u|y, s_2) - \mu\mathfrak{p}(x, s_1, s_2)]\right]}{\sum_u \exp\left[\sum_{s_2,x,y} p(s_2|s_1)q'(x|s_1, u)p(y|x, s_1, s_2)[\log Q(u|y, s_2) - \mu\mathfrak{p}(x, s_1, s_2)]\right]} \tag{17}$$

and the maximum is given by

$$F_c(q^*, q', Q) = \sum_{s_1} p(s_1) \max_u \sum_{s_2,x,y} \left[p(s_2|s_1) q'(x|u,s_1) p(y|x,s_1,s_2) \left(\log \frac{Q(u|y,s_2)}{q^*(u|s_1)} - \mu \mathfrak{p}(s_1,s_2,x) \right) \right].$$

The optimality conditions in Lemmas 4 and 5 are summarized by the following corollary.

Corollary 3 (Optimality condition A_{Fc}) For fixed q', $F_c(q, q', Q)$ is maximized by q^* and Q^*. That is

$$F_c(q^*, q', Q^*) = \sum_{s_1} p(s_1) \max_u \sum_{s_2,x,y} \left[p(s_2|s_1) q'(x|u,s_1) p(y|x,s_1,s_2) \left(\log \frac{Q^*(u|y,s_2)}{q^*(u|s_1)} - \mu \mathfrak{p}(s_1,s_2,x) \right) \right] \triangleq A_{Fc}.$$

To optimize q' for fixed q and Q, we have the following lemma.

Lemma 6 (Optimization of q') For fixed q and Q, the optimum q'^* satisfies

$$q'^*(x|u,s_1) = \begin{cases} 1, & \text{if } f_c(x,u,s_1) = \max_{x'} f_c(x',u,s_1), \\ 0, & \text{otherwise,} \end{cases} \tag{18}$$

and the achieved maximum is

$$F_c(q, q'^*, Q) = \sum_{s_1,u} p(s_1) q(u|s_1) \max_x f_c(x,u,s_1) \triangleq B_{Fc}, \tag{19}$$

where $f_c(x,u,s_1) \triangleq \sum_{s_2,y} \left[p(s_2|s_1) p(y|x,s_1,s_2) \left(\log \frac{Q(u|y,s_2)}{q(u|s_1)} - \mu \mathfrak{p}(s_1,s_2,x) \right) \right]$.

As in the unconstrained case, the optimum q' for fixed u and s_1 may not be unique. For given PMFs q and Q let $S_{q'}(Q,q)$ be the set of q''s that achieve the maximum in (18); then $||S_{q'}(Q,q)|| \leq ||\mathcal{X}||^{||\mathcal{U} \times \mathcal{S}_1||}$. Combining (19) and Corollary 3, we have the following corollary similar to Corollary 2.

Corollary 4 (Optimality condition C_{Fc})

$$F_c(q, q', Q) \leq \sum_{s_1} p(s_1) \max_u \max_x \sum_{s_2,y} \left[p(s_2|s_1) p(y|x,s_1,s_2) \left(\log \frac{Q(u|y,s_2)}{q(u|s_1)} - \mu \mathfrak{p}(s_1,s_2,x) \right) \right] \triangleq C_{Fc},$$

and equality holds if q' maximizes $F_c(q, q', Q)$ with the rest two variables fixed, and q and Q maximize $F_c(q, q', Q)$ with q' fixed.

As in the case without power constraint, $F_c(q, q', Q) = C_{Fc}$ might not imply $F_c(q, q', Q)$ to be the global optimum since there are more than one optimal q''s in general. However, if $F_c(q, q', Q) = C_{Fc}$ for all $q' \in S_{q'}(Q,q)$, then $C(P) = F_c(q, q', Q) + \mu P$.

C. Rate-Distortion Function

The iterative algorithm for computing the rate-distortion function with two-sided state information is similar to that for capacity-power computation. Using the standard Lagrange multiplier technique, we convert (2) into

$$R(D) = \min_{q(u|s_1,x),q'(\hat{x}|s_2,u)} I(U;X,S_1) - I(U;S_2) + \mu(E[\mathfrak{d}(X,\hat{X})] - D), \tag{20}$$

where μ, the Lagrange multiplier, rather than D, is the actual input of computation. Both D and $R(D)$ are generated at the point where the $R(D)$ curve has slope $-\mu$. After optimization, D can be computed as $D = \sum_{s_1,s_2,x,u,\hat{x}} q^*(u|s_1,x)q'^*(\hat{x}|s_2,u)\mathfrak{d}(x,\hat{x})$, where $q^*(u|s_1,x)$ and $q'^*(\hat{x}|s_2,u)$ are the optimum conditional PMFs, i.e., PMFs that minimize $R(D)$. Define $Q_0(u|s_2) \triangleq \frac{\sum_{s_1,x,\hat{x}} p(s_1,s_2,x)q(u|s_1,x)q'(\hat{x}|s_2,u)}{\sum_{s_1,x,\hat{x},u} p(s_1,s_2,x)q(u|s_1,x)q'(\hat{x}|s_2,u)}$, then we can rewrite (20) as

$$R(D) = \min_{q(u|s_1,x),q'(\hat{x}|s_2,u)} \sum_{s_1,s_2,x,u,\hat{x}} p(s_1,s_2,x)q(u|s_1,x)q'(\hat{x}|s_2,u)\log\frac{q(u|s_1,x)}{Q_0(u|s_2)}$$
$$+ \mu\left(\sum_{s_1,s_2,x,u,\hat{x}} p(s_1,s_2,x)q(u|s_1,x)q'(\hat{x}|s_2,u)\mathfrak{d}(x,\hat{x}) - D\right).$$

Define the functional

$$G(q,q',Q) = \sum_{s_1,s_2,x,u,\hat{x}} p(s_1,s_2,x)q(u|s_1,x)q'(\hat{x}|s_2,u)\log\frac{q(u|s_1,x)}{Q(u|s_2)}$$
$$+ \mu \sum_{s_1,s_2,x,u,\hat{x}} p(s_1,s_2,x)q(u|s_1,x)q'(\hat{x}|s_2,u)\mathfrak{d}(x,\hat{x}),$$

then we have the following lemma similar to Lemmas 1 and 4.

Lemma 7: $R(D) = \min_{q(u|s_1,x),q'(\hat{x}|s_2,u)} \min_{Q(u|s_2)} G(q,q',Q) - \mu D$.

Just as Lemmas 1 and 4, Lemma 7 is the key step of the rate-distortion computation algorithm. For a given distortion constraint D, we can now find the minimum rate R by optimizing variables q, q', and Q one at a time alternatively. From Lemma 7, it follows that the optimal value of Q is Q_0. Now to optimize q, we have the following lemma.

Lemma 8 (Optimization of q) For fixed q' and Q, $G(q,q',Q)$ is minimized by

$$q^*(u|s_1,x) = \frac{\exp\left[\sum_{s_2} p(s_2|s_1,x)\log Q(u|s_2) - \mu\sum_{s_2,\hat{x}} p(s_2|s_1,x)q'(\hat{x}|s_2,u)\mathfrak{d}(x,\hat{x})\right]}{\sum_u \exp\left[\sum_{s_2} p(s_2|s_1,x)\log Q(u|s_2) - \mu\sum_{s_2,\hat{x}} p(s_2|s_1,x)q'(\hat{x}|s_2,u)\mathfrak{d}(x,\hat{x})\right]} \tag{21}$$

and the minimum is given by

$$G(q^*,q',Q) = \sum_{s_1,x} p(s_1,x)\min_u \left[\sum_{s_2,\hat{x}} p(s_2|s_1,x)q'(\hat{x}|s_2,u)\log\frac{q^*(u|s_1,x)}{Q(u|s_2)}\right.$$
$$\left. + \mu\sum_{s_2,\hat{x}} p(s_2|s_1,x)q'(\hat{x}|s_2,u)\mathfrak{d}(x,\hat{x})\right]. \tag{22}$$

Optimality conditions in Lemmas 7 and 8 can be summarized by the following corollary.

Corollary 5 (Optimality condition A_G) For fixed q', $G(q, q', Q)$ is minimized by q^* and Q^*. That is

$$\begin{aligned} G(q^*, q', Q^*) = \sum_{s_1, x} p(s_1, x) \min_u \Bigg[& \sum_{s_2, \hat{x}} p(s_2|s_1, x) q'(\hat{x}|s_2, u) \log \frac{q^*(u|s_1, x)}{Q^*(u|s_2)} \\ & + \mu \sum_{s_2, \hat{x}} p(s_2|s_1, x) q'(\hat{x}|s_2, u) \mathfrak{d}(x, \hat{x}) \Bigg] \triangleq A_G. \end{aligned} \tag{23}$$

To optimize q' for fixed q and Q, we have the following lemma.

Lemma 9 (Optimization of q') For fixed q and Q, the optimum q'^* satisfies

$$q'^*(\hat{x}|u, s_2) = \begin{cases} 1, & \text{if } g(\hat{x}, u, s_2) = \min_{\hat{x}'} g(\hat{x}', u, s_2), \\ 0, & \text{otherwise,} \end{cases} \tag{24}$$

and the achieved minimum is

$$G(q, q'^*, Q) = \sum_{u, s_2} \min_{\hat{x}} g(\hat{x}, u, s_2) \triangleq B_G, \tag{25}$$

where $g(\hat{x}, u, s_2) \triangleq \sum_{s_1, x} p(s_1, s_2, x) q(u|s_1, x) \left(\log \frac{q(u|s_1,x)}{Q(u|s_2)} + \mu \mathfrak{d}(x, \hat{x}) \right)$.

Similarly to the previous cases, the optimum q' might not be unique. For given PMFs q and Q, let $S_{q'}(Q, q)$ be the set of q''s that achieve the minimum in (24); then $||S_{q'}(Q, q)|| \leq ||\hat{\mathcal{X}}||^{||\mathcal{U} \times \mathcal{S}_2||}$.

Since there is no simple way to combine (23) and (25), unlike in capacity computation, we need to verify both conditions, $G = A_G$ and $G = B_G$, for optimality. But, even when both conditions are satisfied, $G(q, q', Q)$ might not be the global optimum because for fixed q and Q, there might be more than one q'^*. However, if the above two conditions are satisfied for all $q' \in S_{q'}(Q, q)$, then $R(D) = G(q, q', Q) - \mu D$.

III. Proof of Convergence

We show in this section that the algorithms described in Section II converge to global optimums. Our proof is based on a result of Yeung in [16, Chapter 10], which shows that a two-step iterative maximization (minimization) algorithm converges to the global optimum if the optimization function is concave (convex). We will show in the following (Lemmas 10-12) that for fixed q', F and F_c are concave and G is convex. Therefore, for fixed q', q^*, and Q^*, all three algorithms converge to the corresponding global optimums. Once q and Q are optimized, q''s are updated by (12), (18), or (25). Since F and F_c strictly increase and G strictly decreases after the updates, and the number of possible optimal q''s ($q' \in S_{q'}(Q, q)$) is finite, F, F_c, and G will ultimately converge to the global optimums.

Lemma 10 (Concavity of F for fixed q') $F(q, q', Q)$ is concave over q and Q for fixed q'.

Proof: By the log-sum inequality, for an arbitrary positive $\gamma \leq 1$ and $\bar{\gamma} = 1 - \gamma$,

$$\begin{aligned}(\gamma q_1(u|s_1) + \bar{\gamma} q_2(u|s_1)) \log \frac{\gamma q_1(u|s_1) + \bar{\gamma} q_2(u|s_1)}{\gamma Q_1(u|y, s_2) + \bar{\gamma} Q_2(u|y, s_2)} \\ \leq \gamma q_1(u|s_1) \log \frac{q_1(u|s_1)}{Q_1(u|y, s_2)} + \bar{\gamma} q_2(u|s_1) \log \frac{q_2(u|s_1)}{Q_2(u|y, s_2)}.\end{aligned} \tag{26}$$

Multiplying both sides by $-p(s_1, s_2) q'(x|u, s_1) p(y|x, s_1, s_2)$ and summing over s_1, s_2, u, x, and y, we obtain

$$F(\gamma q_1 + \bar{\gamma} q_2, q', \gamma Q_1 + \bar{\gamma} Q_2) \geq \gamma F(q_1, q', Q_1) + \bar{\gamma} F(q_2, q', Q_2).$$

■

Lemma 11 (Concavity of F_c for fixed q') $F_c(q, q', Q)$ is concave over q and Q for fixed q'.

Proof: From Lemma 10, $F(q, q', Q)$ is concave. Since $F_c(q, q', Q) = F(q, q', Q) - \mu E[\mathfrak{p}(S_1, S_2, X)]$, and $\mu E[\mathfrak{p}(S_1, S_2, X)]$ is linear with respect to q and Q, $F_c(q, q', Q)$ is concave. ■

Lemma 12 (Convexity of G for fixed q') $G(q, q', Q)$ is convex over q and Q for fixed q'.

Proof: Using the log-sum inequality, we can show that $\sum_{s_1, s_2, x, u, \hat{x}} p(s_1, s_2, x) q(u|s_1, x) q'(\hat{x}|s_2, u) \log \frac{q(u|s_1, x)}{Q(u|s_2)}$ is convex over q and Q for fixed q'. Since $\mu \sum_{s_1, s_2, x, u, \hat{x}} p(s_1, s_2, x) q(u|s_1, x) q'(\hat{x}|s_2, u) \mathfrak{d}(x, \hat{x})$ is linear with respect to q and Q, the sum of the two expressions, i.e., $G(q, q', Q)$, is convex. ■

IV. Numerical Examples

In this section, we provide numerical examples for our iterative algorithms. Though the setups of these examples are rather simple, the results are highly non-trivial.

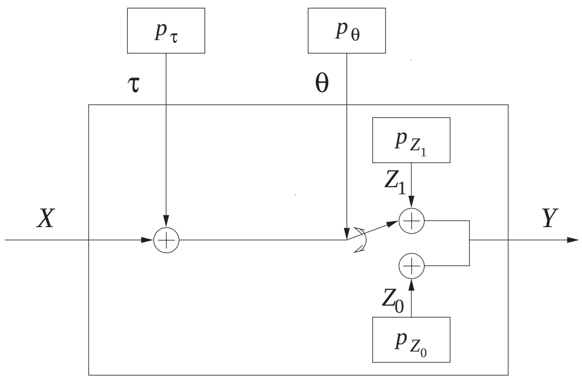

Fig. 2. Binary symmetric channel with two-sided channel state information θ and τ.

Example 1: Binary Symmetric Channel with Channel State Information. Consider a binary symmetric channel $Y = X \oplus \tau \oplus Z$ as shown in Fig. 2, where X is the channel input and τ and Z are the channel noises. The transition probability of τ is fixed to be p_τ, whereas the transition probability of Z can take two different values and is controlled by a binary random variable θ with $p(\theta = 1) = p_\theta$ as follows:

$$p_Z = \begin{cases} p_{Z_1}, & \text{if } \theta = 1, \\ p_{Z_0}, & \text{if } \theta = 0. \end{cases}$$

Consider θ and/or τ as channel state information that may be available to the encoder and decoder. Since each coder can have access either to both side information θ and τ, or only to θ, or only to τ, or to none of them, there are 16 different situations.

We use the algorithm described in Section II-A with $p_\tau = 0.5, p_{Z_1} = 0.001$, and $p_{Z_0} = 0.3$. Since $p_\tau = 0.5$, when τ is given to neither the encoder nor decoder, X and Y are effectively independent, and hence for all four possible cases (θ available or not at the encoder and/or decoder), the channel capacity is simply zero. This is verified in our result. More interestingly, the 16 cases can be grouped into three cases only as shown in Table I; the capacity for each case is shown in Fig. 3. Furthermore, when τ is available at at least one coder, we can reach the higher capacity C_2 only if θ is available at the decoder.

TABLE I

Channel capacities for different cases in Example 1; $\mathbb{S}_1$ and $\mathbb{S}_2$ are the sets of side information available at the encoder and decoder, respectively.

Capacity	Cases	
0	$\mathbb{S}_1 = \emptyset, \mathbb{S}_2 = \emptyset$;	$\mathbb{S}_1 = \{\theta\}, \mathbb{S}_2 = \emptyset$;
	$\mathbb{S}_1 = \emptyset, \mathbb{S}_2 = \{\theta\}$;	$\mathbb{S}_1 = \{\theta\}, \mathbb{S}_2 = \{\theta\}$
C_1	$\mathbb{S}_1 = \emptyset, \mathbb{S}_2 = \{\tau\}$;	$\mathbb{S}_1 = \{\tau\}, \mathbb{S}_2 = \emptyset$;
	$\mathbb{S}_1 = \{\theta\}, \mathbb{S}_2 = \{\tau\}$;	$\mathbb{S}_1 = \{\tau\}, \mathbb{S}_2 = \{\tau\}$;
	$\mathbb{S}_1 = \{\theta, \tau\}, \mathbb{S}_2 = \emptyset$;	$\mathbb{S}_1 = \{\theta, \tau\}, \mathbb{S}_2 = \{\tau\}$
C_2	$\mathbb{S}_1 = \emptyset, \mathbb{S}_2 = \{\theta, \tau\}$;	$\mathbb{S}_1 = \{\tau\}, \mathbb{S}_2 = \{\theta\}$;
	$\mathbb{S}_1 = \{\theta\}, \mathbb{S}_2 = \{\theta, \tau\}$;	$\mathbb{S}_1 = \{\tau\}, \mathbb{S}_2 = \{\theta, \tau\}$;
	$\mathbb{S}_1 = \{\theta, \tau\}, \mathbb{S}_2 = \{\theta\}$;	$\mathbb{S}_1 = \{\theta, \tau\}, \mathbb{S}_2 = \{\theta, \tau\}$

Example 2: Binary Symmetric Source with Side Information. Consider the source Y generated by passing an all-zero sequence X through the binary symmetric channel described in Example 1 (see Fig. 2), and assume the same numerical setting with $p_\tau = 0.5$, $p_{Z_1} = 0.01$, and $p_{Z_0} = 0.3$. We compute the rate-distortion functions for this source when $p_\theta = 0.5$. Like in the previous example, τ and/or θ may be provided to the source encoder and decoder as side information, and hence we have totally 16 different

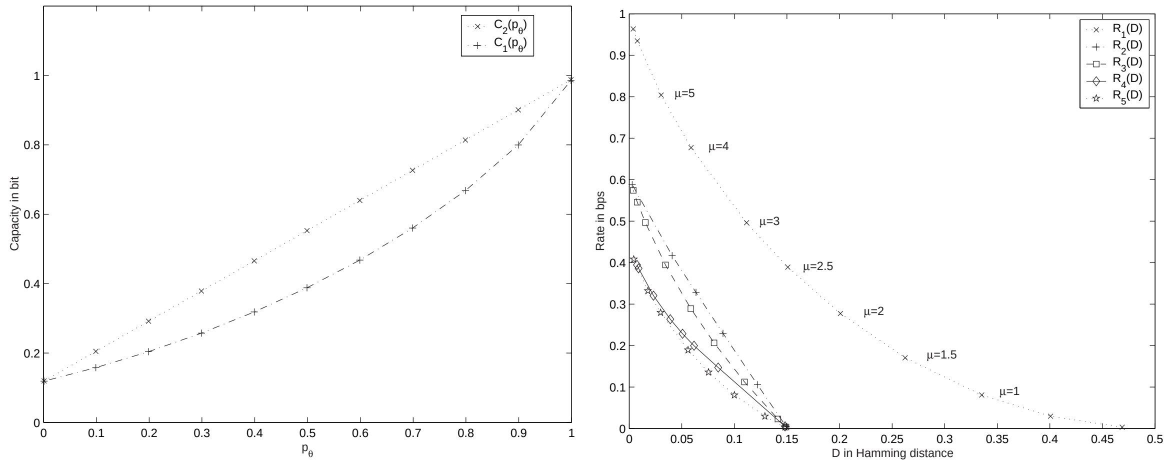

Fig. 3. Left figure: channel capacity C versus p_θ for two different cases in Example 1; right figure: rate-distortion functions for five different cases in Example 2.

cases. Interestingly, these 16 cases can be grouped into five cases only as shown in Table II. The reason for this is apparent. Indeed, for instance, if τ is given to neither the encoder nor the decoder, the source is just a binary symmetric source regardless of the availability of θ. Hence, the rate-distortion function for these cases should be the same as that for a binary symmetric source with no side information. Another interesting observation is that side information is not helpful if it is provided to the encoder alone; for example, the case $\mathbb{S}_1 = \emptyset, \mathbb{S}_2 = \emptyset$ and the case $\mathbb{S}_1 = \{\theta, \tau\}, \mathbb{S}_2 = \emptyset$ have the same rate-distortion function. This is consistent with the classic result of Berger [17].

References

[1] A. Wyner and J. Ziv, "The rate-distortion function for source coding with side information at the decoder," *IEEE Trans. Inform. Theory*, pp. 1–10, Jan. 1976.

[2] S. Gel'fand and M. Pinsker, "Coding for channel with random parameters," *Problems of Control and Information Theory*, no. 1, pp. 19–31, 1980.

[3] B. Chen and G. Wornell, "Quantization index modulation: a class of provably good methods for digital watermarking and information embedding," *IEEE Trans. Inform. Theory*, vol. 47, pp. 1423–1443, May 2001.

[4] G. Caire and S. Shamai, "On the achievable throughput of a multiantenna Gaussian broadcast channel," *IEEE Trans. Inform. Theory*, pp. 1691–1706, July 2003.

[5] T. M. Cover and M. Chiang, "Duality between channel capacity and rate distortion with two-sided state information," *IEEE Trans. Inform. Theory*, pp. 1629–1638, June 2002.

[6] R. Blahut, "Computation of channel capacity and rate-distortion functions," *IEEE Trans. Inform. Theory*, vol. 18, pp. 460–472, July 1972.

[7] S. Arimoto, "An algorithm for calculating the capacity of an arbitrary discrete memoryless channel," *IEEE Trans. Inform. Theory*, vol. 18, pp. 14–20, Jan. 1972.

TABLE II

RATE-DISTORTION FUNCTION FOR FIVE DIFFERENT CASES IN EXAMPLE 2.

R-D function	Cases	
$R_1(D)$	$\mathbb{S}_1 = \emptyset, \mathbb{S}_2 = \emptyset$;	$\mathbb{S}_1 = \{\theta\}, \mathbb{S}_2 = \emptyset$;
	$\mathbb{S}_1 = \emptyset, \mathbb{S}_2 = \{\theta\}$;	$\mathbb{S}_1 = \{\theta\}, \mathbb{S}_2 = \{\theta\}$;
	$\mathbb{S}_1 = \{\tau\}, \mathbb{S}_2 = \emptyset$;	$\mathbb{S}_1 = \{\tau\}, \mathbb{S}_2 = \{\theta\}$;
	$\mathbb{S}_1 = \{\theta, \tau\}, \mathbb{S}_2 = \emptyset$;	$\mathbb{S}_1 = \{\theta, \tau\}, \mathbb{S}_2 = \{\theta\}$
$R_2(D)$	$\mathbb{S}_1 = \emptyset, \mathbb{S}_2 = \{\tau\}$;	$\mathbb{S}_1 = \{\theta\}, \mathbb{S}_2 = \{\tau\}$
$R_3(D)$	$\mathbb{S}_1 = \{\tau\}, \mathbb{S}_2 = \{\tau\}$;	$\mathbb{S}_1 = \{\tau, \theta\}, \mathbb{S}_2 = \{\tau\}$
$R_4(D)$	$\mathbb{S}_1 = \emptyset, \mathbb{S}_2 = \{\theta, \tau\}$;	$\mathbb{S}_1 = \{\theta\}, \mathbb{S}_2 = \{\theta, \tau\}$
$R_5(D)$	$\mathbb{S}_1 = \{\tau\}, \mathbb{S}_2 = \{\theta, \tau\}$;	$\mathbb{S}_1 = \{\theta, \tau\}, \mathbb{S}_2 = \{\theta, \tau\}$

[8] I. Csiszár and G. Tusnády, "Information geometry and alternating minimization procedures," *Statistics and Decisions*, Supplement Issue 1, pp. 205-237, 1984.

[9] P. O. Vontobel, "A generalized Blahut-Arimoto algorithm," in *Proc. ISIT'03*, Yokohama, Japan, July 2003, p. 53.

[10] A. Kavčić, "On the capacity of Markov sources over noisy channels," in *Proc. IEEE Globecom*, San Antonio, TX, Nov. 2001, pp. 2997–3001.

[11] F. Dupuis, W. Yu, and F. Willems, "Blahut-Arimoto algorithms for computing channel capacity and rate-distortion with side information," in *Proc. ISIT'04*, Chicago, IL, June 2004, p. 179.

[12] C. Heegard and A. A. E. Gamal, "On the capacity of computer memory with defects," *IEEE Trans. Inform. Theory*, vol. 29, pp. 731–739, Sept. 1983.

[13] M. Rezaeian and A. Grant, "A generalization of Arimoto-Blahut algorithm," in *Proc. ISIT'04*, Chicago, IL, June 2004, p.181.

[14] M. Chiang and S. Boyd, "Geometric programming duals of channel capacity and rate distortion," *IEEE Trans. Inform. Theory*, pp. 245–259, Feb 2004.

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

[16] R. W. Yeung, *A first course in information theory.* New York: Kluwer Academic / Plenum Publishers, 2002.

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