# Extension of the Blahut–Arimoto Algorithm for Maximizing Directed Information

Iddo Naiss and Haim H. Permuter, *Member, IEEE*

***Abstract*—In this paper, we extend the Blahut–Arimoto algorithm for maximizing Massey's directed information. The algorithm can be used for estimating the capacity of channels with delayed feedback, where the feedback is a deterministic function of the output. In order to maximize the directed information, we apply the ideas from the regular Blahut–Arimoto algorithm, i.e., the alternating maximization procedure, to our new problem. We provide both upper and lower bound sequences that converge to the optimum global value. Our main insight in this paper is that in order to find the maximum of the directed information over a causal conditioning probability mass function, one can use a backward index time maximization combined with the alternating maximization procedure. We give a detailed description of the algorithm, showing its complexity and the memory needed, and present several numerical examples.**

***Index Terms*—Alternating maximization procedure, backward index time maximization, Blahut–Arimoto algorithm, causal conditioning, channels with feedback, directed information, finite-state channels (FSCs), Ising channel, trapdoor channel.**

## I. Introduction

In his seminal work, Shannon [1] showed that the capacity of a memoryless channel is given by the optimization problem

$$C = \max_{p(x)} I(X;Y) \tag{1}$$

where

$$I(X;Y) = \sum_{x,y} p(x)p(y|x) \log \frac{p(x|y)}{p(x)} \tag{2}$$

and $p(x|y)$ is induced by the joint distribution $p(x)p(y|x)$. Since the set of all $p(x)$ is not of finite cardinality, an optimization method is required to find the capacity $C$. In order to obtain an efficient way to calculate the global maximum in (1), the well-known Blahut–Arimoto algorithm (referred to as BAA) was introduced by Blahut [2] and Arimoto [3] in 1972. The main idea is that we can find the optimum value of (1) by calculating the right-hand side (RHS) of the equality

$$\max_{p(x)} I(X;Y) = \max_{p(x),p(x|y)} I(X;Y). \tag{3}$$

On the left-hand side (LHS) of (3), the maximization of $I(X;Y)$ as defined in (2) is only over $p(x)$, where $p(y|x)$ is fixed and $p(x|y)$ is induced by $p(x)p(y|x)$. On the RHS of (3), the maximization of $I(X;Y)$ as defined in (2) is over $p(x)$ and $p(x|y)$, where $p(y|x)$ is fixed, namely, $p(x|y)$ is a parameter rather then being induced by the joint probability $p(x)p(y|x)$. The maximization is then achieved using the alternating maximization procedure. The convergence of the alternating maximization procedure to the global maximum was proven in detail by Csiszár and Tusnady in [4]. Yeung [5, Ch. 9.1] provided a different proof, which we use later on.

In this paper, we find an efficient algorithm for optimizing the directed information which is used to estimate or bound the capacity of channels with feedback. A general channel with feedback is shown in Fig. 1. We note that a channel is defined by a sequence of causal conditioning probabilities $\{p(y_i|x^i, y^{i-1})\}_{i\geq 1}$ and no restriction is imposed. Equivalently, the channels is defined by *causally conditioned* probability mass function (PMF) (definitions in Section II) given by

$$p(y^n\|x^n) = \prod_{i=1}^{n} p(y_i|y^{i-1}, x^i). \tag{4}$$

It was shown by Massey [6], Kramer [7], Tatikonda and Mitter [8], Permuter *et al*. [9] and Kim [10] that the expression

$$C_n = \frac{1}{n} \max_{p(x^n\|y^{n-1})} I(X^n \to Y^n)$$

has an important role in characterizing the feedback capacity, where

$$I(X^n \to Y^n) = \sum_{y^n,x^n} p(y^n, x^n) \log \frac{p(y^n\|x^n)}{p(y^n)}$$

is the *directed information*. In some special cases, the limit of the sequence $C_n$ is, in fact, the capacity of the channel, where, for the general case, $C_n$ is used in an expression that bounds the capacity.

Since in the maximization we deal with causally conditioned PMFs, trying to follow the regular BAA will result in difficulties. This is due to the fact that a causally conditioned PMF is the result of multiplications of conditioned PMFs as seen in (4). In the regular BAA, we maximize over $p(x^n)$, and thus, the constraints are simply $\sum_{x^n} p(x^n) = 1$ and $p(x^n) \geq 0$. However, in our problem, we have no efficient way of optimizing the

Manuscript received December 21, 2010; revised April 05, 2012; accepted July 25, 2012. Date of publication September 11, 2012; date of current version December 19, 2012. This work was supported in part by the Marie Curie Reintegration Fellowship Program under a U.S.–Israel Binational Science Foundation Grant 2008402 and a German–Israeli Foundation for Scientific Research and Development Grant 2275/2010.

I. Naiss was with the Department of Electrical and Computer Engineering, Ben-Gurion University of the Negev, Beer-Sheva 84105, Israel. He is now with Samsung Electronics, Tel-Aviv 61131, Israel (e-mail: naiss@bgu.ac.il).

H. Permuter is with the Department of Electrical and Computer Engineering, Ben-Gurion University of the Negev, Beer-Sheva 84105, Israel (e-mail: haimp@bgu.ac.il).

Communicated by T. Uyematsu, Associate Editor for Shannon Theory.

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIT.2012.2214202

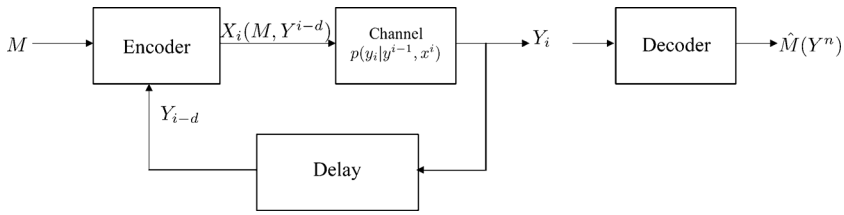


Fig. 1. Feedback-channel model.

directed information over $p(x^n\|y^{n-1})$ under all the necessary constraints, since we need $n$ affine constraints—one for each factor of $p(x^n\|y^{n-1})$, i.e., for all $i$

$$\sum_{x_i} p(x_i|x^{i-1}, y^{i-1}) = 1,\ \forall x^{i-1}, y^{i-1}$$
$$p(x_i|x^{i-1}, y^{i-1}) \geq 0,\ \forall x^i, y^{i-1}.$$

Another difficulty is that although the equality

$$I(X^n \to Y^n) = \sum_{i=1}^{n} I(X_i; Y_i^n|X^{i-1}, Y^{i-1})$$

given by Kim [10, eq. 10], holds, we cannot translate the given problem into

$$\sum_{i=1}^{n} \max_{p(x_i|x^{i-1}, y^{i-1})} I(X_i; Y_i^n|X^{i-1}, Y^{i-1})$$

since $p(x_i|x^{i-1}, y^{i-1})$ influences all terms $\{I(X_j; Y_j^n|X^{j-1}, Y^{j-1})\}_{j=i}^n$. A solution could be to maximize backward from $i = n$ to $i = 1$ over $p(x_i|x^{i-1}, y^{i-1})$ and it can be shown that in each maximization, the noncausal probability $p(x_i|x^{i-1}, y^n)$ is determined only by the previous $p(x_j|x^{j-1}, y^{j-1})$ for $j \geq i$. In our solution, we maximize the entire expression $I(X^n \to Y^n)$ as a function of $\{p(x_1), p(x_2|x_1, y_1), \ldots, p(x_n|x^{n-1}, y^{n-1}), p(x^n|y^n)\}$. Each time, we maximize over a specific $p(x_i|x^{i-1}, y^{i-1})$ starting from $i = n$ and moving backward to $i = 1$, where all but $p(x_i|x^{i-1}, y^{i-1})$ are fixed.

Before we present the extension of the BAA to the directed information, let us present some existing extensions of this algorithm. In 2004, Matz and Duhamel [11] proposed two Blahut–Arimoto-type algorithms that often converge significantly faster than the standard BAA. These algorithms rely on a special gradient form called the "natural gradient" rather than maximizing per variable. During that year, Rezaeian and Grant [12] generalized the regular BAA for multiple access channels and Yu *et al*. extended the BAA for channels with side information [13]. They used the fact that the input is a deterministic function of the auxiliary variable and the side information, and then extended the input alphabet. Another solution to the side information problem was given by Heegard and El Gamal [14], where they did not expand the alphabet, but included an additional step to optimize over $p(x|u, s)$. Also, the BAA was used by Markavian *et al*. [15] to decode Reed–Solomon codes. In 2005, Dauwels [16] showed how the BAA can be used to calculate the capacity of continuous channels. Dauwels's main idea is based on the use of sequential Monte-Carlo integration methods known as the "particle filters." In 2008, Arnold *et al*. [17] extended the regular BAA to estimate the capacity of finite-state channels (FSCs) where the input is Markovian. Sumszyk and Steinberg [18] gave a single letter characterization of the capacity of an information embedding channel and provided a BA-type algorithm for the case where the channel is independent of the host, when the input is given. In 2009, Niesen *et al*. [19] provided an extension to the alternating optimization procedure where the parameters of the underlying problem change over time, thus requiring an adaptive algorithm.

Recently, a few papers related to the maximization of the directed information using control theory and dynamic programming have been published. In [20], Kavcic *et al*. maximized the directed information to estimate the feedback capacity of finite-state machine channels where the state is a deterministic function of the previous state and input. Chen and Berger [21] maximized the directed information for the case where the state of the channel is known to the encoder and decoder in addition to the feedback link. Later, Permuter *et al*. [22] maximized the directed information and found the capacity of the trapdoor channel with feedback. In [23], Gorantla and Coleman estimated the maximum of directed information where they considered a dynamical system, whose state is an input to a memoryless channel. The state of the dynamical system is affected by its past, an exogenous input and causal feedback from the channel's output.

The remainder of this paper is organized as follows. In Section II, we present the notations we use throughout the paper and outline the alternating maximization procedure as given by Yeung [5, Ch. 9.1]. In Section III, we give a description of the algorithm for solving the optimization problem—$\max_{p(x^n\|y^{n-1})} I(X^n \to Y^n)$, calculate the complexity of the algorithm and memory needed, and compare it with those of the regular BAA. In Section IV, we derive the algorithm using the alternating maximization procedure and show the convergence of our algorithm to the optimum value. Numerical examples for channel capacity with feedback are presented in Section V. In Appendix A, we give a wider perspective on the feedback channel problem, where the feedback of the channel is a deterministic function $f$ of the output with some delay $d$, namely, we derive the algorithm for the optimization problem $\max_{p(x^n\|z^{n-d})} I(X^n \to Y^n)$, where $z_i = f(y_i)$ and $d \geq 1$. In Appendix B, we prove an upper bound for $\max_{p(x^n\|y^{n-d})} I(X^n \to Y^n)$, which converges to the directed information from above and helps to determine the stopping point of the algorithm.

## II. Preliminaries

### A. Directed Information and Causal Conditioning

In this section, we present the definitions of directed information and causally conditioned PMF, which were originally introduced by Massey [6] (who was inspired by Marko's work [24] on bidirectional communication) and by Kramer [7]. These definitions are necessary in order to address channels with memory. We denote by $X_1^n$ the source vector $(X_1, X_2, \ldots, X_n)$, where the source alphabet of each $X_i$ is a finite set denoted as $\mathcal{X}$. The channel output alphabet is denoted as $\mathcal{Y}$. Usually, we use the notation $X^n = X_1^n$ for short. Furthermore, when writing a PMF, we simply write $P_X(x) = p(x)$. Let us denote as $p(x^n \| y^{n-d})$ the PMF of $X^n$ *causally conditioned* on $Y^{n-d}$, given by

$$p(x^n \| y^{n-d}) \stackrel{\triangle}{=} \prod_{i=1}^{n} p(x_i | x^{i-1} y^{i-d}). \tag{5}$$

Here, we have to point out that when $d > n$, the notation $X^{n-d}$ indicates the empty set, denoted as $\emptyset$. Two straightforward properties of the causal conditioning PMF that we use throughout the paper are

$$\sum_{x_n} p(x^n \| y^{n-d}) = p(x^{n-1} \| y^{n-d-1}) \tag{6}$$

and

$$p(x_i | x^{i-1} y^{i-d}) = \frac{p(x^i \| y^{i-d})}{p(x^{i-1} \| y^{i-d-1})}. \tag{7}$$

Another elementary property is the chain rule for causal conditioning PMF, given in [9, Lemma 1]

$$p(x^n \| y^{n-1}) p(y^n \| x^n) = p(x^n, y^n). \tag{8}$$

The aforementioned definitions lead to the causally conditioned entropy $H(X^n \| Y^n)$, which is defined by

$$H(X^n \| Y^n) \stackrel{\triangle}{=} - \mathbb{E}\left[\log p(X^n \| Y^n)\right].$$

Moreover, the directed information from $X^n$ to $Y^n$ is defined as

$$\begin{aligned} I(X^n \to Y^n) &\stackrel{\triangle}{=} H(Y^n) - H(Y^n \| X^n) \\ &= \sum_{y^n, x^n} p(y^n \| x^n) r(x^n \| y^{n-1}) \log \frac{p(y^n \| x^n)}{p(y^n)}. \end{aligned}$$

Note that this equality does not require maximization on either side of the equation and the expressions on both sides depend on a specific input distribution $\mathbf{r}$.

It is easy to show that the directed information can be written as a function of $q(x^n|y^n),\ r(x^n \| y^{n-1})$. This follows from the chain rule of causal conditioning, i.e., $p(y^n \| x^n) r(x^n \| y^{n-1}) = p(y^n) q(x^n | y^n)$, and hence

$$\begin{aligned} I(X^n \to Y^n) &= \sum_{y^n, x^n} p(y^n \| x^n) r(x^n \| y^{n-1}) \log \frac{p(y^n \| x^n)}{p(y^n)} \\ &= \sum_{y^n, x^n} p(y^n \| x^n) r(x^n \| y^{n-1}) \log \frac{q(x^n | y^n)}{r(x^n \| y^{n-1})}. \end{aligned} \tag{9}$$

We refer to this form in Lemma 3 while using the alternating maximization procedure since $\{\mathbf{r} = r(x^n \| y^{n-1}),\ \mathbf{q} = q(x^n | y^n)\}$ are the variables we optimize over where $p(y^n \| x^n)$ is fixed. For convenience, we use from now on the notation of

$$I(X^n \to Y^n) = \mathcal{I}(\mathbf{r}, \mathbf{q}) \tag{10}$$

when required. With these definitions, we follow the alternating maximization procedure given by Yeung [5, Ch. 9.1] in order to maximize the directed information.

### B. Alternating Maximization Procedure

Here, we present the alternating maximization procedure on which our algorithm is based. Let $f(u_1, u_2)$ be a real function, and let us consider the optimization problem given by

$$\sup_{u_1 \in A_1, u_2 \in A_2} f(u_1, u_2) = f^*$$

where $A_1$and $A_2$ are the sets we optimize over. We denote by $c_2(u_1)$ the point that achieves $\sup_{u_2 \in A_2} f(u_1, u_2)$, and by $c_1(u_2)$ the one that achieves $\sup_{u_1 \in A_1} f(u_1, u_2)$. The algorithm is performed by iterations, where in each iteration we maximize over one of the variables. Let $(u_1^0, u_2^0)$ be an arbitrary point in $A_1 \times A_2$. For $k \geq 0$ let

$$(u_1^k, u_2^k) = (c_1(u_2^{k-1}), c_2(c_1(u_2^{k-1})))$$

and let $f^k = f(u_1^k, u_2^k)$ be the value in the current iteration. The following lemma describes the conditions the problem needs to meet in order for $f^k$ to converge to $f^*$ as $k$ goes to infinity.

*Lemma 1 Convergence of the Alternating Maximization Procedure [5, Lemmas 9.4 and 9.5]:* Let $f(u_1, u_2)$ be a real, concave, bounded-from-above function that is continuous and has continuous partial derivatives, and let the sets $A_1$and$A_2$, which we maximize over, be convex. Further, assume that $c_2(u_1) \in A_2$ and $c_1(u_2) \in A_1$ for all $u_1 \in A_1,\ u_2 \in A_2$ and that $c_2(u_1), c_1(u_2)$ are unique. Under these conditions, $\lim_{k \to \infty} f^k = f^*$, where $f^*$ is the global maximum.

In Section III, we give a detailed description of the algorithm that computes $\max_{p(x^n \| y^{n-1})} I(X^n \to Y^n)$ based on the alternating maximization procedure. In Section IV, we show that the conditions in Lemma 1 hold, and therefore, the algorithm we suggest, which is based on the alternating maximization procedure, converges to the global optimum.

## III. Description of the Algorithm

In this section, we describe an algorithm for maximizing the directed information. In addition, we compute the complexity of the algorithm per iteration, and compare it to the complexity of the regular BAA. The complexity calculation is in terms of additions and multiplications. The amount of memory needed by the algorithm is also given.

### A. Algorithm for Channel With Feedback

In Algorithm 1, we present the steps required to maximize the directed information where the channel $p(y^n \| x^n)$ is fixed and the delay is $d = 1$. Note that this algorithm is similar in many

ways to the regular BAA given in [2] and [3] and we present the main steps of the regular BAA at the end of this section.

**Algorithm 1** Iterative algorithm for calculating $C_n = \max_{p(x^n\|y^{n-1})} I(X^n \to Y^n)$. The inputs to the algorithm are the channel probability $p(y^n\|x^n)$ and $\epsilon$. The outputs are lower bound $I_L$, and upper bound $I_U$ of $C_n$ that satisfies $I_U - I_L \leq \epsilon$ and the probability $r(x^n\|y^{n-1})$ that achieves $I_L$.

a) Start from a random point $q(x^n|y^n)$. Usually, we start from a uniform distribution, i.e., $q(x^n|y^n) = |\mathcal{X}|^{-n}$ for every $(x^n, y^n)$. Also, set $i = n$.
b) Calculate $r(x_i|x^{i-1}, y^{i-1})$ using the formula
$$r(x_i|x^{i-1}, y^{i-1}) = \frac{r'(x^i, y^{i-1})}{\sum_{x_i} r'(x^i, y^{i-1})} \tag{11}$$
where (12) shown at the bottom of the page holds and do so backwards until $i = 1$.
c) Once you find $\{r(x_i|x^{i-1}, y^{i-1})\}_{i=1}^n$, compute $r(x^n\|y^{n-1}) = \prod_{i=1}^n r(x_i|x^{i-1}, y^{i-1})$.
d) Compute $q(x^n|y^n)$ using the formula
$$q(x^n|y^n) = \frac{r(x^n\|y^{n-1})p(y^n\|x^n)}{\sum_{x^n} r(x^n\|y^{n-1})p(y^n\|x^n)}. \tag{13}$$
e) Calculate $I_U - I_L$, where the equation shown at the bottom of the page holds.
f) If $(I_U - I_L) \geq \epsilon$, set $i = i - 1$ and go to (b).
g) $C_n = I_L$.

Note that Algorithm 1 has a structure similar to that of the regular BAA, where step (b) is an additional backward loop. Its purpose is to maximize over the input causal probability, which is not necessary in the regular BAA.

Now, let us present a special case and a few extensions for Algorithm 1.

1) *Regular BAA*, i.e., $n = 1$: For $n = 1$, the algorithm suggested here agrees with the original BAA, where instead of steps (b) and (c) we have
$$r(x) = \frac{\prod_y q(x|y)^{p(y|x)}}{\sum_x \prod_y q(x|y)^{p(y|x)}} \tag{14}$$
and step (d) is replaced by
$$q(x|y) = \frac{r(x)p(y|x)}{\sum_x r(x)p(y|x)}. \tag{15}$$
The bounds $I_L,\ I_U$ agree with the regular BAA as well, and are of the form
$$I_U = \max_x \sum_y p(y|x) \log \frac{p(y|x)}{\sum_{x'} p(y|x') \cdot r(x')}$$
$$I_L = \sum_{y,x} p(y|x)r(x) \log \frac{q(x|y)}{r(x)}.$$

2) *Feedback with general delay $d$*: We can generalize the algorithm in order to compute $\max_{r(x^n\|y^{n-d})} I(X^n \to Y^n)$, which is used to estimate the capacity of a channel with feedback where the output is known to the encoder with delay $d$. We would like to emphasis that for the delayed feedback case, the channel $p(y^n\|x^n)$ remains the same, i.e., $p(y^n\|x^n) = \prod_{i=1}^n p(y_i|y^{i-1}, x^i)$ and is not influenced by the delayed feedback. In that case, in step (b), we have (16) shown at the bottom of the page, and step (d) will be replaced by
$$q(x^n|y^n) = \frac{r(x^n\|y^{n-d})p(y^n\|x^n)}{\sum_{x^n} r(x^n\|y^{n-d})p(y^n\|x^n)}. \tag{17}$$
The bounds $I_L$ and $I_U$ are of the form
$$I_U = \frac{1}{n} \max_{x^d} \sum_{y_1} \max_{x_{d+1}} \cdots \sum_{y_{n-d}} \max_{x_n} \sum_{y_{n-d+1}^n} p(y^n\|x^n) \cdot \log \frac{p(y^n\|x^n)}{\sum_{x'^n} p(y^n\|x'^n) \cdot r(x'^n\|y^{n-d})}$$
$$I_L = \frac{1}{n} \sum_{y^n, x^n} p(y^n\|x^n)r(x^n\|y^{n-d}) \log \frac{q(x^n|y^n)}{r(x^n\|y^{n-d})}.$$

3) *Feedback as a function of the output with general delay*. In Appendix A, we generalize the algorithm in order to compute $\max_{r(x^n\|z^{n-d})} I(X^n \to Y^n)$, where the feedback,

$$r'(x^i, y^{i-1}) = \prod_{x_{i+1}^n, y_i^n} \left[ \frac{q(x^n|y^n)}{\prod_{j=i+1}^n r(x_j|x^{j-1}, y^{j-1})} \right]^{p(y_i|x^i, y^{i-1}) \prod_{j=i+1}^n r(x_j|x^{j-1}, y^{j-1}) p(y_j|x^j, y^{j-1})} \tag{12}$$

$$I_U = \frac{1}{n} \max_{x_1} \sum_{y_1} \max_{x_2} \cdots \sum_{y_{n-1}} \max_{x_n} \sum_{y_n} p(y^n\|x^n) \log \frac{p(y^n\|x^n)}{\sum_{x'^n} p(y^n\|x'^n) \cdot r(x'^n\|y^{n-1})}$$
$$I_L = \frac{1}{n} \sum_{y^n, x^n} p(y^n\|x^n)r(x^n\|y^{n-1}) \log \frac{q(x^n|y^n)}{r(x^n\|y^{n-1})}$$

$$r'(x^i, y^{i-d}) = \prod_{x_{i+1}^n, y_{i-d+1}^n} \left[ \frac{q(x^n|y^n)}{\prod_{j=i+1}^n r(x_j|x^{j-1}, y^{j-d})} \right]^{\prod_{j=i-d+1}^n p(y_j|x^j, y^{j-1}) \prod_{j=i+1}^n r(x_j|x^{j-1}, y^{j-d})} \tag{16}$$

$z^{n-d} = f(y^{n-d})$, is a deterministic function of the delayed output, where $f$ is given in advanced. The result of the maximization above characterizes the capacity of channels with time-invariant feedback [9]. In that case, in step (b), we have (18) shown at the bottom of the page, where we define the set $A_{i,d,z} \stackrel{\triangle}{=} \{y^{i-d} : z^{i-d} = f(y^{i-d})\}$ as the set of pre-images of $z^{i-d}$ under $f$, and step (d) will be replaced by

$$q(x^n|y^n) = \frac{r(x^n\|z^{n-d})p(y^n\|x^n)}{\sum_{x^n} r(x^n\|z^{n-d})p(y^n\|x^n)}. \tag{19}$$

The bounds $I_L$and$I_U$ are of the form

$$I_U = \frac{1}{n}\max_{x^d}\sum_{z_1}\max_{x_{d+1}}\cdots\sum_{z_{n-d}}\max_{x_n}\sum_{A_{n,d,z}}\sum_{y_{n-d+1}^n} p(y^n\|x^n)\cdot \log\frac{p(y^n\|x^n)}{\sum_{x'^n} p(y^n\|x'^n)\cdot r(x'^n\|z^{n-d})}$$

$$I_L = \frac{1}{n}\sum_{y^n,x^n} p(y^n\|x^n)r(x^n\|z^{n-d})\log\frac{q(x^n|y^n)}{r(x^n\|z^{n-d})}.$$

Note, that for $d = n$, the vector $z^{n-d} = \emptyset$. Hence, for every $i = 1,\ldots,n$, $r(x_i|x^{i-1}, z^{i-d}) = r(x_i|x^{i-1})$, where $r'(x^i)$ is in (20), shown at the bottom of the page. Furthermore, the following equality holds for the causal conditioning PMF $r(x^n\|z^{n-d})$:

$$r(x^n\|z^{n-d}) = \prod_{i=1}^{n} r(x_i|x^{i-1}) = r(x^n).$$

Also note that when $f(y) = \text{const.}$, $r(x^n\|z^{n-d}) = r(x^n)$, $A_{i,d,z} = \{\mathcal{Y}\}^{i-d}$, and $\sum_{y^{i-d}}\prod_{j=1}^{i-d} p(y_j|x^j, y^{j-1}) = 1$. In each of the aforementioned cases ($d = n$ or $f(y) = \text{const.}$), in step (d), we have

$$q(x^n|y^n) = \frac{r(x^n)p(y^n\|x^n)}{\sum_{x^n} r(x^n)p(y^n\|x^n)}$$

and we obtain another version of the regular BAA for channel capacity, where the maximization is done over each $r(x_i|x^{i-1})$ via backward maximization instead of over $r(x^n)$ immediately. Furthermore, if $f(y) = y$, then case 3) agrees with all the equations of case 2).

### B. Complexity and Memory Needed

Here, we give an expression for the computation complexity of one iteration in the algorithm and then compare it to regular BAA complexity. Further, we find the amount of memory needed by both algorithms. The complexity calculation is done in two parts, one for each step in the iteration.

1) Complexity of computing $q(x^n|y^n)$ as given in (13): For each $y^n$, we need $|\mathcal{X}|^n$ multiplications for a specific $x^n$ and use the denominator computed for every other $x^n$, thus obtaining $O(|\mathcal{X}|^n)$ operations. Doing so for all $y^n$ achieves $O(|\mathcal{X}|^n|\mathcal{Y}|^n) = O((|\mathcal{X}||\mathcal{Y}|)^n)$.
2) Complexity of computing $r(x^n\|y^{n-1})$: First, we compute the complexity of each $r(x_i|x^{i-1}, y^{i-1})$ as given in (12), assuming that an exponent is a constant number of computations, i.e., $O(1)$. Simple computations will lead to the conclusion that the entire numerator takes about $O((n-i)(|\mathcal{X}||\mathcal{Y}|)^{n-i})$ computations. The denominator is a summation over $|\mathcal{X}|^i$ variables and, as with $q(x^n|y^n)$, we can use the denominator for every other $x^i$. Hence, we obtain $O((n-i)(|\mathcal{X}||\mathcal{Y}|)^n)$ computations for every $i \in \{1,\ldots,n\}$. Summing over $i$ will achieve $O((n+n^2)(|\mathcal{X}||\mathcal{Y}|)^n) = O(n^2(|\mathcal{X}||\mathcal{Y}|)^n)$ computations. Multiplying all $r(x_i|x^{i-1}, y^{i-1})$s is a constant number of computations for every $(x_i, y_i)$. Finally, in order to compute $r(x^n\|y^{n-1})$, we need $O((n^2+n)(|\mathcal{X}||\mathcal{Y}|)^n)$ computations.

To conclude, each iteration requires about $O(n^2(|\mathcal{X}||\mathcal{Y}|)^n)$ computations.

Comparing to regular BAA complexity: Since BAA computes the capacity of memoryless channels, we only need to compute $r(x)$ and $q(x|y)$. In much the same way, we can find its complexity and obtain $O((|\mathcal{X}||\mathcal{Y}|))$ computations. However, if we want to compare it to BAA for channels with memory, we replace $X \Leftrightarrow X^n, Y \Leftrightarrow Y^n$. However, $|\mathcal{X}^n| = |\mathcal{X}|^n$ and so we obtain $O((|\mathcal{X}||\mathcal{Y}|)^n)$ computations.

The memory needed by the algorithm is very much dependent on the manner in which one implements the algorithm. However, the obligatory memory needed is for $\mathbf{q}$, $p(y^n\|x^n)$, and $\mathbf{r}$ and its factors; thus, we need at least $n(|\mathcal{X}||\mathcal{Y}|)^n$ cells of type double. Computation complexity and memory needed are presented in Table I.

$$r'(x^i, z^{i-d}) = \prod_{x_{i+1}^n, y_{i-d+1}^n}\prod_{A_{i,d,z}}\left[\frac{q(x^n|y^n)}{\prod_{j=i+1}^n r(x_j|x^{j-1}, z^{j-d})}\right]^{\frac{p(y^n\|x^n)\prod_{j=i+1}^n r(x_j|x^{j-1},z^{j-d})}{\sum_{A_{i,d,z}}\prod_{j=1}^{i-d} p(y_j|x^j,y^{j-1})}} \tag{18}$$

$$r'(x^i) = \prod_{x_{i+1}^n, y^n}\left[\frac{q(x^n|y^n)}{\prod_{j=i+1}^n r(x_j|x^{j-1})}\right]^{\prod_{j=1}^n p(y_j|x^j,y^{j-1})\prod_{j=i+1}^n r(x_j|x^{j-1})} \tag{20}$$

TABLE I
MEMORY AND OPERATIONS NEEDED FOR REGULAR AND EXTENDED BAA FOR CHANNEL CODING WITH FEEDBACK

| | Operation | Memory |
|---|---|---|
| $\max_{p(x)} \left(\frac{1}{n} I(X^n;Y^n)\right)$, regular BAA for channel capacity | $O((\lvert\mathcal{X}\rvert\lvert\mathcal{Y}\rvert)^n)$ | $(\lvert\mathcal{X}\rvert\lvert\mathcal{Y}\rvert)^n$ |
| $\max_{p(x^n\Vert y^{n-1})} \left(\frac{1}{n} I(X^n \to Y^n)\right)$, Alg. 1 | $O(n^2(\lvert\mathcal{X}\rvert\lvert\mathcal{Y}\rvert)^n)$ | $n(\lvert\mathcal{X}\rvert\lvert\mathcal{Y}\rvert)^n$ |

## IV. DERIVATION OF ALGORITHM 1

In this section, we derive Algorithm 1 using the alternating maximization procedure and derive its convergence to the global optimum using Lemma 1. Throughout the paper, note that the channel $p(y^n\|x^n)$ is fixed in all maximization calculations. For this purpose, we present several lemmas that will assist in proving our main goal, which is to develop an algorithm for calculating $\max I(X^n \to Y^n)$. In Lemma 2, we show that the directed information function has the properties required for Lemma 1. In Lemma 3, we show that we are allowed to maximize the directed information over $r(x^n\|y^{n-1})$ and $q(x^n|y^n)$ together, rather than just over $r(x^n\|y^{n-1})$, thus creating an opportunity to use the alternating maximization procedure for achieving the optimum value. Lemma 4 is a supplementary claim that helps us prove Lemma 3, in which we find an expression for $q(x^n|y^n)$ that maximizes the directed information where $r(x^n\|y^{n-1})$ is fixed. In Lemma 5, we find an explicit expression for $r(x^n\|y^{n-1})$ that maximizes the directed information where $q(x^n|y^n)$ is fixed. Theorem 1 combines all lemmas to show that the alternating maximization procedure as described by $I_L$ in Algorithm 1 exists and converges. We end with Theorem 2 that proves the existence of the upper bound $I_U$.

*Lemma 2:* For a fixed channel, $p(y^n\|x^n)$, the directed information given by

$$I(X^n \to Y^n) = \sum_{y^n,x^n} p(y^n\|x^n) r(x^n\|y^{n-1}) \log \frac{q(x^n|y^n)}{r(x^n\|y^{n-1})} \tag{21}$$

as a function of $\{\mathbf{r} = r(x^n\|y^{n-1}),\ \mathbf{q} = q(x^n|y^n)\}$ is concave, continuous, and with continuous partial derivatives.

*Proof:* First, we remind that we have shown in (9) that we can write the directed information in the aforementioned form. Now, we can recall the log-sum inequality [25, Th. 2.7.1] given by

$$\sum_{i=1}^{n} a_i \log \frac{a_i}{b_i} \geq \left(\sum_{i=1}^{n} a_i\right) \log \frac{\sum_{i=1}^{n} a_i}{\sum_{i=1}^{n} b_i}. \tag{22}$$

We define the sets

$$\begin{aligned} A_1 &= \{r(x^n\|y^{n-1}) : r(x^n\|y^{n-1}) > 0 \text{ is a causally conditioned PMF}\} \\ A_2 &= \{q(x^n|y^n) : q(x^n|y^n) \text{ is a conditioned PMF}\} \end{aligned} \tag{23}$$

as the sets over which we maximize.

Now, we follow the proof in [5, Ch. 9.3.2] for $(\mathbf{r}_1, \mathbf{q}_1)$, $(\mathbf{r}_2, \mathbf{q}_2)$ in $A = A_1 \times A_2$ and $\lambda \in [0,1]$, and use the aforementioned log-sum inequality given to derive that

$$\begin{aligned} &(\lambda r_1 + (1-\lambda) r_2) \log \frac{\lambda r_1 + (1-\lambda) r_2}{\lambda q_1 + (1-\lambda) q_2} \\ &\qquad \leq \lambda r_1 \log \frac{r_1}{q_1} + (1-\lambda) r_2 \log \frac{r_2}{q_2}. \end{aligned}$$

Taking the reciprocal of the logarithms yields

$$\begin{aligned} &(\lambda r_1 + (1-\lambda) r_2) \log \frac{\lambda q_1 + (1-\lambda) q_2}{\lambda r_1 + (1-\lambda) r_2} \\ &\qquad \geq \lambda r_1 \log \frac{q_1}{r_1} + (1-\lambda) r_2 \log \frac{q_2}{r_2}. \end{aligned}$$

Multiplying by $p(y^n\|x^n)$ and summing over all $x^n,\ y^n$, and letting $\mathcal{I}(\mathbf{r}, \mathbf{q})$ be the directed information as in (10), we obtain

$$\begin{aligned} &\mathcal{I}(\lambda \mathbf{r}_1 + (1-\lambda)\mathbf{r}_2, \lambda \mathbf{q}_1 + (1-\lambda)\mathbf{q}_2 \\ &\qquad \geq \lambda \mathcal{I}(\mathbf{r}_1, \mathbf{q}_1) + (1-\lambda)\mathcal{I}(\mathbf{r}_2, \mathbf{q}_2). \end{aligned}$$

Further, since the function $\log(x)$ is continuous with continuous partial derivatives, and the directed information is a summation of functions of type $\log(x)$, $\mathcal{I}(\mathbf{r}, \mathbf{q})$ has the same properties as well. ■

We note that it is simple to verify that the sets $A_1$and $A_2$ are both convex, and we can conclude that all conditions in Lemma 1 hold for the directed information. Hence, the alternating maximization procedure described here converges to the global maximum, as stated in Lemma 1.

Recall that in the alternating maximization procedure, we maximize over $\{r(x^n\|y^{n-1}),\ \ q(x^n|y^n)\}$ instead of over $r(x^n\|y^{n-1})$ alone, and thus need the following lemma.

*Lemma 3:* For any discrete random variables $X^n,\ Y^n$, the following holds:

$$\max_{\mathbf{r}} I(X^n \to Y^n) = \max_{\mathbf{r},\mathbf{q}} I(X^n \to Y^n). \tag{24}$$

The proof will be given after the following supplementary claim, in which we calculate the specific $q(x^n|y^n)$ that maximizes the directed information where $r(x^n\|y^{n-1})$ is fixed.

*Lemma 4:* For fixed $r(x^n\|y^{n-1})$, there exists a unique $c_2(r) = q^*(x^n|y^n)$ that achieves $\max_{q(x^n|y^n)} I(X^n \to Y^n)$, and is given by

$$q^*(x^n|y^n) = \frac{r(x^n\|y^{n-1}) p(y^n\|x^n)}{\sum_{x^n} r(x^n\|y^{n-1}) p(y^n\|x^n)}.$$

*Proof for Lemma 4:* Let $\mathbf{q}^* = q^*(x^n|y^n)$. For any $\mathbf{q} = q(x^n|y^n)$, and fixed $\mathbf{r} = r(x^n\|y^{n-1})$

$$\begin{aligned}
&\mathcal{I}(\mathbf{r},\mathbf{q}^*) - \mathcal{I}(\mathbf{r},\mathbf{q}) \\
&= \sum_{x^n,y^n} r(x^n\|y^{n-1})p(y^n\|x^n)\log\frac{q^*(x^n|y^n)}{r(x^n\|y^{n-1})} \\
&\quad - \sum_{x^n,y^n} r(x^n\|y^{n-1})p(y^n\|x^n)\log\frac{q(x^n|y^n)}{r(x^n\|y^{n-1})} \\
&= \sum_{x^n,y^n} r(x^n\|y^{n-1})p(y^n\|x^n)\log\frac{q^*(x^n|y^n)}{q(x^n|y^n)} \\
&= D\left(\mathbf{r}\cdot p(y^n\|x^n) \parallel \mathbf{q}\cdot\sum_{x^n}\mathbf{r}\cdot p(y^n\|x^n)\right) \\
&\overset{(a)}{\geq} 0
\end{aligned}$$

where (a) follows from the nonnegativity of the divergence. Finally, divergence of two PMFs is zero if and only if the PMFs are identical. Therefore, we can conclude that inequality (a) holds only if

$$r(x^n\|y^{n-1})p(y^n\|x^n) = q(x^n|y^n)\sum_{x^n} r(x^n\|y^{n-1})p(y^n\|x^n) \quad \forall x^n\in\mathcal{X}^n, y^n\in\mathcal{Y}^n \tag{25}$$

which implies that $\mathbf{q} = \mathbf{q}^*$, namely, we obtained the uniqueness of $\mathbf{q}^*$. ■

*Proof of Lemma 3:* After finding the PMF $\mathbf{q}$ that maximizes $\mathcal{I}(\mathbf{r},\mathbf{q})$ where $\mathbf{r}$ is fixed, we can see that $q(x^n|y^n)$ is the one that corresponds to the joint distribution $r(x^n\|y^{n-1})p(y^n\|x^n)$ in the sense that

$$\begin{aligned}
q(x^n|y^n) &= \frac{p(x^n,y^n)}{p(y^n)} \\
&= \frac{p(x^n,y^n)}{\sum_{x^n}p(x^n,y^n)} \\
&= \frac{r(x^n\|y^{n-1})p(y^n\|x^n)}{\sum_{x^n}r(x^n\|y^{n-1})p(y^n\|x^n)}
\end{aligned}$$

and thus, the lemma is proven. ■

In the following lemma, we find an explicit expression for $\mathbf{r}$ that achieves $\max_{r(x^n\|y^{n-1})} I(X^n\to Y^n)$, where $\mathbf{q}$ is fixed.

*Lemma 5:* For fixed $q(x^n|y^n)$, there exists a unique $c_1(q) = r^*(x^n\|y^{n-1})$ that achieves $\max_{r(x^n\|y^{n-1})} I(X^n\to Y^n)$, and is given by the products

$$r^*(x^n\|y^{n-1}) = \prod_{i=1}^{n} r(x_i|x^{i-1},y^{i-1})$$

where

$$r(x_i|x^{i-1},y^{i-1}) = \frac{r'(x^i,y^{i-1})}{\sum_{x_i} r'(x^i,y^{i-1})} \tag{26}$$

and (27) shown at the bottom of the page.

*Proof:* In order to find the requested $\mathbf{r}$, we find all of its components, namely $\{r(x_i|x^{i-1},y^{i-1})\}_{i=1}^n$, by maximizing the directed information over all of them. With regard to this, we present the following claim.

Maximizing the directed information over $\mathbf{r}$ is equivalent to maximizing it over the set of factors $\{r(x_i|x^{i-1},y^{i-1})\}_{i=1}^n$ denoted as $\{r_i\}$, i.e.,

$$\max_{\mathbf{r}} I(X^n\to Y^n) = \max_{\{r_i\}_{i=1}^n} I(X^n\to Y^n).$$

This follows from [9, Lemma 3], which states that there is a one-to-one correspondence between the causally conditioned PMF $\mathbf{r}$ and the set of its factors $\{r_i\}_{i=1}^n$.

Since in Lemma 2 we showed that $I(X^n\to Y^n)$ is concave in $\{\mathbf{r},\mathbf{q}\}$, it is concave in all $r_i$s separately. Moreover, the constraints of the optimization problem are affine, and we can use the Lagrange multipliers method with the Karush–Kuhn–Tucker (KKT) conditions [26, Ch. 5.3.3] in order to find the optimal $r_i$s. Furthermore, we can arrange the maximization order from $i=n$ to $i=1$. In short, we use the equality

$$\max_{\mathbf{r}} I(X^n\to Y^n) = \max_{r_1}\max_{r_2}\cdots\max_{r_n} I(X^n\to Y^n) \tag{28}$$

and find the optimal $\mathbf{r}$ by solving the RHS of the aforementioned equation. It appears that by maximizing the directed information in the order stated in (28), we would need to maximize over only one factor, $r_i$, at a time.

For convenience, let us further use for short $p_i \overset{\triangle}{=} p(y_i|x^i,y^{i-1})$. Now, for every $i$, we define the following concave optimization problem:

$$\max_{r_i} I(X^n\to Y^n) \tag{29}$$

such that for every $x^{i-1},\ y^{i-1}$

$$\sum_{x_i} r(x_i|x^{i-1},y^{i-1}) = 1.$$

Therefore, we define the Lagrangian as follows:

$$J = \sum_{x^n,y^n}\left(p(y^n\|x^n)\left(\prod_{i=1}^n r_i\right)\log\left(\frac{q(x^n|y^n)}{\prod_{j=1}^n r_j}\right)\right) + \sum_{x^{i-1},y^{i-1}}\nu_{i,(x^{i-1},y^{i-1})}\left(\sum_{x_i} r_i - 1\right). \tag{30}$$

$$r'(x^i,y^{i-1}) = \prod_{x_{i+1}^n,y_i^n}\left[\frac{q(x^n|y^n)}{\prod_{j=i+1}^n r(x_j|x^{j-1},y^{j-1})}\right]^{\prod_{j=i}^n p(y_j|x^j,y^{j-1})\prod_{j=i+1}^n r(x_j|x^{j-1},y^{j-1})} \tag{27}$$

Hence, for every $i \in \{1, \ldots, n\}$ any fixed $x^i$ and $y^{i-1}$, we must find $r_i$ s.t. $\frac{\partial J}{\partial r_i} = 0$, i.e.,

$$\begin{aligned}\frac{\partial J}{\partial r_i} &\overset{(a)}{=} \sum_{x_{i+1}^n, y_i^n} \left( p(y^n \| x^n) \left( \prod_{\substack{j=1 \\ j \neq i}}^{n} r_j \right) \right. \\ &\quad \left. \left[ \log \frac{q(x^n|y^n)}{\prod_{j=1}^n r_j} - 1 \right] \right) + \nu_{i,(x^{i-1}, y^{i-1})} \\ &= \left( \prod_{j=1}^{i-1} r_j \right) \sum_{x_{i+1}^n, y_i^n} \left( p(y^n \| x^n) \left( \prod_{j=i+1}^{n} r_j \right) \cdot \right. \\ &\quad \left. \left[ \log \frac{q(x^n|y^n)}{\prod_{j=i+1}^n r_j} - \log \prod_{j=1}^{i-1} r_j - \log r_i - 1 \right] \right) \\ &\quad + \nu_{i,(x^{i-1}, y^{i-1})} \\ &= 0 \end{aligned} \tag{31}$$

where (a) is a result of derivative over a particular $r(x_i|x^{i-1}, y^{i-1})$; hence, the summation over $x^i$ and $y^{i-1}$ is not required. Note that since $\nu_i$ is a function of $(x^{i-1}, y^{i-1})$, we can divide the whole equation by $\prod_{j=1}^{i-1} r_j$, and get a new relation

$$\nu^*_{i,(x^{i-1}, y^{i-1})} = \frac{\nu_i}{\prod_{j=1}^{i-1} r_j}. \tag{32}$$

Moreover, we can see that three of the terms in the sum, i.e., $\{\log \prod_{j=1}^{i-1} r_j,\ \log r_i,\ 1\}$, do not depend on $(x_{i+1}^n, y_i^n)$, thus leaving their coefficient in (31) to be

$$\begin{aligned}&\sum_{x_{i+1}^n, y_i^n} \left[ p(y^n \| x^n) \prod_{j=i+1}^{n} r(x_j|x^{j-1}, y^{j-1}) \right] \\ &\quad = \prod_{j=1}^{i-1} p(y_j|x^j, y^{j-1}). \end{aligned} \tag{33}$$

Therefore, we obtain from (31) that

$$\begin{aligned}&\sum_{x_{i+1}^n, y_i^n} \left( p(y^n \| x^n) \left( \prod_{j=i+1}^{n} r_j \right) \log \frac{q(x^n|y^n)}{\prod_{j=i+1}^n r_j} \right) \\ &\quad - \left( \prod_{j=1}^{i-1} p_j \right) \left( \log \prod_{j=1}^{i-1} r_j + \log r_i + 1 \right) + \nu^*_{i,(x^{i-1}, y^{i-1})} = 0. \end{aligned} \tag{34}$$

Using algebraic manipulations, we can obtain that

$$\begin{aligned}&\log \left[ \prod_{x_{i+1}^n, y_i^n} \left( \frac{q(x^n|y^n)}{\prod_{j=i+1}^n r_j} \right)^{\frac{p(y^n \| x^n) \prod_{j=i+1}^n r_j}{\prod_{j=1}^{i-1} p_j}} \right] \\ &\quad - \log r_i - \log \nu^{**}_{i,(x^{i-1}, y^{i-1})} = 0, \end{aligned} \tag{35}$$

where

$$\log \nu^{**}_{i,(x^{i-1}, y^{i-1})} = \log \prod_{j=1}^{i-1} r_j + 1 - \frac{\nu^*_{i,(x^{i-1}, y^{i-1})}}{\prod_{j=1}^{i-1} p_j}. \tag{36}$$

Thus, we can see that

$$r_i = \log \left[ \prod_{x_{i+1}^n, y_i^n} \left( \frac{q(x^n|y^n)}{\prod_{j=i+1}^n r_j} \right)^{\frac{p(y^n \| x^n) \prod_{j=i+1}^n r_j}{\prod_{j=1}^{i-1} p_j}} \right] \cdot (\nu^{**})^{-1} \tag{37}$$

and due to the condition $\sum_{x_i} r(x_i|x^{i-1}, y^{i-1}) = 1$, we can find $\nu^{**}$ and have our objective

$$r(x_i|x^{i-1}, y^{i-1}) = \frac{r'(x^i, y^{i-1})}{\sum_{x_i} r'(x^i, y^{i-1})} \tag{38}$$

where $r'(x^i, y^{i-1})$ is as in (27) above.

From equation (Rprime1), we can see that for every $i$, $r_i$ depends on $q(x^n|y^n)$ and the set $\{r_j\}_{j=i+1}^n$ (and hence $r_n$ is a function of $q(x^n|y^n)$ alone). Therefore, after finding $r_n^*$ that maximizes $I(X^n \to Y^n)$, we can place it in the equation we have for $i = n-1$, thus also obtaining $r_{n-1}^*$ depend on $q(x^n|y^n)$ alone. Now we do the same for $r_{n-2}^*$ and so on until $r_1^*$. We name this method *backward maximization*. Hence, we obtained $r^*(x^n \| y^{n-1}) = \prod_{i=1}^n r_i^*$ which maximizes the directed information where $q(x^n|y^n)$ is fixed (i.e., $c_1(q)$).

Finally, we need to show that $r^*(x^n \| y^{n-1})$ that maximize the directed information for a fixed $q(x^n|y^n)$ is unique. Since there is a one-to-one correspondence [9, Lemma 3] between the causally conditioned PMF $\mathbf{r}$ and the set of its factors $\{r_i\}_{i=1}^n$, it suffices to show that the set $\{r_i\}_{i=1}^n$ that achieves $\max_{\{r_i\}_{i=1}^n} I(X^n \to Y^n)$ is unique. The $\{r_i\}_{i=1}^n$ were obtained from the KKT condition through the set of equalities (31)–(37) and therefore $r_i$ given by (37) is the unique solution to the KKT condition. Furthermore, since KKT condition are necessary condition, there exist no other $r_i$ that maximize (29), and therefore, we conclude that $r^*(x^n \| y^{n-1})$ is unique. ■

Having Lemmas 2–5, we can now state and prove our main theorem.

*Theorem 1:* For a fixed channel, $p(y^n \| x^n)$, there exists an alternating maximization procedure, such as $I_L$ in Algorithm 1 to compute

$$C_n = \frac{1}{n} \max_{p(x^n \| y^{n-1})} I(X^n \to Y^n).$$

*Proof:* To prove Theorem 1, we first have to show the existence of a double maximization problem, i.e., an equivalent problem where we maximize over two variables instead of one, and this was shown in Lemma 3. Now, in order for the alternating maximization procedure to work on this optimization problem, we need to show that the conditions given in Lemma 1 hold here, and this was shown in Lemmas 2, 4, and 5. Thus, we have an algorithm for calculating

$$C_n = \frac{1}{n} \max_{r(x^n \| y^{n-1})} I(X^n \to Y^n)$$

that is equal to $\lim_{k\to\infty} I_L(k)$, where $I_L(k)$ is the value of $I_L$ in the $k$th iteration as in Algorithm 1. Hence, the theorem is proven. ■

Our last step in proving the convergence of Algorithm 1 is to show why $I_U$ is a tight upper bound. For that reason, we state the following theorem.

*Theorem 2:* For the value of

$$C_n = \frac{1}{n} \max_{p(x^n\|y^{n-1})} I(X^n \to Y^n)$$

the inequality

$$C_n \le I_U \tag{39}$$

where

$$I_U = \frac{1}{n} \min_r \max_{x_1} \sum_{y_1} \max_{x_2} \cdots \max_{x_n} \sum_{y_n} p(y^n\|x^n) \cdot \log \frac{p(y^n\|x^n)}{\sum_{x'^n} p(y^n\|x'^n) \cdot r(x'^n\|y^{n-1})}$$

holds. Furthermore, if $r(x^n\|y^{n-1})$ achieves the maximum in the $C_n$ expression, then we have an equality in (39).

The proof is given in Appendix B for the general case of delay $d$. However, we omit the proof of the upper bound for the case where the feedback is a deterministic function of the delayed output, as described in Appendix A, since it is similar to the one in Appendix B.

## V. Numerical Examples for Calculating Feedback Channels Capacities

In this section, we present some examples of Algorithm 1 performance over various channels. We start with a memoryless channel to see whether feedback improves the capacity of such channels, and continue with specific FSCs such as the Trapdoor channel and the Ising channel. Since Algorithm 1 is applicable on FSCs, we now describe the class of such channels and their properties. Gallager [27] defined the FSC as one in which the influence of the previous input and output sequence, up to a given point, may be summarized using a *state* with finite cardinality. The FSC is stationary and characterized by the conditional PMF $p(y_i, s_i|x_i, s_{i-1})$ that satisfies

$$p(y_i, s_i|x^i, y^{i-1}, s^{i-1}) = p(y_i, s_i|x_i, s_{i-1})$$

and the initial state $p(s_0)$. Since both channels are symmetrical with regard to the initial state, it suffices to choose one instead of minimizing over it.

The causal conditioning probability of the output given the input is defined by

$$p(y^n\|x^n, s_0) = \sum_{s^n} \prod_{i=1}^{n} p(y_i, s_i|x_i, s_{i-1})$$

and

$$p(y^n\|x^n) = \sum_{s_0} p(y^n\|x^n, s_0) p(s_0).$$

Note that a memoryless channel, i.e., a channel where the output at any given time is dependent on the input at that time alone is an FSC with one state.

It was shown in [9] that the capacity $C$ of an FSC with feedback is bounded between

$$\underline{C}_n \le C \le \overline{C}_n \tag{40}$$

where

$$\overline{C}_n = \frac{1}{n} \max_{p(x^n\|y^{n-1})} \max_{s_0} I(X^n \to Y^n|s_0) + \frac{\log|\mathcal{S}|}{n} \tag{41}$$

$$\underline{C}_n = \frac{1}{n} \max_{p(x^n\|y^{n-1})} \min_{s_0} I(X^n \to Y^n|s_0) - \frac{\log|\mathcal{S}|}{n}. \tag{42}$$

If we require that the probability of error tends to zero for every initial state $s_0$, then

$$C = \lim_{n\to\infty} \underline{C}.$$

Since these bounds are obtained via maximization of the directed information, we can calculate them using Algorithm 1 as presented in Section III, thus estimating the capacity.

Our first example shows the convergence of Algorithm 1 to the analytical capacity of a memoryless channel.

### A. Binary Symmetric Channel (BSC)

Consider a memoryless BSC with a transition probability of $p = 0.3$ as in Fig. 2. The capacity of this BSC (without feedback) is known to be $C = 1 - H(0.3) = 0.1187$ bits/channel-use. In Fig. 3, we present the directed information upper $I_U$ and lower $I_L$ bounds as a function of the iteration (as given in Algorithm 1) and compare it to the capacity that is known analytically. Shannon showed [28] that for memoryless channels, feedback does not increase the capacity. Thus, we can expect the numerical solution given in Algorithm 1 to achieve the same value as in the no-feedback case. We can see that as the iterations number increases, the algorithm approaches the true value and converges. Furthermore, the causally conditioned probability, $r(x^n\|y^{n-1})$, that Algorithm 1 produces is not uniform, but each marginal PMF $r(x_i)$ is uniform. We remind that since this is a memoryless channel, we can achieve the capacity using a uniform distribution of $\mathbf{r}$. This does not imply that we achieved only a local optimum distribution point, because the directed information is not strictly concave in $\mathbf{r}$. The optimum $\mathbf{r}$ found by Algorithm 1 depends on the initial point. Indeed, if we set the initial $\mathbf{r}$ to be uniform, the algorithm does not require more than the first iteration and the optimum distribution remains uniform.

### B. Trapdoor Channel

*1) Trapdoor Channel With 2 States:* The trapdoor channel was introduced by Blackwell in 1961 [29] and later on by Ash [30]. One can look at this channel as follows: consider a binary channel modulated by a box that contains a single bit referred to as the state. In every step, an input bit is fed to the channel, which then transmits either that bit or the one already contained in the box, each with probability $\frac{1}{2}$. The bit that was not transmitted remains in the box for future steps as the state of the channel.

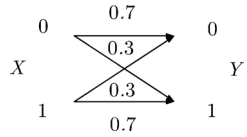


Fig. 2. BSC.

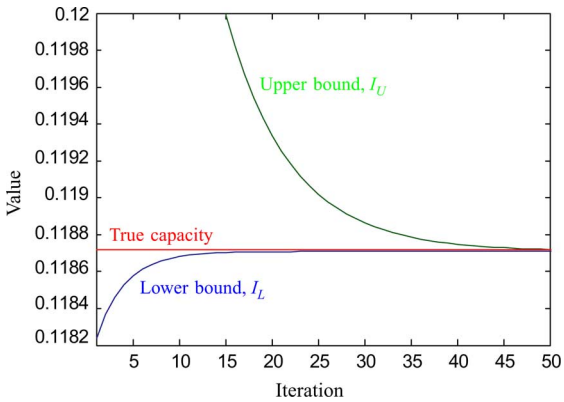


Fig. 3. Performance of Algorithm 1 over BSC(0.3). The lower and upper lines are the bounds in each iteration in Algorithm 1, whereas the horizontal line is the analytical calculation of the capacity.

The state, thus, is the bit in the box, and since it can be "0" or "1," we conclude that $|\mathcal{S}| = 2$ or $\log|\mathcal{S}| = 1$.

In order to use Algorithm 1, we first have to calculate the channel probability, $p(y^n\|x^n, s_0)$. For that purpose, we find $p(y_i|x^i, y^{i-1}, s_0)$ analytically. Note that $p(y_i|x^i, y^{i-1}, s_0) = p(y_i|x_i, s_{i-1})$. Thus, first we find the deterministic function for $s_{i-1}$ given the past input, output, and initial state, i.e., $(x^{i-1}, y^{i-1}, s_0)$, and then the function for $p(y_i|x^i, y^{i-1}, s_0) = p(y_i|x_i, s_{i-1})$. An examination of the truth table in Table II yields the following formula for $s_{i-1}$, where $\oplus$ is for addition modulo 2:

$$\begin{aligned} s_{i-1} &= x_{i-1} \oplus y_{i-1} \oplus s_{i-2} \\ &\overset{m=i-1}{=} \bigoplus_{m=1} (x_m \oplus y_m) \oplus s_0. \end{aligned}$$

Note that in Table II, $(0, 0, 1)$ and $(1, 1, 0)$ are not possible since the output is neither the input bit nor the bit in the box; thus, we may assign to $s_{i-1}$ whatever value we choose, in order to simplify the formula. As for the conditional probability, $p(y_i|x^i, y^{i-1}, s_0)$, we assume that $s_0 = 0$, and because of the channel's symmetry, the outcome for $s_0 = 1$ is easily calculated. Looking at Table III, we can see that the formula for $p(y_i|x^i, y^{i-1}, s_0 = 0)$ is given by

$$\begin{aligned} p(y_i|x^i, y^{i-1}, s_0 = 0) = \frac{1}{2}(x_i \oplus s_{i-1}) \\ + \overline{(x_i \oplus s_{i-1})} \wedge \overline{(x_i \oplus y_i)} \end{aligned}$$

where we know that $s_{i-1}$ is a function of $(x^{i-1},\ y^{i-1},\ s_0)$. The overline denotes a logical NOT, and $\wedge$ denotes logical AND.

Now that we have $p(y^n\|x^n, s_0 = 0)$, we use Algorithm 1 for estimating the capacity of the channel as we run the algorithm

TABLE II
$s_{i-i}$ AS A FUNCTION OF $x_{i-1}$, $s_{i-2}$ AND $y_{i-1}$

| $x_{i-1}$ | $s_{i-2}$ | $y_{i-1}$ | $s_{i-1}$ |
|---|---|---|---|
| 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | $\phi$ |
| 0 | 1 | 0 | 1 |
| 0 | 1 | 1 | 0 |
| 1 | 0 | 0 | 1 |
| 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | $\phi$ |
| 1 | 1 | 1 | 1 |

TABLE III
$p(y_i|s_{i-1}, x_i)$

| $x_i$ | $s_{i-1}$ | $y_i$ | $p(y_i\|x^i, y^{i-1}, s_0 = 0)$ |
|---|---|---|---|
| 0 | 0 | 0 | 1 |
| 0 | 0 | 1 | 0 |
| 0 | 1 | 0 | 0.5 |
| 0 | 1 | 1 | 0.5 |
| 1 | 0 | 0 | 0.5 |
| 1 | 0 | 1 | 0.5 |
| 1 | 1 | 0 | 0 |
| 1 | 1 | 1 | 1 |

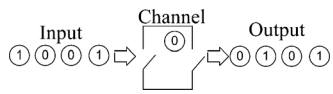


Fig. 4. Trapdoor channel [30].

to find the upper and lower bounds for every $n \in \{1, \ldots, 12\}$, where

$$\overline{C}_n = \max_{s_0} \max_{r(x^n\|y^{n-1})} \frac{1}{n} I(X^n \to Y^n|s_0) + \frac{1}{n} \tag{43}$$

$$\underline{C}_n = \max_{r(x^n\|y^{n-1})} \min_{s_0} \frac{1}{n} I(X^n \to Y^n|s_0) - \frac{1}{n}. \tag{44}$$

Note that (43) is calculated via Algorithm 1 and $s_0 = 0$ due to the channel's symmetry. However, calculating (44) is more difficult, since we have to maximize over all the probabilities $r(x^n\|y^{n-1})$, and at the same time minimize over the initial state. Hence, we use another lower bound denoted by $\underline{C}^*$, for which $r(x^n\|y^{n-1})$ is fixed and is the one that achieves the maximum at (43), and we only minimize over $s_0$. Clearly, $\underline{C}^* \leq \underline{C}$. Fig. 5 presents the capacity estimation, and the upper and lower bound, as a function of the block length $n$. In [22], the capacity of the Trapdoor channel is calculated analytically, and given by

$$C = \lim_{n\to\infty} C_n = \log\left(\frac{1+\sqrt{5}}{2}\right) \approx 0.69424191. \tag{45}$$

We see from the simulation that the upper and lower bounds of the capacity approach the limit in (45), and the estimated capacity at block length $n = 12$ is $C_{12} = 0.6706533$.

*2) Directed Information Rate as a Different Estimator for the Capacity:* We now consider an estimator to the feedback capacity of an FSC by calculating $(n+1)C_{n+1} - nC_n$, named the "directed information rate estimator." The justification for this estimator is based on the following lemma.

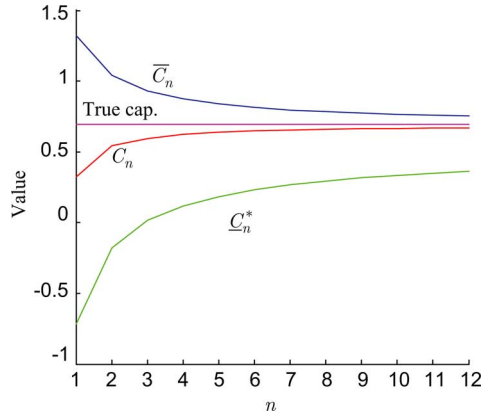


Fig. 5. Plot of $\overline{C}_n$, $C_n$, $\underline{C}_n^*$, and the true capacity of the Trapdoor channel with two states and feedback with delay 1.

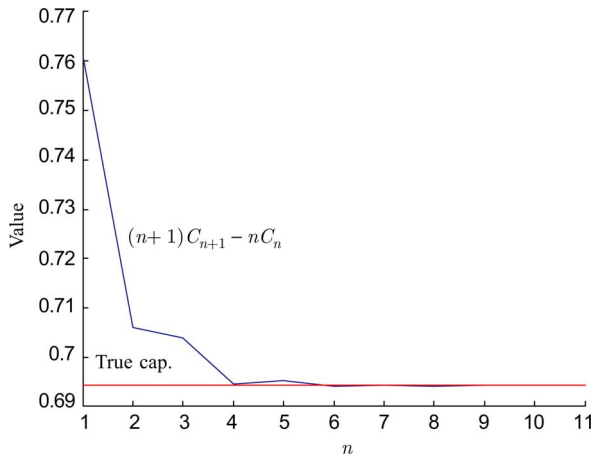


Fig. 6. Upper line is $(n+1)C_{n+1}-nC_n$ calculated using Algorithm 1 and the horizontal line is the analytical calculation, for the Trapdoor channel with two states and feedback with delay 1.

*Lemma 6:* Let

$$a_n = I(X^n \to Y^n) - I(X^{n-1} \to Y^{n-1}).$$

If the sequence $\{a_n\}$ converges, then

$$\lim a_n = \lim \frac{1}{n} I(X^n \to Y^n).$$

*Proof:* The proof is based on the Cesaro mean (also called Cesaro averages) property of a sequence

$$\begin{aligned}\lim a_n &\overset{(a)}{=} \lim \frac{1}{n}\sum_{i=1}^{n} a_n \\ &= \lim \frac{1}{n}\left(I(X^n \to Y^n) - I(X^1 \to Y^1)\right) \\ &= \lim \frac{1}{n} I(X^n \to Y^n)\end{aligned}$$

where (a) follows from the Cesaro mean property. ■

Fig. 6 presents the directed information rate estimator using the aforementioned lemma, and its comparison to the true capacity. One can see that the convergence of $(n+1)C_{n+1}-nC_n$ is faster than of $C_n$ and the upper and lower bounds as seen in Fig. 5, and achieves the value 0.6942285 when we calculate the directed information rate estimator for $n = 11$.

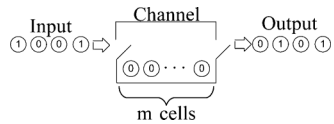


Fig. 7. Trapdoor channel with m states.

*3) m-State Trapdoor Channel:* We generalize the Trapdoor channel to an *m*-state one. In the previous example, we had $m = 2$ cells in the box, one for the state bit, and one for the input bit. One can consider the state to be the number of "1"s in the channel before a new input is inserted. We can expand this notation by letting the "box" contain more than two cells, as presented in Fig. 7. Here, the state at any given time will express the number of $1's$ that are in the box at that time and each cell has equal probability to be chosen for the output. In this case, $m$ cells in the box are equivalent to $m$ states of the channel. By that definition, we can see that the state $s_{i-1}$ as a function of past input, output, and the initial state is given by

$$\begin{aligned}s_{i-1} &= x_{i-1} + s_{i-2} - y_{i-1} \\ &= s_0 + \sum_{j=1}^{i-1}(x_j - y_j).\end{aligned}$$

Moreover, for calculating the channel probability, $p(y_i = 1|x^i, y^{i-1}, s_0)$, we add $s_{i-1}$ to $x_i$ and divide the sum by the number of cells, i.e.,

$$p(y_i = 1|x^i, y^{i-1}, s_0) = \frac{s_{i-1} + x_i}{m}.$$

Now that we have $p(y^n\|x^n, s_0)$, we use Algorithm 1 for calculating $C_n$ for every $n \in \{1, 2, \ldots, 12\}$. Fig. 8 presents the directed information rate estimator $(n+1)C_{n+1} - nC_n$ for the Trapdoor channel with $m = 3$ cells. Note that in Fig. 8 we achieve the value 0.5423984 in the 11th difference: thus, we can assume that the capacity of a three-state trapdoor channel is approximately 0.542.

*4) Influence of the Number of Cells on the Capacity:* To summarize the Trapdoor channel example, we examine the way the number of cells affects the capacity. The estimations we use are the directed information rate estimator and the upper bound, $\overline{C}_n = C_n + \frac{1}{n}$, with $n = 12$. In Fig. 9, we can see that as the number of cells increases, the capacity decreases.

### C. Ising Channel

The Ising model is a mathematical model of ferromagnetism in statistical mechanics. It was originally proposed by the physicist Wilhelm Lenz who gave it as a problem to his student Ernst Ising, after whom it is named. The model consists of discrete variables called spins that can be in one of two states. The spins are arranged in a lattice or graph, and each spin interacts only with its nearest neighbors.

The Ising channel is based on its physical model, and simulates intersymbol interference where the state of the channel at time $i$ is the current input and the output is determined by

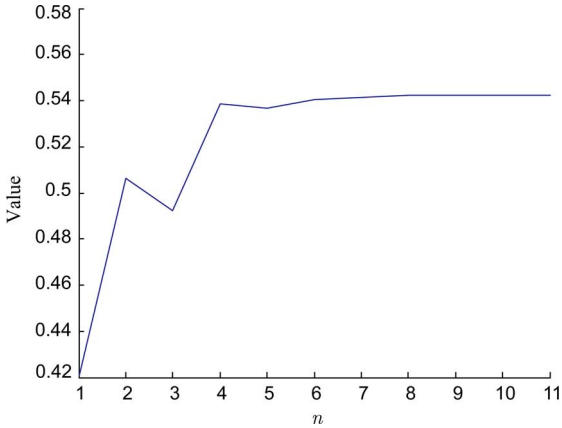


Fig. 8. Plot of $(n+1)C_{n+1} - nC_n$ for the Trapdoor channel with three cells and feedback with delay 1.

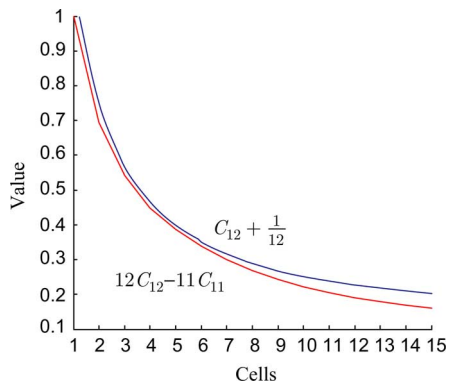


Fig. 9. Change of the upper bound, $\overline{C}_{12} = C_{12} + \frac{1}{12}$, and the estimator, $12C_{12} - 11C_{11}$, over the number of cells in the Trapdoor channel with feedback and delay 1.

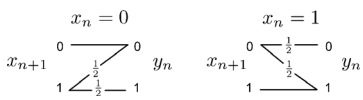


Fig. 10. Ising channel. [31].

the input at time $i+1$. The channel (without feedback) was introduced by Berger and Bonomi [31] and depicted in Fig. 10. In their paper, they proved the existence of bounds for the no-feedback case. In addition, they showed that the zero-error capacity without feedback is 0.5.

*1) Ising Channel With Delay $d = 2$:* We estimate the capacity of the Ising channel with feedback. Since the output at time $i$ is determined by the input at times $i,\ i+1$, we define the channel PMF as $p(y_0^{n-1} \| x^n, s_0)$. Therefore, the feedback at time $i$ must be the output at time $i-2$, since we cannot have $y_{i-1}$ before $x_{i-1}$ is sent. Thus, looking at the Ising channel with delay $d = 1$ is not a practical example and we did not examine it. We ran our algorithm on the Ising channel, with delayed feedback of $d = 2$; the results are presented in Fig. 11 at the top of the column. In Fig. 11(a), we obtain $C_{12} = 0.5459$, and in (b) we achieve $12C_{12} - 11C_{11} = 0.5563$ in the 11th difference.

*2) Effects the Delay has on the Capacity:* Here, we investigate how the delay influences the capacity. We do so by computing the directed information rate estimator of the Ising channel with blocks of length 12, over the feedback delay

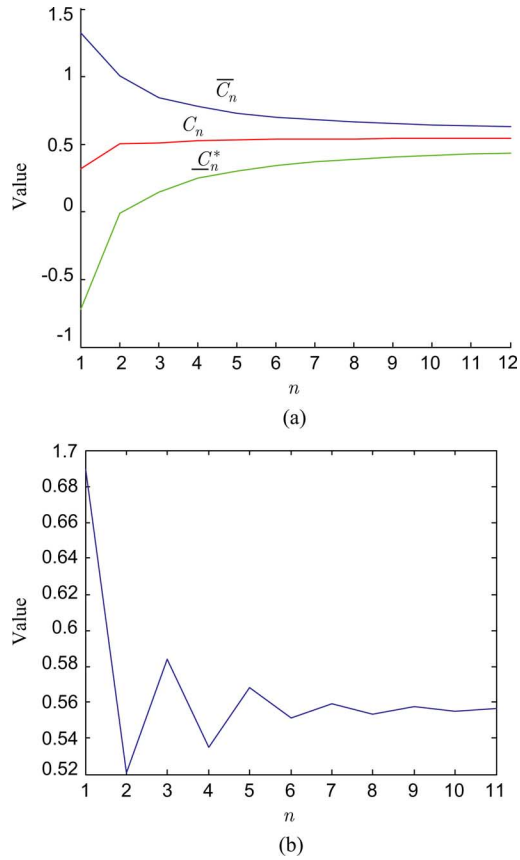


Fig. 11. Performance of Algorithm 1 on the Ising channel with feedback delay of $d = 2$. (a) $\overline{C}_n,\ C_n,\ \underline{C}_n^*$, and (b) $(n+1)C_{n+1} - nC_n$.

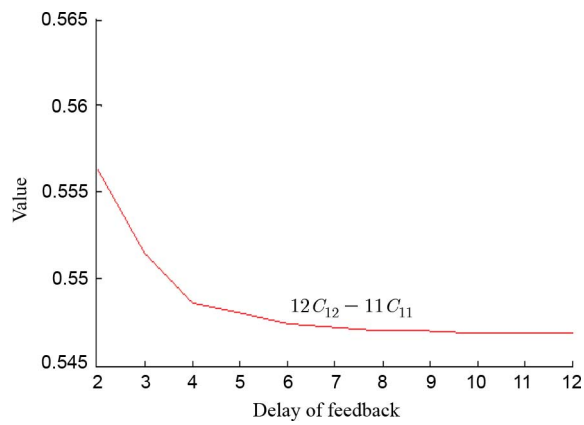


Fig. 12. Change of $12C_{12} - 11C_{11}$ over the delay of the feedback on the Ising channel.

$d = \{2, 3, \ldots, 12\}$. The formulas for estimating the capacity when the delay is bigger than 1 is given in Section III, (16), and (17). In Fig. 12, we can see that, as expected, the capacity decreases as the delay increases. This is due to the fact that we have less knowledge of the output to use.

## VI. Conclusion

In this paper, we generalized the classical BAA for maximizing the directed information over causal conditioning, i.e., we calculate

$$C_n = \frac{1}{n} \max_{p(x^n \| y^{n-1})} I(X^n \to Y^n).$$

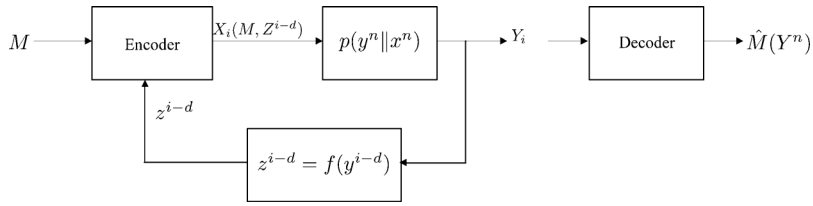


Fig. 13. Channel with delayed feedback as a function of the output.

The optimizing of the directed information is necessary for estimating the capacity of an FSC with feedback. As we attempted to solve this problem, we found that difficulties arose regarding the causal conditioning probability that we were trying to optimize over. We overcame this barrier by using an additional backward loop to find all components of the causally conditioned probability separately.

Another application of optimizing the directed information is to estimate the rate distortion function for source coding with feedforward as presented in [32]–[34]. In our future work [35], we address the source coding with feedforward problem, and derive bounds for stationary and ergodic sources. We also present and prove a BA-type algorithm for obtaining a numerical solution that computes these bounds.

## Appendix A
## General Case for Channel With Feedback That Is a Function of the Delayed Output

Here, we extend Algorithm 1, given in Section IV, for channels where the encoder has specific information about the delayed output. In this case, the input probability is given by $r(x^n\|z^{n-d})$, where $z_i = f(y_i)$ is the feedback, and $f$ is deterministic. In other words, we solve the optimization problem given by

$$\max_{r(x^n\|z^{n-d})} I(X^n \to Y^n).$$

The optimization problem is described in Fig. 13, shown in the following page.

The proof for this case is similar to that of Theorem 1, except for the steps that follow from Lemmas 4 and 5. Lemma 4 proves the existence of an argument $q(x^n|y^n)$ that maximizes the directed information where $r(x^n\|y^{n-1})$ is fixed. The modification of this lemma is presented here, where we find the argument $q(x^n|y^n)$ that maximizes the directed information where $r(x^n\|z^{n-d})$ is fixed; the proof is omitted. Therefore, the maximization over $q(x^n|y^n)$ where $r(x^n\|z^{n-d})$ is fixed is given by

$$q^*(x^n|y^n) = \frac{r(x^n\|z^{n-d})p(y^n\|x^n)}{\sum_{x^n} r(x^n\|z^{n-d})p(y^n\|x^n)}.$$

Lemma 5 proves the existence of an argument $r(x^n\|y^{n-1})$ that maximizes the directed information where $q(x^n|y^n)$ is fixed. We replace this lemma by Lemma 7.

*Lemma 7:* For fixed $q(x^n|y^n)$, there exists $c_1(q)$ that achieves $\max_{r(x^n\|z^{n-d})} I(X^n \to Y^n)$ and is given by

$$r(x^n\|z^{n-d}) = \prod_{i=1}^{n} r(x_i|x^{i-1}, z^{i-d})$$

where

$$r(x_i|x^{i-1}, z^{i-d}) = \frac{r'(x^i, z^{i-d})}{\sum_{x_i} r'(x^i, z^{i-d})} \tag{46}$$

and $r'(x^i, z^{i-d})$ is given in (47), shown at the bottom of the page.

*Proof:* We find the products of $r(x^n\|z^{n-d})$ that achieve the maximum of the directed information where $\mathbf{q}$ is fixed. For convenience, let us use for short: $r_i \stackrel{\triangle}{=} r(x_i|x^{i-1}, z^{i-d})$, and $p_i \stackrel{\triangle}{=} p(y_i|x^i, y^{i-1})$. As in Lemma 2, we can omit the proof that $I(X^n \to Y^n)$ is concave in $\{r_i\}$. Furthermore, the constraints of the optimization problem are affine, and we can use the Lagrange multipliers method with the KKT conditions for optimizing over each of the $r_i$s. We note that this proof is very similar to the one of Lemma 5, and hence presented here in less detail.

For every $i = 1$ to $n$, let us define the optimization problem as

$$\max_{r_i} I(X^n \to Y^n)$$

such that for every $x^{i-1}, z^{i-d}$

$$\sum_{x_i} r(x_i|x^{i-1}, z^{i-1}) = 1.$$

Hence, we define the Lagrangian as follows:

$$J = \sum_{x^n, y^n} \left( p(y^n\|x^n) \left( \prod_{i=1}^{n} r_i \right) \log \left( \frac{q(x^n|y^n)}{\prod_{j=1}^{n} r_j} \right) \right) + \sum_{x^{i-1}, z^{i-d}} \nu_{i,(x^{i-1}, z^{i-d})} \left( \sum_{x_i} r_i - 1 \right).$$

$$r'(x^i, z^{i-d}) = \prod_{x_{i+1}^n, y_{i-d+1}^n} \prod_{A_{i,d,z}} \left[ \frac{q(x^n|y^n)}{\prod_{j=i+1}^{n} r(x_j|x^{j-1}, z^{j-d})} \right]^{\frac{p(y^n\|x^n) \prod_{j=i+1}^{n} r(x_j|x^{j-1}, z^{j-d})}{\sum_{A_{i,d,z}} \prod_{j=1}^{i-d} p(y_j|x^j, y^{j-1})}} \tag{47}$$

Now, for every $i \in \{1 \dots n\}$ we find $r_i > 0$ s.t.

$$\begin{aligned}\frac{\partial J}{\partial r_i} &= \sum_{x_{i+1}^n, y_{i-d+1}^n, A_{i,d,z}} \left( p(y^n \| x^n) \left( \prod_{\substack{j=1 \\ j \neq i}}^{n} r_j \right) \cdot \right. \\ &\quad \left. \left[ \log \frac{q(x^n|y^n)}{\prod_{j=1}^n r_j} - 1 \right] \right) + \nu_{i,(x^{i-1}, z^{i-d})} \\ &= \sum_{A_{i,d,z}} \left[ \left( \prod_{j=1}^{i-1} r_j \right) \sum_{x_{i+1}^n, y_{i-d+1}^n} \left( p(y^n \| x^n) \left( \prod_{j=i+1}^{n} r_j \right) \cdot \right. \right. \\ &\quad \left. \left. \left[ \log \frac{q(x^n|y^n)}{\prod_{j=i+1}^n r_j} - \log \prod_{j=1}^{i-1} r_j - \log r_i - 1 \right] \right) \right] \\ &\quad + \nu_{i,(x^{i-1}, z^{i-d})} \\ &= 0\end{aligned}$$

where the set $A_{i,d,z} = \{y^{i-d} : z^{i-d} = f(y^{i-d})\}$ stands for all output sequences $y^{i-d}$ s.t. the function in the delay maps them to the same sequence, $z^{i-d}$, which is the feedback. It is easy to verify that the solution for $r_i = 0$ automatically satisfies the condition $\frac{\partial J}{\partial r_i} = 0$. Note that since $\prod_{j=1}^{i-1} r_j$ does not depend on $A_{i,d,z}$, we can take this term out of the sum. Furthermore, since $\nu_i$ is a function of $(x^{i-1}, z^{i-d})$, we can divide the whole equation by the aforementioned product, and get a new $\nu^*_{i,(x^{i-1}, z^{i-d})}$. Moreover, we can see that three of the expressions in the sum, i.e., $\{\log \prod_{j=1}^{i-1} r_j,\ \log r_i,\ 1\}$, do not depend on $(x_{i+1}^n, y_{i-d+1}^n)$, thus leaving their coefficient in the equation to be

$$\sum_{x_{i+1}^n, y_{i-d+1}^n, A_{i,d,z}} p(y^n \| x^n) \prod_{j=i+1}^{n} r_j = \sum_{A_{i,d,z}} \prod_{j=1}^{i-d} p_j.$$

Hence, we obtain

$$\begin{aligned}&\log \left[ \prod_{x_{i+1}^n, y_{i-d+1}^n, A_{i,d,z}} \left( \frac{q(x^n|y^n)}{\prod_{j=i+1}^n r_j} \right)^{\frac{p(y^n \| x^n) \prod_{j=i+1}^n r_j}{\sum_{A_{i,d,z}} \prod_{j=1}^{i-d} p_j}} \right] \\ &\quad - \log r_i - \log \nu^{**}_{i,(x^{i-1}, z^{i-d})} = 0\end{aligned}$$

where

$$\log \nu^{**}_{i,(x^{i-1}, z^{i-d})} = 1 + \log \prod_{j=1}^{i-1} r_j - \frac{\nu^*_{i,(x^{i-1}, z^{i-d})}}{\sum_{A_{i,d,z}} \prod_{j=1}^{i-d} p_j}.$$

Therefore, we are left with the expression

$$r(x_i | x^{i-1}, z^{i-d}) = \frac{r'(x^i, z^{i-d})}{\sum_{x_i} r'(x^i, z^{i-d})}$$

where $r'(x^i, z^{i-d})$ as is in (47).

As in Section IV, we can see that for all $i$, $r_i$ is dependent on $q(x^n|y^n)$ and $\{r_{i+1}, r_{i+2}, \dots, r_n\}$, and $r_n$ is a function of $q(x^n|y^n)$ alone. Thus, we use the *backward maximization* method. After calculating $r_i$ for all $i = 1, \dots, n$, we obtain $r(x^n \| z^{n-d}) = \prod_{i=1}^n r_i$ that maximizes the directed information where $q(x^n|y^n)$ is fixed, i.e., $c_1(q)$. ■

As mentioned, by replacing Lemmas 4 and 5 by those given here, we can follow the outline of Theorem 1 and conclude the existence of an alternating maximization procedure, i.e., we can compute

$$C_n = \frac{1}{n} \max_{r(x^n \| z^{n-d})} I(X^n \to Y^n)$$

that is equal to $\lim_{k \to \infty} I_L(k)$, where $I_L(k)$ is the value of $I_L$ in the $k$th iteration in the extended algorithm. One more step is required in order to prove the extension of Algorithm 1 to the case presented here; the existence of $I_U$. This part is presented in Appendix B.

## Appendix A
## Proof of Theorem 2

Here, we prove the existence of an upper bound $I_U$ that converges to $C_n$ from above simultaneously with the convergence of $I_L$ to it from below, as in Algorithm 1. To this purpose, we present and prove a few lemmas that assist in obtaining our main goal. We start with Lemma 8 that gives an inequality for the directed information. This inequality is used in Lemma 9 to prove the existence of our upper bound, which Lemma 10 proves to be tight. Theorem 2 combines Lemmas 9 and 10. Before we start, we present a new notation. Since the directed information is a function of $r(x^n \| y^{n-1})$ alone (note that $q(x^n|y^n)$ is a function of the joint $p(y^n \| x^n) r(x^n \| y^{n-1})$), we denote it by $I_r(X^n \to Y^n)$.

*Lemma 8:* Let $I_{r_1}(X^n \to Y^n)$ correspond to $r_1(x^n \| y^{n-d})$; then, for every $r_0(x^n \| y^{n-d})$

$$\begin{aligned}I_{r_1}(X^n \to Y^n) \leq \sum_{x^n, y^{n-d}} r_1(x^n \| y^{n-d}) \sum_{y_{n-d+1}^n} p(y^n \| x^n) \cdot \\ \log \frac{p(y^n \| x^n)}{\sum_{x'^n} p(y^n \| x'^n) \cdot r_0(x'^n \| y^{n-d})}.\end{aligned}$$

*Proof:* For any $r_1(x^n \| y^{n-d})$, $r_0(x^n \| y^{n-d})$, consider the chain of inequalities in (48) shown at the bottom of the next page, where in (a), $p_0(y^n)$ and $p_1(y^n)$ are the PMFs of $y^n$ that corresponds to $r_0(x'^n \| y^{n-d})$ and $r_1(x'^n \| y^{n-d})$, and (b) follows from the nonnegativity of the divergence. Thus, the lemma is proven. ■

Our next lemma uses the inequality in Lemma 8 to show the existence of the upper bound, which is the first step in proving Theorem 2.

*Lemma 9:* For every $r_0(x^n\|y^{n-d})$

$$C_n \le I_U$$

where

$$I_U = \frac{1}{n}\max_{x^d}\sum_{y_1}\max_{x_{d+1}}\sum_{y_2}\cdots\max_{x_n}\sum_{y_{n-d+1}^n} p(y^n\|x^n)\cdot \log\frac{p(y^n\|x^n)}{\sum_{x'^n} p(y^n\|x'^n)\cdot r_0(x'^n\|y^{n-d})}.$$

*Proof:* To prove this lemma, we first use lemma 8. For every $r_1(x^n\|y^{n-d})$, $r_0(x^n\|y^{n-d})$, consider the chain of inequalities in (49) shown at the bottom of the next page, where (a) follows Lemma 8, (b) follows from maximizing an expression over $x_n$, and (c) follows from the fact that the expression in the under-brace is a function of $x^{n-1}, y^{n-d}$, and we can take it out of the summation over $x_n$ and use $\sum_{x_n} r(x_n|x^{n-1}, y^{n-d}) = 1$. The rest of the steps are the same as (b) and (c), where we refer to a different $x_i$.

Since the aforementioned inequality is true for every $r_1(x^n\|y^{n-d})$, we replace it by $r_c(x^n\|y^{n-d})$ that achieves $C_n$; hence, the LHS of the inequality is $I_{r_c} = C_n$, and thus for every $r_0(x^n\|y^{n-d})$

$$C_n \le \frac{1}{n}\max_{x^d}\sum_{y_1}\max_{x_{d+1}}\sum_{y_2}\cdots\max_{x_n}\sum_{y_{n-d+1}^n} p(y^n\|x^n)\cdot \log\frac{p(y^n\|x^n)}{\sum_{x'^n} p(y^n\|x'^n)\cdot r_0(x'^n\|y^{n-d})}. \quad (50)$$

■

Equation (50) is also true for every $r_0(x^n\|y^{n-d})$ and hence for the minimum over all $r_0(x^n\|y^{n-d})$, we obtain

$$C_n \le \frac{1}{n}\min_{r_0}\max_{x^d}\sum_{y_1}\max_{x_{d+1}}\sum_{y_2}\cdots\max_{x_n}\sum_{y_{n-d+1}^n} p(y^n\|x^n)\cdot \log\frac{p(y^n\|x^n)}{\sum_{x'^n} p(y^n\|x'^n)\cdot r_0(x'^n\|y^{n-d})}.$$

In Lemma 9, we showed only half of the proof of the theorem, i.e., the existence of an upper bound. The next part of Theorem 2 is to show that the bound is tight.

*Lemma 10:* The upper bound in Lemma 9 is tight, and is obtained by $r_0(x^n\|y^{n-d})$ that achieves the capacity.

*Proof:* To prove this lemma, we need to show that this inequality is tight. For that purpose, we use the Lagrange multipliers method with the KKT conditions with respect to each $r(x_i|x^{i-1}, y^{i-d})$ separately. Thus, we can use the same arguments in Lemma 5 to apply the KKT conditions.

Hence, our optimization problem for every $i$ is

$$\max_{r_i} I(X^n \to Y^n)$$

such that

$$\forall x^{i-1}, y^{i-1} : \sum_{x_i} r_i = 1$$
$$\forall x^i, y^{i-1} : r_i >= 0.$$

Therefore, we define the Lagrangian as

$$J = \sum_{x^n,y^n} r(x^n\|y^{n-d})\cdot p(y^n\|x^n)\cdot \log\frac{p(y^n\|x^n)}{\sum_{x'^n} p(y^n\|x'^n)\cdot r(x'^n\|y^{n-d})} - \sum_{x^{i-1},y^{i-d}} \nu_{i,(x^{i-1},y^{i-d})}\Big(\sum_{x_i} r(x_i|x^{i-1}, y^{i-d}) - 1\Big) + \sum_{x^i,y^{i-d}} h_{i,(x^i,y^{i-d})} r(x_i|x^{i-1}, y^{i-d}).$$

We note that the parameter $h$ is introduced to take care of the inequalities in the optimization problem. Now, we differentiate over $r(x_i|x^{i-1}, y^{i-d})$ and obtain (without proof) (51) shown at

$$\begin{aligned}
&\sum_{x^n,y^{n-d}} r_1(x^n\|y^{n-d}) \sum_{y_{n-d+1}^n} p(y^n\|x^n)\log\frac{p(y^n\|x^n)}{\sum_{x'^n} p(y^n\|x'^n)\cdot r_0(x'^n\|y^{n-d})} - I_{r_1}(X^n \to Y^n)\\
&= \sum_{x^n,y^n} r_1(x^n\|y^{n-d})\cdot p(y^n\|x^n)\log\frac{p(y^n\|x^n)}{\sum_{x'^n} p(y^n\|x'^n)\cdot r_0(x'^n\|y^{n-d})}\\
&\quad - \sum_{x^n,y^n} r_1(x^n\|y^{n-d})\cdot p(y^n\|x^n)\log\frac{p(y^n\|x^n)}{\sum_{x'^n} p(y^n\|x'^n)\cdot r_1(x'^n\|y^{n-d})}\\
&= \sum_{x^n,y^n} r_1(x^n\|y^{n-d})\cdot p(y^n\|x^n)\log\frac{\sum_{x'^n} p(y^n\|x'^n)\cdot r_1(x'^n\|y^{n-d})}{\sum_{x'^n} p(y^n\|x'^n)\cdot r_0(x'^n\|y^{n-d})}\\
&= \sum_{y^n} p_1(y^n)\log\frac{p_1(y^n)}{p_0(y^n)}\\
&\overset{(a)}{=} D\left(p_1(y^n)\|p_0(y^n)\right)\\
&\overset{(b)}{\ge} 0.
\end{aligned} \quad (48)$$

the bottom of the page, where $\nu^*_{i,(x^{i-1},y^{i-d})}$ and $h^*_{i,(x^{i-1},y^{i-d})}$ are the modification of $\nu_{i,(x^{i-1},y^{i-d})}$ and $h_{i,(x^{i-1},y^{i-d})}$ after development of the equation, i.e.,

$$\begin{aligned}\nu^*_{i,(x^{i-1},y^{i-d})} &= \frac{\nu_{i,(x^{i-1},y^{i-d})}}{\prod_{j<i} r(x_j|x^{j-1},y^{j-d})}\\ h^*_{i,(x^{i-1},y^{i-d})} &= \frac{h_{i,(x^{i-1},y^{i-d})}}{\prod_{j<i} r(x_j|x^{j-1},y^{j-d})}\\ &\quad - \prod_{j<i-d+1} p(y_j|y^{j-1},x^{j-1}).\end{aligned}$$

We refer to $\nu^*$ as $\nu$ from now on. Setting $\frac{\partial J}{\partial r(x_i|x^{i-1},y^{i-d})} = 0$, we are left with two cases. For $r(x_i|x^{i-1},y^{i-d}) > 0$, the KKT conditions requires us to set $h^*_i = 0$ and (51) turns into

$$\begin{aligned}&\sum_{x_{i+1},y_{i-d+1}} r(x_{i+1}|x^i,y^{i-d+1}) \cdots \sum_{x_n,y_{n-d}} r(x_n|x^{n-1},y^{n-d}) \cdot\\ &\quad \sum_{y^n_{n-d+1}} p(y^n\|x^n) \log \frac{p(y^n\|x^n)}{\sum_{x'^n} p(y^n\|x'^n)\cdot r(x'^n\|y^{n-d})}\\ &\quad = \nu_{i,(x^{i-1},y^{i-d})}\end{aligned} \tag{52}$$

whereas for $r(x_i|x^{i-1},y^{i-d}) = 0$, we set $h^*_i > 0$ and the equality in (52) becomes an inequality.

$$\begin{aligned}I_{r_1}(X^n \to Y^n) &\overset{(a)}{\le} \sum_{x^n,y^{n-d}} r_1(x^n\|y^{n-d}) \sum_{y^n_{n-d+1}} p(y^n\|x^n) \log \frac{p(y^n\|x^n)}{\sum_{x'^n} p(y^n\|x'^n)\cdot r_0(x'^n\|y^{n-d})}\\
&\overset{(b)}{\le} \sum_{x^n,y^{n-d}} \prod_{i=1}^{n} r_1(x_i|x^{i-1},y^{i-d}) \underbrace{\max_{x_n} \sum_{y^n_{n-d+1}} p(y^n\|x^n) \log \frac{p(y^n\|x^n)}{\sum_{x'^n} p(y^n\|x'^n)\cdot r_0(x'^n\|y^{n-d})}}_{f(x^{n-1},y^{n-d})}\\
&\overset{(c)}{=} \sum_{x^{n-1},y^{n-d-1}} \prod_{i=1}^{n-1} r_1(x_i|x^{i-1},y^{i-d}) \underbrace{\sum_{y_{n-d}} \max_{x_n} \sum_{y^n_{n-d+1}} p(y^n\|x^n) \log \frac{p(y^n\|x^n)}{\sum_{x'^n} p(y^n\|x'^n)\cdot r_0(x'^n\|y^{n-d})}}_{f(x^{n-1},y^{n-d-1})}\\
&\le \sum_{x^{n-1},y^{n-d-1}} \prod_{i=1}^{n-1} r_1(x_i|x^{i-1},y^{i-d}) \underbrace{\max_{x_{n-1}} \sum_{y_{n-d}} \max_{x_n} \sum_{y^n_{n-d+1}} p(y^n\|x^n) \log \frac{p(y^n\|x^n)}{\sum_{x'^n} p(y^n\|x'^n)\cdot r_0(x'^n\|y^{n-d})}}_{f(x^{n-2},y^{n-d-1})}\\
&\vdots\\
&\le \sum_{x^d} \prod_{i=1}^{d} r_1(x_i|x^{i-1},y^{i-d}) \underbrace{\sum_{y_1} \max_{x_{d+1}} \sum_{y_2} \cdots \max_{x_n} \sum_{y^n_{n-d+1}} p(y^n\|x^n) \log \frac{p(y^n\|x^n)}{\sum_{x'^n} p(y^n\|x'^n)\cdot r_0(x'^n\|y^{n-d})}}_{f(x^d)}\\
&\le \underbrace{\sum_{x^d} \prod_{i=1}^{d} r_1(x_i|x^{i-1},y^{i-d})}_{=1} \underbrace{\max_{x^d} \sum_{y_1} \max_{x_{d+1}} \sum_{y_2} \cdots \max_{x_n} \sum_{y^n_{n-d+1}} p(y^n\|x^n) \log \frac{p(y^n\|x^n)}{\sum_{x'^n} p(y^n\|x'^n)\cdot r_0(x'^n\|y^{n-d})}}_{g\in\mathbb{R}}\\
&= \max_{x^d} \sum_{y_1} \max_{x_{d+1}} \sum_{y_2} \cdots \max_{x_n} \sum_{y^n_{n-d+1}} p(y^n\|x^n) \log \frac{p(y^n\|x^n)}{\sum_{x'^n} p(y^n\|x'^n)\cdot r_0(x'^n\|y^{n-d})}\end{aligned} \tag{49}$$

$$\begin{aligned}\frac{\partial J}{\partial r(x_i|x^{i-1},y^{i-d})} &= \sum_{x_{i+1},y_{i-d+1}} r(x_{i+1}|x^i,y^{i-d+1}) \cdots \sum_{x_n,y_{n-d}} r(x_n|x^{n-1},y^{n-d}) \cdot\\
&\quad \sum_{y^n_{n-d+1}} p(y^n\|x^n) \log \frac{p(y^n\|x^n)}{\sum_{x'^n} p(y^n\|x'^n)\cdot r(x'^n\|y^{n-d})} - \nu^*_{i,(x^{i-1},y^{i-d})} + h^*_{i,(x^{i-1},y^{i-d})}\end{aligned} \tag{51}$$

We now analyze our results for the case where $r(x_i|x^{i-1}, y^{i-d}) > 0$. First, we note that for $i = n$, we have from (52)

$$\sum_{y_{n-d+1}^n} p(y^n\|x^n) \log \frac{p(y^n\|x^n)}{\sum_{x'^n} p(y^n\|x'^n) \cdot r(x'^n\|y^{n-d})} = \nu_{n,(x^{n-1},y^{n-d})}. \tag{53}$$

Since $\nu_n$ does not depend on $x_n$, the LHS does not depend on $x_n$ as well, and thus is constant for every $x_n$. As a result, for $i = n-1$, we have

$$\begin{aligned}
&\nu_{n-1,(x^{n-2},y^{n-d-1})} \\
&= \sum_{x_n, y_{n-d}} r(x_n|x^{n-1}, y^{n-d}) \cdot \underbrace{\sum_{y_{n-d+1}^n} p(y^n\|x^n) \log \frac{p(y^n\|x^n)}{\sum_{x'^n} p(y^n\|x'^n) \cdot r(x'^n\|y^{n-d})}}_{(*)} \\
&\overset{(a)}{=} \sum_{y_{n-d}} \max_{x_n} \sum_{y_{n-d+1}^n} p(y^n\|x^n) \cdot \log \frac{p(y^n\|x^n)}{\sum_{x'^n} p(y^n\|x'^n) \cdot r(x'^n\|y^{n-d})}
\end{aligned} \tag{54}$$

where (a) follows from the fact that $(*)$ does not depend on $x_n$ as seen in (53), and the sum of $r_n$ over $x_n$ is 1.

Again, for $i = n-1$, $\nu_{n-1,(x^{n-2},y^{n-d-1})}$ does not depend on $x_{n-1}$, and hence, the RHS of (54) is constant for $x_{n-1}$. Thus, we can continue backward and obtain for $i = 1$, the chain of inequalities in (55) shown at the bottom of the page, where (a) follows from (54) and (b) follows from the fact that $\nu_{n-1}$ does not depend on $x_{n-1}$, (c) follows from the fact that $\nu_2$ does not depend on $x_2$, and (d) from the fact that $\nu_1$ does not depend on $x_1$. Using the aforementioned analysis, we find an expression for $C_n$ using $r(x^n\|y^{n-d})$ that achieves it. Note that in the following equations, we can assume that $r(x^n\|y^{n-d}) > 0$ since otherwise, for the specific $x^n, y^n$, the expression for $C_n$ will contribute 0 to the summation

$$\begin{aligned}
C_n &= \frac{1}{n} \sum_{x^n, y^n} r(x^n\|y^{n-d}) \cdot p(y^n\|x^n) \cdot \log \frac{p(y^n\|x^n)}{\sum_{x'^n} p(y^n\|x'^n) \cdot r(x'^n\|y^{n-d})} \cdot \\
&= \frac{1}{n} \sum_{x_1} r(x_1) \sum_{x_2} r(x_2|x_1) \cdots \sum_{x_d} r(x_d|x^{d-1}) \cdot \sum_{x_{d+1}, y_1} r(x_{d+1}|x^d, y_1) \cdots r(x_n|x^{n-1}, y^{n-d}) \\
&\quad \sum_{y_{n-d+1}^n} p(y^n\|x^n) \log \frac{p(y^n\|x^n)}{\sum_{x'^n} p(y^n\|x'^n) \cdot r(x'^n\|y^{n-d})} \\
&\overset{(a)}{=} \frac{1}{n} \sum_{x_1} r(x_1) \max_{x^d} \sum_{y_1} \max_{x_{d+1}} \cdots \max_{x_n} \sum_{y_{n-d+1}^n} p(y^n\|x^n) \cdot \log \frac{p(y^n\|x^n)}{\sum_{x'^n} p(y^n\|x'^n) \cdot r(x'^n\|y^{n-d})} \\
&= \frac{1}{n} \max_{x^d} \sum_{y_1} \max_{x_{d+1}} \cdots \max_{x_n} \sum_{y_{n-d+1}^n} p(y^n\|x^n) \cdot \log \frac{p(y^n\|x^n)}{\sum_{x'^n} p(y^n\|x'^n) \cdot r(x'^n\|y^{n-d})}
\end{aligned}$$

$$\begin{aligned}
\nu_1 &= \sum_{x_2} r(x_2|x_1) \cdots \sum_{x_d} r(x_d|x^{d-1}) \sum_{x_{d+1}, y_1} r(x_{d+1}|x^d, y_1) \cdots \sum_{x_n, y_{n-d}} r(x_n|x^{n-1}, y^{n-d}) \cdot \\
&\quad \sum_{y_{n-d+1}^n} p(y^n\|x^n) \log \frac{p(y^n\|x^n)}{\sum_{x'^n} p(y^n\|x'^n) \cdot r(x'^n\|y^{n-d})} \\
&\overset{(a)}{=} \sum_{x_2} r(x_2|x_1) \cdots \sum_{x_d} r(x_d|x^{d-1}) \sum_{x_{d+1}, y_1} r(x_{d+1}|x^d, y_1) \cdots \sum_{x_{n-1}, y_{n-d-1}} r(x_{n-1}|x^{n-2}, y^{n-d-1}) \nu_{n-1,(x^{n-2},y^{n-d-1})} \\
&\overset{(b)}{=} \sum_{x_2} r(x_2|x_1) \cdots \sum_{x_d} r(x_d|x^{d-1}) \sum_{x_{d+1}, y_1} r(x_{d+1}|x^d, y_1) \cdots \sum_{x_{n-1}, y_{n-d-1}} r(x_{n-1}|x^{n-2}, y^{n-d-1}) \max_{x_{n-2}} \nu_{n-1,(x^{n-2},y^{n-d-1})} \\
&= \sum_{x_2} r(x_2|x_1) \cdots \sum_{x_d} r(x_d|x^{d-1}) \sum_{x_{d+1}, y_1} r(x_{d+1}|x^d, y_1) \cdots \sum_{x_{n-2}, y_{n-d-2}} r(x_{n-2}|x^{n-3}, y^{n-d-2}) \nu_{n-2,(x^{n-3},y^{n-d-2})} \\
&\vdots \\
&= \sum_{x_2} r(x_2|x_1) \underbrace{\max_{x_3^d} \sum_{y_1} \max_{x_{d+1}} \cdots \max_{x_n} \sum_{y_{n-d+1}^n} p(y^n\|x^n) \log \frac{p(y^n\|x^n)}{\sum_{x'^n} p(y^n\|x'^n) \cdot r(x'^n\|y^{n-d})}}_{\nu_{2,(x_1)}} \\
&\overset{(c)}{=} \underbrace{\max_{x_2^d} \sum_{y_1} \max_{x_{d+1}} \cdots \max_{x_n} \sum_{y_{n-d+1}^n} p(y^n\|x^n) \log \frac{p(y^n\|x^n)}{\sum_{x'^n} p(y^n\|x'^n) \cdot r(x'^n\|y^{n-d})}}_{\nu_1} \\
&\overset{(d)}{=} \max_{x^d} \sum_{y_1} \max_{x_{d+1}} \cdots \max_{x_n} \sum_{y_{n-d+1}^n} p(y^n\|x^n) \log \frac{p(y^n\|x^n)}{r \sum_{x'^n} p(y^n\|x'^n) \cdot r(x'^n\|y^{n-d})}
\end{aligned} \tag{55}$$

where (a) is due to the aforementioned analysis for $i = 1$. We showed that the upper bound is tight, and thus the lemma is proven. ■

Now, we combine both lemmas to conclude our main theorem.

*Proof of Theorem 2:* As showed in Lemma 9, there exists an upper bound for $C_n$. Lemma 10 showed that this upper bound is tight, when using the PMF $r(x^n \| y^{n-d})$ that achieves $C_n$. Thus, the theorem is proven. ■

*Generalization of Theorem 2:* We generalize Theorem 2 for the case where the feedback is a delayed function of the output (as presented in Appendix A). We recall that the optimization problem for this model is

$$\max_{r(x^n \| z^{n-d})} I(X^n \to Y^n).$$

While solving this optimization problem, we defined the following set: $A_{i,d,z} = \{y^{i-d} : z^{i-d} = f(y^{i-d})\}$, namely, all output sequences $y^{i-d}$ s.t. the function in the delay sends them to the same sequence $z^{i-d}$. We use this notation for the upper bound. In that case, the upper bound is of the form

$$I_U = \frac{1}{n} \max_{x^d} \sum_{z_1} \max_{x_{d+1}} \cdots \sum_{z_{n-d}} \max_{x_n} \sum_{A_{n,d,z}} \sum_{y_{n-d+1}^n} p(y^n \| x^n) \cdot \log \frac{p(y^n \| x^n)}{\sum_{x'^n} p(y^n \| x'^n) \cdot r(x'^n \| z^{n-d})}.$$

The proof for this upper bound is omitted due to its similarity to the case where $z_i = y_i$ for all $i$, i.e., Theorem 2. Moreover, one can see that this is a generalization, since if indeed $z_i = y_i$, then $A_{n,d,z}$ has only one sequence, $y_{n-d}$, and the equation for $I_U$ coincides with the one in Theorem 2.

## ACKNOWLEDGMENT

The authors would like to thank the Associate Editor and anonymous referees for their comments, which significantly helped to improve the content and the organization of the paper. The authors gratefully acknowledge Yossef Steinberg for very helpful discussions.

## REFERENCES

[1] C. E. Shannon, "A mathematical theory of communication," *Bell Syst. Tech. J.*, vol. 27, pp. 623–656, 1948.
[2] R. Blahut, "An algorithm for computing the capacity of arbitrary discrete memoryless channels," *IEEE Trans. Inf. Theory*, vol. 18, no. 1, pp. 14–20, Jan. 1972.
[3] S. Arimoto, "Computation of channel capacity and rate-distortion functions," *IEEE Trans. Inf. Theory*, vol. 18, no. 4, pp. 460–473, Jul. 1972.
[4] I. Csiszár and G. Tusnady, "Information geometry and alternating minimization procedures," *Statist. Decis.*, vol. 1, pp. 205–237, 1984.
[5] R. W. Yeung, *Information Theory and Network Coding*. New York: Springer-Verlag, 2008.
[6] J. Massey, "Causality, feedback and directed information," in *Proc. Int. Symp. Inf. Theory Appl.*, Waikiki, HI, Nov. 1990, pp. 303–305.
[7] G. Kramer, "Directed information for channels with feedback," Ph.D. dissertation, Swiss Federal Institute of Technology, Zurich, Switzerland, 1998.
[8] S. Tatikonda and S. Mitter, "The capacity of channels with feedback," *IEEE Trans. Inf. Theory*, vol. 55, no. 1, pp. 323–349, Jan. 2009.
[9] H. H. Permuter, T. Weissman, and A. J. Goldsmith, "Finite state channels with time-invariant deterministic feedback," *IEEE Trans. Inf. Theory*, vol. 55, no. 2, pp. 644–662, Feb. 2009.
[10] Y. H. Kim, "A coding theorem for a class of stationary channels with feedback," *IEEE Trans. Inf. Theory*, vol. 54, no. 4, pp. 1488–1499, Apr. 2008.
[11] G. Matz and P. Duhamel, "Information geometric formulation and interpretation of accelerator Blahut–Arimoto-type algorithms," in *Proc. IEEE Inf. Theory Workshop*, San Antonio, TX, Oct. 2004, pp. 66–70.
[12] M. Rezaeian and A. Grant, "Computation of total capacity for discrete memoryless multiple-access channels," *IEEE Trans. Inf. Theory*, vol. 50, no. 11, pp. 2779–2784, Nov. 2004.
[13] W. Yu, F. Dupuis, and F. Willems, "Arimoto–Blahut algorithms for computing channel capacity and rate-distortion with side information," presented at the Int. Symp. Inf. Theory, 2004.
[14] C. Heegard and A. A. El Gamal, "On the capacity of computer memory with defects," *IEEE Trans. Inf. Theory*, vol. 29, no. 5, pp. 731–739, Sep. 1983.
[15] G. Markavian, S. Egorov, and K. Pickavance, "A modified Blahut algorithm for decoding Reed Solomon codes beyond half the minimum distance," *IEEE Trans. Inf. Theory*, vol. 52, no. 12, pp. 2052–2056, Dec. 2004.
[16] J. Dauwels, "Numercal computation of the capacity of continuous memoryless channels," presented at the 26th Symp. Inf. Theory, 2005.
[17] D. Arnold, H.-A. Loeliger, P. O. Vontobel, and A. Kavcic, "Capacity of finite-state machine channels," *IEEE Trans. Inf. Theory*, vol. 54, no. 5, pp. 1887–1918, May 2008.
[18] O. Sumszyk and Y. Steinberg, "Information embedding with reversible stegotext," in *ISIT*, Novo mesto, Slovenia, Jul. 2009, pp. 2728–2732.
[19] U. Niesen, D. Shah, and G. W. Wornell, "Adaptive alternating minimization algorithms," *IEEE Trans. Inf. Theory*, vol. 55, no. 3, pp. 1423–1429, Mar. 2009.
[20] A. Kavcic, S. Yang, and S. Tatikonda, "Feedback capacity of finite-state machine channels," *IEEE Trans. Inf. Theory*, vol. 51, no. 3, pp. 799–810, Mar. 2005.
[21] J. Chen and T. Berger, "The capacity of finite-state Markov channels with feedback," *IEEE Trans. Inf. Theory*, vol. 51, no. 3, pp. 780–789, Mar. 2005.
[22] H. H. Permuter, P. Cuff, B. Van Roy, and T. Weissman, "Capacity of the trapdoor channel with feedback," *IEEE Trans. Inf. Theory*, vol. 54, no. 7, pp. 3150–3165, Jul. 2008.
[23] S. K. Gorantla and T. P. Coleman, "On reversible Markov chains and maximization of directed information," in *Proc. Int. Symp. Inf. Theory*, Jun. 2010, pp. 216–220.
[24] H. Marko, "The bidirectional communication theory—A generalization of information theory," *IEEE Trans. Commun.*, vol. COM-21, no. 12, pp. 1335–1351, Dec. 1973.
[25] T. M. Cover and J. A. Thomas, *Elements of Information Theory*, 2nd ed. New-York: Wiley, 2006.
[26] S. Boyd and L. Vandenberghe, *Convex Optimization*. New York: Cambridge Univ. Press, 2004.
[27] R. G. Gallager, *Information Theory and Reliable Communication*. New York: Wiley, 1968.
[28] C. E. Shannon, "The zero error capacity of a noisy channel," *IEEE Trans. Inf. Theory*, vol. IT-2, no. 3, pp. 8–19, Sep. 1956.
[29] D. Blackwell, Modern Mathematics for Engineer ser. II, 1961, pp. 183–193.
[30] R. Ash, *Information Theory*. New York: Wiley, 1965.
[31] T. Berger and F. Bonomi, "Capacity and zero-error capacity of Ising channels," *IEEE Trans. Inf. Theory*, vol. 36, no. 1, pp. 173–180, Jan. 1990.
[32] T. Weissman and N. Merhav, "On competitive prediction and its relation to rate-distortion theory," *IEEE Trans. Inf. Theory*, vol. 49, no. 12, pp. 3185–3194, Dec. 2003.
[33] R. Venkataramanan and S. S. Pradhan, "Source coding with feed-forward: Rate-distortion theorems and error exponents for a general source," *IEEE Trans. Inf. Theory*, vol. 53, no. 6, pp. 2154–2179, Jun. 2007.
[34] R. Venkataramanan and S. S. Pradhan, "On evaluating the rate-distortion function of sources with feed-forward and the capacity of channels with feedback," CoRR 2007, http://arxiv.org/abs/cs/0702009.
[35] H. Permuter and I. Naiss, in *Bounds on rate distortion with feed forward for stationary and ergodic sources*, Jul.–Aug. 2011, pp. 385–389.

**Iddo Naiss** received his B.Sc. (summa cum laude) degrees in Electrical and Computer Engineering and in Mathematics from the Ben-Gurion University, Israel, in 2010. He also received his M.Sc. degree on the fast track program for honor students in Electrical and Computer Engineering from the Ben-Gurion University, Israel, in 2012. Currently Iddo is working with Samsung-Israel on coding for flash memories.

**Haim H. Permuter** (M'08) received his B.Sc. (summa cum laude) and M.Sc. (summa cum laude) degree in Electrical and Computer Engineering from the Ben-Gurion University, Israel, in 1997 and 2003, respectively, and Ph.D. degrees in Electrical Engineering from Stanford University, California in 2008.

Between 1997 and 2004, he was an officer at a research and development unit of the Israeli Defense Forces. He is currently a senior lecturer at Ben-Gurion university.

Dr. Permuter is a recipient of the Fullbright Fellowship, the Stanford Graduate Fellowship (SGF), Allon Fellowship, Marie Curie Reintegration fellowship, and the 2009 U.S.-Israel Binational Science Foundation Bergmann Memorial Award.