Extension of the Blahut-Arimoto algorithm for maximizing directed information

Iddo Naiss and Haim Permuter

arXiv:1012.5071v2 [cs.IT] 27 Dec 2010

Abstract

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 do so, we apply the ideas from the regular Blahut-Arimoto algorithm, i.e., the alternating maximization procedure, onto our new problem. We provide both upper and lower bound sequences that converge to the optimum value. Our main insight in this paper is that in order to find the maximum of the directed information over causal conditioning probability mass function (PMF), one can use a backward index time maximization combined with the alternating maximization procedure. We give a detailed description of the algorithm, its complexity, the memory needed, and several numerical examples.

Index Terms

Alternating maximization procedure, Backwards index time maximization, Blahut-Arimoto algorithm, Causal conditioning, Channels with feedback, Directed information, Finite state channels, Ising Channel, Trapdoor channel.

I. INTRODUCTION

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

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

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 calculate the optimum value using the equality

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

Iddo Naiss and Haim Permuter are with the Department of Electrical and Computer Engineering, Ben-Gurion University of the Negev, Beer-Sheva, Israel. Emails: naiss@bgu.ac.il, haimp@bgu.ac.il.

i.e., we can maximize over $p(x)$ and $p(x|y)$, instead of just $p(x)$ alone. The maximization is then achieved using the alternating maximization procedure. The convergence of the alternating maximization procedure to the global maximum was proven by Csiszar and Tusnady [4], and later by Yeung [5].

In this paper, we find an efficient way to estimate the capacity of channels with feedback. It was shown by Massey [6], Kramer [7], Tatikonda and Mitter [8], Permuter, Weissman, and Goldsmith [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*, and $p(y^n||x^n)$ is a *causally conditioned* PMF (definitions in Section II) given by

$$p(y^n||x^n) = \prod_{i=1}^{n} p(y_i|y^{i-1}, x^i). \tag{2}$$

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 causal conditioned PMF is the result of multiplications of conditioned PMFs as seen in (2). While in the regular BAA we maximize over $p(x^n)$, and thus the constraints are simply $\sum_{x^n} p(x^n) = 1$ and $p(x^n) \geq 0$, in our extended problem we have no efficient way of writing all the constraints necessary for a causally conditioned PMF. In fact, we need n simple constraints, one for each product of $p(x^n||y^{n-1})$. Another difficulty is that although the equality

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

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 $p(x_i|x^{i-1}, y^{i-1})$ influence all terms $\{I(X_j; Y_j^n | X^{j-1}, Y^{j-1})\}_{j=i}^{n}$. A solution could be to maximize backwards 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 non-causal 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 $\{p(x_1), p(x_2|x_1, y_1), ..., p(x_n|x^{n-1}, y^{n-1}), p(x^n|y^n)\}$. Each time we maximize over a specific $p(x_i|x^{i-1}, y^{i-1})$ starting from $i = n$ and moving backwards 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 of the other extensions of this algorithm. In 2004, Matz and Duhamel [11] proposed two Blahut-Arimoto-type algorithms that often converge significantly faster than the standard Blahut-Arimoto algorithm, which relied on following the natural

gradient rather than maximizing per variable. During that year, Rezaeian and Grant [12] generalized the regular BAA for multiple access channels, and Dupuis, Yu, and Willems 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 El Gamal and Heegard [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 Egorov, Markavian, and Pickavance [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 the use of sequential Monte-Carlo integration methods known as the "particle filters". In 2008 Vontobel, Kavčić, Arnold, and Loeliger [17] extended the regular BAA to estimate the capacity of finite state channels 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 given the input.

Recently, few papers about the maximization of the directed information using control theory and dynamic programming were published. In [19], Yang, Kavcic and Tatikonda 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 [20] 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, Cuff, Van Roy and Weissman [21] maximized the directed information and found the capacity of the trapdoor channel with feedback. In [22], 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 the paper is organized as follows. In Section II we present the notations we use throughout the paper, and give the outline for the alternating maximization procedure as given by Yeung [5]. 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 angle 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 determining the stoping iteration 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, originally introduced by Massey [6] (who was inspired by Marko's work [23] 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 vector $(X_1, X_2, ... X_n)$. Usually we use the notation $X^n = X_1^n$ for short. Further, when writing a PMF we simply write $P_X(X = x) = p(x)$. Let us denote as $p(x^n||y^{n-d})$ the probability mass function (PMF) of X^n *causally conditioned* on Y^{n-d}, given by

$$p(x^n||y^{n-d}) \triangleq \prod_{i=1}^{n} p(x_i|x^{i-1}y^{i-d}). \tag{3}$$

Here we have to establish that when $d > n$, the vector $X^{n-d} = \emptyset$. Two straight forward properties of the causal conditioning PMF that we use throughout the paper are

$$\sum_{x_n} p(x^n||y^{n-d}) = p(x^{n-1}||y^{n-d-1}), \tag{4}$$

and

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

Another elementary property is the chain rule for directed information

$$p(x^n||y^{n-1})p(y^n||x^n) = p(x^n, y^n). \tag{6}$$

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

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

Moreover, the directed information from X^n to Y^n is given by

$$I(X^n \to Y^n) \triangleq H(Y^n) - H(Y^n||X^n). \tag{7}$$

It is possible to show, that we can write the directed information as such:

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

We refer to this form when 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

$$I(X^n \to Y^n) = \mathcal{I}(\mathbf{r}, \mathbf{q}) \tag{8}$$

when required. With these definitions, we follow the alternating maximization procedure given by Yeung [5] 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

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

We denote by $c_2(u_1) \in A_2$ the point that achieves $\sup_{u_2 \in A_2} f(u_1, u_2)$, and by $c_1(u_2) \in A_1$ the one that achieves $\sup_{u_1 \in A_1} f(u_1, u_2)$. The algorithm is defined 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 if the present 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 (Lemmas 9.4, 9.5 in [5], Convergence of the alternating maximization procedure) . 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, 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 $u_1 \in A_1$, $u_2 \in A_2$. Under these conditions, $\lim_{k \to \infty} f^k = f^*$.

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 memory calculation is also given.

A. The 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 the regular BAA has a structure similar to that of Algorithm 1, 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 Alg. 1.

Algorithm 1 Iterative algorithm for calculating $\max_{p(x^n||y^{n-1})} I(X^n \to Y^n)$, where $p(y^n||x^n)$ is fixed.

(a) Start from a random point $q(x^n|y^n)$. Usually we start from a uniform distribution, i.e., $q(x^n|y^n) = 2^{-n}$ for every (x^n, y^n)

(b) Starting from $i = n$, calculate $r(x_i|x^{i-1}, y^{i-1})$ using the formula

$$r(x_i|x^{i-1}, y^{i-1}) = \frac{r'(x^i, y^{i-1})}{\sum_{x_i} r'(x^i, y^{i-1})}, \tag{9}$$

where

$$r'(x^i, y^{i-1}) = \prod_{x_{i+1}^n, y_i^n} \left[\frac{q(x^n|y^n)}{\prod_{j=i+1}^n r(x_j|x^{j-1}, y^{j-1})} \right]^{p(y_i|x^i, y^{i-1}) \prod_{j=i+1}^n r(x_j|x^{j-1}, y^{j-1}) p(y_j|x^j, y^{j-1})}, \tag{10}$$

and do so backwards until $i = 1$.

(c) Once you have $r(x_i|x^{i-1}, y^{i-1})$ for all $i \in \{1, ..., n\}$, 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

$$q(x^n|y^n) = \frac{r(x^n||y^{n-1})p(y^n||x^n)}{\sum_{x^n} r(x^n||y^{n-1})p(y^n||x^n)}. \tag{11}$$

(e) Calculate $I_U - I_L$, where

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

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

(f) Return to (b) if $(I_U - I_L) \geq \epsilon$.

(g) $C_n = I_L$.

---

(1) *Regular BAA, i.e., $n = 1$.* For $n = 1$, the algorithm suggested here agrees with the original BAA, where instead of steps (b), (c) we have

$$r(x) = \frac{\prod_y q(x|y)^{p(y|x)}}{\sum_x \prod_y q(x|y)^{p(y|x)}}, \tag{12}$$

and step (d) is replaced by

$$q(x|y) = \frac{r(x)p(y|x)}{\sum_x r(x)p(y|x)}. \tag{13}$$

The bounds I_L, I_U agree with the regular BAA as well, and are of the form

$$I_L = \sum_{y,x} p(y|x)r(x) \log \frac{q(x|y)}{r(x)},$$

$$I_U = \max_x \sum_y p(y|x) \log \frac{p(y|x)}{\sum_{x'} p(y|x') \cdot r(x')}.$$

(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), where the feedback is the output with delay d. In that case, in step (b) we have

$$r'(x^i, y^{i-d}) = \prod_{x_{i+1}^n, y_{i-d+1}^n} \left[\frac{q(x^n|y^n)}{\prod_{j=i+1}^n r(x_j|x^{j-1}, y^{j-d})} \right]^{\prod_{j=i-d+1}^n p(y_j|x^j, y^{j-1}) \prod_{j=i+1}^n r(x_j|x^{j-1}, y^{j-d})}, \quad (14)$$

and step (d) will be replaced by

$$q(x^n|y^n) = \frac{r(x^n||y^{n-d})p(y^n||x^n)}{\sum_{x^n} r(x^n||y^{n-d})p(y^n||x^n)}. \quad (15)$$

The bounds I_L, I_U are of the form

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

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

(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 z^{n-d} is a deterministic function of the delayed output. The expression characterizes the capacity of channels with time-invariant feedback [9]. In that case, in step (b) we have

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

where we define the set $A_{i,d,z} \triangleq \{y^{i-d} : z^{i-d} = f(y^{i-d})\}$ as the set of output sequences that f transforms to z^{i-d}, and step (d) will be replaced by

$$q(x^n|y^n) = \frac{r(x^n||z^{n-d})p(y^n||x^n)}{\sum_{x^n} r(x^n||z^{n-d})p(y^n||x^n)}. \quad (17)$$

The bounds I_L, I_U are of the form

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

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

Note, that for $d = n$, the vector $z^{n-d} = \emptyset$, hence $r(x_i|x^{i-1}, z^{i-d}) = r(x_i|x^{i-1})$, and

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

Also note that when $f(y) = const$, $r(x^n||z^{n-d}) = r(x^n)$, $A_{i,d,z} = y^{i-d}$, and $\sum_{y^{i-d}} \prod_{j=1}^{i-d} p(y_j|x^j, y^{j-1}) = 1$. In each of the cases above ($d = n$ or $f(y) = const.$), 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 a different version of the regular BAA for channel capacity, where the maximization is done over all $r(x_i|x^{i-1})$ instead of over $r(x^n)$ at once. 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. This will be done in two parts, one for each step in the iteration.

(1) Complexity of computing $q(x^n|y^n)$ as given in (11). 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 (10), assuming that an exponent is a constant number of computations, i.e., $O(1)$. Simple computations will conclude 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 $i \in \{1..n\}$. 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: 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 have its complexity and achieve $O((|\mathcal{X}||\mathcal{Y}|))$ computations. However, if we want to compare it to BAA for channels with memory, we replace $X \Leftrightarrow X^n$, $Y \Leftrightarrow Y^n$ But, $|\mathcal{X}^n| = |\mathcal{X}|^n$ and so we obtain $O((|\mathcal{X}||\mathcal{Y}|)^n)$ computations. The memory needed for the algorithm is very much dependent on the manner in which one implements the algorithm. However, the obligatory memory needed is for q, p, and r and its products; 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.

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	$O((|\mathcal{X}||\mathcal{Y}|)^n)$	$(|\mathcal{X}||\mathcal{Y}|)^n$
$\max_{p(x^n||y^{n-1})} \left(\frac{1}{n}I(X^n \to Y^n)\right)$, Alg. 1	$O(n^2(|\mathcal{X}||\mathcal{Y}|)^n)$	$n(|\mathcal{X}||\mathcal{Y}|)^n$

IV. Derivation of Algorithm 1

In this section, we derive Algorithm 1 using the alternating maximization procedure, and conclude 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: an algorithm for calculating $\max I(X^n \rightarrow Y^n)$. 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 $q(x^n|y^n)$ combined, 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 Alg. 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

$$I(X^n \rightarrow Y^n) = \sum_{y^n, x^n} p(y^n||x^n) r(x^n||y^{n-1}) \log \frac{q(x^n|y^n)}{r(x^n||y^{n-1})} \tag{18}$$

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

Proof: First we need to show that the directed information can be written as above by using the causal conditioning chain rule.

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

Then we recall the log-sum inequality [24, Theorem 2.7.1] given by

$$\sum_{i=1}^{n} a_i \log \frac{a_i}{b_i} \geq \left(\sum_{i=1}^{n} a_i \right) \log \frac{\sum_{i=1}^{n} a_i}{\sum_{i=1}^{n} b_i}. \tag{19}$$

We define the sets

$$A_1 = \{r(x^n||y^{n-1}) : r(x^n||y^{n-1}) > 0 \text{ is a causally conditioned PMF}\},$$

$$A_2 = \{q(x^n|y^n) : q(x^n|y^n) \text{ is a conditioned PMF}\}, \tag{20}$$

as the sets over which we maximize. Now, for $(\mathbf{r}_1, \mathbf{q}_1), \ (\mathbf{r}_2, \mathbf{q}_2)$ in $A = A_1 \times A_2$ and $\lambda \in [0, 1]$, by using the

log-sum inequality given above we derive that

$$(\lambda r_1 + (1-\lambda)r_2)\log\frac{\lambda r_1 + (1-\lambda)r_2}{\lambda q_1 + (1-\lambda)q_2} \le \lambda r_1 \log\frac{r_1}{q_1} + (1-\lambda)r_2 \log\frac{r_2}{q_2}.$$

Taking the reciprocal of the logarithms yields

$$(\lambda r_1 + (1-\lambda)r_2)\log\frac{\lambda q_1 + (1-\lambda)q_2}{\lambda r_1 + (1-\lambda)r_2} \ge \lambda r_1 \log\frac{q_1}{r_1} + (1-\lambda)r_2 \log\frac{q_2}{r_2}.$$

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 (8), we obtain

$$\mathcal{I}(\lambda\mathbf{r}_1 + (1-\lambda)\mathbf{r}_2, \lambda\mathbf{q}_1 + (1-\lambda)\mathbf{q}_2 \ge \lambda\mathcal{I}(\mathbf{r}_1,\mathbf{q}_1) + (1-\lambda)\mathcal{I}(\mathbf{r}_2,\mathbf{q}_2).$$

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. Moreover, it is simple to verify that the sets A_1, A_2 are both convex, and we can conclude that all conditions in Lemma 1 hold for the directed information. ∎

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

$$\max_{r(x^n||y^{n-1})} I(X^n \to Y^n) = \max_{r(x^n||y^{n-1}),q(x^n|y^n)} I(X^n \to Y^n). \tag{21}$$

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 $c_2(r) = q^*(x^n|y^n)$ that achieves $\max_{q(x^n|y^n)} I(X^n \to Y^n)$, and 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 $\mathbf{r} = r(x^n||y^{n-1})$

$$\mathcal{I}(\mathbf{r},\mathbf{q}^*) - \mathcal{I}(\mathbf{r},\mathbf{q})$$
$$= \sum_{x^n,y^n} r(x^n||y^{n-1})p(y^n||x^n)\log\frac{q^*(x^n|y^n)}{r(x^n||y^{n-1})} - \sum_{x^n,y^n} r(x^n||y^{n-1})p(y^n||x^n)\log\frac{q(x^n|y^n)}{r(x^n||y^{n-1})}$$
$$= \sum_{x^n,y^n} r(x^n||y^{n-1})p(y^n||x^n)\log\frac{q^*(x^n|y^n)}{q(x^n|y^n)}$$
$$= D\left(r(x^n||y^{n-1})p(y^n||x^n) \,\|\, q(x^n|y^n)\sum_{x^n} r(x^n||y^{n-1})p(y^n||x^n)\right)$$
$$\overset{(a)}{\ge} 0,$$

where (a) follows from the non-negativity of the divergence. ∎

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 $c_1(q) = r^*(x^n||y^{n-1})$ 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

$$
r(x_i|x^{i-1}, y^{i-1}) = \frac{r'(x^i, y^{i-1})}{\sum_{x^i} r'(x^i, y^{i-1})}, \tag{22}
$$

and

$$
r'(x^i, y^{i-1}) = \prod_{x_{i+1}^n, y_i^n} \left[\frac{q(x^n|y^n)}{\prod_{j=i+1}^{n} r(x_j|x^{j-1}, y^{j-1})} \right]^{\prod_{j=i}^{n} p(y_j|x^j, y^{j-1}) \prod_{j=i+1}^{n} r(x_j|x^{j-1}, y^{j-1})}. \tag{23}
$$

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 each of them. For convenience, let us use for short: $r_i \triangleq r(x_i|x^{i-1}, y^{i-1})$, and $p_i \triangleq p(y_i|x^i, y^{i-1})$. Since in Lemma 2 we showed that $I(X^n \to Y^n)$ is concave in $\{\mathbf{r}, \mathbf{q}\}$ and the constraints of the optimization problem are affine, we can use the Lagrange multipliers method with the Karush-Kuhn-Tucker conditions [25, Ch. 5.3.3]. We define the Lagrangian as:

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

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

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

$$= 0.$$

Note that since ν_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 $\nu_{i,(x^{i-1},y^{i-1})}^*$.

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^n), thus leaving their coefficient in the equation to be

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

Hence we obtain:

$$\log \left[\prod_{x_{i+1}^n, y_i^n} \left(\frac{q(x^n | y^n)}{\prod_{j=i+1}^n r_j} \right)^{\frac{p(y^n \| x^n) \prod_{j=i+1}^n r_j}{\prod_{j=1}^{i-1} p_j}} \right] - \log r_i - \log \nu_{i,(x^{i-1},y^{i-1})}^{**} = 0,$$

where

$$\log \nu_{i,(x^{i-1},y^{i-1})}^{**} = \prod_{j=1}^{i-1} p_j \left(1 + \log \prod_{j=1}^{i-1} r_j \right) - \nu_{i,(x^{i-1},y^{i-1})}^*.$$

Finally, we are left with the expression:

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

where

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

$$= \prod_{x_{i+1}^n, y_i^n} \left[\frac{q(x^n | y^n)}{\prod_{j=i+1}^n r(x_j | x^{j-1}, y^{j-1})} \right]^{\prod_{j=i}^n p(y_j | x^j, y^{j-1}) \prod_{j=i+1}^n r(x_j | x^{j-1}, y^{j-1})}. \tag{24}$$

We can see that for every i, r_i depends on $q(x^n | y^n)$ and $\{r_{i+1}, r_{i+2}, ..., r_n\}$, and r_n is a function of $q(x^n | y^n)$ alone. Therefore, we can place r_n in the function we have for r_{n-1}, thus making r_{n-1} depend on $q(x^n | y^n)$ alone as well. Now we do the same for r_{n-2} and so on until for all i, r_i is dependent on $q(x^n | y^n)$ alone. We name this method *Backwards maximization*. Finally, we obtain $r(x^n \| y^{n-1}) = \prod_{i=1}^n r_i$ that maximizes the directed information where $q(x^n | y^n)$ is fixed, i.e., $c_1(q)$, and the lemma is proven. ∎

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 Alg. 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 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 Lemma 2, 4 and 5. Thus, we have an algorithm for calculating

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

that is equal to $\lim_{k \rightarrow \infty} I_L(k)$, where $I_L(k)$ is the value of I_L in the kth iteration as in Alg. 1. Hence, the theorem is proven. ∎

Our last step in proving the convergence of Alg. 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 $C_n = \frac{1}{n} \max_{p(x^n||y^{n-1})} I(X^n \rightarrow Y^n)$, the inequality

$$C_n \leq I_U, \tag{25}$$

where

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

holds. Furthermore, if $r(x^n||y^{n-1})$ achieves C_n, then we have equality in (25).

The proof is given in Appendix B for the general case of delay d. We also 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.

V. NUMERICAL EXAMPLES FOR CALCULATING FEEDBACK CHANNEL'S CAPACITIES

In this section we present some examples of Alg. 1 performances 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 Alg. 1 is applicable on Finite State Channels (FSC), we describe this class of such channels and their properties. Gallager [26] 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 $p(s_0)$.

The causal conditioning probability of the output given the input is given 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., 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 of an FSC with feedback is bounded between

$$\underline{C}_N - \frac{\log|\mathcal{S}|}{N} \leq C_N \leq \overline{C}_N + \frac{\log|\mathcal{S}|}{N}, \tag{26}$$

where

$$\overline{C} = \frac{1}{N} \max_{p(x^n||y^{n-1})} \max_{s_0} I(X^n \to Y^n|s_0), \tag{27}$$

$$\underline{C} = \frac{1}{N} \max_{p(x^n||y^{n-1})} \min_{s_0} I(X^n \to Y^n|s_0). \tag{28}$$

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 Alg. 1 as presented in Section III, thus estimating the capacity.

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

A. Binary Symmetric Channel

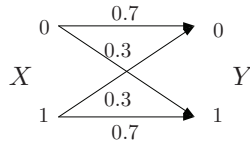Consider a memoryless BSC with probability of $p = 0.3$ as in Fig. 1. The capacity of this BSC is known to be

Fig. 1: Binary Symmetric Channel

$C = 1 - H(0.3) = 0.1187$. In Fig. 2 we present the directed information upper I_U and lower I_L bounds as a function of the iteration (as given in Alg. 1) and compare it to the capacity that is known analytically. Shannon showed [27] that for memoryless channels, feedback does not increase the capacity. Thus, we can expect the numerical solution given in Alg. 1 to achieve the same value as in the no-feedback case. Indeed, 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 Alg. 1 achieves is actually $r(x^n)$, i.e., does not depend on the feedback. We note here that we can achieve the capacity of the channel using a uniform distribution or $r(x^n)$. This does not imply that there is only one optimum distribution, and indeed the one that Alg. 1 achieves is not uniform.
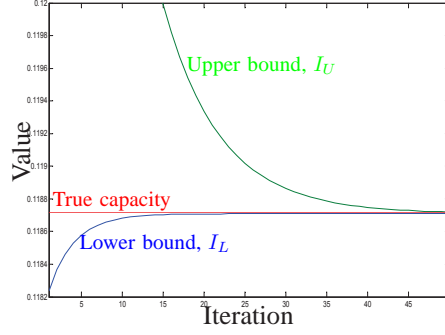
Fig. 2: Performance of Alg. 1 over BSC(0.3). The lower and upper lines are the bounds in each iteration in Alg. 1, whereas the horizontal line is the analytical calculation of the capacity.

B. Trapdoor Channel

1) Trapdoor channel with 2 states: The trapdoor channel was introduced by David Blackwell in 1961 [28] and later on by Ash [29]. One can look at this channel as such: Consider a binary channel modulated by a box that
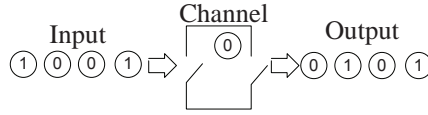

Fig. 3: Trapdoor Channel [29]

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. 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 Alg. 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 formula for s_{i-1} as

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

Note that in Table II, the input series $(0,0,1)$ and $(1,1,0)$ are not possible since the output is not one of the bits 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 $p(y_i|x^i, y^{i-1}, s_0 = 0)$

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	ϕ
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	ϕ
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

is given by

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

where we know that s_{i-1} is a function of (x^{i-1}, y^{i-1}, s_o), and $\wedge$ denotes AND.

Now that we have $p(y^n||x^n, s_0 = 0)$, we use Alg. 1 for estimating the capacity of the channel as we run the algorithm to find the upper and lower bound for every $n \in \{1..12\}$, where

$$\overline{C}_n = \max_{s_0} \max_{r(x^n||y^{n-1})} \frac{1}{n} I(X^n \to Y^n|s_0) + \frac{1}{n}, \tag{29}$$

$$\underline{C}_n = \max_{r(x^n||y^{n-1})} \min_{s_0} \frac{1}{n} I(X^n \to Y^n|s_0) - \frac{1}{n}. \tag{30}$$

Note that (29) is calculated via Alg. 1 and $s_0 = 0$ due to the channel's symmetry. However, calculating (30) 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 (29), and we only minimize over s_0. Clearly, $\underline{C}^* \leq \underline{C}$. Fig. 4 presents the capacity estimation, and the upper and lower bound, as a function of the block length n. In [21], the capacity of
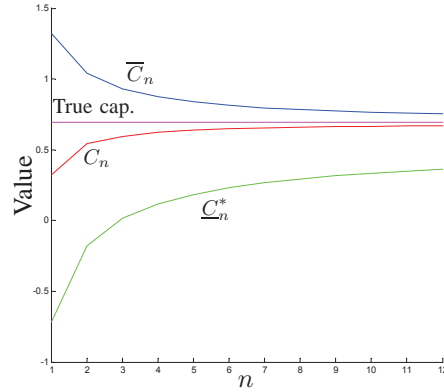

Fig. 4: Plot of $\overline{C}_n$, C_n, $\underline{C}_n^*$ and the true capacity of the trapdoor channel with 2 states and feedback with delay 1.

the trapdoor channel is calculated analytically, and given by

$$C = \lim_{n\to\infty} C_n = \log\left(\frac{1+\sqrt{5}}{2}\right) \approx 0.69424191. \tag{31}$$

We see from the simulation that the upper and lower bounds of the capacity approach the limit in (31), 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 $(n+1)C_{n+1} - nC_n$. The justification for this estimator is based on the following lemma.

Lemma 6 . If $\lim_{n\to\infty} I(X^n; Y_n|Y^{n-1})$ exists, then

$$\lim_{n\to\infty} \frac{1}{n}I(X^n \to Y^n) = \lim_{n\to\infty} \left(I(X^n \to Y^n) - I(X^{n-1} \to Y^{n-1})\right),$$

i.e.,

$$\lim_{n\to\infty} C_n = \lim_{n\to\infty} (n+1)C_{n+1} - nC_n.$$

Proof: If we suppose that the limit above exists, then

$$\lim_{n\to\infty} \left(I(X^n \to Y^n) - I(X^{n-1} \to Y^{n-1})\right) = \lim_{n\to\infty} I(X^n; Y_n|Y^{n-1})$$
$$\overset{(a)}{=} \lim_{n\to\infty} \frac{1}{n}\sum_{i=1}^{n} I(X^i; Y_i|Y^{i-1})$$
$$= \lim_{n\to\infty} \frac{1}{n}I(X^n \to Y^n),$$

where (a) follows from the fact that if the limit of the sequence $\{a_n\}$ exists, then the average of the sequence converges to the same limit. Further, a result from [7] provides that if the joint process $\{X_i, Y_i\}$ is stationary, then the limit $\lim_{n\to\infty} I(X^n; Y_n|Y^{n-1})$ exists. ∎

Fig. 5 presents the directed information rate estimator using the lemma above, and its comparison to the true capacity. One can see that the convergence of $(n+1)C_{n+1} - nC_n$ is faster than C_n and the upper and lower
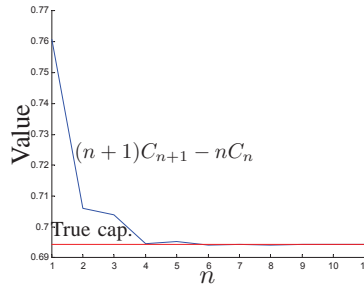

Fig. 5: The upper line is $(n+1)C_{n+1} - nC_n$ calculated using Alg. 1 and the horizontal line is the analytical calculation, for the trapdoor channel with 2 states and feedback with delay 1.

bounds as seen in Fig. 4, and achieves the value 0.6942285 when we calculate the 11th difference. Furthermore, the convergence of the directed information rate stabilizes faster.

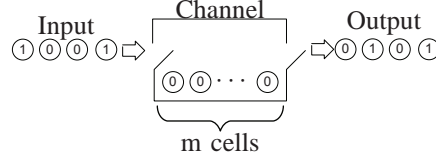*3) M-State Trapdoor channel:* We generalize the trapdoor channel to an M-state one. In the previous example

Fig. 6: Trapdoor channel with M states.

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 2 cells as presented in Fig. 6. 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 even 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 Alg. 1 for calculating C_n for every $n \in \{1, 2, ..., 12\}$. Fig. 7 presents the directed information rate estimator $(n+1)C_{n+1} - nC_n$ for the trapdoor channel with $M = 3$ cells. Note, that in
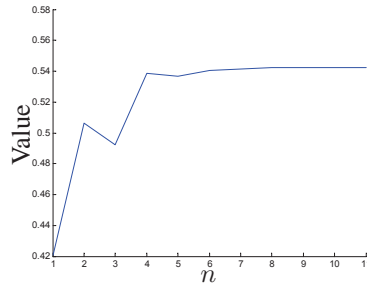

Fig. 7: Plot of $(n+1)C_{n+1} - nC_n$ for the trap door channel with 3 cells and feedback with delay 1.

Fig. 7 we achieve the value 0.5423984 in the 11th difference, thus we can assume that the capacity of a 3-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 estimation use is the directed information rate, with $n = 12$.
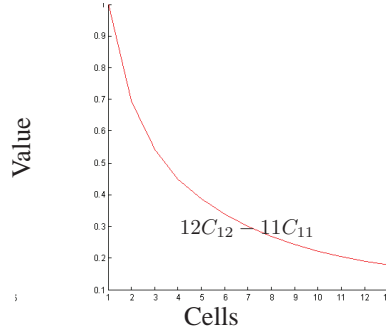

Fig. 8: Change of $12C_{12} - 11C_{11}$ over the number of cells in the trapdoor channel with feedback with delay 1.

In Fig. 8 we can see that the capacity decreases as the number of cells increases and approaches zero.

C. The 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 the input at time $i+1$. The channel (without
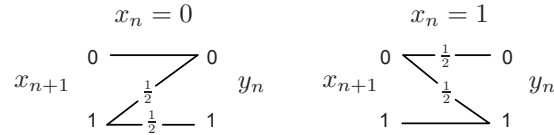

Fig. 9: The Ising Channel. [30]

feedback) was introduced by Berger and Bonomi [30] and is depicted in Fig. 9. 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. 10. In Fig. 10 (a), we obtain $C_{12} = 0.5459$, and in (b) we achieve $12C_{12} - 11C_{11} = 0.5563$ in the 11th difference.
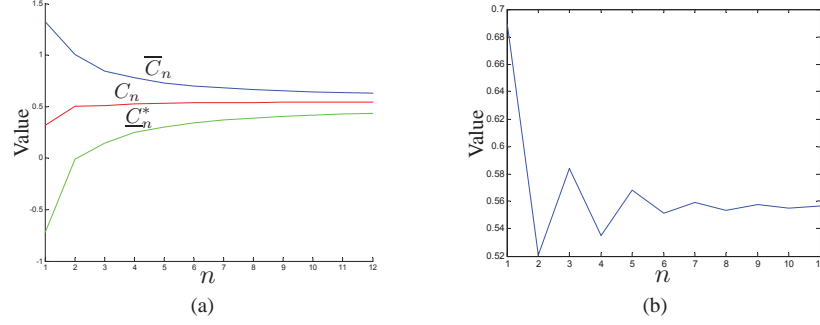
Fig. 10: Performance of Alg. 1 on the Ising channel with feedback delay of $d = 2$. In (a) we present $\overline{C}_n$, C_n, $\underline{C}_n^*$, and in (b) we have $(n+1)C_{n+1} - nC_n$.

2) The 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 $d = \{2, 3, ..., 12\}$. The formulas for estimating the capacity when the delay is bigger than 1 is given in Section III, equations (14), (15). In Fig. 11 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.
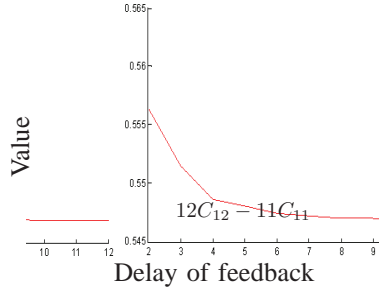

Fig. 11: Change of $12C_{12} - 11C_{11}$ over the delay of the feedback on the Ising channel.

VI. CONCLUSIONS

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

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

The optimizing 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 we tried to optimize over. We overcame this barrier by using an additional backwards loop to find all components of the causal conditioned probability, separately.

Another application of optimizing the directed information is to estimate the rate distortion function for source coding with feed forward as presented in [31], [32], [33]. In our future work [34], 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 CODING-FEEDBACK THAT IS A FUNCTION OF THE DELAYED OUTPUT

Here we extend Alg. 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

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

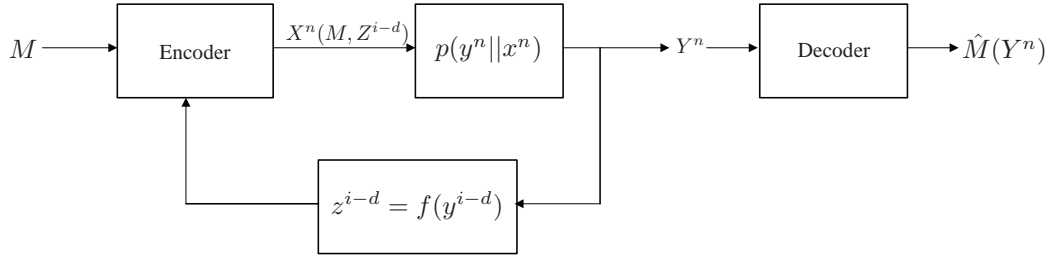The optimization problem is associated to Fig. 12.

Fig. 12: Channel with delayed feedback as a function of the output.

The proof for this case is similar to that of Theorem 1, except 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 given by

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

where

$$r(x_i|x^{i-1}, z^{i-d}) = \frac{r'(x^i, z^{i-d})}{\sum_{x^i} r'(x^i, z^{i-d})}, \tag{32}$$

and

$$r'(x^i, z^{i-d}) = \prod_{x_{i+1}^n, y_{i-d+1}^n} \prod_{A_{i,d,z}} \left[\frac{q(x^n|y^n)}{\prod_{j=i+1}^n r(x_j|x^{j-1}, z^{j-d})} \right]^{\frac{p(y^n||x^n) \prod_{j=i+1}^n r(x_j|x^{j-1}, z^{j-d})}{\Sigma_{A_{i,d,z}} \prod_{j=1}^{i-d} p(y_j|x^j, y^{j-1})}}. \tag{33}$$

Proof: We find the products of $r(x^n||z^{n-d})$ that achieve maximum for the directed information. For convenience, let us use for short: $r_i \triangleq r(x_i|x^{i-1}, z^{i-d})$, and $p_i \triangleq p(y_i|x^i, y^{i-1})$. As in Lemma 2 we can omit that $I(X^n \to Y^n)$ is concave in $\{r(x^n||y^{n-d}), q(x^n|y^n)\}$. Furthermore, the constraints of the optimization problem are affine, and we can use the Lagrange multipliers method with the Karush-Kuhn-Tucker conditions. We define the Lagrangian as:

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

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

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

$$= \sum_{A_{i,d,z}} \prod_{j=1}^{i-1} r_j \sum_{x_{i+1}^n, y_{i-d+1}^n} \left(p(y^n||x^n) \prod_{j=i+1}^n r_j \left[\log \frac{q(x^n|y^n)}{\prod_{j=i+1}^n r_j} - \log \prod_{j=1}^{i-1} r_j - \log r_i - 1 \right] \right) + \nu_{i,(x^{i-1}, z^{i-d})}$$

$$= 0,$$

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.

Note that since $\prod_{j=1}^{i-1} r_j$ does not depend on $A_{i,d,z}$, we can take the product out of the sum. Furthermore, since ν_i is a function of (x^{i-1}, z^{i-d}) we can divide the whole equation by the product above, 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:

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

where

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

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

$$r'(x^i, z^{i-d}) = \prod_{x_{i+1}^n, y_{i-d+1}^n, A_{i,d,z}} \left[\frac{q(x^n|y^n)}{\prod_{j=i+1}^n r(x_j|x^{j-1}, z^{j-d})} \right]^{\frac{p(y^n||x^n) \prod_{j=i+1}^n r(x_j|x^{j-1}, z^{j-d})}{\sum_{A_{i,d,z}} \prod_{j=1}^{i-d} p(y_j|x^j, y^{j-1})}}. \tag{34}$$

As in Section IV, we can see that for all i, r_i is dependent on $q(x^n|y^n)$ and $\{r_{i+1}, r_{i+2}, ..., r_n\}$, and r_n is a function of $q(x^n|y^n)$ alone. Thus, we use the *Backwards maximization* method. After calculating r_i for all $i = 1, ..., n$, 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., $c_1(q)$ and the lemma is proven. ∎

As mentioned, by replacing Lemmas 4, 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 kth iteration in the extended algorithm. One more step is required in order to prove the extension of Alg. 1 to the case presented here; the existence of I_U. This part is presented in Appendix B.

APPENDIX B
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 on I_L to it from below, as in Alg. 1. To this purpose, we present and prove 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, 10.

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

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

Proof: For any $r_1(x^n||y^{n-d})$, $r_0(x^n||y^{n-d})$,

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

$$= \sum_{x^n, y^n} r_1(x^n||y^{n-d}) \cdot p(y^n||x^n) \log \frac{p(y^n||x^n)}{\sum_{x'^n} p(y^n||x'^n) \cdot r_0(x'^n||y^{n-d})}$$

$$-\sum_{x^n, y^n} r_1(x^n||y^{n-d}) \cdot p(y^n||x^n) \log \frac{p(y^n||x^n)}{\sum_{x'^n} p(y^n||x'^n) \cdot r_1(x'^n||y^{n-d})}$$

$$= \sum_{x^n, y^n} r_1(x^n||y^{n-d}) \cdot p(y^n||x^n) \log \frac{\sum_{x'^n} p(y^n||x'^n) \cdot r_1(x'^n||y^{n-d})}{\sum_{x'^n} p(y^n||x'^n) \cdot r_0(x'^n||y^{n-d})}$$

$$= \sum_{y^n} p_1(y^n) \log \frac{p_1(y^n)}{p_0(y^n)}$$

$$\overset{(a)}{=} D\left(p_1(y^n)||p_0(y^n)\right)$$

$$\overset{(b)}{\geq} 0,$$

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 non negativity 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 $r_0(x^n||y^{n-d})$,

$$C_n \leq I_U,$$

where

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

Proof: To prove this lemma, we first use lemma 8. For every $r_1(x^n||y^{n-d})$, $r_0(x^n||y^{n-d})$,

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

$$\overset{(b)}{\leq} \sum_{x^n, y^{n-d}} \prod_{i=1}^{n} r_1(x_i|x^{i-1}, y^{i-d}) \max_{x_n} \underbrace{\sum_{y_{n-d+1}^n} p(y^n||x^n) \log \frac{p(y^n||x^n)}{\sum_{x'^n} p(y^n||x'^n) \cdot r_0(x'^n||y^{n-d})}}_{f(x^{n-1}, y^{n-d})}$$

$$\overset{(c)}{=} \sum_{x^{n-1}, y^{n-d-1}} \prod_{i=1}^{n-1} r_1(x_i|x^{i-1}, y^{i-d}) \underbrace{\sum_{y_{n-d}} \max_{x_n} \sum_{y_{n-d+1}^n} p(y^n||x^n) \log \frac{p(y^n||x^n)}{\sum_{x'^n} p(y^n||x'^n) \cdot r_0(x'^n||y^{n-d})}}_{f(x^{n-1}, y^{n-d-1})}$$

$$\leq \sum_{x^{n-1}, y^{n-d-1}} \prod_{i=1}^{n-1} r_1(x_i|x^{i-1}, y^{i-d}) \underbrace{\max_{x_{n-1}} \sum_{y_{n-d}} \max_{x_n} \sum_{y_{n-d+1}^n} p(y^n||x^n) \log \frac{p(y^n||x^n)}{\sum_{x'^n} p(y^n||x'^n) \cdot r_0(x'^n||y^{n-d})}}_{f(x^{n-2}, y^{n-d-1})}$$

$$\vdots$$

$$\leq \sum_{x^d} \prod_{i=1}^{d} r_1(x_i|x^{i-1}, y^{i-d}) \underbrace{\sum_{y_1} \max_{x_{d+1}} \sum_{y_2} \cdots \max_{x_n} \sum_{y_{n-d+1}^n} p(y^n||x^n) \log \frac{p(y^n||x^n)}{\sum_{x'^n} p(y^n||x'^n) \cdot r_0(x'^n||y^{n-d})}}_{f(x^d)}$$

$$\leq \underbrace{\sum_{x^d} \prod_{i=1}^{d} r_1(x_i|x^{i-1}, y^{i-d})}_{=1} \underbrace{\max_{x^d} \sum_{y_1} \max_{x_{d+1}} \sum_{y_2} \cdots \max_{x_n} \sum_{y_{n-d+1}^n} p(y^n||x^n) \log \frac{p(y^n||x^n)}{\sum_{x'^n} p(y^n||x'^n) \cdot r_0(x'^n||y^{n-d})}}_{\in \mathbb{R}}$$

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

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 inequality above is true for every $r_1(x^n||y^{n-d})$, we can use it on $r_c(x^n||y^{n-d})$ that achieves C_n, and thus for every $r_0(x^n||y^{n-d})$

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

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

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

and the lemma is proven. ■

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 $r(x^n||y^{n-d})$ that achieves the capacity.

Proof: In Lemma 9, we showed only half of the proof of the theorem, i.e., the existence of an upper bound. 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 all $r(x_i|x^{i-1}, y^{i-d})$s. We can use the KKT conditions since the directed information is a concave function in all $r(x_i|x^{i-1}, y^{i-d})$s, as seen in Lemma 3.

We define the Lagrangian as

$$J = \sum_{x^n, y^n} r(x^n||y^{n-d}) \cdot p(y^n||x^n) \log \frac{p(y^n||x^n)}{\sum_{x'^n} p(y^n||x'^n) \cdot r(x'^n||y^{n-d})}$$

$$- \sum_{i=1}^{n} \sum_{x^{i-1}, y^{i-d}} \nu_{i,(x^{i-1}, y^{i-d})} \left(\sum_{x_i} r(x_i|x^{i-1}, y^{i-d}) - 1 \right) + \sum_{i=1}^{n} \sum_{x^{i-1}, y^{i-d}} h_{i,(x^{i-1}, y^{i-d})} r(x_i|x^{i-1}, y^{i-d}).$$

Now, for every $r(x_i|x^{i-1}, y^{i-d})$, we have

$$\frac{\partial J}{\partial r(x_i|x^{i-1}, y^{i-d})} = \sum_{x_{i+1}, y_{i-d+1}} r(x_{i+1}|x^i, y^{i-d+1}) \cdots \sum_{x_n, y_{n-d}} r(x_n|x^{n-1}, y^{n-d}).$$

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

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 we obtain

$$\sum_{x_{i+1},y_{i-d+1}} r(x_{i+1}|x^i, y^{i-d+1}) \cdots \sum_{x_n,y_{n-d}} r(x_n|x^{n-1}, y^{n-d}) \sum_{y_{n-d+1}^n} p(y^n||x^n) \log \frac{p(y^n||x^n)}{\sum_{x'^n} p(y^n||x'^n) \cdot r(x'^n||y^{n-d})} = \nu_i,$$

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

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 that

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

and thus constant for every x_n. As a result, for $i = n - 1$ we have

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

$$= \sum_{y_{n-d}} \max_{x_n} \sum_{y_{n-d+1}^n} p(y^n||x^n) \log \frac{p(y^n||x^n)}{\sum_{x'^n} p(y^n||x'^n) \cdot r(x'^n||y^{n-d})}$$

$$= \nu_{n-1,(x^{n-2},y^{n-d-1})}$$

that again, is constant for every x_{n-1}. We can move backwards and obtain that for $i = 1$,

$$\sum_{x_2} r(x_2|x_1) \cdots \sum_{x_d} r(x_d|x^{d-1}) \sum_{x_{d+1},y_1} r(x_{d+1}|x^d, y_1) \cdots r(x_n|x^{n-1}, y^{n-d}).$$

$$\sum_{y_{n-d+1}^n} p(y^n||x^n) \log \frac{p(y^n||x^n)}{\sum_{x'^n} p(y^n||x'^n) \cdot r(x'^n||y^{n-d})}$$

$$= \sum_{x_2} r(x_2|x_1) \max_{x_3^d} \sum_{y_1} \max_{x_{d+1}} \cdots \max_{x_n} \underbrace{\sum_{y_{n-d+1}^n} p(y^n||x^n) \log \frac{p(y^n||x^n)}{\sum_{x'^n} p(y^n||x'^n) \cdot r(x'^n||y^{n-d})}}_{\nu_{2,(x_1)}}$$

$$= \max_{x_2^d} \sum_{y_1} \max_{x_{d+1}} \cdots \max_{x_n} \underbrace{\sum_{y_{n-d+1}^n} p(y^n||x^n) \log \frac{p(y^n||x^n)}{\sum_{x'^n} p(y^n||x'^n) \cdot r(x'^n||y^{n-d})}}_{\nu_1}$$

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

Using the analysis above, we find an expression for C_n where $r(x^n||y^{n-d})$ achieves it. 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.

$$C_n = \frac{1}{n} \sum_{x^n,y^n} r(x^n||y^{n-d}) \cdot p(y^n||x^n) \log \frac{p(y^n||x^n)}{\sum_{x'^n} p(y^n||x'^n) \cdot r(x'^n||y^{n-d})}$$

$$= \frac{1}{n} \sum_{x_1} r(x_1) \sum_{x_2} r(x_2|x_1) \cdots \sum_{x_d} r(x_d|x^{d-1}) \sum_{x_{d+1},y_1} r(x_{d+1}|x^d, y_1)$$

$$\cdots r(x_n|x^{n-1}, y^{n-d}) \sum_{y_{n-d+1}^n} p(y^n||x^n) \log \frac{p(y^n||x^n)}{\sum_{x'^n} p(y^n||x'^n) \cdot r(x'^n||y^{n-d})}$$

$$= \frac{1}{n} \sum_{x_1} r(x_1) \max_{x^d} \sum_{y_1} \max_{x_{d+1}} \cdots \max_{x_n} \sum_{y_{n-d+1}^n} p(y^n||x^n) \log \frac{p(y^n||x^n)}{\sum_{x'^n} p(y^n||x'^n) \cdot r(x'^n||y^{n-d})}$$

$$\overset{(a)}{=} \frac{1}{n} \max_{x^d} \sum_{y_1} \max_{x_{d+1}} \cdots \max_{x_n} \sum_{y_{n-d+1}^n} p(y^n||x^n) \log \frac{p(y^n||x^n)}{\sum_{x'^n} p(y^n||x'^n) \cdot r(x'^n||y^{n-d})},$$

where (a) is due to the analysis above 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 to 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

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

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.

References

[1] C. E. Shannon. A mathematical theory of communication. *Bell Syst. Tech. J.*, 27:379–423 and 623–656, 1948.

[2] R. Blahut. An algorithm for computing the capacity of arbitrary discrete memoryless channels. *IEEE Trans. Information Theory*, pages 14–20, 1972.

[3] S. Arimoto. Computation of channel capacity and rate-distortion functions. *IEEE Trans. Information Theory*, pages 160–473, 1972.

[4] I. Csiszár and G. Tusnady. Information geometry and alternating minimization procedures. *Statistics and Decisions*, Supplementary Issue 1:205–237, 1984.

[5] R. W. Yeung. *Information theory and network coding*. Springer, 2008.

[6] J. Massey. Causality, feedback and directed information. *Proc. Int. Symp. Inf. Theory Applic. (ISITA-90)*, pages 303–305, Nov. 1990.

[7] G. Kramer. *Directed information for channels with feedback*. Ph.D. dissertation, Swiss Federal Institute of Technology (ETH) Zurich, 1998.

[8] S. Tatikonda and S. Mitter. The capacity of channels with feedback. *IEEE Trans. Inf. Theory*, 55:323–349, January 2009.

[9] H. H. Permuter, T. Weissman, and A. J. Goldsmith. Finite state channels with time-invariant deterministic feedback. *IEEE Trans. Inf. Theory*, 55(2):644–662, February 2009.

[10] Y. H. Kim. A coding theorem for a class of stationary channels with feedback. *IEEE Trans. Information Theory*, pages 1488–1499, April 2008.

[11] G. Matz and P. Duhamel. Information geometric formulation and interpretation of accelerater blahut-arimoto-type algorithms. *Proc. 2004 IEEE Information Theory Workshop*. San Antonio, TX, USA, Oct. 2004.

[12] M. Rezaeian and A. Grant. A generalization of arimoto-blahut algorithm. *IEEE Trans. Information Theory*, pages 2779–2784, 2004.

[13] W. Yu F. Dupuis and F. Willems. Arimoto-blahut algorithms for computing channel capacity and rate-distortion with sideinformation. *in ISIT*, 2004.

[14] C. Heegard and A. A. El Gamal. On the capacity of computer memory with defects. *IEEE Transactions on Information Theory*, 29(5):731–739, 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. Information Theory*, pages 2052–2056, 2004.

[16] J. Dauwels. Numercal computation of the capacity of continuous memoryless channels. *Proc. of the 26th Symposium on Information Theory in the BENELUX*, 2005.

[17] D. Arnold H.-A. Loeliger P. O. Vontobel, A. Kavčić. Capacity of finite-state machine channels. *IEEE Trans. Information Theory*, pages 1887–1918, May 2008.

[18] O. Sumszyk and Y. Steinberg. Information embedding with reversible stegotext. *ISIT*, July 2009.

[19] A. Kavcic S. Yang and S. Tatikonda. Feedback capacity of finite-state machine channels. *IEEE Trans. Inf. Theory*, 51(3):799–810, March 2005.

[20] J. Chen and T. Berger. The capacity of finite-state Markov channels with feedback. *IEEE Trans. Inf. Theory*, 51:780–789, 2005.

[21] H. H. Permuter, P. Cuff, B. Van Roy, and T. Weissman. Capacity of the trapdoor channel with feedback. *IEEE Trans. Inf. Theory*, 54(7):3150–3165, July 2008.

[22] S. K. Gorantla and T. P. Coleman. On reversible markov chains and maximization of directed information. *ISIT*, June 2010.

[23] H. Marko. The bidirectional communication theory- a generalization of information theory. *IEEE Trans. on communication*, COM-21:1335–1351, 1973.

[24] T. M. Cover and J. A. Thomas. *Elements of Information Theory*. Wiley, New-York, 2nd edition, 2006.

[25] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, New-York, 2004.

[26] R. G. Gallager. *Information theory and reliable communication*. Wiley, New York, 1968.

[27] C. E. Shannon. The zero error capacity of a noisy channel. *IEEE Trans. Inf. Theory*, IT-2:8–19, 1956.

[28] D. Blackwell. Information theory. *Modern Mathematics for Engineer: Second Series*, pages 183–193, 1961.

[29] R. Ash. *Information Theory*. Wiley, New-York, 1965.

[30] T. Berger and F. Bonomi. Capacity and zero-error capacity of ising channels. *IEEE Trans. Inf. Theory*, 36:173–180, 1990.

[31] T. Weissman and N. Merhav. On competitive prediction and its relation to rate-distortion theory. *IEEE Trans. Inf. Theory*, 49(12):3185–3194, 2003.

[32] R. Venkataramanan and S. S. Pradhan. Source coding with feed-forward: Rate-distortion theorems and error exponents for a general source. *IEEE Transactions on Information Theory*, 53(6):2154–2179, 2007.

[33] 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*, abs/cs/0702009, 2007.

[34] H. Permuter and I. Naiss. Bounds on rate distortion with feedforward for stationary and ergodic sources. in preperation, nov 2010.