Efficient Approximation of Channel Capacities

Tobias Sutter,[1, *] David Sutter,[2, †] Peyman Mohajerin Esfahani,[1, *] and John Lygeros[1, *]

[1]*Automatic Control Laboratory, ETH Zurich, Switzerland*
[2]*Institute for Theoretical Physics, ETH Zurich, Switzerland*

We propose an iterative method for approximately computing the capacity of discrete memoryless channels, possibly under additional constraints on the input distribution. Based on duality of convex programming, we derive explicit upper and lower bounds for the capacity. The presented method requires $O(\frac{M^2 N \sqrt{\log N}}{\varepsilon})$ to provide an estimate of the capacity to within ε, where N and M denote the input and output alphabet size; a single iteration has a complexity $O(MN)$. We also show how to approximately compute the capacity of memoryless channels having a bounded continuous input alphabet and a countable output alphabet under some mild assumptions on the decay rate of the channel's tail. It is shown that discrete-time Poisson channels fall into this problem class. As an example, we compute sharp upper and lower bounds for the capacity of a discrete-time Poisson channel with a peak-power input constraint.

1. INTRODUCTION

A discrete memoryless channel (DMC) comprises a finite input alphabet $\mathcal{X} = \{1, 2, \ldots, N\}$, a finite output alphabet $\mathcal{Y} = \{1, 2, \ldots, M\}$, and a conditional probability mass function expressing the probability of observing the output symbol y given the input symbol x, denoted by $W(y|x)$. In his seminal 1948 paper [1], Shannon proved that the channel capacity for a DMC is

$$C(W) = \max_{p \in \Delta_N} I(p, W), \tag{1}$$

where $\Delta_N := \{x \in \mathbb{R}^N : x \geq 0, \sum_{i=1}^N x_i = 1\}$ denotes the N-simplex and $I(p, W) := \sum_{x \in \mathcal{X}} p(x) D(W(\cdot|x) \| (pW)(\cdot))$ the mutual information. $W(y|x) = \mathbb{P}[Y = y | X = x]$ describes the channel law and $(pW)(\cdot)$ is the probability distribution of the channel output induced by p and W, i.e., $(pW)(y) := \sum_{x \in \mathcal{X}} p(x) W(y|x)$. $D(\cdot \| \cdot)$ denotes the relative entropy that is defined as $D(W(\cdot|x) \| (pW)(\cdot)) := \sum_{y \in \mathcal{Y}} W(y|x) \log \left(\frac{W(y|x)}{(pW)(y)} \right)$. Shannon also showed that in case of an additional average cost constraint on the input distribution of the form $\mathbb{E}[s(X)] \leq S$, where $s : \mathcal{X} \to \mathbb{R}_{\geq 0}$ denotes a cost function and $S \geq 0$, the capacity is given by

$$C_S(W) = \begin{cases} \max_p & I(p, W) \\ \text{s.t.} & \mathbb{E}[s(X)] \leq S \\ & p \in \Delta_N. \end{cases} \tag{2}$$

For a few DMCs it is known that the capacity can be computed analytically, however in general there is no closed-form solution. It is therefore of interest to have an algorithm that solves (2) in a reasonable amount of time. Since for a fixed channel the mutual information is a concave function in p, the optimization problem (2) is a finite dimensional convex optimization problem. Solving

---

[*] {sutter, mohajerin, lygeros}@control.ee.ethz.ch
[†] suttedav@phys.ethz.ch

(2) with convex programming solvers, however, turned out to be computationally inefficient even for small alphabet sizes [2].

Shannon's formula for the capacity of a DMC generalizes to the case of memoryless channels with continuous input and output alphabets, i.e. $\mathcal{X} = \mathcal{Y} = \mathbb{R}$. However, when considering such channels, it is essential to introduce additional constraints on the channel input to obtain physically meaningful results, more details can be found in [3, Chapter 7]. In addition to average cost type constraints, peak-power constraints are also often considered. A peak-power constraint demands that $X \in \mathbb{A}$ for some compact set $\mathbb{A} \subset \mathcal{X}$ with probability one. For such a setup, i.e., having average and peak-power constraints, the capacity is given by

$$
C_{\mathbb{A},S}(W) = \left\{
\begin{array}{ll}
\sup_{p} & I(p, W) \\
\text{s.\,t.} & \mathbb{E}[s(X)] \leq S \\
& p \in \mathcal{P}(\mathbb{A}),
\end{array}
\right.
\tag{3}
$$

where $\mathcal{P}(\mathbb{A})$ denotes the set of all probability distributions on the Borel σ-algebra $\mathcal{B}(\mathbb{A})$ and the mutual information is defined as $I(p, W) := \int_{\mathbb{A}} D(W(\cdot|x)\|(pW)(\cdot))\,p(\mathrm{d}x)$. The channel is described by a transition density defined by $\mathbb{P}[Y \in \mathrm{d}y | X = x] = W(y|x)\mathrm{d}y$ and $(pW)(\cdot)$ is the probability distribution of the channel output induced by p and W which is given by $(pW)(y) := \int_{\mathbb{A}} W(y|x)p(\mathrm{d}x)$ and the relative entropy that is defined as $D(W(\cdot|x)\|(pW)(\cdot)) := \int_{\mathcal{Y}} W(y|x)\log\left(\frac{W(y|x)}{(pW)(y)}\right)\mathrm{d}y$. The optimization problem (3) is an infinite dimensional convex optimization problem and as such in general computationally intractable (NP-hard).

Previous Work and Contributions.— Historically one of the first attempts to numerically solve (2) is the so-called *Blahut-Arimoto algorithm* [2, 4], that exploits the special structure of the mutual information and approximates iteratively the capacity of any DMC. Each iteration step has a computational complexity $O(MN^2)$. It was shown that this algorithm, in case of no additional input constraints has an *a priori* error bound of the form $|C(W) - C_{\text{approx}}^{(n)}(W)| \leq O(\frac{\log(N)}{n})$, where n denotes the number of iterations [4, Corollary 1]. Hence, the overall computational complexity of finding an ε-solution is given by $O(\frac{MN^2 \log(N)}{\varepsilon})$. As such the computational cost required for an acceptable accuracy for channels with large input alphabets can be considerable. This undesirable property together with the complexity per iteration prevents the algorithm from being useful for a large class of channels, e.g., a Rayleigh channel with a discrete input alphabet [5]. There have been several improvements of the Blahut-Arimoto algorithm [6–8], which achieve a better convergence for certain channels. However, since they all rely on the original Blahut-Arimoto algorithm they inherit its overall computational complexity as well as its complexity per iteration step. Therefore, even with improved Blahut-Arimoto algorithms, approximating the capacity for channels having large input alphabets remains computationally expensive.

A more recent approach towards approximating (2) is presented in [9] by Mung and Boyd, where they introduce an efficient method to derive upper bounds on the channel capacity problem, based on geometric programming. Huang and Meyn [10] developed a different approach based on cutting plane methods, where the mutual information is iteratively approximated by linear functionals and in each iteration step, a finite dimensional linear program is solved. It has been shown that this method converges to the optimal value, however no rate of convergence is provided.

In this article, we present a new approach to solve (2) that is based on its dual formulation. It turns out that the dual problem of (2) has a particular structure that allows us to apply Nesterov's smoothing method [11]. In the absence of input cost constraints, this leads to an a priori error bound of the order $|C(W) - C_{\text{approx}}^{(n)}(W)| \leq O(\frac{M\sqrt{\log(N)}}{n})$, where n denotes the number of iterations and each iteration step has a computational complexity of $O(NM)$. Thus, the overall

computational complexity of finding an ε-solution is given by $O(\frac{M^2 N \sqrt{\log(N)}}{\varepsilon})$. In particular for large input alphabets our method has a considerable computational advantage over the Blahut-Arimoto algorithm. In addition the novel method provides primal and dual optimizers leading to an *a posteriori* error which is often much smaller than the a priori error.

Due to the favorable structure of the capacity problem and its dual formulation, the presented method can be extended to approximate the capacity of memoryless channels having a bounded continuous input alphabet and a countable output alphabet, under some assumptions on the tail of $W(\cdot|x)$, i.e., problem (3) is addressed for a countable output alphabet. As a concrete example, this is demonstrated on the discrete time Poisson channel with a peak-power constraint. To the best of our knowledge, for this scenario up to now only lower bounds exist [12].

Structure.— Section 2 introduces our method for approximating the channel capacity for DMCs. We provide a priori and a posteriori bounds for the approximation error and present two numerical examples that illustrate its computational performance compared to the Blahut-Arimoto algorithm. In Section 3, we generalize the approximation scheme to channels having bounded continuous input alphabets and countable output alphabets. We then show how the presented results can be used to compute the capacity of discrete-time Poisson channels under a peak-power constraint and possibly average-power constraints on the input. We conclude in Section 4 with a summary and potential subjects of further research. In the interest of readability, some of the technical proofs and details are given in the appendices.

Notation.— The logarithm with basis 2 is denoted by $\log(\cdot)$ and the natural logarithm by $\ln(\cdot)$. In Section 2 we consider DMCs with a finite input alphabet $\mathcal{X} = \{1, 2, \ldots, N\}$ and a finite output alphabet $\mathcal{Y} = \{1, 2, \ldots, M\}$. The channel law is summarized in a matrix $\mathsf{W} \in \mathbb{R}^{N \times M}$, where $\mathsf{W}_{ij} := \mathbb{P}[Y = j | X = i] = W(j|i)$. We define the standard $n-$simplex as $\Delta_n := \{x \in \mathbb{R}^n : x \geq 0, \sum_{i=1}^n x_i = 1\}$. The input and output probability mass functions are denoted by the vectors $p \in \Delta_N$ and $q \in \Delta_M$. The input cost constraint can be written as $\mathbb{E}[s(X)] = p^\top s \leq S$, where $s \in \mathbb{R}_{\geq 0}^N$ denotes the cost vector and $S \in \mathbb{R}_{\geq 0}$ is the given total cost. The binary entropy function is denoted by $H_{\mathrm{b}}(\alpha) := -\alpha \log(\alpha) - (1 - \alpha) \log(1 - \alpha)$, for $\alpha \in [0, 1]$. For a probability mass function $p \in \Delta_N$ we denote the entropy by $H(p) := \sum_{i=1}^N -p_i \log(p_i)$. It is convenient to introduce an additional variable for the conditional entropy of Y given $\{X = i\}$ as $r \in \mathbb{R}^N$, where $r_i = -\sum_{j=1}^M \mathsf{W}_{ij} \log(\mathsf{W}_{ij})$. For a probability density p supported at a measurable set $B \subset \mathbb{R}$ we denote the differential entropy by $h(p) = -\int_B p(x) \log(p(x)) \mathrm{d}x$. For two vectors $x, y \in \mathbb{R}^n$, we denote the canonical inner product by $\langle x, y \rangle := x^\top y$. We denote the maximum (resp. minimum) between a and b by $a \vee b$ (resp. $a \wedge b$). For $\mathbb{A} \subset \mathbb{R}$ and $1 \leq p \leq \infty$, let $\mathrm{L}^p(\mathbb{A})$ denote the space of L^p-functions on the measure space $(\mathbb{A}, \mathcal{B}(\mathbb{A}), \mathrm{d}x)$, where $\mathcal{B}(\mathbb{A})$ denotes the Borel σ-algebra and $\mathrm{d}x$ the Lebesgue measure. The capacity of a channel W is denoted by $C(W)$. For the channel law matrix $\mathsf{W} \in \mathbb{R}^{N \times M}$ we consider the norm $\|\mathsf{W}\| := \max_{\lambda \in \mathbb{R}^M, \, p \in \mathbb{R}^N} \{\langle \mathsf{W}\lambda, p \rangle \, : \, \|\lambda\|_2 = 1, \, \|p\|_1 = 1\}$, and note that an upper bound is given by

$$\|\mathsf{W}\| = \max_{\|p\|_1=1} \max_{\|\lambda\|_2=1} \lambda^\top \mathsf{W}^\top p \leq \max_{\|p\|_1=1} \|\mathsf{W}^\top p\|_2 \leq \max_{\|p\|_1=1} \|\mathsf{W}^\top p\|_1 = \max_{\|p\|_1=1} \|p\|_1 = 1. \tag{4}$$

2. DISCRETE MEMORYLESS CHANNEL

To keep notation simple we consider a single average-input cost constraint as the extension to multiple average-input cost constraints is straightforward. In a first step, we introduce the output distribution $q \in \Delta_M$ as an additional decision variable, as done in [9, 13, 14].

Lemma 2.1. *Let $\mathcal{F} := \arg\max\limits_{p \in \Delta_N} I(p, W)$ and $S_{\max} := \min\limits_{p \in \mathcal{F}} s^\top p$. If $S \geq S_{\max}$ the optimization problem* (2) *has the same optimal value as*

$$
\mathsf{P}: \quad \begin{cases} \max\limits_{p,q} & -r^\top p + H(q) \\ \mathrm{s.\,t.} & \mathsf{W}^\top p = q \\ & p \in \Delta_N, \; q \in \Delta_M. \end{cases} \tag{5}
$$

If $S < S_{\max}$ the optimization problem (2) *has the same optimal value as*

$$
\mathsf{P}: \quad \begin{cases} \max\limits_{p,q} & -r^\top p + H(q) \\ \mathrm{s.\,t.} & \mathsf{W}^\top p = q \\ & s^\top p = S \\ & p \in \Delta_N, \; q \in \Delta_M. \end{cases} \tag{6}
$$

Proof. The proof can be found in Appendix A 1. □

Note that we later add an assumption on our channel (Assumption 2.3) that guarantees unique-ness of the optimizer maximizing the mutual information, i.e., $\mathcal{F}$ is a singleton. In this case the optimizer to (6) (resp. (5)) is also feasible for the original problem (2). Computing $S_{\max}$ is straight-forward once $\mathcal{F}$ is known. The singleton $\mathcal{F}$ can be seen as the maximizer of a channel capacity problem with no additional input cost constraint and can as such be computed with the scheme we present in this article.

For the rest of the section we restrict attention to (6), since the less constrained problem (5) can be solved in a similar, more direct way. We tackle this optimization problem through its Lagrangian dual problem. The dual function turns out to be a non-smooth function. As such, it is known that the efficiency estimate of a black-box first-order method is of the order $O\left(\frac{1}{\varepsilon^2}\right)$ if no specific problem structure is used, where ε is the desired abolute accuracy of the approximate solution in function value [15]. We show, however, that P has a certain structure that allows us to use Nesterov's approach for approximating non-smooth problems with smooth ones [11] leading to an efficiency estimate of the order $O\left(\frac{1}{\varepsilon}\right)$. This, together with the low complexity of each iteration step in the approximation scheme leads to a numerical method for the channel capacity problem that has a very attractive computational complexity.

A. Input Constrained Case

In the following we focus on the input constrained channel capacity problem (6) and the scenario of no input constraints (5) is discussed as a special case within this section. As a preliminary result for some $c \in \mathbb{R}^N$ we consider the following optimization problem, that, if feasible, has an analytical solution

$$
\begin{cases} \max\limits_{p} & J(p) := H(p) - c^\top p \\ \mathrm{s.t.} & s^\top p = S \\ & p \in \Delta_N. \end{cases} \tag{7}
$$

Lemma 2.2. *Let $p^\star = [p_1^\star, \ldots, p_N^\star]$ with $p_i^\star = 2^{\mu_1 - c_i + \mu_2 s_i}$, where μ_1 and μ_2 are chosen such that $p^\star$ satisfies the constraints in* (7). *Then $p^\star$ uniquely solves* (7).

Proof. See Appendix A 2. □

Consider the convex optimizaton problem (6), whose optimal value, according to Lemma 2.1 is the capacity $C_S(W)$. The Lagrange dual program to (6) is

$$\mathsf{D}: \quad \begin{cases} \min\limits_{\lambda} & G(\lambda) + F(\lambda) \\ \text{s.t.} & \lambda \in \mathbb{R}^M, \end{cases} \tag{8}$$

where $F, G : \mathbb{R}^M \to \mathbb{R}$ are given by

$$G(\lambda) = \begin{cases} \max\limits_{p} & -r^\top p + \lambda^\top \mathsf{W}^\top p \\ \text{s.t.} & s^\top p = S \\ & p \in \Delta_N \end{cases} \quad \text{and} \quad F(\lambda) = \begin{cases} \max\limits_{q} & H(q) - \lambda^\top q \\ \text{s.t.} & q \in \Delta_M \end{cases}. \tag{9}$$

Note that since the coupling constraint $\mathsf{W}^\top p = q$ in the primal program (6) is affine, the set of optimal solutions to the dual program (8) is nonempty [16, Proposition 5.3.1] and as such the optimum is attained. To ensure that the set of dual optimizers is compact, we need to impose the following assumption on the channel matrix W, that we will maintain for the remainder of Section 2.

Assumption 2.3. $\gamma := \min\limits_{i,j} \mathsf{W}_{ij} > 0$

Assumption 2.3 excludes situations where the channel matrix has zero entries. Even though this may seem restrictive at first glance, it holds for a large class of channels. Moreover, in a finite dimensional setting, for a fixed input distribution, the mutual information is well known to be continuous in the channel matrix entries. Therefore, singular cases where the channel matrix contains zero entries can be avoided by slight perturbations of those entries. Under Assumption 2.3 for a fixed channel, the mutual information can be seen to be a strictly concave function in the input distribution. Therefore, the capacity achieving input distribution is unique. With Assumption 2.3 one can derive an explicit bound on the norm of the dual optimizers, which is crucial in the subsequent derivation of the main result in this section, namely Theorem 2.9.

Lemma 2.4. *Under Assumption 2.3, the dual program (8) is equivalent to*

$$\begin{cases} \min\limits_{\lambda} & G(\lambda) + F(\lambda) \\ \text{s.t.} & \lambda \in Q, \end{cases} \tag{10}$$

where $Q := \left\{ \lambda \in \mathbb{R}^M \ : \ \|\lambda\|_2 \leq M \left(\log(\gamma^{-1}) \vee \frac{1}{\ln 2} \right) \right\}$.

Proof. See Appendix A 3. □

Lemma 2.5. *Strong duality holds between (6) and (8).*

Proof. The proof follows by a standard strong duality result of convex optimization, see [16, Proposition 5.3.1, p. 169]. □

Note that the optimization problem defining $F(\lambda)$ is of the form given in (7). Hence, according to Lemma 2.2, $F(\lambda)$ has a unique optimizer $q^\star$ with components $q_j^\star = 2^{\mu - \lambda_j}$, where $\mu \in \mathbb{R}$ needs to be chosen such that $q^\star \in \Delta_M$, i.e.,

$$\mu = -\log\left(\sum_{i=1}^{M} 2^{-\lambda_i} \right).$$

Therefore,

$$F(\lambda) = \sum_{i=1}^{M} \left(-q_i^\star \log(q_i^\star) - \lambda_i q_i^\star \right) = -\sum_{i=1}^{M} \mu \, 2^{\mu - \lambda_i} = -\mu \, 2^\mu \sum_{i=1}^{M} 2^{-\lambda_i} = \log \left(\sum_{i=1}^{M} 2^{-\lambda_i} \right). \tag{11}$$

$F(\lambda)$ is a smooth function with gradient

$$(\nabla F(\lambda))_i = \frac{-2^{-\lambda_i}}{\sum_{j=1}^{M} 2^{-\lambda_j}}. \tag{12}$$

According to [11, Theorem 1] and the fact that the negative entropy is strongly convex with convexity parameter 1 [11, Lemma 3], $\nabla F(\lambda)$ is Lipschitz continuous with Lipschitz constant 1. The main difficulty in solving (10) efficiently is that $G(\cdot)$ is non-smooth. Following Nesterov's smoothing technique [11], we alleviate this difficulty by approximating $G(\cdot)$ by a function with a Lipschitz continuous gradient. This smoothing step is efficient in our case because of the particular structure of (10). Following [11], consider

$$G_\nu(\lambda) = \begin{cases} \max_{p} & \lambda^\top \mathsf{W}^\top p - r^\top p + \nu H(p) - \nu \log(N) \\ \text{s.t.} & s^\top p = S \\ & p \in \Delta_N, \end{cases} \tag{13}$$

with smoothing parameter $\nu \in \mathbb{R}_{>0}$ and denote by $p_\nu(\lambda)$ the optimizer to (13), which is unique because the objective function is strictly concave. Clearly for any $p \in \Delta_N$, $G_\nu(\lambda)$ is a uniform approximation of the non-smooth function $G(\lambda)$, since $G_\nu(\lambda) \leq G(\lambda) \leq G_\nu(\lambda) + \nu \log(N)$. Using Lemma 2.2, the optimizer $p_\nu(\lambda)$ to (13) is analytically given by

$$p_\nu(\lambda, \mu)_i = 2^{\mu_1 + \frac{1}{\nu}(\mathsf{W}\lambda - r)_i + \mu_2 s_i}, \tag{14}$$

where $\mu_1, \mu_2 \in \mathbb{R}$ have to be chosen so that $s^\top p_\nu(\lambda, \mu) = S$ and $p_\nu(\lambda, \mu) \in \Delta_N$; for this choice of μ_1, μ_2 we denote the solution by $p_\nu(\lambda)$.

Remark 2.6. In case of no input constraints, the unique optimizer to (13) is given by

$$p_\nu(\lambda)_i = \frac{2^{\frac{1}{\nu}(\mathsf{W}\lambda - r)_i}}{\sum_{i=1}^{N} 2^{\frac{1}{\nu}(\mathsf{W}\lambda - r)_i}} \quad \text{for } i = 1, \dots, N,$$

whose straightforward evaluation is numerically difficult for small ν. One can circumvent this problem, however, by following the numerically stable technique that we present in Remark 2.11.

Remark 2.7. In case of additional input constraints, we need an efficient method to find the coefficients μ_1 and μ_2 in (14). In particular if there are multiple input constraints (leading to multiple μ_i) the efficiency of the method computing them becomes important. Instead of solving a system of non-linear equations, one can show that the μ_i can be found by solving the following convex optimization problem [17, p. 257 ff.]

$$\sup_{\mu \in \mathbb{R}^2} \left\{ y^\top \mu - \sum_{i=1}^{N} p_\nu(\lambda, \mu)_i \right\}, \tag{15}$$

where $y := (1, S)$. Note that (15) is an unconstrained maximization of a concave function, whose gradient and Hessian can be easily computed, allowing the use of efficient second-order methods.

Finally, we can show that the uniform approximation $G_\nu(\lambda)$ is smooth and has a Lipschitz continuous gradient, with known Lipschitz constant.

Proposition 2.8. $G_\nu(\lambda)$ *is well defined and continuously differentiable at any* $\lambda \in Q$. *Moreover, it is convex and its gradient* $\nabla G_\nu(\lambda) = \mathsf{W}^\top p_\nu(\lambda)$ *is Lipschitz continuous with Lipschitz constant* $\frac{1}{\nu}$.

Proof. The proof follows directly from the proof of Theorem 1 and Lemma 3 in [11] together with (4). $\square$

We consider the smooth, convex optimization problem

$$\mathsf{D}_\nu : \quad \begin{cases} \min_{\lambda} & F(\lambda) + G_\nu(\lambda) \\ \text{s.t.} & \lambda \in Q, \end{cases} \tag{16}$$

whose objective function has a Lipschitz continuous gradient with Lipschitz constant $1 + \frac{1}{\nu}$. As such D_ν can be be approximated with Nesterov's optimal scheme for smooth optimization [11], which is summarized in Algorithm 1, where $\pi_Q(x)$ denotes the projection operator of the set Q, defined in Lemma 2.4, with $R := M \left(\log(\gamma^{-1}) \vee \frac{1}{\ln 2} \right)$

$$\pi_Q(x) := \begin{cases} R \frac{x}{\|x\|_2}, & \|x\|_2 > R \\ x, & \text{otherwise.} \end{cases}$$

Algorithm 1: Optimal scheme for smooth optimization

Choose some $\lambda_0 \in Q$

For $k \geq 0$ **do*** **Step 1:** Compute $\nabla F(x_k) + \nabla G_\nu(x_k)$
Step 2: $y_k = \pi_Q \left(-\frac{1}{L_\nu} \left(\nabla F(x_k) + \nabla G_\nu(x_k) \right) + x_k \right)$
Step 3: $z_k = \pi_Q \left(-\frac{1}{L_\nu} \sum_{i=0}^{k} \frac{i+1}{2} \left(\nabla F(x_i) + \nabla G_\nu(x_i) \right) \right)$
Step 4: $x_{k+1} = \frac{2}{k+3} z_k + \frac{k+1}{k+3} y_k$

[*The stopping criterion is explained in Remark 2.10]

The following theorem provides explicit error bounds for the solution provided by Algorithm 1 after n iterations. Define the constants $D_1 := \frac{1}{2} (M \log(\gamma^{-1}) \vee \frac{1}{\ln 2})^2$ and $D_2 := \log(N)$.

Theorem 2.9 ([11]). *Under Assumption 2.3, for* $n \in \mathbb{N}$ *consider a smoothing parameter*

$$\nu = \nu(n) = \frac{2}{n+1} \sqrt{\frac{D_1}{D_2}}.$$

Then after n *iterations of Algorithm 1 we can generate the approximate solutions to the problems* (8) *and* (2), *namely,*

$$\hat{\lambda} = y_n \in Q \qquad \text{and} \qquad \hat{p} = \sum_{k=0}^{n} \frac{2(k+1)}{(n+1)(n+2)} p_\nu(x_k) \in \Delta_N, \tag{17}$$

which satisfy

$$0 \leq F(\hat{\lambda}) + G(\hat{\lambda}) - I(\hat{p}, W) \leq \frac{4}{n+1} \sqrt{D_1 D_2} + \frac{4 D_1}{(n+1)^2}. \tag{18}$$

Thus, the complexity of finding an ε-*solution to the problems* (8) *and* (2) *does not exceed*

$$4\sqrt{D_1 D_2} \, \frac{1}{\varepsilon} + 2\sqrt{\frac{D_1}{\varepsilon}}. \tag{19}$$

Proof. The proof follows along the lines of [11, Theorem 3] and in particular requires Lemma 2.4, Lemma 2.5 and Proposition 2.8. □

Note that Theorem 2.9 provides an explicit error bound (18), also called *a priori error*. In addition this theorem gives an approximation to the optimal input distribution (17), i.e., the optimizer of the primal problem. Thus, by comparing the values of the primal and the dual optimization problem, one can also compute an *a posteriori error* which is the difference of the dual and the primal problem, namely $F(\hat{\lambda}) + G(\hat{\lambda}) - I(\hat{p}, W)$.

Remark 2.10 (Stopping criterion of Algorithm 1). There are two immediate approaches to define a stopping criterion for Algorithm 1.

(i) *A priori stopping criterion*: Choose an a priori error $\varepsilon > 0$. Setting the right hand side of (18) equal to ε defines a number of iterations n_ε required to ensure an ε-close solution.

(ii) *A posteriori stopping criterion*: Choose an a posteriori error $\varepsilon > 0$. Choose the smoothing parameter $\nu(n_\varepsilon)$ for n_ε as defined above in the a priori stopping criterion. Fix a (small) number of iterations ℓ that are run using Algorithm 1. Compute the a posteriori error $e_\ell := F(\hat{\lambda}) + G(\hat{\lambda}) - I(\hat{p}, \rho)$ according to Theorem 2.9. If $e_\ell \leq \varepsilon$ terminate the algorithm otherwise continue with another ℓ iterations. Continue until the a posteriori error is below ε.

Remark 2.11 (Computational stability). In the special case of no input cost constraints, one can derive an analytical expression for $G_\nu(\lambda)$ and its gradient as

$$G_\nu(\lambda) = \nu \log \left(\sum_{i=1}^{N} 2^{\frac{1}{\nu}(\mathsf{W}\lambda - r)_i} \right) - \nu \log(N)$$

$$\nabla G_\nu(\lambda) = \frac{1}{S(\lambda)} \sum_{i=1}^{N} 2^{\frac{1}{\nu}(\mathsf{W}\lambda - r)_i} \mathsf{W}_{i,\cdot}, \tag{20}$$

where $S(\lambda) := \sum_{i=1}^{N} 2^{\frac{1}{\nu}(\mathsf{W}\lambda - r)_i}$. In order to achieve an ε-precise solution the smoothing factor ν has to be chosen in the order of ε, according to Theorem 2.9. A straightforward computation of $\nabla G_\nu(\lambda)$ via (20) for a small enough ν is numerically difficult. In the light of [11, p. 148], we present a numerically stable technique for computing $\nabla G_\nu(\lambda)$. By considering the functions $\mathbb{R}^M \ni \lambda \mapsto f(\lambda) = \mathsf{W}\lambda - r \in \mathbb{R}^N$ and $\mathbb{R}^N \ni x \mapsto R_\nu(x) = \nu \log \left(\sum_{i=1}^{N} 2^{\frac{x_i}{\nu}} \right) \in \mathbb{R}$ it is clear that $\nabla_\lambda R_\nu(f(\lambda)) = \nabla G_\nu(\lambda)$. The basic idea is to define $\bar{f}(\lambda) := \max_{1 \leq i \leq N} f_i(\lambda)$ and then consider a function $g : \mathbb{R}^M \to \mathbb{R}^N$ given by $g_i(\lambda) = f_i(\lambda) - \bar{f}(\lambda)$, such that all components of $g(\lambda)$ are non-positive. One can show that

$$\nabla_\lambda R_\nu(f(\lambda)) = \nabla_\lambda R_\nu(g(\lambda)) + \nabla \bar{f}(\lambda),$$

where the term on the right-hand side can be computed with a small numerical error.

Remark 2.12 (Computational complexity). In case of no input cost constraint, one can see by (20) that the computational complexity of a single iteration step of Algorithm 1 is $O(MN)$. Furthermore, according to (19), the complexity in terms of number of iterations to achieve an ε-precise solution is $O\left(\frac{M\sqrt{\log N}}{\varepsilon} \right)$. This finally gives a computational complexity for finding an ε-solution of $O(\frac{M^2 N \sqrt{\log N}}{\varepsilon})$.

B. Simulation Results

This section presents two examples to illustrate the theoretical results developed in the preceding sections and their performance. All the simulations in this section are performed on a 2.3 GHz Intel Core i7 processor with 8 GB RAM.

Example 2.13. Consider a DMC W having a channel matrix $\mathsf{W} \in \mathbb{R}^{N \times M}$ with $N = 10000$ and $M = 100$, such that $\mathsf{W}_{ij} = \frac{V_{ij}}{\sum_{j=1}^{M} V_{ij}}$, where V_{ij} is chosen i.i.d. uniformly distributed in $[0, 1]$ for all $1 \leq i \leq N$ and $1 \leq j \leq M$. The parameter γ happens to be $1.0742 \cdot 10^{-8}$. Figure 2.13 and Table I compare the performance of the Blahut-Arimoto algorithm with that of Algorithm 1, which has the a priori error bound predicted by Theorem 2.9, namely

$$C_{\mathrm{UB}}(W) - C_{\mathrm{LB}}(W) \leq \frac{2M\sqrt{2\log(N)}}{n+1}\left(\log(\gamma^{-1}) \vee \tfrac{1}{\ln 2})\right) + \frac{2M^2}{(n+1)^2}\left(\log(\gamma^{-1}) \vee \tfrac{1}{\ln 2})\right)^2,$$

where n denotes the number of iterations and γ is equal to the smallest entry in the channel matrix W. Recall that the Blahut-Arimoto algorithm has an a priori error bound of the form $C(W) - C_{\mathrm{LB}}(W) \leq \frac{\log(N)}{n}$ [4, Corollary 1]. Moreover, the new method provides us with an a posteriori error, which the Blahut-Ariomoto algorithm does not.
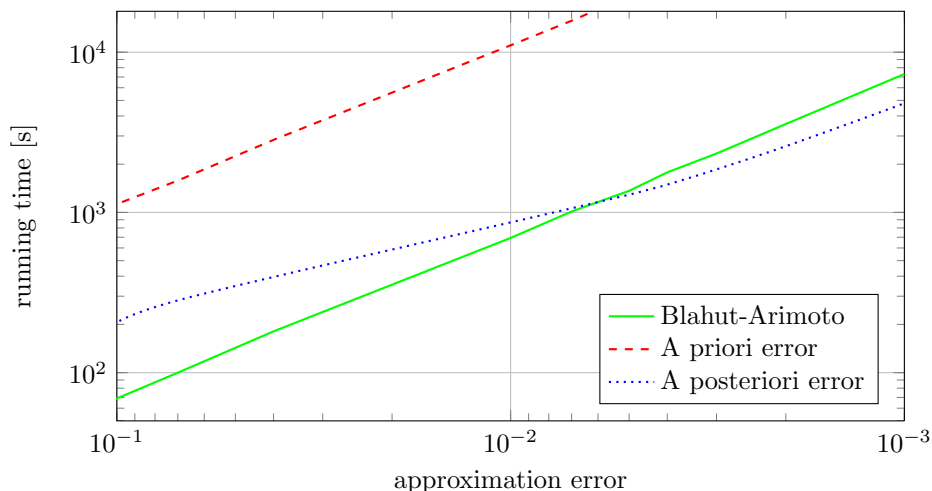

FIG. 1. For Example 2.13, this plot depicts the runtime of Algorithm 1 with respect to the a priori and a posteriori stopping criterion, as explained in Remark 2.10. As a reference, the runtime of the Blahut-Arimoto algorithm is shown.

TABLE I. Some specific simulation points of Example 2.13.

	Blahut-Arimoto Algorithm				Fast-Gradient Method			
A priori error	1	0.1	0.01	0.001	1	0.1	0.01	0.001
$C_{\mathrm{UB}}(W)$	—	—	—	—	0.4419	0.4131	0.4092	0.4088
$C_{\mathrm{LB}}(W)$	0.2930	0.4008	0.4088	0.4088	0.3094	0.4069	0.4088	0.4088
A posteriori error	—	—	—	—	0.1325	0.0063	$4.0\cdot10^{-4}$	$3.7\cdot10^{-5}$
Time [s]	7.4	69	693	7306	114	1127	11 036	110 987
Iterations	14	133	1329	13 288	27 797	273 447	2 729 860	27 294 000

Example 2.14. Consider a DMC W having a channel matrix $\mathsf{W} \in \mathbb{R}^{N \times M}$ with $N = 100000$ and $M = 10$, such that $\mathsf{W}_{ij} = \frac{V_{ij}}{\sum_{j=1}^{M} V_{ij}}$, where V_{ij} is chosen i.i.d. uniformly distributed in $[0,1]$ for all $1 \leq i \leq N$ and $1 \leq j \leq M$. The parameter γ happens to be $3.7140 \cdot 10^{-7}$. Table II shows the performance of Algorithm 1 with a priori and a posteriori errors given by Theorem 2.9. Note that for the Blahut-Arimoto algorithm we were not able to run a single iteration, as the computation run out of memory.

TABLE II. Some specific simulation points of Example 2.14.

A priori error	1	0.1	0.01	0.001
$C_{\mathrm{UB}}(W)$	1.0938	1.0543	1.0516	1.0514
$C_{\mathrm{LB}}(W)$	1.0100	1.0512	1.0514	1.0514
A posteriori error	0.0837	0.0031	$2.5 \cdot 10^{-4}$	$2.4 \cdot 10^{-5}$
Time [s]	46	456	4553	45617
Iterations	2498	24659	246264	2462310

3. CHANNELS WITH CONTINUOUS INPUT AND COUNTABLE OUTPUT ALPHABETS

In this section we generalize the approximation scheme introduced in Section 2 to memoryless channels with continuous input and countable output alphabets. The class of discrete-time Poisson channels is an example of such channels with particular interest in applications, for example to model direct detection optical communication systems [18, 19]. Consider $\mathcal{X} \subseteq \mathbb{R}$ as the input alphabet set and $\mathcal{Y} = \mathbb{N}_0$ as the output alphabet set. The channel is described by the conditional probability $W(i|x) := \mathbb{P}[Y = i \mid X = x]$. Given a channel W and an integer M, we introduce an M-truncated version of the channel by

$$W_M(i|x) := \begin{cases} W(i|x) + \frac{1}{M} \sum_{j \geq M} W(j|x), & i \in \{0, 1, \ldots, M-1\} \\ 0, & i \geq M. \end{cases} \tag{21}$$

W_M can be seen as a channel with input alphabet $\mathcal{X}$ and output alphabet $\{0, 1, \ldots, M-1\}$. Figure 2 shows a pictorial representation of a channel and its M-truncated counterpart. The finiteness of the output alphabet of W_M allows us to deploy an approximation scheme similar to the one developed in Section 2 to numerically approximate $C(W_M)$.
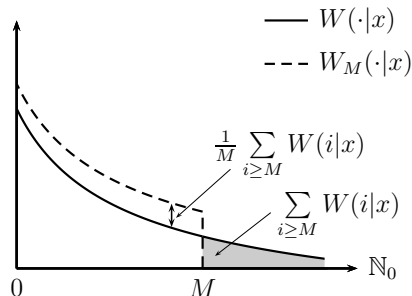

FIG. 2. Pictorial representation of the M-truncated channel counterpart

The following definition is a key feature of the channel required for the theoretical results developed in this section which, roughly speaking, imposes a certain decay rate for the output distribution uniformly in the input alphabet.

Definition 3.1 (Polynomial tail)**.** The channel W features a *k-ordered polynomial tail* if for $M \in \mathbb{N}_0$ and $k \in \mathbb{R}_{\geq 0}$

$$R_k(M) := \sum_{i \geq M} \big(\sup_{x \in \mathcal{X}} W(i|x) \big)^k < \infty. \tag{22}$$

The following assumptions hold throughout this section.

Assumption 3.2.

  (i) The channel W has a k-ordered polynomial tail for some $k \in (0, 1)$ in the sense of Definition 3.1.

  (ii) The mapping $x \mapsto W(i|x)$ is Lipschitz continuous for any $i \in \mathbb{N}_0$ with Lipschitz constant L.

Assumption 3.2 allows us to relate the capacity of the original channel to that of its truncated counterpart.

Theorem 3.3. *Suppose channel W satisfies Assumption 3.2(i) with the order $k \in (0, 1)$. Then, for any $M \in \mathbb{N}_0$ we have*

$$\big| C(W) - C(W_M) \big| \leq \frac{2 \log(\mathrm{e})}{\mathrm{e}(1 - k)} \Big[M^{1-k} \big(R_1(M) \big)^k + R_k(M) \Big],$$

where $R_k(M)$ is as defined in (22).

Proof. See Appendix A 4. ☐

We consider two types of input cost constraints: a peak-power constraint $\mathbb{P}[X \in \mathbb{A}] = 1$ for a compact set $\mathbb{A} \subseteq \mathcal{X}$ and an average-power constraint $\mathbb{E}[s(X)] \leq S$ for $S \in \mathbb{R}_{\geq 0}$ and a continuous function s on $\mathcal{X}$. The primal capacity problem for the channel W_M is given by

$$C_{\mathbb{A},S}(W_M) = \begin{cases} \sup\limits_{p} \ I(p, W_M) \\ \mathrm{s.\,t.} \quad \mathbb{E}[s(X)] \leq S \\ \qquad p \in \mathcal{P}(\mathbb{A}), \end{cases} \tag{23}$$

where $\mathcal{P}(\mathbb{A})$ denotes the space of all probability distributions supported on $\mathbb{A}$, cf. (3). Our method always requires a peak-power constraint, whereas the average-power constraint is optimal. The following proposition allows us to restrict the optimization variables from probability distributions to probability densites.

Proposition 3.4. *The optimization problem (23) is equivalent to*

$$C_{\mathbb{A},S}(W_M) = \begin{cases} \sup\limits_{p} \ I(p, W_M) \\ \mathrm{s.\,t.} \quad \mathbb{E}[s(X)] \leq S \\ \qquad p \in \mathcal{D}(\mathbb{A}), \end{cases}$$

where $\mathcal{D}(\mathbb{A})$ is the set of probability densities functions, i.e., $\mathcal{D}(\mathbb{A}) := \{ f \in \mathrm{L}^1(\mathbb{A}) \ : \ f \geq 0, \int_{\mathbb{A}} f(x) \mathrm{d}x = 1 \}$.

Proof. See Appendix A 5. ☐

We consider the pair of vector spaces $(L^1(\mathbb{A}), L^\infty(\mathbb{A}))$ along with the bilinear form

$$\langle f, g \rangle := \int_{\mathcal{X}} f(x)g(x)\mathrm{d}x.$$

In the light of [20, Theorem 243G] this is a dual pair of vector spaces; we refer to [21, Section 3] for the details of the definition of dual pairs of vector spaces. Considering the standard inner product as a bilinear form on the dual pair $(\mathbb{R}^M, \mathbb{R}^M)$, we define the linear operator $\mathcal{W} : \mathbb{R}^M \to L^\infty(\mathbb{A})$ and its adjoint operator $\mathcal{W}^\star : L^1(\mathbb{A}) \to \mathbb{R}^M$, given by

$$\mathcal{W}\lambda(x) := \sum_{i=1}^M W_M(i-1|x)\lambda_i, \qquad (\mathcal{W}^\star p)_i := \int_X W_M(i-1|x)p(x)\mathrm{d}x.$$

Let $S_{\max} := \inf_{p \in \mathcal{D}(\mathbb{A})} \{\langle p, s \rangle : I(p, W_M) = \sup_{q \in \mathcal{D}(\mathbb{A})} I(q, W_M)\}$. Following similar lines as in Lemma 2.1, one can deduce that in problem (23) the inequality input constraint can be replaced by equality (resp. removed) is $S < S_{\max}$ (resp. $S \geq S_{\max}$). That is, in view of Proposition 3.4, Lemma 2.1 and the discussion there, problem (23) (under Assumption 3.6, that we require later) is equivalent to

$$\mathsf{P}: \quad \begin{cases} \displaystyle\sup_{p,q} & -\langle p, r \rangle + H(q) \\ \text{s.t.} & \mathcal{W}^\star p = q \\ & \langle p, s \rangle = S \\ & p \in \mathcal{D}(\mathbb{A}), \ q \in \Delta_M, \end{cases} \tag{24}$$

where $r(\cdot) := -\sum_{j=0}^{M-1} W_M(j|\cdot) \log(W_M(j|\cdot))$ is an element in $L^\infty(\mathbb{A})$ by Assumption 3.2(ii). For the rest of the section we restrict attention to (24), since unconstrained problem can be solved in a similar way. We call (24) the primal program. Thanks to the dual vector space framework, the Lagrange dual program of P is given by

$$\mathsf{D}: \quad \begin{cases} \displaystyle\inf_\lambda & G(\lambda) + F(\lambda) \\ \text{s.t.} & \lambda \in \mathbb{R}^M, \end{cases} \tag{25}$$

where

$$G(\lambda) = \begin{cases} \displaystyle\sup_p & \langle p, \mathcal{W}\lambda \rangle - \langle p, r \rangle \\ \text{s.t.} & \langle p, s \rangle = S \\ & p \in \mathcal{D}(\mathbb{A}) \end{cases} \quad \text{and} \quad F(\lambda) = \begin{cases} \displaystyle\max_q & H(q) - \lambda^\top q \\ \text{s.t.} & q \in \Delta_M. \end{cases}$$

Lemma 3.5. *Strong duality holds between* (24) *and* (25).

Proof. The proof follows from standard strong duality results of convex optimization, see [22, Theorem 6]. $\qquad\square$

In the remainder of this article we impose the following assumption on the channel.

Assumption 3.6. $\gamma_M := \displaystyle\min_{y \in \{0,1,\ldots,M-1\}} \min_{x \in \mathbb{A}} W_M(y|x) > 0$

In case $\sum_{j \geq M} W(j|x) > 0$ for all x, Assumption 3.6 holds according to (21) and a lower bound can be given by $\gamma_M \geq \frac{1}{M} \min_x \sum_{j \geq M} W(j|x)$. Under Assumption 3.6 we can show that we can again assume without loss of generality that λ takes values in a compact set.

Lemma 3.7. *Under Assumption 3.6, the dual program* (25) *is equivalent to*

$$\begin{cases} \min\limits_{\lambda} & G(\lambda) + F(\lambda) \\ \text{s.t.} & \lambda \in Q, \end{cases}$$

where $Q := \left\{ \lambda \in \mathbb{R}^M \; : \; \|\lambda\|_2 \leq M \left(\log(\gamma_M^{-1}) \vee \frac{1}{\ln 2} \right) \right\}$.

Proof. The proof follows the same lines as in the proof of Lemma 2.4. $\qquad\square$

Note that $F(\lambda)$ is the same as in Section 2 and therefore given by (11) and its gradient by (12). As in Section 2, we consider the smooth approximation

$$G_\nu(\lambda) = \begin{cases} \sup\limits_{p} & \langle p, \mathcal{W}\lambda \rangle - \langle p, r \rangle + \nu h(p) - \nu \log(\rho) \\ \text{s.t.} & \langle p, s \rangle = S \\ & p \in \mathcal{D}(\mathbb{A}), \end{cases} \tag{26}$$

with smoothing parameter $\nu \in \mathbb{R}_{>0}$ and ρ denoting the Lebesgue measure of $\mathbb{A}$. To analyze the properties of $G_\nu(\lambda)$ we need one more auxiliary lemma.

Lemma 3.8. *The function* $\mathcal{D}(\mathbb{A}) \ni p \mapsto -h(p) + \log(\rho) \in \mathbb{R}_{\geq 0}$ *is strongly convex with convexity parameter* $\sigma = 1$.

Proof. See Appendix A 6. $\qquad\square$

Furthermore, we can show that the uniform approximation $G_\nu(\lambda)$ is smooth and has a Lipschitz continuous gradient, with known constant. The following result is a generalization of Proposition 2.8.

Proposition 3.9. $G_\nu(\lambda)$ *is well defined and continuously differentiable at any* $\lambda \in \mathbb{R}^M$. *Moreover, this function is convex and its gradient* $\nabla G_\nu(\lambda) = \mathcal{W}^\star p_\nu^\lambda$ *is Lipschitz continuous with constant* $L_\nu = \frac{1}{\nu}$.

Proof. See Appendix A 7. $\qquad\square$

We denote by p_ν^λ the optimizer to (26), that is unique since the objective function is strictly concave. To analyze the solution to (26) we consider the following optimization problem, that, if feasible, has a closed form solution

$$\begin{cases} \sup\limits_{p} & h(p) + \langle p, c \rangle \\ \text{s.t.} & \langle p, s \rangle = S \\ & p \in \mathcal{D}(\mathbb{A}), \end{cases} \tag{27}$$

with $c, s \in \mathrm{L}^\infty(\mathbb{A})$.

Lemma 3.10. *Let* $p^\star(x) = 2^{\mu_1 + c(x) + \mu_2 s(x)}$, *where* $\mu_1, \mu_2 \in \mathbb{R}$ *are chosen such that* $p^\star$ *satisfies the constraints in* (27). *Then* $p^\star$ *uniquely solves* (27).

The proof directly follows from [23, p. 409] and the proof of Lemma 2.2. Hence, $G_\nu(\lambda)$ has a (unique) analytical optimizer

$$p_\nu^\lambda(x, \mu) = 2^{\mu_1 + \frac{1}{\nu}(\mathcal{W}\lambda(x) - r(x)) + \mu_2 s(x)}, \quad x \in \mathcal{X}, \tag{28}$$

where $\mu_1, \mu_2 \in \mathbb{R}$ have to be chosen such that $\langle p_\nu^\lambda(\cdot, \mu), s \rangle = S$ and $p_\nu^\lambda(\cdot, \mu) \in \mathcal{D}(\mathbb{A})$; for this choice of μ_1, μ_2 we denote the solution by $p_\nu^\lambda(\cdot)$.

Remark 3.11. In case of no input constraints, the unique optimizer to (26) is given by

$$p_\nu^\lambda(x) = \frac{2^{\frac{1}{\nu}(\mathcal{W}\lambda(x)-r(x))}}{\int_\mathbb{A} 2^{\frac{1}{\nu}(\mathcal{W}\lambda(x)-r(x))}\mathrm{d}x},$$

whose numerical evaluation can be done in a stable way by following Remark 2.11.

Remark 3.12. As in Remark 2.7, in case of additional input constraints we need an efficient method to find the coefficients μ_i in (28). This problem can again be reduced to the finite dimensional convex optimization problem [17, p. 257 ff.]

$$\sup_{\mu\in\mathbb{R}^2}\left\{\langle y,\mu\rangle - \int_\mathbb{A} p_\nu^\lambda(x,\mu)\mathrm{d}x\right\}, \tag{29}$$

where $y := (1, S)$ and $\mu := (\mu_1, \mu_2)$. Note that (29) is an unconstrained maximization of a concave function. Unlike the finite input alphabet case, the evalutation of the gradient and the Hessian of this function involves computing moments of the measure $p_\nu^\lambda(x, \mu)\mathrm{d}x$, for which efficient numerical schemes based on semidefinite programming have been proposed; see [17, p. 259 ff.] for details.

Note that the differential entropy $h(p) \le \log(\rho)$ for all $p \in \mathcal{D}(\mathbb{A})$ and that there exists a function $\iota : \mathbb{R}_{>0} \to \mathbb{R}_{\ge 0}$ such that

$$G_\nu(\lambda) \le G(\lambda) \le G_\nu(\lambda) + \iota(\nu) \text{ for all } \lambda \in Q, \tag{30}$$

i.e., $G_\nu(\lambda)$ is a uniform approximation of the non-smooth function $G(\lambda)$. The following lemma, Lemma 3.15, provides an explicit expression for the function ι in (30) under some Lipschitz continuity assumptions, implying in particular that $\iota(\nu) \to 0$ as $\nu \to 0$.

Lemma 3.13. *Under Assumption 3.2(ii) and Assumption 3.6 the function $f_\lambda(\cdot) := \mathcal{W}\lambda(\cdot) - r(\cdot)$ is Lipschitz continuous uniformly in $\lambda \in Q$ with constant $L_f = LM^2(\log\frac{1}{\gamma_M} \vee \frac{1}{\ln 2}) + ML|\log\frac{1}{\gamma_M} - \frac{1}{\ln 2}|$.*

Proof. See Appendix A 8. □

Assumption 3.14 (Lipschitz continuity of the average-power constraint function)**.** The average-power constraint function $s(\cdot)$ is Lipschitz continuous with constant L_s.

Lemma 3.15. *Under Assumptions 3.2(ii), 3.6 and 3.14 a possible choice of the function ι in (30) is given by*

$$\iota(\nu) = \begin{cases} \nu\left(\log\left(\frac{T_1}{\nu} + T_2\right) + 1\right), & \nu < \frac{T_1}{1-T_2} \text{ or } T_2 > 1 \\ \nu, & \text{otherwise,} \end{cases}$$

where $T_1 := L_f\rho + 2L_fL_s\rho^2\left(\frac{1}{-\underline{s}} \vee \frac{1}{\overline{s}}\right)$, $T_2 := L_s\rho(\underline{\mu} \vee \overline{\mu})$, $\underline{\mu} := \frac{2}{-\underline{s}}\log\left(\frac{2L_s\rho}{-\underline{s}} \vee 1\right)$, $\overline{\mu} := \frac{2}{\overline{s}}\log\left(\frac{2L_s\rho}{\overline{s}} \vee 1\right)$, $\rho := \int_\mathbb{A} \mathrm{d}x$, $\underline{s} := -S + \min_{x\in\mathbb{A}} s(x)$ and $\overline{s} := -S + \max_{x\in\mathbb{A}} s(x)$.

Proof. See Appendix A 9. □

We consider the smooth, finite dimensional, convex optimization problem

$$\mathsf{D}_\nu : \begin{cases} \inf_\lambda & F(\lambda) + G_\nu(\lambda) \\ \text{s.t.} & \lambda \in Q, \end{cases} \tag{31}$$

whose solution can be approximated with Algorithm 1 presented in Section 2, as follows. Define the constant $D_1 := \frac{1}{2}(M\log(\gamma^{-1}) \vee \frac{1}{\ln 2})^2$.

Theorem 3.16. *Under Assumptions 3.2(ii), 3.6 and 3.14, let $\alpha := 2(T_1 + T_2 + 1)$ where T_1 and T_2 are as defined in Lemma 3.15. Given precision $\varepsilon \in (0, \frac{\alpha}{4})$, we set the smoothing parameter $\nu = \frac{\varepsilon/\alpha}{\log(\alpha/\varepsilon)}$ and number of iterations $n \geq \frac{1}{\varepsilon}\sqrt{8D_1\alpha}\sqrt{\log(\varepsilon^{-1}) + \log(\alpha) + \frac{1}{4}}$. Consider*

$$\hat{\lambda} = y_n \in Q \qquad and \qquad \hat{p} = \sum_{k=0}^{n} \frac{2(i+1)}{(n+1)(n+2)}p_\nu^{x_k} \in \mathcal{D}(\mathbb{A}), \tag{32}$$

where y_k computed at the kth iteration of Algorithm 1 and $p_\nu^{x_k}$ is the analytical solution in (28). Then, $\hat{\lambda}$ and $\hat{p}$ are the approximate solutions to the problems (25) and (23), i.e.,

$$0 \leq F(\hat{\lambda}) + G(\hat{\lambda}) - I(\hat{p}, W_M) \leq \varepsilon. \tag{33}$$

Therefore, Algorithm 1 requires $O\left(\frac{1}{\varepsilon}\sqrt{\log\left(\varepsilon^{-1}\right)}\right)$ iterations to find an ε-solution to the problems (25) and (23).

Proof. See Appendix A 10. □

Hence, under Assumption 3.14 we can quantify the approximation error of the presented method to find the capacity of any channel W, satisfying Assumptions 3.2 and 3.6, by

$$\left|C(W) - C_{\text{approx}}^{(n)}(W_M)\right| \leq \underbrace{|C(W) - C(W_M)|}_{(\star)} + \underbrace{\left|C(W_M) - C_{\text{approx}}^{(n)}(W_M)\right|}_{(\star\star)},$$

where $(\star)$ and $(\star\star)$ are addressed by Theorem 3.3 and Theorem 3.16, respectively. Let us highlight that for the term $(\star\star)$ we have two different quantitative bounds: First, the *a priori* bound ε for which Theorem 3.16 prescribes a lower bound for the required number of iterations; second, the *a posteriori* bound $F(\hat{\lambda}) + G(\hat{\lambda}) - I(\hat{p}, W_M)$ which can be computed after a number of iterations have been executed. In practice, the a posteriori bound often approaches ε much faster than the a priori bound. Note also that by (30) and Theorem 3.16

$$0 \leq F(\hat{\lambda}) + G_\nu(\hat{\lambda}) + \iota(\nu) - I(\hat{p}, W_M) \leq \iota(\nu) + \varepsilon,$$

which shows that $F(\hat{\lambda}) + G_\nu(\hat{\lambda}) + \iota(\nu)$ is an upper bound for the channel capacity with a priori error $\iota(\nu) + \varepsilon$. This bound can be particularly helpful in cases where an evaluation of $G(\lambda)$ for a given λ is hard.

Remark 3.17 (Optimal tail truncation)**.** Given a fixed number of iterations, the term $(\star\star)$ above is effected by the truncation level M for two reasons: the higher M the larger the size of the output as well as the lower the parameter γ_M. Therefore, term $(\star\star)$ increases as M increases, which can be quantified by (A19). On the other hand, term $(\star)$ obviously has the opposite behavior. Namely, the higher M leads to the better approximation of the channel W by the truncated version W_M as quantified in Theorem 3.3. Hence, given a channel W with the polynomial tail order k, there is an optimal value for the truncation parameter M, which thanks to the monotonicity explained above can be effectively computed in practice by techniques such as bisection.

Note, that this truncation procedure could also be applied to a finite output alphabet, given that the channel satisfies Assumption 3.2(i), and for example improve the performance of the method presented in Section 2.

Remark 3.18 (Without average-power constraint). In case of considering only a peak-power constraint and no average-power constraint, our proposed methodology allows us to access a closed form expression for $G_\nu(\lambda)$ and its gradient,

$$G_\nu(\lambda) = \nu \log \left(\int_{\mathbb{A}} 2^{\frac{1}{\nu}(\mathcal{W}\lambda(x)-r(x))} \mathrm{d}x \right) - \nu \log(\rho) \tag{34}$$

$$\nabla G_\nu(\lambda) = \frac{\int_{\mathbb{A}} 2^{\frac{1}{\nu}(\mathcal{W}\lambda(x)-r(x))} W_M(\cdot|x) \mathrm{d}x}{\int_{\mathbb{A}} 2^{\frac{1}{\nu}(\mathcal{W}\lambda(x)-r(x))} \mathrm{d}x}.$$

Discrete-Time Poisson Channel

The discrete-time Poisson channel is a mapping from $\mathbb{R}_{\geq 0}$ to $\mathbb{N}_0$, such that conditioned on the input $x \geq 0$ the output is Poisson distributed with mean $x + \eta$, i.e.,

$$W(y|x) = e^{-(x+\eta)} \frac{(x+\eta)^y}{y!}, \quad y \in \mathbb{N}_0, \ x \in \mathbb{R}_{\geq 0}, \tag{35}$$

where $\eta \geq 0$ denotes a constant sometimes referred to as *dark current*. A peak-power constraint on the transmitter is given by the peak-input constraint $X \leq A$ with probability one, i.e., $\mathbb{A} = [0, A]$ and an average-power constraint on the transmitter is considered by $\mathbb{E}[X] \leq S$.

Up to now, no analytic expression for the capacity of a discrete-time Poisson channel is known. However, for different scenarios lower and upper bounds exist. Brady and Verdú derived a lower and upper bound in the presence of only an average-power constraint [24]. Later, for $\eta = 0$ and only an average-power constraint, Martinez introduced better upper and lower bounds [25]. Lapidoth and Moser derived a lower bound and an asymptotic upper bound, which is valid only when the available peak and average power tend to infinity with their ratio held fixed, for the presence of a peak and average-power constraint [12]. Lapidoth *et al.* computed the asymptotic capacity of the discrete-time Poisson channel when the allowed average-input power tends to zero with the allowed peak power — if finite — held fixed and the dark current is constant or tends to zero proportionally to the average power [26].

Here, we numerically approximate the capacity of a discrete-time Poisson channel. For simplicity we consider the case where only a peak power constraint is imposed; the case where an additional average power constraint is present can be treated similarly.

The following proposition provides an upper bound for the k-polynomial tail for the Poisson channel W as defined in (35).

Proposition 3.19 (Poisson tail). *The Poisson channel* (35) *having a bounded input alphabet* $\mathcal{X} = [0, A]$ *and* dark current *parameter* η *has a k-polynomial tail for any $k \in (0, 1]$ in the sense of Definition 3.1, which is upper bounded for all $M \geq A + \eta$ by*

$$R_k(M) \leq \left(\alpha \mathrm{e}^{(\alpha-1)(A+\eta)} \frac{(A+\eta)^M}{M!} \right)^k, \qquad \alpha := 2^{(k^{-1}-1)}.$$

Proof. See Appendix A 11. □

In the following we present an example to illustrate the theoretical results developed in the preceding sections and their performance. Note that for the discrete-time Poisson channel Assumption 3.6 clearly holds.

Example 3.20. We consider a discrete-time Poisson channel W as defined in (35) with a peak-power constraint A and dark current $\eta = 1$. Up to now, the best known lower bound for the capacity is given by [12, Theorem 4]

$$C(W) \geq \frac{1}{\ln 2} \left(\frac{1}{2} \ln A + \left(\frac{A}{3} + 1 \right) \ln \left(1 + \frac{3}{A} \right) - 1 - \sqrt{\frac{\eta + \frac{1}{12}}{A}} \left(\frac{\pi}{4} + \frac{1}{2} \ln 2 \right) - \frac{1}{2} \ln \frac{\pi e}{2} \right). \quad (36)$$

To the best of our knowledge no upper bound for the capacity is known. In [12] an asymptotic upper bound is given which includes an unknown error term that is vanishing in the limit $A \to \infty$. According to Theorems 3.3 and 3.16, the algorithm introduced in this article leads to an approximation error after n iterations that is given by

$$\left| C_{\text{approx}}^{(n)}(W_M) - C(W) \right| \leq \left| C_{\text{approx}}^{(n)}(W_M) - C(W_M) \right| + |C(W_M) - C(W)|$$
$$\leq F(\hat{\lambda}) + G(\hat{\lambda}) - I(\hat{p}, W) + \mathcal{E},$$

where $\mathcal{E} = \frac{2 \log(e)}{e(1-k)} \left[M^{1-k} \left(R_1(M) \right)^k + R_k(M) \right]$, $R_\ell(M) = \left(\alpha e^{(\alpha-1)(A+\eta)} \frac{(A+\eta)^M}{M!} \right)^\ell$ and $\alpha := 2^{(\ell^{-1}-1)}$ for any $k \in (0, 1)$ and $\ell \in (0, 1]$. The truncation parameter M was determined as described in Remark 3.17. This finally leads to the following upper and lower bounds on $C(W)$

$$2I(\hat{p}, W) - \left(F(\hat{\lambda}) + G(\hat{\lambda}) \right) - \mathcal{E} \leq C(W) \leq 2 \left(F(\hat{\lambda}) + G(\hat{\lambda}) \right) - I(\hat{p}, W) + \mathcal{E}. \quad (37)$$

Figure 3.20 compares the two bounds (36) and (37) for different values of A. Further details on the simulation can be found in Appendix B.

4. CONCLUSION AND FUTURE WORK

We introduced a new approach to approximate the capacity of DMCs possibly having constraints on the input distribution. The dual problem of Shannon's capacity formula turns out to have a particular structure such that the Lagrange dual function admits a closed form solution. Applying smoothing techniques to the non-smooth dual function enables us to solve the dual problem efficiently. This new approach, in the case of no constraints on the input distribution, has a computational complexity per iteration step of $O(MN)$, where N is the input alphabet size and M the size of the output alphabet. In comparison, the Blahut-Arimoto algorithm has a computational cost of $O(MN^2)$ per iteration step. More precisely for no input power constraint, the total computational cost to find an ε-close solution is $O(\frac{M^2 N \sqrt{\log(N)}}{\varepsilon})$ for the algorithm developed in this article, whereas the Blahut-Arimoto algorithm requires $O(\frac{MN^2 \log(N)}{\varepsilon})$. We would like to emphasize that the computational cost of the smallest unit, i.e. the cost of one iteration is strictly better for the algorithm introduced in this article. As highlighted by Example 2.14, this can make a crucial difference especially for large input alphabets. Another strength of the new approach is that it provides an a posteriori error, i.e., after having run a certain number of iterations we can precisely estimate the actual error in the current approximation. This is computationally appealing as explicit (or a priori) error bounds often are conservative in practice. By exploiting this a posteriori bound we can stop the computation once the desired accuracy has been reached.

As a second contribution, we have shown how similar ideas can be used to approximate the capacity of memoryless channels with continuous bounded input alphabets and countable output alphabets under a mild assumption on the channels tail. This assumption holds, for example, for discrete-time Poisson channels, allowing us to efficiently approximate their capacity. As an
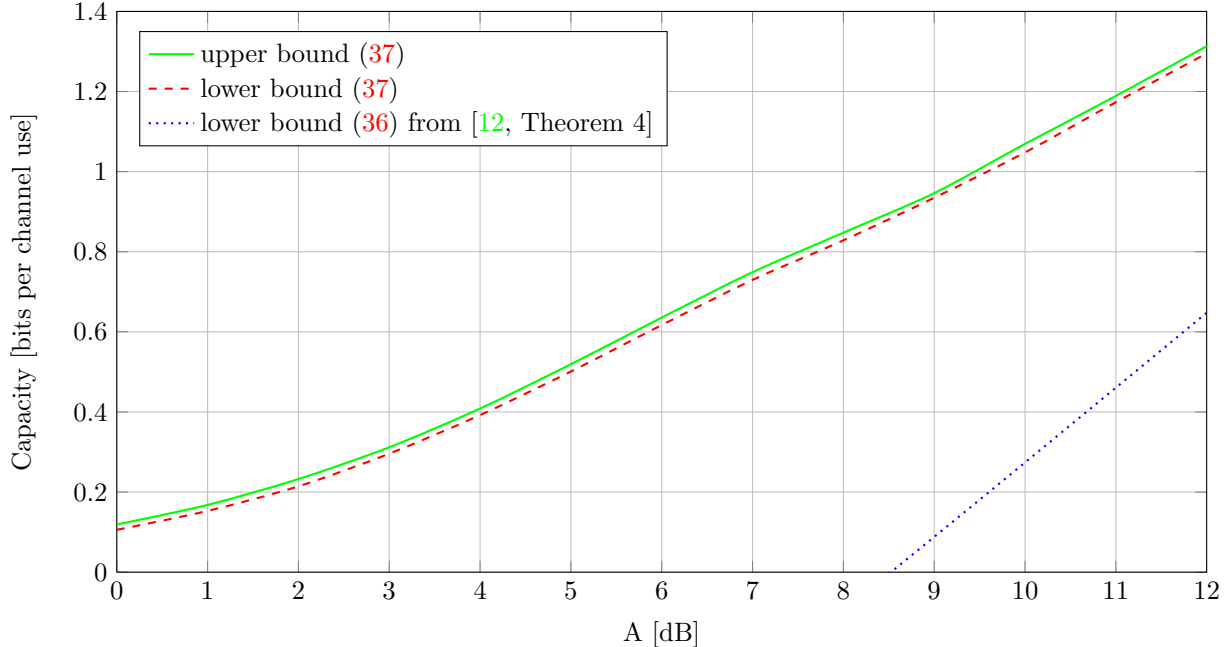
FIG. 3. This plot depicts the capacity of a discrete-time Poisson channel with dark current $\eta = 1$ as a function of the peak-power constraint parameter A. The red (resp. green) line shows the lower (resp. upper) bound (37) obtained for a moderate number of iterations, see Appendix B. As a comparison we plot the lower bound of [12], which to the best of our knowledge is the tightest lower bound available to date (blue line). The parameter A is given in decibels where $A[\mathrm{dB}] = 10 \log_{10}(A)$.

example we derived upper and lower bounds for a discrete-time Poisson channel having a peak-power constraint at the input.

The presented optimization method highly depends on the Lipschitz constant estimate of the objective's gradient. The worse this estimate the more steps the method requires for an a priori ε-precision. For future work, we aim to study the derivation of local Lipschitz constants of the gradient. This technique has recently been shown to be very efficient in practice (up to three orders of magnitude reduction of computation time), while preserving the worst-case complexity [27].

The approach introduced in this article can be used to efficiently approximate the capacity of classical-quantum channels, i.e., channels that have classical input and quantum mechanical output, with a discrete or bounded continuous input alphabet. Using the idea of a universal encoder allows us to compute close upper and lower bounds for the Holevo capacity [28].

Appendix A: Proofs

This appendix collects the technical proofs omitted above.

1. Proof of Lemma 2.1

The mutual information $I(p, W)$ can be expressed as

$$I(p, W) = \sum_{i=1}^{N} \sum_{j=1}^{M} \mathsf{W}_{ij} p_i \log \left(\frac{\mathsf{W}_{ij}}{\sum_{k=1}^{N} \mathsf{W}_{kj} p_k} \right)$$
$$= \sum_{i=1}^{N} \sum_{j=1}^{M} \left[p_i \mathsf{W}_{ij} \log(\mathsf{W}_{ij}) - p_i \mathsf{W}_{ij} \log \left(\sum_{k=1}^{N} \mathsf{W}_{kj} p_k \right) \right].$$

By adding the constraint $\sum_{i=1}^{N} p_i \mathsf{W}_{ij} = q_j$ for all $j = 1, \dots, M$,

$$I(p, W) = \sum_{i=1}^{N} \sum_{j=1}^{M} [p_i \mathsf{W}_{ij} \log(\mathsf{W}_{ij}) - p_i \mathsf{W}_{ij} \log(q_j)]$$
$$= \sum_{i=1}^{N} \sum_{j=1}^{M} p_i \mathsf{W}_{ij} \log(\mathsf{W}_{ij}) - \sum_{j=1}^{M} q_j \log(q_j)$$
$$= -r^\top p + H(q),$$

where $p \in \Delta_N$. Since $q = \mathsf{W}^\top p$ and $\mathsf{W}^\top$ is a stochastic matrix, this implies $q \in \Delta_M$. By definition of $S_{\max}$ it is obvious that the input cost constraint $s^\top p \leq S$ is inactive for $S \geq S_{\max}$, leading to the first optimization problem in Lemma 2.1. It remains to show that for $S < S_{\max}$, the input constraint can be written with equality, leading to the second optimization problem in Lemma 2.1. In oder to keep the notation simple we define $\mathsf{C}(S) := C_S(W)$ for a fixed channel W. We show that $\mathsf{C}(S)$ is concave in S for $S \in [0, S_{\max}]$. Let $S^{(1)}, S^{(2)} \in [0, S_{\max}]$, $0 \leq \lambda \leq 1$ and $p^{(i)}$ probability mass functions that achieve $\mathsf{C}(S^{(i)})$ for $i \in \{1, 2\}$. Consider the probability mass function $p^{(\lambda)} = \lambda p^{(1)} + (1 - \lambda) p^{(2)}$. We can write

$$s^\top p^{(\lambda)} = \lambda s^\top p^{(1)} + (1 - \lambda) s^\top p^{(2)}$$
$$\leq \lambda S^{(1)} + (1 - \lambda) S^{(2)}$$
$$=: S^{(\lambda)} \in [0, S_{\max}]. \tag{A1}$$

Using the concavity of the mutual information in the input distribution, we obtain

$$\lambda \mathsf{C}(S^{(1)}) + (1 - \lambda) \mathsf{C}(S^{(2)}) = \lambda I\left(p^{(1)}, W\right) + (1 - \lambda) I\left(p^{(2)}, W\right)$$
$$\leq I\left(p^{(\lambda)}, W\right)$$
$$\leq \mathsf{C}(S^{(\lambda)}),$$

where the final inequality follows by Shannon's formula for the capacity given in (1). $\mathsf{C}(S)$ clearly is non-decreasing in S since enlarging S relaxes the input cost constraint. Furthermore, we show that

$$\mathsf{C}(S_{\max} - \varepsilon) < \mathsf{C}(S_{\max}), \quad \text{for all } \varepsilon > 0. \tag{A2}$$

Suppose $\mathsf{C}(S_{\max} - \varepsilon) = \mathsf{C}(S_{\max})$ and denote $\mathsf{C}^\star := \max_{p \in \Delta_N} I(p, W)$. This then implies that there exists $\bar{p} \in \Delta_N$ such that $I(\bar{p}, W) = \mathsf{C}^\star$ and $s^\top \bar{p} \leq S_{\max} - \varepsilon$, which contradicts the definition of $S_{\max}$. Hence, the concavity of $\mathsf{C}(S)$ together with the non-decreasing property and (A2) imply that $\mathsf{C}(S)$ is strictly increasing in S. $\qquad \square$

2.   Proof of Lemma 2.2

This proof is similar to the proof given in [23, Theorem 12.1.1]. Let q satisfy the constraints in (7). Then

$$J(q) = H(q) - c^\top q = -\sum_{i=1}^{N} q_i \log(q_i) - c^\top q$$

$$= -\sum_{i=1}^{N} q_i \log\left(\frac{q_i}{p_i^\star} p_i^\star\right) - c^\top q = -D(q\|p^\star) - \sum_{i=1}^{N} q_i \log(p_i^\star) - c^\top q$$

$$\leq -\sum_{i=1}^{N} q_i \log(p_i^\star) - c^\top q \tag{A3}$$

$$= -\sum_{i=1}^{N} q_i \left(\mu_1 + \mu_2 s_i\right) \tag{A4}$$

$$= -\sum_{i=1}^{N} p_i^\star \left(\mu_1 + \mu_2 s_i\right) - c^\top p^\star + c^\top p^\star \tag{A5}$$

$$= -\sum_{i=1}^{N} p_i^\star \log(p_i^\star) - c^\top p^\star = J(p^\star).$$

The inequality follows form the non-negativity of the relative entropy. Equality (A4) follows by the definition of $p^\star$ and (A5) uses the fact that both $p^\star$ and q satisfy the constraints in (7). Note that equality holds in (A3) if and only if $q = p^\star$. This proves the uniqueness.   □

3.   Proof of Lemma 2.4

Consider the following two convex optimization problems

$$\mathsf{P}_\beta : \begin{cases} \max\limits_{p,q,\varepsilon} & -r^\top p + H(q) - \beta\varepsilon \\ \text{s.t.} & \|\mathsf{W}^\top p - q\|_\infty \leq \varepsilon \\ & s^\top p = S \\ & p \in \Delta_N, \ q \in \Delta_M, \ \varepsilon \in \mathbb{R}_{\geq 0} \end{cases} \quad \text{and} \quad \mathsf{D}_\beta : \begin{cases} \min\limits_{\lambda} & F(\lambda) + G(\lambda) \\ \text{s.t.} & \|\lambda\|_1 \leq \beta \\ & \lambda \in \mathbb{R}^M. \end{cases}$$

Claim A.1. *Strong duality holds between* P_β *and* D_β.

Proof. According to the identity $\|\mathsf{W}^\top p - q\|_\infty = \max_{\|\lambda\|_1 \leq 1} \lambda^\top (\mathsf{W}^\top p - q)$ [29, p. 7] the optimization problem P_β can be rewritten as

$$\mathsf{P}_\beta : \begin{cases} \max\limits_{p,q} & -r^\top p + H(q) + \min\limits_{\|\lambda\|_1 \leq \beta} \lambda^\top (\mathsf{W}^\top p - q) \\ \text{s.t.} & s^\top p = S \\ & p \in \Delta_N, \ q \in \Delta_M, \end{cases}$$

whose dual program, where strong duality holds according to [16, Proposition 5.3.1, p. 169] is given by

$$\begin{cases} \min\limits_{\|\lambda\|_1 \leq \beta} \max\limits_{p,q} & -r^\top p + H(q) + \lambda^\top (\mathsf{W}^\top p - q) \\ \text{s.t.} & s^\top p = S \\ & p \in \Delta_N, \ q \in \Delta_M, \end{cases} \quad .$$

which clearly is equivalent to D_β with $F(\cdot)$ and $G(\cdot)$ as given in (9). $\qquad\square$

Denote by $\varepsilon^\star(\beta)$ the optimizer of P_β with the respective optimal value $J_\beta^\star$. We show that for a sufficiently large β the optimizer $\varepsilon^\star(\beta)$ of P_β is equal to zero. Hence, in light of the duality relation, the constraint $\|\lambda\|_1 \leq \frac{\beta}{2}$ in D_β is inactive and as such D_β is equivalent to D in equation (8). Note that for

$$
J(\varepsilon) := \begin{cases} \max\limits_{p,q} & -r^\top p + H(q) \\ \text{s.t.} & \|\mathsf{W}^\top p - q\|_\infty \leq \varepsilon \\ & s^\top p = S \\ & p \in \Delta_N, \ q \in \Delta_M \end{cases}, \tag{A6}
$$

the mapping $\varepsilon \mapsto J(\varepsilon)$, the so-called perturbation function, is concave [30, p. 268]. In the next step we write the optimization problem (A6) in another equivalent form

$$
J(\varepsilon) = \begin{cases} \max\limits_{p,v} & -r^\top p + H(\mathsf{W}^\top p + \varepsilon v) \\ \text{s.t.} & \|v\|_\infty \leq 1 \\ & s^\top p = S \\ & p \in \Delta_N, \ v \in \mathsf{Im}(\mathsf{W}^\top) \subset \mathbb{R}^M \end{cases}. \tag{A7}
$$

By using Taylor's theorem, there exists $y_\varepsilon \in [0,\varepsilon]$ such that the entropy term in the objective function of (A7) can be bounded as

$$
H(\mathsf{W}^\top p + \varepsilon v) = H(\mathsf{W}^\top p) - \big(\log(\mathsf{W}^\top p) + \tfrac{1}{\ln 2}\mathbf{1}\big)^\top v\varepsilon - \sum_{j=1}^{M} \frac{v_j^2}{\sum_{i=1}^{N} \mathsf{W}_{ij} p_i + y_\varepsilon v_j} \varepsilon^2 \tfrac{1}{\ln 2}
$$

$$
\leq H(\mathsf{W}^\top p) - \big(\log(\mathsf{W}^\top p) + \tfrac{1}{\ln 2}\mathbf{1}\big)^\top v\varepsilon + \frac{M}{\gamma \ln 2}\varepsilon^2. \tag{A8}
$$

Thus, the optimal value of problem P_β can be expressed as

$$
J_\beta^\star \leq \max_\varepsilon \big\{ J(\varepsilon) - \beta\varepsilon \big\}
$$

$$
\leq \max_\varepsilon \left\{ \max_{p,v} \left[-r^\top p + H(\mathsf{W}^\top p) - \big(\log(\mathsf{W}^\top p) + \tfrac{1}{\ln 2}\mathbf{1}\big)^\top v\varepsilon \ : \ s^\top p = S \right] + \frac{M}{\gamma \ln 2}\varepsilon^2 - \beta\varepsilon \right\} \tag{A9a}
$$

$$
\leq \max_\varepsilon \left\{ \max_{p,v} \left[-r^\top p + H(\mathsf{W}^\top p) \ : \ s^\top p = S \right] + (\rho - \beta)\varepsilon + \frac{M}{\gamma \ln 2}\varepsilon^2 \right\} \tag{A9b}
$$

$$
= J(0) + \max_\varepsilon \left\{ (\rho - \beta)\varepsilon + \frac{M}{\gamma \ln 2}\varepsilon^2 \right\}, \tag{A9c}
$$

where $\rho = M\big(\log(\gamma^{-1}) \vee \tfrac{1}{\ln 2}\big)$. Note that (A9a) follows from (A7) and (A8). The equation (A9b) uses the fact that for $\|v\|_\infty \leq 1$, $-\big(\log(\mathsf{W}^\top p) + \tfrac{1}{\ln 2}\mathbf{1}\big)^\top v \leq M\big(\log(\gamma^{-1}) \vee \tfrac{1}{\ln 2}\big)$. Thus, for $\beta > \rho$ and $\varepsilon_1 = \frac{\gamma \ln 2}{M}(\beta - \rho)$, we have $\max\limits_{\varepsilon \leq \varepsilon_1}\left\{ (\rho - \beta)\varepsilon + \frac{M}{\gamma \ln 2}\varepsilon^2 \right\} = 0$. Therefore, (A9c) together with the concavity of the mapping $\varepsilon \mapsto J(\varepsilon)$ imply that $J(0)$ is the global optimum of $J(\varepsilon)$ and as such $\varepsilon^\star(\beta) = 0$ for $\beta > \rho$, indicating that P_β is equivalent to P in the sense that $J_\beta^\star = J_0^\star$. By strong duality this implies that the constraint $\|\lambda\|_1 \leq \beta$ in D_β is inactive. Finally, $\|\lambda\|_2 \leq \|\lambda\|_1$ concludes the proof. $\qquad\square$

4. Proof of Theorem 3.3

To prove Theorem 3.3 we need a preliminary lemma.

Lemma A.2. *Given $k \in (0,1)$ and $p \in [0,1]$, we have for all $x \in [0, 1-p]$*

$$\big|(p+x)\log(p+x) - p\log(p)\big| \leq \frac{\log(\mathrm{e})}{\mathrm{e}(1-k)}x^k.$$

Proof. Note that for a fixed $x \in [0,1]$, the mapping $p \mapsto (p+x)\log(p+x) - p\log(p)$ is non-decreasing; observe that the derivative of the mapping is non-negative for all $x \in [0,1]$. Therefore, it suffices to verify the claim for $p \in \{0,1\}$. For $p = 1$ and accordingly $x = 0$, Lemma A.2 holds trivially. Let $p = 0$ and $h(x) := \frac{\log(\mathrm{e})}{\mathrm{e}(1-k)}x^{k-1} + \log(x)$. Note that $h(1) = \frac{\log(\mathrm{e})}{\mathrm{e}(1-k)} > 0$ and $h(x) \to \infty$ as $x \to 0$. Hence, by setting $\frac{\mathrm{d}}{\mathrm{d}x}h(x^\star) = 0$, it can be easily seen that

$$\min_{x \in (0,1]} h(x) = h(x^\star) = 0, \qquad x^\star := \mathrm{e}^{\frac{1}{k-1}}.$$

Thus $h(x) \geq 0$, and consequently $xh(x) \geq 0$ for all $x \in (0,1]$, which concludes the proof. $\quad\square$

Proof of Theorem 3.3. Note that

$$|C(W) - C(W_M)| = \big|\max_{p \in \mathcal{P}(\mathcal{X})} I(p, W) - \max_{p \in \mathcal{P}(\mathcal{X})} I(p, W_M)\big| \leq \max_{p \in \mathcal{P}(\mathcal{X})} \big|I(p, W) - I(p, W_M)\big|.$$

It thus suffices to bound the mutual information difference uniformly in the input probability distribution $p \in \mathcal{P}(\mathcal{X})$. Observe that

$$\big|I(p, W) - I(p, W_M)\big|$$
$$= \bigg|\int_{\mathcal{X}} \Big[-h\big(W(\cdot, x)\big) + h\big(W_M(\cdot, x)\big)\Big]p(\mathrm{d}x) + h\Big(\int_{\mathcal{X}} W(\cdot, x)p(\mathrm{d}x)\Big) - h\Big(\int_{\mathcal{X}} W_M(\cdot, x)p(\mathrm{d}x)\Big)\bigg|$$
$$= \bigg|\int_{\mathcal{X}} \Big[\sum_{i \in \mathbb{N}_0} W(i|x)\log(W(i|x)) - W_M(i|x)\log(W_M(i|x))\Big]p(\mathrm{d}x)$$
$$+ \sum_{i \in \mathbb{N}_0} -\Big(\int_{\mathcal{X}} W(i|x)p(\mathrm{d}x)\Big)\log\Big(\int_{\mathcal{X}} W(i|x)p(\mathrm{d}x)\Big)$$
$$+ \Big(\int_{\mathcal{X}} W_M(i|x)p(\mathrm{d}x)\Big)\log\Big(\int_{\mathcal{X}} W_M(i|x)p(\mathrm{d}x)\Big)\bigg|.$$

By the definition of the truncated channel in (21) and applying Lemma A.2 to the above relation, we have

$$|I(p, W) - I(p, W_M)| \leq \frac{\log(\mathrm{e})}{\mathrm{e}(1-k)}\bigg(\int_{\mathcal{X}} \Big[\sum_{i < M}\Big(\frac{1}{M}\sum_{j \geq M} W(j|x)\Big)^k + \sum_{i \geq M}\big(W(i|x)\big)^k\Big]p(\mathrm{d}x)$$
$$+ \sum_{i < M}\Big(\frac{1}{M}\sum_{j \geq M}\int_{\mathcal{X}} W(j|x)p(\mathrm{d}x)\Big)^k + \sum_{i \geq M}\Big(\int_{\mathcal{X}} W(i|x)p(\mathrm{d}x)\Big)^k\bigg)$$
$$\leq \frac{2\log(\mathrm{e})}{\mathrm{e}(1-k)}\Big(M\big(\frac{R_1(M)}{M}\big)^k + R_k(M)\Big),$$

which concludes the proof. $\quad\square$

5.   Proof of Proposition 3.4

We show that the optimization problem (23) is equivalent to

$$C_{\mathbb{A},S}(W_M) = \sup_{p \in \mathfrak{D}(\mathbb{A})} \{ I(p, W_M) \ : \ \mathbb{E}[s(X)] \leq S \},$$

where $\mathfrak{D}(\mathbb{A})$ is the space of probability measures that are absolutely continuous with respect to the Lebesgue measure. This completes the proof since optimizing over $\mathfrak{D}(\mathbb{A})$ is equivalent to optimizing over the space of probability densities $\mathcal{D}(\mathbb{A})$ according to the Radon-Nikodým Theorem [31, Theorem 3.8, p. 90].

It is known that the mapping $p \mapsto I(p, W_M)$ is weakly lower semicontinuous [32]. It then suffices to show that $\mathfrak{D}(\mathbb{A})$ is weakly dense in $\mathcal{P}(\mathbb{A})$. Let $\mathbb{B}$ be a countable dense subset of $\mathbb{A}$, and $\Delta(\mathbb{B})$ be the family of probability measures whose supports are finite subsets of $\mathbb{B}$. It is well known that $\Delta(\mathbb{B})$ is weakly dense in $\mathcal{P}(\mathbb{A})$, i.e., $\mathcal{P}(\mathbb{A}) = \overline{\Delta(\mathbb{B})}$ [33, Theorem 4, p. 237], where $\overline{\Delta}$ is the weak closure of Δ. Moreover, thanks to the Lebesgue differentiation theorem [31, Theorem 3.21, p. 98], we know that for any $b \in \mathbb{B}$ the point measure $\delta_{\{b\}} \in \Delta(\mathbb{B})$ can be arbitrarily weakly approximated by measures in $\mathfrak{D}(\mathbb{A})$, i.e., $\delta_{\{b\}} \in \overline{\mathfrak{D}(\mathbb{A})}$. Hence, we have $\Delta(\mathbb{B}) = \overline{\mathfrak{D}(\mathbb{A})}$, which in light of the preceding assertion implies $\mathcal{P}(\mathbb{A}) = \overline{\mathfrak{D}(\mathbb{A})}$. $\qquad\square$

6.   Proof of Lemma 3.8

The proof follows the ideas of [11]. It can easily be shown that for $d(p) := -h(p) + \log(\rho)$

$$\langle d''(p) \cdot g, g \rangle = \int_{\mathbb{A}} \frac{g(x)^2}{p(x)} \mathrm{d}x.$$

Cauchy-Schwarz then implies

$$\langle d''(p) \cdot g, g \rangle \geq \frac{\left(\int_{\mathbb{A}} g(x) \mathrm{d}x \right)^2}{\int_{\mathbb{A}} p(x) \mathrm{d}x} = \|g\|^2.$$

$\qquad\square$

7.   Proof of Proposition 3.9

It is known, according to Theorem 5.1 in [34], that $G_\nu(\lambda)$ is well defined and continuously differentiable at any $\lambda \in \mathbb{R}^M$ and that this function is convex and its gradient $\nabla G_\nu(\lambda) = \mathcal{W}^\star p_\nu^\lambda$ is Lipschitz continuous with constant $L_\nu = \frac{1}{\nu} \|\mathcal{W}\|^2$, where we have also used Lemma 3.8. The operator norm can be simplified to

$$
\begin{aligned}
\|\mathcal{W}\| &= \sup_{\lambda \in \mathbb{R}^M, \, p \in \mathrm{L}^1(\mathbb{A})} \left\{ \langle p, \mathcal{W}\lambda \rangle \ : \ \|\lambda\|_2 = 1, \ \|p\|_1 = 1 \right\} \\
&\leq \sup_{\lambda \in \mathbb{R}^M, \, p \in \mathrm{L}^1(\mathbb{A})} \left\{ \|\mathcal{W}^\star p\|_2 \|\lambda\|_2 \ : \ \|\lambda\|_2 = 1, \ \|p\|_1 = 1 \right\} \qquad\text{(A10)} \\
&\leq \sup_{p \in \mathrm{L}^1(\mathbb{A})} \left\{ \|\mathcal{W}^\star p\|_1 \ : \ \|p\|_1 = 1 \right\} \\
&= \sup_{p \in \mathrm{L}^1(\mathbb{A})} \left\{ \sum_{i=0}^{M-1} \int_{\mathcal{X}} W_M(i|x) p(x) \mathrm{d}x \ : \ \|p\|_1 = 1 \right\}
\end{aligned}
$$

$$
= \sup_{p \in L^1(\mathbb{A})} \left\{ \int_{\mathcal{X}} \|W_M(\cdot|x)\|_1 \, p(x)\mathrm{d}x \;:\; \|p\|_1 = 1 \right\}
$$

$$
\leq \sup_{x \in \mathbb{A}} \|W_M(\cdot|x)\|_1
$$

$$
\leq 1,
$$

where (A10) is due to Cauchy-Schwarz. □

8. Proof of Lemma 3.13

Let $x_1, x_2 \in \mathcal{X}$, then by definition of $f_\lambda(\cdot)$ we obtain

$$
|f_\lambda(x_1) - f_\lambda(x_2)|
$$

$$
= \left| \sum_{i=1}^{M} W_M(i-1|x_1)\lambda_i + \sum_{j=1}^{M} W_M(j-1|x_1) \log W_M(j-1|x_1) \right.
$$

$$
\left. - \sum_{i=1}^{M} W_M(i-1|x_2)\lambda_i - \sum_{j=1}^{M} W_M(j-1|x_2) \log W_M(j-1|x_2) \right|
$$

$$
\leq \left| \sum_{i=1}^{M} \left(W_M(i-1|x_1) - W_M(i-1|x_2) \right) \lambda_i \right| + |H(W_M(\cdot|x_1)) - H(W_M(\cdot|x_2))| \tag{A11a}
$$

$$
\leq \sum_{i=1}^{M} |\left(W_M(i-1|x_1) - W_M(i-1|x_2) \right) \lambda_i| + |H(W_M(\cdot|x_1)) - H(W_M(\cdot|x_2))| \tag{A11b}
$$

$$
\leq LM\|\lambda\|_1 |x_1 - x_2| + |H(W_M(\cdot|x_1)) - H(W_M(\cdot|x_2))| \tag{A11c}
$$

$$
\leq LM^2 \left(\log \frac{1}{\gamma_M} \vee \frac{1}{\ln 2} \right) |x_1 - x_2| + |H(W_M(\cdot|x_1)) - H(W_M(\cdot|x_2))| \tag{A11d}
$$

$$
\leq LM^2 \left(\log \frac{1}{\gamma_M} \vee \frac{1}{\ln 2} \right) |x_1 - x_2| + ML \left| \log \frac{1}{\gamma_M} - \frac{1}{\ln 2} \right| |x_1 - x_2|. \tag{A11e}
$$

Inequalities (A11a) and (A11b) use the triangle inequality. Inequality (A11c) follows by Assumption 3.2(ii) and (A11d) can be derived by following the proof of Lemma 3.7, which is similar to the one of Lemma 2.4. Finally, (A11e) follows from the fact that the function $\Delta_n \ni x^n \mapsto H(x^n) \in \mathbb{R}_{\geq 0}$ with $\min_{1 \leq i \leq n} x_i < c$ is Lipschitz continuous with constant $n \left| \log \frac{1}{c} - \frac{1}{\ln 2} \right|$ and from Assumption 3.2(ii). □

9. Proof of Lemma 3.15

We start by the following definitions that simplify the proof below

$$
f_{\lambda,\nu}(x) := \mathcal{W}\lambda(x) - r(x) + \nu\mu_\nu s(x), \qquad \bar{f}_{\lambda,\nu} := \max_{x \in \mathbb{A}} f_{\lambda,\nu}(x)
$$

$$
B_{\lambda,\nu}(\varepsilon) := \left\{ x \in \mathbb{A} \mid \bar{f}_{\lambda,\nu} - f_{\lambda,\nu}(x) < \varepsilon \right\}, \qquad \eta_{\lambda,\nu}(\varepsilon) := \int_{B_{\lambda,\nu}(\varepsilon)} \mathrm{d}x.
$$

By the Lipschitz continuity of $f_\lambda(\cdot)$ and $s(\cdot)$ we get the uniform lower bound

$$
\eta_{\lambda,\nu}(\varepsilon) \geq \frac{\varepsilon}{L_f + |\nu\mu_\nu|L_s} \wedge \rho. \tag{A12}
$$

By using the solution to $G_\nu(\lambda)$, according to (28) we can write

$$G_\nu(\lambda) = -\nu \log(\rho) + \nu \log \left(\int_{\mathbb{A}} 2^{\frac{1}{\nu} f_{\lambda,\nu}(x)} \mathrm{d}x \right) \tag{A13a}$$

$$\leq \inf_{\ell \in \mathbb{R}} \max_{x \in \mathbb{A}} \{ f_\lambda(x) + \ell s(x) \} \tag{A13b}$$

$$= G(\lambda), \tag{A13c}$$

where the equality (A13c) follows as (A13b) is the dual program to $G(\lambda)$ and strong duality holds. The inequality (A13b) then is due to $G_\nu(\lambda) \leq G(\lambda)$ for any λ, see (30). Therefore,

$$G(\lambda) - G_\nu(\lambda) \leq \bar{f}_{\lambda,\nu} - G_\nu(\lambda) \tag{A14a}$$

$$= \nu \left(-\log \left(\int_{B_{\lambda,\nu}(\varepsilon)} 2^{\frac{1}{\nu}(f_{\lambda,\nu}(x) - \bar{f}_{\lambda,\nu})} \mathrm{d}x + \int_{B^c_{\lambda,\nu}(\varepsilon)} 2^{\frac{1}{\nu}(f_{\lambda,\nu}(x) - \bar{f}_{\lambda,\nu})} \mathrm{d}x \right) + \log(\rho) \right) \tag{A14b}$$

$$\leq \nu \left(-\log \left(\int_{B_{\lambda,\nu}(\varepsilon)} 2^{\frac{1}{\nu}(f_{\lambda,\nu}(x) - \bar{f}_{\lambda,\nu})} \mathrm{d}x \right) + \log(\rho) \right)$$

$$\leq \nu \left(-\log \left(\eta_{\lambda,\nu}(\varepsilon) 2^{-\frac{\varepsilon}{\nu}} \right) + \log(\rho) \right) \tag{A14c}$$

$$\leq \nu \left(-\log \left(\frac{\varepsilon}{L_f + |\nu\mu_\nu|L_s} \vee \rho \right) + \frac{\varepsilon}{\nu} + \log(\rho) \right) \tag{A14d}$$

$$= \nu \log \left(\frac{(L_f + |\nu\mu_\nu|L_s)\rho}{\varepsilon} \vee 1 \right) + \varepsilon,$$

where (A14a) follows from (A13c) and (A14b) is due to (A13a). The inequality (A14c) results from the definitions of $B_{\lambda,\nu}(\varepsilon)$ and $\eta_{\lambda,\nu}(\varepsilon)$ above and (A14d) is implied by (A12). Finally, it can be seen that for $\nu < (L_f + |\nu\mu_\nu|L_s)\rho$, the optimal choice for ε is ν, which leads to

$$G(\lambda) - G_\nu(\lambda) \leq \nu \left(1 + \log \left(\frac{(L_f + |\nu\mu_\nu|L_s)\rho}{\nu} \vee 1 \right) \right). \tag{A15}$$

It remains to upper bound the term $|\nu\mu_\nu|$. Define $\underline{f} := \min_{x,\lambda} f_\lambda(x)$, $\overline{f} := \max_{x,\lambda} f_\lambda(x)$, $\Delta_f := \overline{f} - \underline{f}$ and note that $\Delta_f \leq L_f\rho$. By (A13a), (30) and the fact that adding an additional constraint to a maximization problem cannot increase its objective value

$$G_\nu(\lambda) = \nu \log \left(\int_{\mathbb{A}} 2^{\frac{1}{\nu}(f_\lambda(x) + \nu\mu_\nu s(x))} \mathrm{d}x \right) - \nu \log(\rho) \leq \overline{f} = \nu \log \left(2^{\frac{1}{\nu}\overline{f}} \right),$$

which is equivalent to $\int_{\mathbb{A}} 2^{\frac{1}{\nu}(f_\lambda(x) - \overline{f} + \nu\mu_\nu s(x))} \mathrm{d}x \leq \rho$ and implies

$$\int_{\mathbb{A}} 2^{\mu_\nu s(x)} \mathrm{d}x \leq \rho \, 2^{\frac{\Delta_f}{\nu}}. \tag{A16}$$

From (A16) two bounds can be derived. First, (A16) implies that $\left(\rho \wedge \frac{\varepsilon}{L_s} \right) 2^{\mu_\nu(\overline{s} - \varepsilon)} \leq \rho 2^{\frac{\Delta_f}{\nu}}$, which by choosing $\varepsilon = \frac{\overline{s}}{2}$ leads to $2^{\mu_\nu \frac{\overline{s}}{2}} \leq \left(\frac{2L_s\rho}{\overline{s}} \vee 1 \right) 2^{\frac{\Delta_f}{\nu}}$ and finally,

$$\nu\mu_\nu \leq \frac{2}{\overline{s}} \log \left(\frac{2L_s\rho}{\overline{s}} \vee 1 \right) \nu + \frac{2\Delta_f}{\overline{s}}. \tag{A17}$$

Similarly one can derive a lower bound

$$\nu\mu_\nu \geq \frac{2}{\underline{s}} \log \left(\frac{2L_s\rho}{-\underline{s}} \vee 1 \right) \nu + \frac{2\Delta_f}{\underline{s}}. \tag{A18}$$

Equation (A15) together with (A17) and (A18) complete the proof. $\qquad\square$

10. Proof of Theorem 3.16

Following [11] and using Lemma 3.7, Lemma 3.5, Propostion 3.9 and Lemma 3.15, after n iterations of Algorithm 1 the following approximation error is obtained

$$0 \leq F(\hat{\lambda}) + G(\hat{\lambda}) - I(\hat{p}, W) \leq \iota(\nu) + \frac{4D_1}{\nu(n+1)^2} + \frac{4D_1}{(n+1)^2} =: \mathsf{err}(\nu, n), \qquad (A19)$$

where for $\nu < \frac{T_1}{1-T_2}$ or $T_2 > 1$ we have $\iota(\nu) = \nu \left(\log \left(\frac{T_1}{\nu} + T_2 \right) + 1 \right)$, which is strictly increasing in ν. Let us redefine the smoothing term by $\nu := \frac{\delta}{\log(\delta^{-1})}$ for $\delta \in (0,1)$ and define the function $g(\delta) := \left(\frac{\log(T_1 \log(\delta^{-1}) + T_2 \delta) + 1}{\log(\delta^{-1})} + 1 \right)$. One can see that $\iota(\nu) = \delta g(\delta)$ and that $\lim_{\delta \to 0} g(\delta) = 1$. Furthermore $\delta \leq 2^{-1} \wedge 2^{-\frac{1}{T_1+T_2}}$ implies

$$g(\delta) - 1 \leq \frac{\log \left(2(T_1 + T_2) \log \left(\delta^{-1} \right) \right)}{\log \left(\delta^{-1} \right)} \leq T_1 + T_2, \qquad (A20)$$

where the first inequality is due to $\delta \leq 2^{-1}$ and the second follows from $\delta \leq 2^{-\frac{1}{T_1+T_2}}$. We seek for a lower bound of n and upper bound δ such that the error term (A19) is smaller than the preassigned $\varepsilon > 0$, i.e.,

$$\mathsf{err}(\tfrac{\delta}{\log(\delta^{-1})}, n) = g(\delta)\delta + \frac{4D_1}{(n+1)^2} \left(\frac{\log \left(\delta^{-1} \right)}{\delta} + 1 \right) \leq \varepsilon \qquad (A21)$$

To this end, we introduce an auxiliary variable $\zeta \in (0,1)$ such that such that $g(\delta)\delta = (1 - \zeta)\varepsilon$ and $\frac{4D_1}{(n+1)^2} \left(\frac{\log(\delta^{-1})}{\delta} + 1 \right) \leq \zeta\varepsilon$, which implies (A21). Observe that $g(\delta)\delta = (1 - \zeta)\varepsilon$ is equivalent to $\delta = \frac{(1-\zeta)}{g(\delta)}\varepsilon =: \beta\varepsilon$. Hence $\zeta = 1 - \beta g(\delta)$ for $\beta \in [0, \frac{1}{g(\delta)}]$. Moreover,

$$\frac{4D_1}{(n+1)^2} \left(\frac{\log \left(\delta^{-1} \right)}{\delta} + 1 \right) = \frac{4D_1}{(n+1)^2} \left(\frac{\log \left((\beta\varepsilon)^{-1} \right)}{\beta\varepsilon} + 1 \right) \leq \zeta\varepsilon$$

is equivalent to

$$4D_1 \left(\frac{\log \left((\beta\varepsilon)^{-1} \right) + \beta\varepsilon}{\beta(1 - g(\delta)\beta)\varepsilon^2} \right) = 4D_1 \left(\frac{\log(\varepsilon^{-1}) + \log 2g(\delta) + \frac{\varepsilon}{2g(\delta)}}{\frac{\varepsilon^2}{4g(\delta)}} \right) \leq (n+1)^2, \qquad (A22)$$

where we have chosen $\beta = \frac{1}{2g(\delta)}$ and as such is equivalent to

$$\frac{4}{\varepsilon} \sqrt{D_1 \left(g(\delta) \log \left(\varepsilon^{-1} \right) + g(\delta) \log \left(2g(\delta) \right) + \frac{\varepsilon}{2} \right)} \leq n + 1$$

Finally, using (A20) implies for $\nu = \frac{\varepsilon/\alpha}{\log(\alpha/\varepsilon)}$, where $\alpha := 2(T_1 + T_2 + 1)$

$$\mathsf{err}(\nu, n) \leq \varepsilon \quad \text{for} \quad n \geq \frac{1}{\varepsilon} \sqrt{8D_1\alpha} \sqrt{\log(\varepsilon^{-1}) + \log(\alpha) + \frac{1}{4}}.$$

$\square$

11. Proof of Proposition 3.19

To prove Proposition 3.19, we need two lemmas.

Lemma A.3. *For any $k \in (0,1]$ and $a, b \geq 0$*

$$a^k + b^k \leq 2^{1-k}(a+b)^k.$$

Proof. Let $g(x) := 2^{1-k}(1+x)^k - x^k$. By setting $\frac{\mathrm{d}}{\mathrm{d}x}h(x^\star) = 0$, one can easily see that $x^\star = 1$ is the minimizer of function h over the interval $[0,1]$, i.e., $g(x) \geq g(1) = 1$ for all $x \in [0,1]$. Suppose, without loss of generality, that $a \geq b$. By virtue of the preceding result of function g, we know that

$$1 \leq g\left(\frac{b}{a}\right) = 2^{1-k}\left(1 + \frac{b}{a}\right)^k - \left(\frac{b}{a}\right)^k,$$

where by multiplying a^k it readily leads to the desired assertion. $\square$

Lemma A.4. *Let $(a_i)_{i\in\mathbb{N}}$ be a non-negative sequence of real numbers. For any $k \in (0,1]$*

$$\sum_{i\in\mathbb{N}} a_i^k \leq \left(\sum_{i\in\mathbb{N}} \alpha^i a_i\right)^k, \qquad \alpha := 2^{(k^{-1}-1)}.$$

Proof. For the proof we make use of an induction argument. Note that for any $a_1 \geq 0$ it trivially holds that $a_1^k \leq 2^{1-k}a_1^k$. We now assume that for any sequence $(a_i)_{i=1}^N \subset \mathbb{R}_{\geq 0}$ we have

$$\sum_{i=1}^N a_i^k \leq \left(\sum_{i=1}^N 2^{(k^{-1}-1)i} a_i\right)^k. \tag{A23}$$

Let $(a_i)_{i=1}^{N+1} \subset \mathbb{R}_{\geq 0}$. Then,

$$\sum_{i=1}^{N+1} a_i^k = a_1^k + \sum_{i=2}^{N+1} a_i^k \leq a_1^k + \left(\sum_{i=2}^{N+1} 2^{(k^{-1}-1)(i-1)} a_i\right)^k \leq 2^{1-k}\left(a_1 + \sum_{i=2}^{N+1} 2^{(k^{-1}-1)(i-1)} a_i\right)^k \tag{A24}$$

$$= \left(2^{(k^{-1}-1)} a_1 + \sum_{i=2}^{N+1} 2^{(k^{-1}-1)i} a_i\right)^k = \left(\sum_{i=1}^{N+1} 2^{(k^{-1}-1)i} a_i\right)^k,$$

where the first (resp. second) inequality in (A24) follows from (A23) (resp. Lemma A.3). $\square$

Proof of Proposition 3.19. It is straightforward to see that

$$\max_{x\in[0,A]} \mathrm{e}^{-x} x^i = \mathrm{e}^{-\min\{A,i\}}\left(\min\{A,i\}\right)^i. \tag{A25}$$

Moreover, based on a Taylor series expansion, it is well known that for all $M \in \mathbb{N}$ and $x \in \mathbb{R}_{\geq 0}$

$$\sum_{i\geq M} \frac{x^i}{i!} \leq \frac{\mathrm{e}^x}{M!} x^M. \tag{A26}$$

Therefore, it follows that

$$R_k(M) := \sum_{i\geq M} \left(\sup_{x\in[0,A]} \mathrm{e}^{-(x+\eta)} \frac{(x+\eta)^i}{i!}\right)^k \leq \sum_{i\geq M} \left(\mathrm{e}^{-(A+\eta)} \frac{(A+\eta)^i}{i!}\right)^k \tag{A27a}$$

$$\le \mathrm{e}^{-k(A+\eta)}\Big(\sum_{i\ge M}\alpha^{(i-M+1)}\frac{(A+\eta)^i}{i!}\Big)^k = \frac{\mathrm{e}^{-k(A+\eta)}}{\alpha^{k(M-1)}}\Big(\sum_{i\ge M}\frac{\big(\alpha(A+\eta)\big)^i}{i!}\Big)^k \tag{A27b}$$

$$\le \frac{\mathrm{e}^{-k(A+\eta)}}{\alpha^{k(M-1)}}\Big(\frac{\mathrm{e}^{\alpha(A+\eta)}}{M!}\alpha^M(A+\eta)^M\Big)^k = \Big(\alpha\mathrm{e}^{(\alpha-1)(A+\eta)}\frac{(A+\eta)^M}{M!}\Big)^k, \tag{A27c}$$

where (A27a) results from (A25) and the assumption $M \ge A+\eta$, and (A27b) (resp. (A27c)) follows from Lemma A.4 (resp. (A26)). $\qquad\square$

Appendix B: Simulation Details

This section provides some further details on the simulation in Example 3.20. The parameters considered are $k = \frac{1}{2}$, $L_f = 0$ and M is chosen according to Table III. All the simulations in this section are performed on a 2.3 GHz Intel Core i7 processor with 8 GB RAM with Matlab.

TABLE III. Simulation details to Example 3.20

A [dB]	0	1	2	3	4	5	6
M	16	17	19	20	22	25	28
Iterations n	$4\cdot10^4$	$4\cdot10^4$	$4\cdot10^4$	$5\cdot10^4$	$6\cdot10^4$	$7\cdot10^4$	$9\cdot10^4$
ν	0.0026	0.0029	0.0036	0.0029	0.0027	0.0029	0.0026
$F(\hat{\lambda}) + G(\hat{\lambda})$	0.1144	0.1626	0.2263	0.3063	0.4029	0.5129	0.6293
$I(\hat{p}, W)$	0.1105	0.1583	0.2206	0.3015	0.3979	0.5072	0.6234
$\mathcal{E}$	$9.3\cdot10^{-4}$	$9.7\cdot10^{-4}$	$4.8\cdot10^{-4}$	$8.5\cdot10^{-4}$	$8.2\cdot10^{-4}$	$4.9\cdot10^{-4}$	$5.0\cdot10^{-4}$

A [dB]	7	8	9	10	11	12
M	31	36	42	49	59	71
Iterations n	$1.2\cdot10^5$	$2\cdot10^5$	$5\cdot10^5$	$2\cdot10^6$	$3\cdot10^6$	$4\cdot10^6$
ν	0.0022	0.0016	$7.1\cdot10^{-5}$	$8.0\cdot10^{-4}$	$8.3\cdot10^{-4}$	$9.7\cdot10^{-4}$
$F(\hat{\lambda}) + G(\hat{\lambda})$	0.7423	0.8410	0.9422	1.0591	1.1835	1.3070
$I(\hat{p}, W)$	0.7365	0.8351	0.9388	1.0547	1.1788	1.3013
$\mathcal{E}$	$9.5\cdot10^{-4}$	$7.5\cdot10^{-4}$	$7.1\cdot10^{-4}$	$8.0\cdot10^{-4}$	$6.2\cdot10^{-4}$	$5.2\cdot10^{-4}$

[1] Claude E. Shannon, "A mathematical theory of communication," Bell System Technical Journal **27**, 379–423 (1948).

[2] Richard E. Blahut, "Computation of channel capacity and rate-distortion functions," IEEE Transactions on Information Theory **18**, 460–473 (1972).

[3] Robert G. Gallager, *Information Theory and Reliable Communication* (John Wiley & Sons, 1968).

[4] Suguru Arimoto, "An algorithm for computing the capacity of arbitrary discrete memoryless channels," IEEE Transactions on Information Theory **18**, 14–20 (1972).

[5] Ibrahim C. Abou-Faycal, Mitchell D. Trott, and Shlomo Shamai, "The capacity of discrete-time memoryless rayleigh-fading channels," IEEE Transactions on Information Theory **47**, 1290–1301 (2001).

[6] Jossy Sayir, "Iterating the arimoto-blahut algorithm for faster convergence," Proceedings IEEE International Symposium on Information Theory (ISIT) , 235 (2000).

[7] Gerald Matz and Pierre Duhamel, "Information geometric formulation and interpretation of accelerated blahut-arimoto-type algorithms," Proceedings Information Theory Workshop (ITW) , 66–70 (2004).

[8] Yaming Yu, "Squeezing the arimoto-blahut algorithm for faster convergence," IEEE Transactions on Information Theory **56**, 3149–3157 (2010).

[9] Chiang Mung and Stephen Boyd, "Geometric programming duals of channel capacity and rate distortion," IEEE Transactions on Information Theory **50**, 245–258 (2004).

[10] Jianyi Huang and Sean P. Meyn, "Characterization and computation of optimal distributions for channel coding," IEEE Transactions on Information Theory **51**, 2336–2351 (2005).

[11] Yurii Nesterov, "Smooth minimization of non-smooth functions," Mathematical Programming **103**, 127–152 (2005).

[12] Amos Lapidoth and Stefan M. Moser, "On the capacity of the discrete-time poisson channel," IEEE Transactions on Information Theory **55**, 303–322 (2009).

[13] Aharon Ben-Tal and Marc Teboulle, "Extension of some results for channel capacity using a generalized information measure," Applied Mathematics and Optimization **17**, 121–132 (1988).

[14] Chiang Mung, "Geometric programming for communication systems," Foundations and Trends in Communications and Information Theory **2**, 1–154 (2005).

[15] Yurii Nesterov, *Introductory Lectures on Convex Optimization: A Basic Course*, Applied Optimization (Springer, 2004).

[16] Dimitri P. Bertsekas, *Convex Optimization Theory*, Athena Scientific optimization and computation series (Athena Scientific, 2009).

[17] Jean B. Lasserre, *Moments, Positive Polynomials and Their Applications*, Imperial College Press optimization series (Imperial College Press, 2009).

[18] Stefan M. Moser, "Duality-based bounds on channel capacity," PhD thesis, ETH Zurich (2005).

[19] Shlomo Shamai, "Capacity of a pulse amplitude modulated direct detection photon channel," IEE Proceedings on Communications, Speech and Vision **137**, 424–430 (1990).

[20] D. H. Fremlin, *Measure theory. Vol. 2* (Torres Fremlin, Colchester, 2010) pp. 563+12 pp. (errata), broad foundations, Second edition January 2010.

[21] Edward J. Anderson and Peter Nash, *Linear programming in infinite-dimensional spaces: theory and applications*, Wiley-Interscience Series in Discrete Mathematics and Optimization (Wiley, 1987).

[22] Sanjoy K. Mitter, "Convex optimization in infinite dimensional spaces," in *Recent advances in learning and control*, Lecture Notes in Control and Inform. Sci., Vol. 371 (Springer, London, 2008) pp. 161–179.

[23] Thomas M. Cover and Joy A. Thomas, *Elements of Information Theory* (Wiley Interscience, 2006).

[24] David Brady and Sergio Verdú, "The asymptotic capacity of the direct detection photon channel with a bandwidth constraint," in *28th Annual Allerton Conference on Communication, Control, and Computing* (1990) pp. 691–700.

[25] Alfonso Martinez, "Spectral efficiency of optical direct detection," J. Opt. Soc. Am. B **24**, 739–749 (2007).

[26] Amos Lapidoth, Jeffrey H. Shapiro, Vinodh Venkatesan, and Ligong Wang, "The discrete-time poisson channel at low input powers," IEEE Transactions on Information Theory **57**, 3260–3272 (2011).

[27] Michel Baes and Michael Bürgisser, "An acceleration procedure for optimal first-order methods," Optimization Methods and Software **29**, 610–628 (2014).

[28] David Sutter, Tobias Sutter, Peyman Mohajerin Esfahani, and Renato Renner, "Efficient approximation of quantum channel capacities," (2014), available at arXiv.

[29] Alexander S. Holevo, *Quantum Systems, Channels, Information* (De Gruyter Studies in Mathematical Physics 16, 2012).

[30] Stephen Boyd and Lieven Vandenberghe, *Convex Optimization* (Cambridge University Press, Cambridge, 2004) pp. xiv+716, sixth printing with corrections, 2008.

[31] Gerald B. Folland, *Real analysis: modern techniques and their applications*, Pure and applied mathematics (Wiley, 1999).

[32] Yihong Wu and Sergio Verdú, "Functional properties of minimum mean-square error and mutual information," IEEE Transactions on Information Theory **58**, 1289–1301 (2012).

[33] Patrick Billingsley, *Convergence of probability measures*, Wiley Series in probability and Mathematical Statistics: Tracts on probability and statistics (Wiley, 1968).

[34] Olivier Devolder, François Glineur, and Yurii Nesterov, "Double smoothing technique for large-scale linearly constrained convex optimization," SIAM Journal on Optimization **22**, 702–727 (2012).