



## Data-Driven Estimation of Capacity Upper Bounds

Downloaded from: <https://research.chalmers.se>, 2025-05-17 12:37 UTC

Citation for the original published paper (version of record):

Häger, C., Agrell, E. (2022). Data-Driven Estimation of Capacity Upper Bounds. IEEE Communications Letters, 26. <http://dx.doi.org/10.1109/LCOMM.2022.3207385>

N.B. When citing this work, cite the original published paper.

© 2022 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, or reuse of any copyrighted component of this work in other works.

# Data-Driven Estimation of Capacity Upper Bounds

Christian Häger, *Member, IEEE* and Erik Agrell, *Fellow, IEEE*

**Abstract**—We consider the problem of estimating an upper bound on the capacity of a memoryless channel with unknown channel law and continuous output alphabet. A novel data-driven algorithm is proposed that exploits the dual representation of capacity where the maximization over the input distribution is replaced with a minimization over a reference distribution on the channel output. To efficiently compute the required divergence maximization between the conditional channel and the reference distribution, we use a modified mutual information neural estimator that takes the channel input as an additional parameter. We numerically evaluate our approach on different memoryless channels and show empirically that the estimated upper bounds closely converge either to the channel capacity or to best-known lower bounds.

**Index Terms**—Autoencoders, channel capacity, divergence estimation, duality, dual capacity representation, mutual information neural estimation, neural networks, upper capacity bounds.

## I. INTRODUCTION

The capacity of a communication channel is the maximum rate that can be reliably transmitted [1]. Even though capacity is of fundamental importance for both theory and practice, exact analytical expressions are only available in relatively few cases. If the underlying channel law is known, numerical techniques can be used to approximately compute capacity such as the well-known Blahut–Arimoto algorithm [2], [3] and its many variations, see, e.g., [4] and references therein.

Recently, there has been significant interest in developing capacity estimation algorithms based on machine learning [5]–[11]. These approaches have their origin in the seminal work [12], where the authors propose to reinterpret the communication problem as a reconstruction task using parameterized transmitters and receivers, similar to autoencoders (AEs) in machine learning. It can be shown that the cross-entropy minimization commonly used to train AEs maximizes a lower bound on mutual information, whereas the transmitter optimization can be regarded as shaping a discrete input distribution. Using this approach, tight lower bounds on the capacity of a nonlinear phase noise (NLPN) channel were for example obtained in [5].

A disadvantage of the AE approach is that it requires a differentiable channel model to compute gradients for the transmitter optimization. To address this problem, [6] proposes to use the sample-based mutual information neural estimation

This work was supported by the Knut and Alice Wallenberg Foundation (grant no. 2018.0090) and the Swedish Research Council (grants no. 2020-04718 and 2021-03709). The computations were enabled by resources provided by the Swedish National Infrastructure for Computing (SNIC) at the Chalmers Centre for Computational Science and Engineering (C3SE) partially funded by the Swedish Research Council (grant no. 2018-05973).

C. Häger and E. Agrell are with the Department of Electrical Engineering, Chalmers University of Technology, 41296 Gothenburg, Sweden (email: {hagerc, agrell}@chalmers.se).

Manuscript received Month xx, 2021, revised Month xx, 2021

(MINE) technique from [13] and train the AE transmitter based on the (differentiable) MINE. Related approaches were subsequently proposed for more general channels that may include feedback and/or memory<sup>1</sup> in [7] and for memoryless multiple-access channels in [9]. Moreover, a hybrid approach that regularizes the cross-entropy-based AE training using MINE is proposed in [10]. Comparisons of various sample-based mutual information estimators similar to MINE can be found in [8] and [11].

All of the above learning-based approaches target the estimation of *lower* capacity bounds using the conventional maximization of mutual information (or directed information in [7]) over the input distribution, see (1) below. In this paper, we follow a different path and consider the problem of estimating an *upper* capacity bound by exploiting the dual representation of channel capacity, which is described in Sec. II. Our work relies on a variation of MINE for estimating the divergence between the conditional channel and a given reference distribution. Compared to the conventional Blahut–Arimoto algorithm, the main advantage of our approach and similar works on neural capacity estimation in [5]–[11] stems from the fact that no knowledge about the underlying channel law is required. As such, the resulting estimators can be used in settings where the channel is only accessible via input–output samples (e.g., in experimental setups) and the precise channel law is unknown.

*Notation:* Random variables are denoted by upper-case letters (e.g.,  $X$ ), realizations by lower-case letters (e.g.,  $x$ ), and sets by calligraphic letters (e.g.,  $\mathcal{X}$ ). The probability distribution of a random variable  $X$  is denoted by  $f_X$ . Expectation is denoted by  $\mathbb{E}[\cdot]$ , mutual information by  $I(\cdot; \cdot)$ , and Kullback–Leibler divergence by  $D(\cdot || \cdot)$ . For an integer  $N$ , we define the set  $[N] = \{1, 2, \dots, N\}$ .

## II. DUAL REPRESENTATION OF CHANNEL CAPACITY

We consider a memoryless channel<sup>2</sup> with input  $X \in \mathcal{X}$  and output  $Y \in \mathcal{Y}$ . The channel law conditioned on a particular input  $x$  is denoted by  $f_{Y|X=x}(y)$ . In general, the input is assumed to be constrained by a cost function  $c : \mathcal{X} \rightarrow \mathbb{R}_{\geq 0}$ . The capacity of such a channel is

$$C = \max_{f_X: \mathbb{E}_{f_X}[c(X)] \leq P} I(X; Y), \quad (1)$$

where  $P$  denotes the maximum average cost. In the following, we work with the dual representation [15, p. 142]

$$C = \min_{\gamma \geq 0} [F(\gamma) + \gamma P], \quad (2)$$

<sup>1</sup>See also the earlier work in [14] based on reinforcement learning, which, however, requires knowledge about the channel law.

<sup>2</sup>To keep the notation simple, we focus on scalar channels. However, our approach generalizes to (block-wise) memoryless channels where the input and outputs are (possibly complex-valued) random vectors, see Sec. VI-C.

where

$$F(\gamma) = \min_{q_Y} \max_{x \in \mathcal{X}} [D(f_{Y|X=x} || q_Y) - \gamma c(x)]. \quad (3)$$

The minimization in (3) is over all distributions  $q_Y$  on  $\mathcal{Y}$ . Note that any fixed choice for the reference distribution  $q_Y$  leads to an upper bound in (3) and hence on the capacity (2).

According to [16], the above dual approach first originated in [17] and was further developed in [18], [15], and [19]. Recent work exploiting this approach has mostly focused on choosing  $q_Y$  to obtain a tractable analytical expression for the resulting upper bound, see, e.g., [16], [19]. In this paper, we will instead use (3) as a blueprint for an iterative optimization procedure that alternates between training a divergence estimator (see Sec. III) and the reference distribution  $q_Y$  (see Sec. IV) based on the obtained estimator. The resulting algorithm is described in Sec. V.

### III. DATA-DRIVEN DIVERGENCE ESTIMATION

We use the approach proposed in [13] to estimate the divergence term in (3). This approach is based on the Donsker-Varadhan (DV) representation [13, Th. 1]

$$D(f_{Y|X=x} || q_Y) = \sup_{T \in \mathcal{T}} \mathbb{E}_{f_{Y|X=x}} [T(Y)] - \log \left( \mathbb{E}_{q_Y} [e^{T(Y)}] \right), \quad (4)$$

where the supremum is over all functions  $T: \mathcal{Y} \rightarrow \mathbb{R}$  such that the expectations in (4) are finite. The idea in [13] is to approximate the class of functions  $\mathcal{T}$  using a neural network (NN)  $T_\theta: \mathcal{Y} \rightarrow \mathbb{R}$ , where  $\theta$  are the NN parameters.  $T_\theta$  is also referred to as the statistics network. For a fixed set of parameters  $\theta$ , the resulting estimator is

$$\hat{D}_\theta = \frac{1}{B} \sum_{i=1}^B T_\theta(y^{(i)}) - \log \left( \frac{1}{B} \sum_{i=1}^B e^{T_\theta(\tilde{y}^{(i)})} \right), \quad (5)$$

where  $B$  is the batch size,  $y^{(1)}, \dots, y^{(B)} \sim f_{Y|X=x}$ , and  $\tilde{y}^{(1)}, \dots, \tilde{y}^{(B)} \sim q_Y$ . This estimator can be iteratively trained by running gradient ascent on (5), see [13, Alg. 1] for details.

*Remark:* We use the above estimator mainly for its simplicity and the fact that it empirically tends to work well (see, e.g., [11]). However, one should be aware that using the lower-bound estimator (5) does not guarantee to result in a true upper bound on capacity, which would in principle require a full optimization over  $\mathcal{T}$ . Moreover, the gradients resulting from (5) are biased [13] and the estimator has high variance, especially if the true underlying divergence is large [20]. We comment on potential alternative approaches in Sec. VII.

Note that the estimator (5) assumes a fixed channel input  $x$ . A different statistics network  $T_\theta$  would thus be required for each input to evaluate the maximization in (3) over the input alphabet. However, this quickly becomes infeasible if the size of the input alphabet is large or infinite. To circumvent this problem, we propose a modified version of (5) where the input  $x$  is taken as an additional input to the statistics network  $T_\theta$ , i.e.,  $T_\theta: \mathcal{Y} \times \mathcal{X} \rightarrow \mathbb{R}$ . The resulting modified estimator is denoted by

$$\hat{D}_\theta(x) = \frac{1}{B} \sum_{i=1}^B T_\theta(y^{(i)}, x) - \log \left( \frac{1}{B} \sum_{i=1}^B e^{T_\theta(\tilde{y}^{(i)}, x)} \right). \quad (6)$$

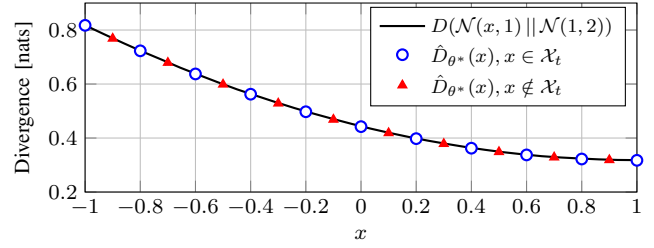


Fig. 1: Accuracy and generalization ability of the trained divergence estimator (6), where  $\mathcal{X}_t = \{-1, -0.8, \dots, +1\}$ .

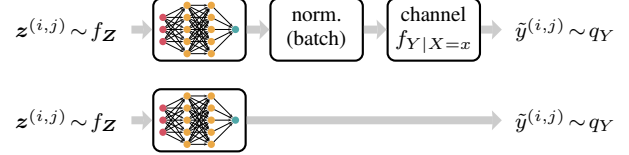


Fig. 2: Two approaches for implementing the neural distribution transformer (NDT) that generates samples from the reference distribution  $q_Y$ .

This formulation allows us to train a single statistics network that works well for a range of different channel inputs. In particular, this can be accomplished by jointly considering multiple inputs  $\{x^{(1)}, \dots, x^{(N_t)}\} = \mathcal{X}_t \subseteq \mathcal{X}$  and running gradient descent on an average loss  $-\frac{1}{N_t} \sum_j \hat{D}_\theta(x^{(j)})$ . However, averaging the logarithms in (6) leads to a geometric mean  $\frac{1}{N_t} \sum_j \log a_j = \log((\prod_j a_j)^{1/N_t})$ . We found that replacing this geometric mean with an arithmetic mean gives a numerically more stable training behavior, resulting in

$$L_\theta = -\frac{1}{N_t B} \sum_{j=1}^{N_t} \sum_{i=1}^B T_\theta(y^{(i,j)}, x^{(j)}) + \log \left( \frac{1}{N_t B} \sum_{j=1}^{N_t} \sum_{i=1}^B e^{T_\theta(\tilde{y}^{(i,j)}, x^{(j)})} \right), \quad (7)$$

where  $y^{(1,j)}, \dots, y^{(B,j)} \sim f_{Y|X=x^{(j)}}$  and  $\tilde{y}^{(1,j)}, \dots, \tilde{y}^{(B,j)} \sim q_Y$ . Note that (7) has the same functional form as the Monte Carlo approximation of the DV representation of  $-I(X; Y)$ , assuming that the input is uniformly distributed over  $\mathcal{X}_t$ .

*Example:* Assume that  $f_{Y|X=x}$  and  $q_Y$  correspond to  $\mathcal{N}(x, 1)$  and  $\mathcal{N}(1, 2)$ , respectively. To optimize the parameters  $\theta$ , we set  $\mathcal{X}_t = \{-1, -0.8, \dots, +1\}$  and train the statistics network<sup>3</sup> using the loss (7). Fig. 1 compares the accuracy of the resulting estimator  $\hat{D}_{\theta^*}(x)$  to the true divergence as a function of  $x$ , where  $\theta^*$  refers to the optimized parameters. Note that  $\hat{D}_{\theta^*}(x)$  generalizes well even to values of  $x$  that were not seen during training, as illustrated by the red triangles.

### IV. REPRESENTATION OF THE REFERENCE DISTRIBUTION

To allow for the gradient-based optimization of the reference distribution  $q_Y$ , two different approaches are described in the following for generating the samples  $\tilde{y}^{(1,j)}, \dots, \tilde{y}^{(B,j)} \sim q_Y$  in (7) using NNs. The corresponding block diagrams are shown in Fig. 2. Similar to [7], we refer to the resulting transformation as the neural distribution transformer (NDT).

<sup>3</sup>The NN architecture and all other training hyperparameters for this example are the same as for case (i) in Tab. I below.

In the first approach (Fig. 2, top), the NDT consists of an NN  $f_\tau$  with parameters  $\tau$ , which is then followed by a batch-wise normalization procedure and transmission over the channel. Thus, this approach generates channel inputs as an intermediate step. The normalization procedure enforces the average cost constraint<sup>4</sup> via

$$\hat{s}^{(i,j)} = \frac{s^{(i,j)}}{c^{-1} \left( \frac{1}{P} \sum_{i=1}^B c(s^{(i,j)}) \right)}, \quad (8)$$

where  $s^{(i,j)} = f_\tau(\mathbf{z}^{(i,j)})$ ,  $i \in [B]$ ,  $j \in [N_t]$  is the NN output and  $\mathbf{z}^{(i,j)} \in \mathbb{R}^l$  is a random vector sampled from a fixed latent probability distribution  $f_{\mathbf{Z}}$ . Note that (8) implicitly assumes that the cost function distributes over division, i.e.,  $c(a/b) = c(a)/c(b)$ , which is the case for all cost functions considered in this paper. The above approach ensures that  $q_Y$  is a valid output distribution for the channel under consideration, given the cost constraint. However, it should be noted that it requires a differentiable channel model in order to compute gradients with respect to  $\tau$ .

In the second approach (Fig. 2, bottom), the NDT directly generates samples from the channel output alphabet and simply consists of an NN  $f_\tau$  as before but without any additional post-processing, i.e.,  $\hat{y}^{(i,j)} = f_\tau(\mathbf{z}^{(i,j)})$ . While this approach does not necessarily ensure that  $q_Y$  is a valid output distribution for cost-constrained channels (which is not required to obtain an upper bound in (3)), it is more universal and can be used even if the channel is only accessible as a black box through input–output samples (e.g., in an experimental setting). On the other hand, we found that this representation typically requires more training steps when optimizing the NN parameters  $\tau$ .

## V. PROPOSED ALGORITHM

The proposed capacity estimation algorithm is detailed in Algorithm 1. It alternates between training the statistics network  $T_\theta$  (lines 3–5) and the NDT network  $f_\tau$  (lines 6–8) for a total of  $M$  iterations. The latter optimizes the reference distribution  $q_Y$  based on the loss function (cf. (3))

$$\begin{aligned} \hat{F}_\tau(\gamma) &= \max_{j \in [N_d]} \left[ \hat{D}_\theta(x^{(j)}) - \gamma c(x^{(j)}) \right] \\ &= \max_{j \in [N_d]} \left[ \frac{1}{B} \sum_{i=1}^B T_\theta(y^{(i,j)}, x^{(j)}) \right. \\ &\quad \left. - \log \left( \frac{1}{B} \sum_{i=1}^B e^{T_\theta(\hat{y}^{(i,j)}, x^{(j)})} \right) - \gamma c(x^{(j)}) \right], \end{aligned} \quad (9)$$

where the dependence of  $\hat{F}_\tau(\gamma)$  on the parameters  $\tau$  is implicit through the samples  $\hat{y}^{(i,j)}$  generated by the NDT. Compared to (3), the outer minimization over  $q_Y$  is encapsulated in the NN parameters  $\tau$ , which are optimized via gradient descent in Algorithm 1.

The definition of the sets  $\mathcal{X}_t$  and  $\{x^{(1)}, \dots, x^{(N_d)}\} = \mathcal{X}_d \subseteq \mathcal{X}$  in lines 3 and 6 depends on whether the input alphabet

---

### Algorithm 1: Estimation of capacity upper bounds.

---

- 1 **Inputs:** number of iterations  $M$ , batch size  $B$ , learning rate  $\beta$ , Lagrange multiplier  $\gamma$  (for cost-constrained channels), input sets  $\mathcal{X}_t$  (see Sec. III) and  $\mathcal{X}_d$  (see Sec. V)
  - 2 **for**  $l = 1, 2, \dots, M$  **do**
  - 3    $\forall x^{(j)} \in \mathcal{X}_t, i \in [B]$ : generate  $y^{(i,j)} \sim f_{Y|X=x^{(j)}}, \tilde{y}^{(i,j)} \sim q_Y$
  - 4    $L_\theta \leftarrow$  compute average loss according to (7)
  - 5    $\theta \leftarrow \theta - \beta \nabla_\theta L_\theta$    /\* update statistics network \*/
  - 6    $\forall x^{(j)} \in \mathcal{X}_d, i \in [B]$ : generate  $y^{(i,j)} \sim f_{Y|X=x^{(j)}}, \tilde{y}^{(i,j)} \sim q_Y$
  - 7    $\hat{F}_\tau(\gamma) \leftarrow$  estimate upper bound according to (9)
  - 8    $\tau \leftarrow \tau - \beta \nabla_\tau \hat{F}_\tau(\gamma)$    /\* update NDT network \*/
  - 9 **return**  $\hat{F}_\tau(\gamma)$
- 

TABLE I: NN parameters for the (i) average-power-constrained AWGN, (ii) peak-power-constrained AWGN, (iii) OI, and (iv) NLPN channel.

	NDT network $f_\tau$			statistics network $T_\theta$			
	layer	input	hidden	output	input	hidden	output
(i)	# neurons	50	2 × 100	1 (linear)	2	2 × 100	1 (linear)
(ii)	# neurons	50	2 × 100	1 (tanh)	2	2 × 100	1 (linear)
(iii)	# neurons	50	2 × 100	1 (sigmoid)	2	2 × 100	1 (linear)
(iv)	# neurons	50	2 × 150	2 (linear)	4	2 × 150	1 (linear)

is discrete or continuous. For channels with discrete input alphabet  $\mathcal{X}$ , we may set  $\mathcal{X}_t = \mathcal{X}_d = \mathcal{X}$ .<sup>5</sup> If the input alphabet of the channel is continuous, we assume that the input space has been appropriately discretized and the resulting set of discretized inputs is given by  $\mathcal{X}_d$ . A native, but more involved, approach for channels with continuous inputs that does not require any input space discretization is suggested in Sec. VII.

## VI. NUMERICAL RESULTS

In this section, we numerically evaluate the proposed approach.<sup>6</sup> Note that for cost-constrained channels, Algorithm 1 estimates the capacity in (3) as a function of the Lagrange multiplier  $\gamma \geq 0$ . In this case, we use a golden-section search to solve the outer one-dimensional minimization over  $\gamma$  in (2).

For all considered cases, we use fully-connected NNs with rectified linear unit activation functions in the hidden layers to represent both the NDT network  $f_\tau$  and the statistics network  $T_\theta$ . The NN parameters are summarized in Tab. I. The number of input neurons for  $f_\tau$  depends on the latent distribution, which we assume to be a multivariate Gaussian distribution  $\mathcal{N}(\mathbf{0}_l, \mathbf{I}_l)$  with  $l = 50$ . We found that the results are relatively insensitive to the choice of  $l$ , which mainly affects the initial distribution for  $q_Y$  before training. Moreover, while we have verified both NDT approaches discussed in Sec. IV, the following numerical results use the first approach (Fig. 2, top) since all channel models below are differentiable.

For the gradient-based optimization steps in Algorithm 1 (lines 5 and 8), we employ the Adam optimizer [21] with learning rate  $\beta = 10^{-3}$  and batch size  $B = 20000$ . Lastly, we always pretrain the statistics network  $T_\theta$  by running 200 initial iterations of lines 3–5 in Algorithms 1, which we found to improve training convergence.

<sup>5</sup>In this case, the data generated in line 3 can be reused in line 6.

<sup>6</sup>The source code to reproduce all numerical results in this paper is available at [https://github.com/chaeger/upper\\_capacity\\_bounds](https://github.com/chaeger/upper_capacity_bounds).

<sup>4</sup>Note that this procedure enforces the average cost constraint with equality, even if the inequality constraint was satisfied before the normalization.

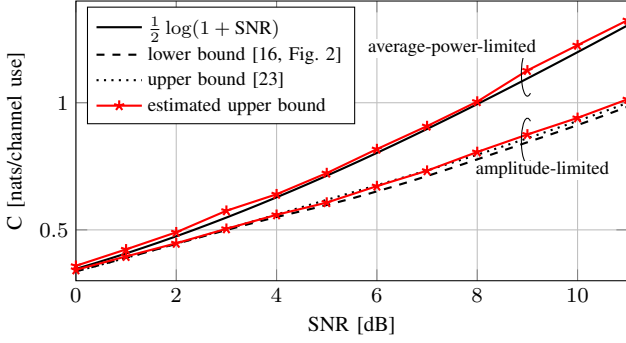


Fig. 3: Results for the AWGN channel.

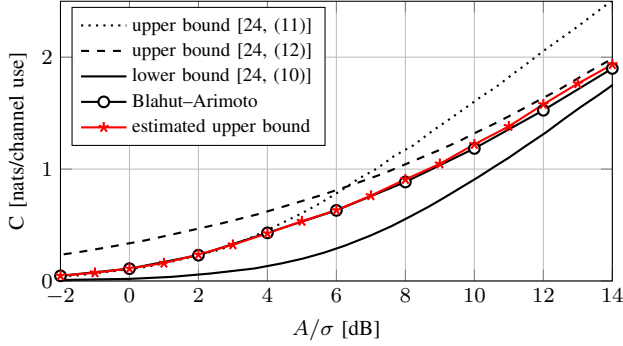


Fig. 4: Results for the OI channel with  $\alpha = 0.4$  (cf. [24, Fig. 2])

### A. AWGN Channel

We start with the additive white Gaussian noise (AWGN) channel  $Y = X + Z$ , where  $X$  is the channel input and  $Z \sim \mathcal{N}(0, \sigma^2)$ . For the average-power-limited case, we have  $c(x) = x^2$  as a cost function, in which case  $C = \frac{1}{2} \log(1 + \text{SNR})$  with  $\text{SNR} = P/\sigma^2$ . For the amplitude-limited case, we instead have no cost constraint,  $|X| \leq A$ , and  $\text{SNR} = A^2/\sigma^2$ . In this case, no closed-form analytical capacity expressions exist, but upper and lower bounds have been derived, see, e.g., [22], [23], [16].

For the numerical estimation, we set  $P = 1$  and  $A = 1$  without loss of generality and vary the SNR by varying  $\sigma^2$ . The input space is discretized using  $N_d = 15$  uniformly spaced grid points in the intervals  $[-2.5, 2.5]$  and  $[-1, 1]$  for the average-power-limited and amplitude-limited case, respectively. In this paper, we always assume for simplicity that  $\mathcal{X}_t = \mathcal{X}_d$ , noting that in general the set  $\mathcal{X}_d$  can be different from the set  $\mathcal{X}_t$  used to train the divergence estimator. Fig. 3 shows the estimated upper bounds after  $M = 500$  iterations, where we compare to the upper bound in [23] and lower bound in [16, Fig. 2] for the amplitude-limited case. Note that the NDT network  $f_\tau$  for the amplitude-limited case uses a tanh activation function in the last layer (cf. Tab. I) to enforce the amplitude constraint. Moreover, due to the absence of a cost constraint, no normalization procedure is applied and the outputs of  $f_\tau$  are directly transmitted over the channel to generate the NDT output samples  $\tilde{y}^{(i,j)}$ .

### B. Optical Intensity Channel

Next, we consider the optical intensity (OI) channel which is defined by  $Y = X + Z$  with  $c(x) = x$ ,  $X \in [0, A]$ , and  $Z \sim \mathcal{N}(0, \sigma^2)$  [24]. We consider the case  $\alpha = P/A = 0.4$ .

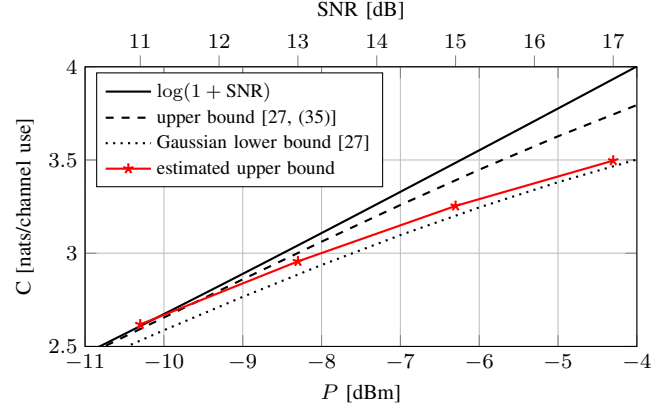


Fig. 5: Results for the NLPN channel.

For the numerical estimation, we set  $P = 1$  and discretize the input space using 15 uniformly spaced grid points in the interval  $[0, A]$ , where  $A = 2.5$ . For this case, the NDT network uses a sigmoid activation (scaled by  $A$ ) in the last layer to ensure that the channel input satisfies the amplitude constraint. Results are shown in Fig. 4 after  $M = 500$  iterations, where we compare to the upper and lower capacity bounds developed in [24], see in particular [24, Fig. 2]. The gap of the estimated upper bound with respect to the lower bound is due to the fact that the latter is not tight. Indeed, to verify that the channel capacity is close to our estimated upper bound, we used the Blahut–Arimoto algorithm for cost-constrained channels [15, p. 140] (black circles). We also note that tighter analytical upper bounds can potentially be obtained by extending the methodology in [23] to this channel model.

### C. Nonlinear Phase Noise Channel

Lastly, we consider an NLPN channel for coherent optical communication which is based on a split-step solution of the nonlinear Schrödinger equation without dispersive effects. The resulting complex-valued channel is defined by the recursion

$$X_{k+1} = X_k e^{j\gamma L |X_k|^2 / K} + N_{k+1}, \quad 0 \leq k \leq K, \quad (10)$$

where  $X = X_0 \in \mathbb{C}$  is the input,  $Y = X_K \in \mathbb{C}$  is the output,  $N_{k+1} \sim \mathcal{CN}(0, \sigma^2/K)$ ,  $\sigma^2$  is the total noise power,  $\gamma$  is a nonlinearity parameter,  $L$  is the transmission distance, and  $c(x) = |x|^2$ . This channel has a long history in terms of capacity analysis, see [25], [26], [5], [27], [28] and references therein. Here, we use the same parameters as in [5], [27], i.e.,  $K = 50$ ,  $\sigma^2 = -21.3$  dBm,  $\gamma = 1.27$  rad/km/W, and  $L = 5000$  km.

For the numerical estimation, we consider a renormalized version of (10), where  $\tilde{X}_k = X_k/\sqrt{P}$ . The input space of the renormalized channel is discretized using 9 uniformly spaced grid points in the interval  $[-1.75, 1.75]$  for both the real and imaginary part, i.e.,  $N_d = 81$  total discretization points. Separating the real and imaginary parts of the channel input and output, respectively, the number of NDT output neurons is increased to 2 and the number of input neurons of the statistics network to 4. Fig. 5 shows the obtained results after  $M = 2500$  iterations as a function of  $P$  (see the top axis for a conversion to  $\text{SNR} = P/\sigma^2$ ). It can be seen that the estimated upper



bound closely follows the lower bound in [27], which is based on a Gaussian input distribution.

## VII. DISCUSSION AND FUTURE WORK

We have proposed a novel data-driven approach for estimating upper bounds on channel capacity. Similar to recent work in [6], [7], the proposed algorithm relies on the DV representation for estimating divergence, with the consequence that the resulting estimates are neither true upper nor lower bounds for a finite sample size [29]. Even assuming an infinite sample size, one cannot guarantee that the resulting estimates are true upper bounds on capacity, which would in principle require a full optimization over the function class  $\mathcal{T}$  (cf. (4)). It is therefore important to properly choose the NN and training parameters. For example, more training iterations were required for the NLPN channel compared to the other cases to ensure convergence of the NDT and statistics networks.

An overview of potential alternative divergence estimation approaches can be found in [8] and [11]. Moreover, [9] recently proposed an approach to obtain outer bounds on the achievable rate region of memoryless multiple-access channels by exploiting the upper bounds based on  $f$ -divergence inequalities from [30]. However, the histogram-based approach to numerically evaluate these bound in [9] does not directly lead to a differentiable loss function. Therefore, additional modifications (e.g., based on ideas similar to [31]) would be required to be able to use such inequalities in conjunction with the NDT optimization in Algorithm 1.

Lastly, another reason why the proposed approach does not necessarily compute true upper bounds for continuous-input channels is that the maximization over  $x$  is only done approximately via discretization. To ensure that the capacity of the resulting input-discretized channel is close to that of the original channel, one approach is to successively increase the number of discretization points  $N_d$  until convergence. For future work, it may be interesting to develop native estimation approaches that do not require such a discretization. This could be done, for example, by considering an auxiliary distribution over the input space and casting the maximization in (3) as an optimization problem over this auxiliary distribution. Similar to the NDT, the auxiliary distribution could then again be parameterized using an NN.

## REFERENCES

- [1] C. E. Shannon, "A mathematical theory of communication," *Bell System Technical Journal*, vol. 27, pp. 379–423, Jul. 1948.
- [2] R. Blahut, "Computation of channel capacity and rate-distortion functions," *IEEE Trans. Inf. Theory*, vol. 18, no. 4, pp. 460–473, Jul. 1972.
- [3] S. Arimoto, "An algorithm for computing the capacity of arbitrary discrete memoryless channels," *IEEE Trans. Inf. Theory*, vol. 18, no. 1, pp. 14–20, Jan. 1972.
- [4] I. Naiss and H. H. Permuter, "Extension of the Blahut–Arimoto Algorithm for Maximizing Directed Information," *IEEE Trans. Inf. Theory*, vol. 59, no. 1, pp. 204–222, Jan. 2013.
- [5] S. Li, C. Häger, N. Garcia, and H. Wymeersch, "Achievable Information Rates for Nonlinear Fiber Communication via End-to-end Autoencoder Learning," in *Proc. European Conf. Optical Communication (ECOC)*, Rome, Italy, 2018.
- [6] R. Fritschek, R. F. Schaefer, and G. Wunder, "Deep Learning for Channel Coding via Neural Mutual Information Estimation," in *Proc. IEEE Int. Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, Cannes, France, 2019.
- [7] Z. Aharoni, D. Tsur, Z. Goldfeld, and H. H. Permuter, "Capacity of Continuous Channels with Memory via Directed Information Neural Estimator," in *Proc. IEEE Int. Symp. Information Theory (ISIT)*, Los Angeles, CA, 2020.
- [8] R. Fritschek, R. F. Schaefer, and G. Wunder, "Neural Mutual Information Estimation for Channel Coding: State-of-the-Art Estimators, Analysis, and Performance Comparison," in *Proc. IEEE Int. Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, Atlanta, GA, 2020.
- [9] F. Mirkarimi and N. Farsad, "Neural Computation of Capacity Region of Memoryless Multiple Access Channels," in *Proc. IEEE Int. Symp. Information Theory (ISIT)*, Melbourne, Australia, 2021.
- [10] N. A. Letizia and A. M. Tonello, "Capacity-Driven Autoencoders for Communications," *IEEE Open J. Commun. Soc.*, vol. 2, pp. 1366–1378, Jun. 2021.
- [11] F. Mirkarimi, S. Rini, and N. Farsad, "Neural Capacity Estimators: How Reliable Are They?" *arXiv:2111.07401 [cs.IT]*, Nov. 2021.
- [12] T. O’Shea and J. Hoydis, "An Introduction to Deep Learning for the Physical Layer," *IEEE Trans. Cogn. Commun. Netw.*, vol. 3, no. 4, pp. 563–575, Dec. 2017.
- [13] M. I. Belghazi, A. Baratin, S. Rajeswar, S. Ozair, Y. Bengio, A. Courville, and R. D. Hjelm, "Mutual Information Neural Estimation," in *Proc. Int. Conf. Mach. Learning (ICML)*, Stockholm, Sweden, 2018.
- [14] Z. Aharoni, O. Sabag, and H. H. Permuter, "Computing the Feedback Capacity of Finite State Channels using Reinforcement Learning," in *Proc. IEEE Int. Symp. Information Theory (ISIT)*, Paris, France, 2019.
- [15] I. Csiszár and J. Körner, *Information Theory: Coding Theorems for Discrete Memoryless Systems*. Academic Press, 1981.
- [16] A. Thangaraj, G. Kramer, and G. Böhcherer, "Capacity Bounds for Discrete-Time, Amplitude-Constrained, Additive White Gaussian Noise Channels," *IEEE Trans. Inf. Theory*, vol. 63, no. 7, pp. 4172–4182, Jul. 2017.
- [17] F. Topsøe, "An information theoretical identity and a problem involving capacity," *Studia Sci. Math. Hungarica*, vol. 2, pp. 291–292, 1967.
- [18] J. Kemperman, "On the Shannon capacity of an arbitrary channel," *Indagationes Math.*, vol. 77, no. 2, pp. 101–115, Jan. 1974.
- [19] A. Lapidoth and S. Moser, "Capacity bounds via duality with applications to multiple-antenna systems on flat-fading channels," *IEEE Trans. Inf. Theory*, vol. 49, no. 10, pp. 2426–2467, Oct. 2003.
- [20] D. McAllester and K. Stratos, "Formal Limitations on the Measurement of Mutual Information," in *Proc. Int. Conf. Artificial Intelligence and Statistics (AISTATS)*, virtual, 2020.
- [21] D. P. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization," in *Proc. Int. Conf. Learning Representations (ICLR)*, San Diego, CA, 2015.
- [22] A. McKellips, "Simple tight bounds on capacity for the peak-limited discrete-time channel," in *Proc. IEEE Int. Symp. Information Theory (ISIT)*, Chicago, IL, 2004.
- [23] B. Rassouli and B. Clerckx, "An Upper bound for the capacity of amplitude-constrained scalar AWGN channel," *IEEE Commun. Lett.*, vol. 20, no. 10, pp. 1924–1926, Oct. 2016.
- [24] A. Lapidoth, S. M. Moser, and M. A. Wigger, "On the Capacity of Free-Space Optical Intensity Channels," *IEEE Trans. Inf. Theory*, vol. 55, no. 10, pp. 4449–4461, Oct. 2009.
- [25] K. S. Turitsyn, S. A. Derevyanko, I. V. Yurkevich, and S. K. Turitsyn, "Information Capacity of Optical Fiber Channels with Zero Average Dispersion," *Phys. Rev. Lett.*, vol. 91, no. 20, p. 203901, Nov. 2003.
- [26] M. I. Yousefi and F. R. Kschischang, "On the per-sample capacity of nondispersive optical fibers," *IEEE Trans. Inf. Theory*, vol. 57, no. 11, pp. 7522–7541, Nov. 2011.
- [27] K. Keykhosravi, G. Durisi, and E. Agrell, "Accuracy Assessment of Nondispersive Optical Perturbative Models through Capacity Analysis," *Entropy*, vol. 21, no. 8, pp. 1–19, Aug. 2019.
- [28] A. V. Reznichenko and I. S. Terekhov, "Path integral approach to nondispersive optical fiber communication channel," *Entropy*, vol. 22, no. 6, pp. 1–30, May 2020.
- [29] B. Poole, S. Ozair, A. Van Den Oord, A. A. Alemi, and G. Tucker, "On variational bounds of mutual information," in *Proc. Int. Conf. Mach. Learning (ICML)*, Long Beach, California, 2019.
- [30] I. Sason and S. Verdú, "f-Divergence Inequalities," *IEEE Trans. Inf. Theory*, vol. 62, no. 11, pp. 5973–6006, Nov. 2016.
- [31] E. Ustinova and V. Lempitsky, "Learning deep embeddings with histogram loss," in *Proc. Advances in Neural Information Processing Systems (NIPS)*, Barcelona, Spain, 2016.