

# Efficiently Computable Converse for Finite-Blocklength Communication

Felipe Areces, Dan Song, and Richard Wesel  
Dept. of Electrical and Computer Engineering  
University of California, Los Angeles  
Los Angeles, CA USA  
Email: {fareces99,dansong,wesel}@ucla.edu

Aaron B. Wagner  
School of Electrical and Computer Engineering  
Cornell University  
Ithaca, NY 14853 USA  
Email: wagner@cornell.edu

**Abstract**—This paper presents a method for computing a finite-blocklength converse for the rate of fixed-length codes with feedback used on discrete memoryless channels (DMCs). The new converse is expressed in terms of a stochastic control problem whose solution can be efficiently computed using dynamic programming and Fourier methods. For channels such as the binary symmetric channel (BSC) and binary erasure channel (BEC), the accuracy of the proposed converse is similar to that of existing special-purpose converse bounds, but the new converse technique can be applied to arbitrary DMCs. We provide example applications of the new converse technique to the binary asymmetric channel (BAC) and the quantized amplitude-constrained AWGN channel.

## I. INTRODUCTION

Consider communicating one of  $M$  messages using a finite blocklength transmission of  $n$  symbols over a discrete memoryless channel (DMC) with average error probability  $\epsilon$ . A fundamental practical problem that arises from this setup is the computation of upper bounds on the size of  $M$  (converse) in the finite-blocklength regime when both  $n$  and  $\epsilon$  are fixed. A hypothesis testing framework can be used to establish converse bounds for arbitrary DMCs [1]–[4]. However, these bounds are generally hard to compute at finite blocklengths due to the high dimensionality of their parameter space. Their computation can be significantly reduced by exploiting symmetry in some cases, such as that of the binary-symmetric channel (BSC), the binary-erasure channel (BEC) [5], and the AWGN channel [6]. Their limits in various asymptotic regimes can also be computed. These insights provides no assistance at finite blocklengths for arbitrary DMCs, however.

Recent advances toward this goal include an algorithm for saddle point identification [7] in the “minimax” version of the bound [5] that is polynomial-time in the blocklength for DMCs, a linear programming formulation of the bound [8] that is also polynomial-time for DMCs, and a method for generating good output distributions for general minimax converses where an optimization over a distribution on the

channel output is required [9]. The complexity of these approaches is exponential in the input alphabet however, making them unsuitable for even moderately sized channels at longer blocklengths. See [10] and [11] for approximations of the bound that are more easily computed.

Stochastic control methods have proven useful in deriving results for communication systems with feedback, such as a general framework for the computation of channel capacity [12] and the timid/bold technique to improve second-order coding performance [13]. In many circumstances, stochastic control problems require the design of an optimal controller and dynamic programming (DP) techniques are often used to solve this problem efficiently [14].

## A. Contributions and organization

In this paper, we develop an efficient algorithm to compute converses at finite blocklength for arbitrary DMCs. The key idea is to consider the channel with feedback, which allows for efficient computation of the resulting hypothesis testing bounds using dynamic programming and Fourier methods. We describe this algorithm, analyze its complexity, and demonstrate its utility on several channels. For the BSC and BEC, we find that it provides bounds that are close to those obtained via channel-specific methods. On the other hand, we also show that our bound scales well to DMCs with moderately large alphabets at moderate blocklengths, for which there is currently no known way of efficiently computing nontrivial converse bounds. Of course, any converse bound for a DMC with feedback also applies to the channel without feedback.

The rest of the paper is organized as follows. In Section II we formulate the converse bound as a DP problem. In Section III we present an algorithm to compute the bound and analyze its complexity. Section IV presents numerical results for the BSC, BEC, BAC, and quantized amplitude constrained AWGN channel, and provides a comparison to existing bounds where possible. If no specific bounds are available we use Lemma 14 in [13] with a capacity achieving distribution to obtain a general achievability result without feedback.

This research is supported by National Science Foundation (NSF) grant CCF-1955660. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect views of the NSF.

## II. DMC CONVERSES VIA DYNAMIC PROGRAMMING

### A. Definitions

1) *Probability of Success/Failure*: Assume a DMC with finite input alphabet  $\mathcal{X}$  and finite output alphabet  $\mathcal{Y}$ . In this case, for an  $(n, R)$  code with ideal feedback and error probability  $\epsilon \in (0, 1)$  we define:

$$M^*(n, \epsilon) := \max\{[\exp(nR)] \in \mathbb{N}_+ : \bar{\mathbf{P}}_{\text{e,fb}}(n, R) \leq \epsilon\}, \quad (1)$$

where  $\bar{\mathbf{P}}_{\text{e,fb}}(n, R)$  is the minimum average error probability achievable by any  $(n, R)$  code with feedback. We follow [13]. In particular, given input and output random vectors

$$\mathbf{X}^n = (X_1, \dots, X_n) \in \mathcal{X}^n \quad (2)$$

$$\mathbf{Y}^n = (Y_1, \dots, Y_n) \in \mathcal{Y}^n, \quad (3)$$

a (stochastic) controller

$$F = F^n \quad (4)$$

$$F^k = (F_1, \dots, F_k) \quad (5)$$

$$F_t : (\mathbf{X}^{t-1}, \mathbf{Y}^{t-1}) \mapsto X_t, \quad (6)$$

a channel  $W \in \mathcal{P}(\mathcal{Y}|\mathcal{X})$ , an information density threshold  $T$ , and an arbitrary output distribution  $Q \in \mathcal{P}(\mathcal{Y})$ , we refer to the event in which the accumulated information density (AID) at the end of the block exceeds  $T$  as a *success*,  $S$ . The probability of success for any controller can be defined in terms of  $\mathbf{X}^n$  and  $\mathbf{Y}^n$  as

$$P(S_{F,Q,T}) = (F \circ W) \left( \log_2 \left[ \frac{W(\mathbf{Y}^n|\mathbf{X}^n)}{Q(\mathbf{Y}^n)} \right] > T \right), \quad (7)$$

where  $F \circ W$  denotes the joint distribution over  $\mathbf{X}^n$  and  $\mathbf{Y}^n$  induced by the controller and the channel. Similarly, the probability of failure is defined as  $P(\bar{S}_{F,Q,T}) = 1 - P(S_{F,Q,T})$ .

2) *Information density*: Given an input distribution to a DMC  $G \in \mathcal{P}(\mathcal{X})$  and an output distribution  $Q \in \mathcal{P}(\mathcal{Y})$  we can define the information density change from one transmission  $(\Delta I)$  as

$$\Delta I_G = \log_2 \left( \frac{W(Y|X)}{Q(Y)} \right) \quad (8)$$

with range  $\mathcal{I}_G$  and PMF  $f_G(\Delta i)$

$$f_G(\Delta i) = \sum_{x \in \mathcal{X}} \sum_{y \in \mathcal{Y}} \mathbb{1} \left( \Delta i = \log_2 \left[ \frac{W(y|x)}{Q(y)} \right] \right) W(y|x) G(x). \quad (9)$$

We finally define the set of all possible AID changes in a single transmission as

$$\mathcal{I} = \left\{ \log_2 \left( \frac{W(y|x)}{Q(y)} \right) \mid x \in \mathcal{X}, y \in \mathcal{Y} \right\} \quad (10)$$

and the set of all possible AID levels attainable by any controller at timestep  $k$  as

$$\Lambda_k = \left\{ \sum_{\gamma \in \mathcal{I}} \alpha_\gamma \gamma \mid \alpha_\gamma \in \mathbb{Z}^+ \text{ and } \sum_{\gamma \in \mathcal{I}} \alpha_\gamma = k \right\}, \quad (11)$$

where  $\mathbb{Z}^+$  is the set of nonnegative integers.

### B. Mathematical formulation

1) *General setup*: From Lemma 15 in [13] and the max-min inequality we know that given parameters  $W, n, \rho > 0, \epsilon > 0$  and  $T = \log_2(\rho)$ ,

$$R_b^*(n, \epsilon) \leq \inf_Q \sup_F \frac{1}{n} \left[ T - \log_2 \left( [P(\bar{S}_{F,Q,T}) - \epsilon]^+ \right) \right], \quad (12)$$

where  $R_b^*(n, \epsilon)$  is the best achievable code rate in bits for the given blocklength  $n$  and error probability  $\epsilon$ .

2) *Computation of maximum probability of success*: From the supremum in (12) we see that we are interested in finding the optimal controller  $F^*$  that minimizes  $P(\bar{S}_{F,Q,T})$  or, equivalently, maximizing  $P(S_{F,Q,T})$  for a fixed  $Q$ . In order to translate this problem into a dynamic programming framework we will focus on the probability of success of the optimal controller at timestep  $k$  and a specific AID level,  $S_k^*(\Gamma)$ , where references to  $Q$  and  $T$  are omitted for simplicity. With this setup every  $S_k^*(\Gamma)$  can be defined in terms of the following recurrence relation as long as the chosen  $Q$  is a product distribution:

$$S_n^*(\Gamma) = \mathbb{1}(\Gamma \geq T) \quad (13)$$

$$S_k^*(\Gamma) = \max_{G \in \mathcal{P}(\mathcal{X})} \sum_{\Delta i \in \mathcal{I}_G} S_{k+1}^*(\Gamma + \Delta i) f_G(\Delta i), \quad k < n. \quad (14)$$

However, we know that the cardinality of the set of all possible achievable AID levels grows polynomially with the blocklength, which makes direct computation infeasible even for moderate values of  $n$ . In order to address this problem we will quantize the AID in a sequence of  $N$  bins  $\beta_i = (\tau_i^L, \tau_i^U]$ ,  $i \in \{L_L, L_L + 1, \dots, L_U\}$  of size  $\delta$  such that

$$\frac{\tau_{L_U}^U}{n} = \Delta I_{\max} = \max_{(x,y) \in \mathcal{X} \times \mathcal{Y}} \log_2 \left( \frac{W(y|x)}{Q(y)} \right) \quad (15)$$

$$\frac{\tau_{L_L}^L}{n} = \Delta I_{\min} = \min_{(x,y) \in \mathcal{X} \times \mathcal{Y} : W(y|x) > 0} \log_2 \left( \frac{W(y|x)}{Q(y)} \right) \quad (16)$$

$$N = \left\lceil \frac{\Delta I_{\max} - \Delta I_{\min}}{\delta} \right\rceil. \quad (17)$$

This definition guarantees that all achievable AID levels will be comprised in this range as no single walk can exceed the bin limits. Our scheme also rounds up any AID realization to the upper bound of its corresponding bin (denoted as  $\lceil t \rceil_\delta = \delta \lceil \frac{t}{\delta} \rceil$ ). Equations (13) and (14) thus become

$$\hat{S}_n^*(\Gamma) = \mathbb{1}(\Gamma \geq T) \quad (18)$$

$$\hat{S}_k^*(\Gamma) = \max_{G \in \mathcal{P}(\mathcal{X})} \sum_{\Delta i \in \mathcal{I}_G} \hat{S}_{k+1}^*(\Gamma + \lceil \Delta i \rceil_\delta) f_G(\Delta i) \quad (19)$$

for  $\Gamma \in \{\tau_{L_L}^U, \tau_{L_L+1}^U, \dots, \tau_{L_U}^U\}$ , where  $\hat{S}_k^*(\Gamma)$  represents the proposed upper bound on  $S_k^*(\Gamma)$  obtained with rounding. Denote the AID random variable

$$\Delta I_k = \log_2 \left( \frac{W(Y_k|X_k)}{Q(Y_k)} \right), \quad (20)$$

which has distribution depending on the controller  $F^k$ . We can now show that this rounding strategy provides an upper

bound on the probability of success of the optimal controller and thus preserves the validity of the converse

$$P(S_{F^*,Q,T}) = \sup_F (F \circ W) \left( \sum_{j=1}^n \Delta I_j \geq T \right) \quad (21)$$

$$= \sup_{F^{n-1}, F_n} \sum_{\Gamma_{n-1} \in \Lambda_{n-1}} \left[ (F^{n-1} \circ W) \left( \sum_{j=1}^{n-1} \Delta I_j = \Gamma_{n-1} \right) \cdot (F_n \circ W) \left( \Delta I_n \geq T - \Gamma_{n-1} \left| \sum_{j=1}^{n-1} \Delta I_j = \Gamma_{n-1} \right. \right) \right] \quad (22)$$

$$= \sup_{F^{n-1}} \sum_{\Gamma_{n-1} \in \Lambda_{n-1}} \left[ (F^{n-1} \circ W) \left( \sum_{j=1}^{n-1} \Delta I_j = \Gamma_{n-1} \right) \cdot \sup_{F_n} (F_n \circ W) (\Delta I_n \geq T - \Gamma_{n-1}) \right] \quad (23)$$

$$\leq \sup_{F^{n-1}} \sum_{\Gamma_{n-1} \in \Lambda_{n-1}} \left[ (F^{n-1} \circ W) \left( \sum_{j=1}^{n-1} \Delta I_j = \Gamma_{n-1} \right) \cdot \sup_{F_n} (F_n \circ W) (\Delta I_n \geq T - \lceil \Gamma_{n-1} \rceil \delta) \right] \quad (24)$$

$$\leq \sup_{F^{n-1}} \sum_{\ell=L_L}^{L_U} \left[ (F^{n-1} \circ W) \left( \left\lceil \sum_{j=1}^{n-1} \Delta I_j \right\rceil_\delta = \ell \delta \right) \cdot \sup_P \sum_x P(x) \sum_y W(y|x) \mathbb{1}(\lceil \Delta I_n + \ell \delta \rceil_\delta \geq T) \right] \quad (25)$$

Since maximization over  $P$  is a linear program over a simplex, the maximum is achieved at a vertex, i.e. the deterministic distribution equal to some  $x$ .

$$= \sup_{F^{n-1}} \sum_{\ell=L_L}^{L_U} \left[ (F^{n-1} \circ W) \left( \left\lceil \sum_{j=1}^{n-1} \Delta I_j \right\rceil_\delta = \ell \delta \right) \cdot \max_x \sum_y W(y|x) \hat{S}_n^*(\lceil \Delta I_n + \ell \delta \rceil_\delta) \right] \quad (26)$$

$$= \sup_{F^{n-1}} \sum_{\ell=L_L}^{L_U} \left[ (F^{n-1} \circ W) \left( \left\lceil \sum_{j=1}^{n-1} \Delta I_j \right\rceil_\delta = \ell \delta \right) \cdot \hat{S}_{n-1}^*(\ell \delta) \right] \quad (27)$$

Continuing,

$$= \sup_{F^1} \sum_{\ell=L_L}^{L_U} \left[ (F^1 \circ W) (\lceil \Delta I_1 \rceil_\delta = \ell \delta) \hat{S}_1^*(\ell \delta) \right] \quad (28)$$

$$= \hat{S}_0^*(0). \quad (29)$$

Thus it suffices to compute  $\hat{S}_0^*(0)$ , which can be done recursively. Note that we are not computing the upper bound in [13] exactly, but rather an upper bound on it. The parameter  $\delta$  controls the tradeoff between the complexity of the algorithm and the weakening of the bound.

### III. ALGORITHMS

#### A. Algorithm for computation of $\hat{S}_0^*(0)$

1) *General setup:* The computation of  $\hat{S}_0^*(0)$  can be carried out efficiently by defining the transition from one bin to another when using the  $j$ -th input as a discrete time-invariant Markov process where the state space is simply the set of all bins. Using this definition we note that the probability of moving from bin  $\beta_m$  given that we start at the upper limit of bin  $\beta_l$  given the AID change distribution induced by the  $j$ -th input only depends on their distance  $(l - m)\delta$ , and we can thus define this probability as  $p_j^{l-m}$ . We can now define the Toeplitz transition matrix for the  $j$ -th input as

$$M_j = \begin{bmatrix} p_j^0 & p_j^1 & \cdots & p_j^{N-2} & p_j^{N-1} \\ p_j^{-1} & p_j^0 & \cdots & p_j^{N-3} & p_j^{N-2} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ p_j^{-N+2} & p_j^{-N+3} & \cdots & p_j^0 & p_j^1 \\ p_j^{-N+1} & p_j^{-N+2} & \cdots & p_j^{-1} & p_j^0 \end{bmatrix}, \quad (30)$$

where the max is taken component-wise. Since by construction the AID can never fall out of the defined range, this stochastic matrix can be used to compute the probability of success vector  $\hat{\mathbf{S}}_k^* = [\hat{S}_k^*(L_L\delta), \dots, \hat{S}_k^*(L_U\delta)]$  induced by the  $j$ -th input for all bins simultaneously as

$$\hat{\mathbf{S}}_k^* = \max_{j \in \{1, \dots, |\mathcal{X}|\}} M_j \hat{\mathbf{S}}_{k+1}^*, \quad (31)$$

where the maximum operation is an element-wise maximum over all  $M_j \hat{\mathbf{S}}_{k+1}^*$  vectors. We can now easily compute  $\hat{S}_0^*(0)$  as shown in algorithm 1.

---

**Algorithm 1:** Computation of probability of success for optimal controller

---

**Input:**  $W, Q, n, T, \delta$

**Output:**  $\hat{S}_0^*(0)$

---

- 1 Compute  $\Delta I_{max}$  and  $\Delta I_{min}$
  - 2 Construct bins with spacing  $\delta$  in  $[n\Delta I_{min}, n\Delta I_{max}]$
  - 3 Construct matrices  $M_j$  for  $j \in \{1, \dots, |\mathcal{X}|\}$
  - 4  $\hat{\mathbf{S}}_n^* \leftarrow \mathbf{1}$  if bin upper bound  $> T$  and 0 otherwise
  - 5 **for**  $k \in \{N-1, \dots, 0\}$  **do**
  - 6      $\hat{\mathbf{S}}_k^* = \mathbf{0}$
  - 7     **for**  $j \in \{1, \dots, |\mathcal{X}|\}$  **do**
  - 8          $\hat{\mathbf{S}}_k^* \leftarrow \text{Element-wise max}(\hat{\mathbf{S}}_k^*, M_j \hat{\mathbf{S}}_{k+1}^*)$
  - 9     **end**
  - 10 **end**
  - 11  $\hat{S}_0^*(0) \leftarrow \text{Bin of } \hat{\mathbf{S}}_0^* \text{ that contains } 0$
  - 12 **return**  $\hat{S}_0^*(0)$
-

2) *Algorithm complexity*: The complexity of Algorithm 1 depends on the technique used to compute the product  $M_j \hat{\mathbf{S}}_{k+1}^*$ . One possible method to perform this computation efficiently relies on the fact that all matrices  $M_j$  in (30) are Toeplitz, as this fact allows the use of the FFT to speed up computation. In this case the algorithm will have time complexity  $O(nN \log(N)|\mathcal{X}|)$  and space complexity  $O(N|\mathcal{X}|)$ .

Another possible method can be derived from the fact that all matrices  $M_j$  are generally sparse as when fixing an input the information density change can take at most  $|\mathcal{Y}|$  values, which translates to at most  $|\mathcal{Y}|$  non-zero elements in each row of the matrix. In this case the algorithm will have time complexity  $O(nN|\mathcal{Y}||\mathcal{X}|)$  and space complexity  $O(N|\mathcal{X}|)$ .

### B. Threshold optimization

1) *Properties of the optimization landscape*: Using algorithm 1 we can now obtain a general converse for any DMC by choosing an arbitrary  $Q$  that enforces independence between timesteps, and solving the optimization problem derived from equation (12)

$$R_b^* \leq \min_T \frac{1}{n} \left[ T - \log_2 \left( [P(\bar{S}_{F,Q,T}) - \epsilon]^+ \right) \right]. \quad (32)$$

This problem can be difficult to solve using traditional solvers because that the optimization landscape is piecewise linear and discontinuous as shown in Fig 1. Given a controller  $F$  and a threshold  $T$  we can express its probability of failure as a sum over all the possible AID outcomes for all controllers using  $\Delta I_k$  as defined in (20)

$$P(\bar{S}_{F,Q,T}) = \sum_{\Gamma \in \Lambda_n} \mathbb{1}(\Gamma \leq T) \cdot (F \circ W) \left( \sum_{j=1}^n \Delta I_k = \Gamma \right) \quad (33)$$

and note that this quantity must be monotonically increasing in  $T$  for any arbitrary controller. Therefore, given an optimal controller  $F^*$  for a threshold  $T$  just above a mass-point  $\lambda_m$  if the value of  $T$  is increased while staying below the next mass-point with non-zero probability  $\lambda_k$  the value of (33) will remain fixed for the original optimal controller as the sum is unchanged and it will still be optimal. Thus the bound would grow linearly until  $T$  exceeds  $\lambda_k$  where the probability of failure will experience a discrete increase causing the overall bound to potentially drop discontinuously.

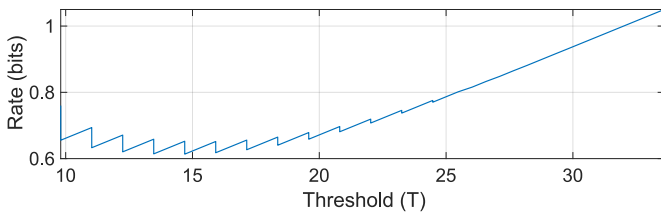


Fig. 1. Optimization landscape for BEC with erasure probability  $\delta = 0.3$  and probability of error  $\epsilon = 10^{-4}$  at  $n = 32$

2) *Heuristics*: It is possible to use the characteristics of the optimization landscape to develop two heuristics that improve the convergence of solvers in certain conditions:

- 1) Our previous discussion shows that a global minimum can only be achieved slightly above a mass-point in  $\Lambda$ , which makes it possible to restrict the solver to only evaluate the function slightly above valid mass-points.
- 2) Intuition and empirical results indicate that the optimal threshold grows approximately linearly with the blocklength. This property allows faster computation when evaluating the converse for an increasing sequence of blocklengths  $[n_1, n_2, \dots]$  as the converses for the first blocklengths can be computed using the full range of thresholds and subsequent converses can be computed by only analyzing a neighborhood around the threshold predicted by a linear regression obtained from the previous values.

Since the converse bound in (12) is valid for any threshold  $T$  the bounds obtained using this process are always valid even if the solver fails to find a minimizing  $T$ .

## IV. NUMERICAL RESULTS AND COMPARISONS

### A. Comparison of DP Converse to Existing Converses

In this section we compare the results from our DP converse (DPC) to the existing converse and achievability results in the SPECTRE toolbox [15] for the BSC and the BEC. These channels are good baselines for the algorithm as they have small input and output alphabets, well known converse and achievability results, and the optimal output distributions are known [5]. In the BSC case our algorithm produces very good results (Fig. 2) as for blocklengths above 300 the result is less than 1% above the channel-specific converse, and this gap closes rapidly as the blocklength increases. The BEC case provides a challenge to the DPC algorithm as [5] shows that the optimal distribution is not a product distribution and our algorithm is restricted to product distributions. However, Fig. 3 shows that the DPC result is still very close to the channel specific converse with a gap of less than 2.4% to the channel specific converse for blocklengths larger than 300. The fact that the optimal output distribution for the BEC is non-product is a possible explanation for the added looseness in the BEC converse when compared to our BSC results, but our algorithm provides good converse results even in this case.

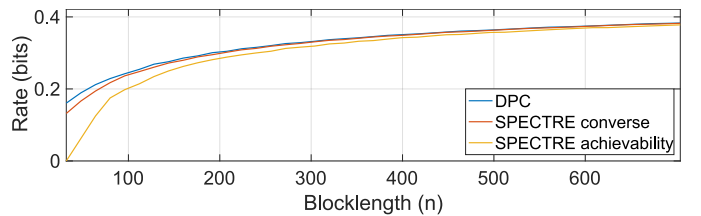


Fig. 2. DPC results for BSC with  $\epsilon = 10^{-4}$  and crossover probability  $p = 0.11$ . The capacity of the channel is approximately 0.5 bits.

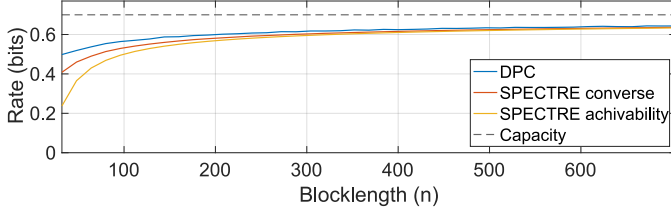


Fig. 3. DPC results for BEC with  $\epsilon = 10^{-4}$  and erasure probability  $p_e = 0.3$

### B. New Converse Bounds Using the DP Approach

Section IV-A shows that the performance of our DP algorithm is comparable to channel specific converses. This section applies the new algorithm to two DMCs not covered by the SPECTRE toolbox. Fig. 5 shows that the DPC produces a reasonable converse for the BAC. For blocklengths above 500 the result is less than 9.9% above the achievable rate computed using Lemma 14 of [13] with a capacity-achieving distribution. It is hard to determine whether this gap originates from looseness of the achievability or converse results, but the behavior of our converse is consistent with the BSC and BEC results.

Recall that existing algorithms for computing converses have complexity that scales with the blocklength  $n$  and input alphabet size  $|\mathcal{X}|$  as  $n^{|\mathcal{X}|}$ . Next we show that our DP algorithm can be used to provide bounds at moderately large  $n$  and  $\mathcal{X}$ . Consider a DMC with  $|\mathcal{X}| = |\mathcal{Y}| = 8$  corresponding to using an AWGN channel with an optimized input alphabet and quantized outputs as described in [16]. The input alphabet is a fixed set of constellation points  $\mathcal{X}$ . The output is quantized using thresholds  $-\infty = q_0 < q_1 < q_2 < \dots < q_7 < q_8 = \infty$ . With input  $x$ , the output is  $i$  if  $x + Z \in (q_{i-1}, q_i]$ , where  $Z \sim \mathcal{N}(0,1)$ . Such channels are of practical interest, and it is known that the optimal input distribution has finite-support [16] given some fixed quantization bins  $\{q_i\}$ . Dynamic Assignment Blahut-Arimoto [17], [18] can be used to find the optimal  $\mathcal{X}$  and input distribution when restricting the input of a channel to be finite-support. Modifying the algorithm for AWGN channel with amplitude constraint from [17] to account for output quantization and performing alternating optimization with  $\{q_i\}$  gives a choice of  $\mathcal{X}$  and  $\{q_i\}$ . Setting  $|\mathcal{Y}| = 8$  and the amplitude constraint to 10 yields an optimized  $\mathcal{X}$  of cardinality 8 as shown in Fig. 6, which yields the DMC described by transitions matrix  $W$  shown in Fig. 4.

$$W \approx \begin{bmatrix} 0.96 & 0.04 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 \\ 0.06 & 0.84 & 0.10 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 \\ 0.00 & 0.11 & 0.79 & 0.10 & 0.00 & 0.00 & 0.00 & 0.00 \\ 0.00 & 0.00 & 0.08 & 0.85 & 0.07 & 0.00 & 0.00 & 0.00 \\ 0.00 & 0.00 & 0.00 & 0.07 & 0.85 & 0.08 & 0.00 & 0.00 \\ 0.00 & 0.00 & 0.00 & 0.00 & 0.10 & 0.79 & 0.11 & 0.00 \\ 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.10 & 0.84 & 0.06 \\ 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.04 & 0.96 \end{bmatrix}$$

Fig. 4. Equivalent DMC representation for the quantized output amplitude constrained AWGN channel.

Fig. 7 shows that the DPC also provides a reasonable converse in this case. For blocklengths above 300 the converse is less than 6.8% above the achievable rate computed using Lemma 14 of [13] with a capacity-achieving distribution.

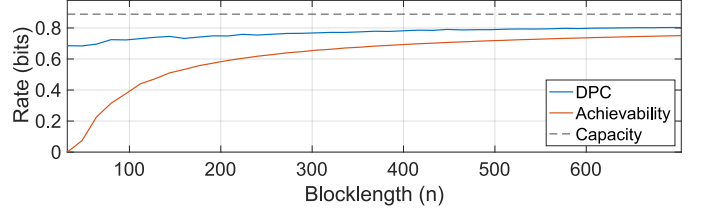


Fig. 5. DPC results for BAC with  $\epsilon = 10^{-4}$  and crossover probabilities  $p_1 = 0.01$ ,  $p_2 = 0.02$ . The achievability curve is derived from using Lemma 14 in [13] with a capacity achieving distribution.

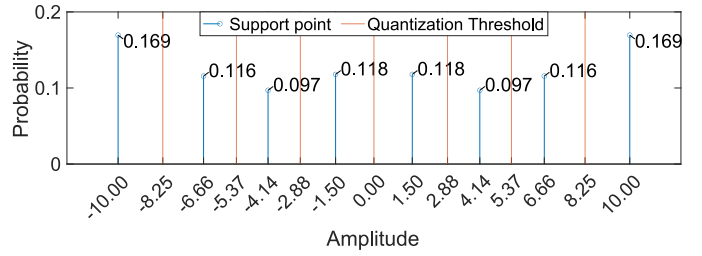


Fig. 6. Constellation and quantization thresholds that maximize mutual information for the unit noise AWGN channel with quantized output under amplitude constraint 10. These induce the DMC given by  $W$  shown above.

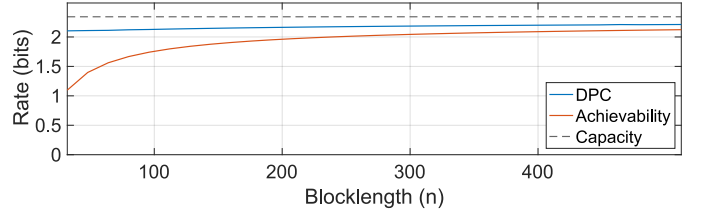


Fig. 7. DPC results for finite-alphabet, quantized-output AWGN channel described by the  $W$  shown above with  $\epsilon = 10^{-4}$ . The achievability curve is derived using Lemma 14 in [13] with a capacity achieving distribution.

## V. CONCLUSION

This paper presents a general technique for computing finite-blocklength converse bounds for an arbitrary DMC. The converse uses dynamic programming to solve a stochastic control formulation of the converse problem. The utility of the approach was verified for the BSC, BEC, BAC and a DMC resulting from an amplitude constrained AWGN channel with quantized output. Since the technique presented provides a valid bound for any product output distribution, the bound can be further improved by optimizing the output distribution over this set. Identifying the best such output distribution is a topic for future work, as is the extension to channels with cost constraints.

## REFERENCES

- [1] R. E. Blahut, "Hypothesis testing and information theory," *IEEE Transactions on Information Theory*, vol. 20, no. 4, pp. 405–417, 1974.
- [2] Y. Altuğ and A. B. Wagner, "Refinement of the sphere-packing bound: Asymmetric channels," *IEEE Trans. Inf. Theory*, vol. 60, no. 3, pp. 1592–1614, Mar. 2014.
- [3] —, "On exact asymptotics of the error probability in channel coding: symmetric channels," *IEEE Trans. Inf. Theory*, vol. 67, no. 2, pp. 844–868, Feb. 2021.
- [4] Y. Polyanskiy, H. V. Poor, and S. Verdú, "Channel coding rate in the finite blocklength regime," *IEEE Transactions on Information Theory*, vol. 56, no. 5, pp. 2307–2359, 2010.
- [5] Y. Polyanskiy, "Saddle point in the minimax converse for channel coding," *IEEE Transactions on Information Theory*, vol. 59, no. 5, pp. 2576–2595, 2013.
- [6] T. Erseghe, "On the evaluation of the polyanskiy-poor-verdu converse bound for finite blocklength coding in AWGN," *CoRR*, vol. abs/1401.7169, 2014. [Online]. Available: <http://arxiv.org/abs/1401.7169>
- [7] N. Elkayam and M. Feder, "On the calculation of the minimax-converse of the channel coding problem," 2016.
- [8] W. Matthews, "A linear program for the finite block length converse of Polyanskiy–Poor–Verdú via nonsignaling codes," *IEEE Transactions on Information Theory*, vol. 58, no. 12, pp. 7036–7044, 2012.
- [9] O. Kosut, "Boosting output distributions in finite blocklength channel coding converse bounds," in *2015 IEEE Information Theory Workshop (ITW)*, 2015, pp. 1–5.
- [10] G. Vazquez-Vilar, A. G. i Fabregas, T. Koch, and A. Lancho, "Saddlepoint approximation of the error probability of binary hypothesis testing," in *2018 IEEE International Symposium on Information Theory (ISIT)*, 2018, pp. 2306–2310.
- [11] T. Erseghe, "Coding in the finite-blocklength regime: Bounds based on laplace integrals and their asymptotic approximations," *IEEE Transactions on Information Theory*, vol. 62, no. 12, pp. 6854–6883, 2016.
- [12] S. Tatikonda and S. Mitter, "The capacity of channels with feedback," *IEEE Transactions on Information Theory*, vol. 55, no. 1, pp. 323–349, 2009.
- [13] A. B. Wagner, N. V. Shende, and Y. Altuğ, "A new method for employing feedback to improve coding performance," *IEEE Transactions on Information Theory*, vol. 66, no. 11, pp. 6660–6681, 2020.
- [14] S. C. Tatikonda, "Control under communication constraints," Ph.D. dissertation, Electrical Engineering and Computer Science, Massachusetts Institute of Technology, Cambridge, MA, 8 2000.
- [15] S. Chen, A. Collins, G. Durisi, T. Erseghe, G. C. Ferrante, V. Kostina, J. Östman, Y. Polyanskiy, I. Tal, and W. Yang, "SPECTRE: short-packet communication toolbox," 2021. [Online]. Available: [github.com/yp-mit/spectre](https://github.com/yp-mit/spectre)
- [16] J. Singh, O. Dabeer, and U. Madhow, "On the limits of communication with low-precision analog-to-digital conversion at the receiver," *IEEE Transactions on Communications*, vol. 57, no. 12, pp. 3629–3639, 2009.
- [17] D. Xiao, L. Wang, D. Song, and R. D. Wesel, "Finite-support capacity-approaching distributions for AWGN channels," in *2020 IEEE Information Theory Workshop (ITW)*, 2021, pp. 1–5.
- [18] N. Farsad, W. Chuang, A. Goldsmith, C. Kominakis, M. Médard, C. Rose, L. Vandenberghe, E. E. Wesel, and R. D. Wesel, "Capacities and optimal input distributions for particle-intensity channels," *IEEE Transactions on Molecular, Biological and Multi-Scale Communications*, vol. 6, no. 3, pp. 220–232, 2020.