

Scaling Laws for Line Networks: From Zero-Error to Min-Cut Capacity

Urs Niesen
Email: uniesen@mit.edu

Christina Fragouli
Email: christina.fragouli@epfl.ch

Daniela Tuninetti
Email: daniela@ece.uic.edu

Abstract— We consider communication through a cascade of L identical Discrete Memoryless Channels (DMCs). The source and destination node are allowed to use coding schemes of arbitrary complexity, but the intermediate relay nodes are restricted to process only blocks of N symbols. We are interested in the set of length- N processing performed at the relays that maximize the achievable rate between the source and the destination.

It is well known [1] that for any L and $N \rightarrow \infty$ the relays can use a capacity achieving code and communicate reliably as long as the rate of this code is below the capacity of the underlying DMC. The capacity of the cascade is hence equal to the network *min-cut capacity*. For finite N and $L \rightarrow \infty$, we showed in [2] that the optimal intermediate processing is the highest rate zero-error code of length N for of the underlying DMC. The capacity of the cascade hence coincides with the rate of this zero-error code, and is always below the *zero-error capacity*. In this work, we characterize how N must scale with L in order to achieve rates in between the zero-error and the min-cut capacity.

In [3] we showed that $N = \Theta(\log L)$ is *sufficient* to achieve any rate below the min-cut capacity. Here, we develop a novel upper bound on the capacity of cascades with optimal intermediate processing that apply for any (N, L) pairs and use it to show that $N = \Theta(\log L)$ is *necessary* to achieve certain rates above the zero-error capacity. Furthermore, we show how our upper bound can be efficiently evaluated, from a computation time point of view, by using well known approximation algorithms developed to solve the so-called Set-Covering Problem in graph theory.

I. INTRODUCTION

Communication systems today are organized in large scale networks, with Internet the most conspicuous example, where information needs to traverse multiple hops in order to reach its destination. Another such example are wireless ad-hoc networks where the average number of hops between a source-destination pair scales as the square root of the number of nodes in the network [4]. Each of the hops may introduce errors, that become more and more pronounced as the size of the network grows.

Motivated by these observations, in [2] we investigated what benefits finite complexity processing at intermediate nodes may offer. We modeled the communication path between the source and the destination as a line network that consists of L cascaded identical DMCs, and allowed each intermediate node to process blocks of N symbols. This is a reasonable definition of complexity as it allows to bound not only processing complexity, but also delay, and memory requirements at intermediate nodes. Moreover, it is well suited to an environment where information is transmitted in packets.

In [2] we showed that if the network length increases ($L \rightarrow \infty$) but the blocklength N is fixed, the optimal processing is identical at each relay and corresponds to an optimal (highest rate) zero-error code of length N . Thus, the capacity of the cascade coincides with the rate of this code and it is hence always below the *zero-error capacity* of the underlying channel. The zero-error capacity is the maximum rate at which information can be communicated over a channel with zero probability of error [5]. An intuitive interpretation of this result is that, as $L \rightarrow \infty$, the zero-error capacity is the only part of the transmitted information rate that we may hope to preserve. On the other hand, when $N \rightarrow \infty$ the relays can use a capacity achieving code and communicate reliably as long as the rate of this code is below the capacity of the DMC. That is, for $N \rightarrow \infty$ we can achieve the *min-cut capacity* [1].

Since the zero-error capacity and the min-cut capacity might differ quite substantially, a natural question to ask is what happens if we allow N to grow with L . In this paper we investigate how fast N needs to scale with L in order to achieve rates above the zero-error capacity. In [3] we showed that $N = \Theta(\log L)$ ¹ is *sufficient to achieve any rate below the min-cut capacity*. Here, we show that $N = \Theta(\log L)$ is *necessary to achieve certain rates above the zero-error capacity*.

In order to prove our result, we start by deriving a novel upper bound on the capacity of cascades with optimal intermediate processing that are valid for any L and N . We decompose the channel transition matrix into a linear combination of two stochastic matrices. We then develop a bound that depends on this decomposition through the smallest rank of these two matrices. With this bound we show that logarithmic scaling of N with L is necessary to achieve certain rates above the zero-error capacity.

We also show that finding the minimum rank decomposition of a channel transition matrix is equivalent to solving a *Set Cover Problem*, for which several approximation algorithms are available in the literature. This means that in polynomial time we can compute a decomposition of channel transition matrix whose gap from the optimal minimum rank decomposition vanishes as N increases (recall that here we are interested in non-finite N). Interestingly, the set-covering problem is the “dual” (in a sense to be made precise later) of the *Maximum*

¹We use Knuth’s notation: $f(n) = O(g(n))$ means that there exists a constant c and integer n_0 such that $f(n) \leq cg(n)$ for $n > n_0$. $f(n) = \Theta(g(n))$ denotes that $f(n) = O(g(n))$ as well as $g(n) = O(f(n))$.

Independent Set Problem, which is equivalent to determine the channel zero-error capacity.

The paper is organized as follows. Section II introduces the network model and briefly reviews our previous results. Section III presents the upper bound on the capacity that is used in Section IV to derive the scaling law. Section V concludes the paper.

II. MODEL AND PREVIOUS RESULTS

A. Model

We consider line networks with $L - 1$ relays as depicted in Figure 1. The source A_0 sends information to the destination A_L via relays $\{A_i\}_{i=1}^{L-1}$. Each link corresponds to the same DMC with finite input alphabet \mathcal{X} , finite output alphabet \mathcal{Y} , and arbitrary transition probability matrix \mathbf{V} . We impose that all the DMCs in the cascade are the same.

We restrict the relays $\{A_i\}_{i=1}^{L-1}$ to perform operations from blocks of N symbols in \mathcal{Y} to blocks of N symbols in \mathcal{X} in a memoryless way across blocks. Using N times the channel \mathbf{V} between A_i and A_{i+1} , amounts to connecting A_i and A_{i+1} through an equivalent DMC with input alphabet \mathcal{X}^N , output alphabet \mathcal{Y}^N , and transition probability matrix $\mathbf{W} \triangleq \mathbf{V}^{\otimes N}$ where \otimes denotes the Kronecker product. For the node A_i , we denote by $\mathbf{X}_i \in \mathcal{X}^N$ what the relay sends and with $\mathbf{Y}_i \in \mathcal{Y}^N$ what the relay receives. The output \mathbf{X}_i is then a (not necessarily deterministic) function of \mathbf{Y}_i . This function can be described by a transition probability matrix \mathbf{M}_i specifying for each realization \mathbf{x} of \mathbf{X}_i and \mathbf{y} of \mathbf{Y}_i the probability $\Pr[\mathbf{X}_i = \mathbf{x} | \mathbf{Y}_i = \mathbf{y}]$.

We allow the source A_0 and the destination A_L to perform coding and decoding of arbitrary complexity, across a possibly infinite number of symbols in \mathcal{X}^N and \mathcal{Y}^N .

We are interested in identifying the set of processings $\{\mathbf{M}_i\}_{i=1}^{L-1}$ that achieve

$$C_{N,L}(\mathbf{V}) \triangleq \max_{\{\mathbf{M}_i\}_{i=1}^{L-1}} \frac{1}{N} C \left(\mathbf{W} \prod_{i=1}^{L-1} (\mathbf{M}_i \mathbf{W}) \right).$$

Here, $C(\mathbf{Q}) = \max_{\mathbf{p}} I(\mathbf{p}, \mathbf{Q})$ where \mathbf{p} is the input distribution and \mathbf{Q} the channel transition matrix.

In this paper, we make connections between $C_{N,L}(\mathbf{V})$ and the zero-error capacity of the underlying channel \mathbf{V} . The zero-error capacity is defined in [5] as the maximum rate at which communication is possible with zero probability of error and can be computed as follows. For a channel with transition matrix \mathbf{V} , we call two input letters k and ℓ adjacent if there exists an output letter j such that $[\mathbf{V}]_{k,j} > 0$ and $[\mathbf{V}]_{\ell,j} > 0$. We then construct a graph $G(\mathbf{V})$ corresponding to the stochastic matrix \mathbf{V} having as vertex set the possible inputs of \mathbf{V} and in which two vertices are connected by an edge if the corresponding input letters are adjacent. Denote by $M_0(\mathbf{V})$ the largest number of input letters of \mathbf{V} no two of which are adjacent. The zero-error capacity of \mathbf{V} is

$$C_0(\mathbf{V}) \triangleq \sup_n \frac{1}{n} \log M_0(\mathbf{V}^{\otimes n}). \quad (1)$$

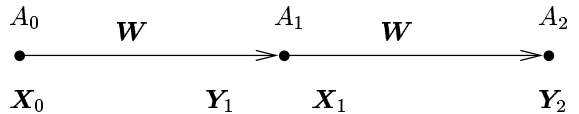


Fig. 1. A line network with two channels and one relay ($L = 2$).

B. Previous Results

Clearly, for any DMC with transition probability matrix \mathbf{V} , any intermediate processing of length N and any network length L , we have

$$\frac{1}{N} M_0(\mathbf{V}^{\otimes N}) \leq C_{N,L}(\mathbf{V}) \leq C(\mathbf{V}). \quad (2)$$

The lower bound in (2) is achievable by using the same zero-error code of length N at each node in the network, while the upper bound in (2) is achievable by using a capacity achieving code at each node in the network. That no other coding strategy can do better than $C(\mathbf{V})$ is clear from the min-cut bound. Hence, the upper bound in (2) is tight for $N \rightarrow \infty$ [1].

Our main result in [2] states that the lower bound in (2) is tight for finite N and $L \rightarrow \infty$:

Theorem II.1 (Allerton 2005).

$$\lim_{L \rightarrow \infty} C_{N,L}(\mathbf{V}) = \frac{1}{N} \log M_0(\mathbf{V}^{\otimes N}) \leq C_0(\mathbf{V}) \quad \square$$

This theorem tells us that the limit $\lim_{L \rightarrow \infty} C_{N,L}(\mathbf{V})$ exists and that the optimal processing is identical at each node in the limit for large L . This processing corresponds to the best, in the sense of highest rate, zero-error code of length N for the channel \mathbf{V} . Thus, the capacity of the infinite cascade equals the rate of this zero-error code and is always upper bounded by the zero-error capacity of \mathbf{V} . Notice that any rate strictly below the zero-error capacity is achievable with *finite length processing* since the limit superior in (1) is a true limit [6].

Since $C_0(\mathbf{V})$ and $C(\mathbf{V})$ might differ quite substantially, a natural question to ask is what happens if both N and L are allowed to grow. In [3] we derived the following lower bound on $C_{N,L}(\mathbf{V})$:

Theorem II.2 (ISITA 2004).

$$C_{N,L}(\mathbf{V}) \geq \max_{r \in [0, C(\mathbf{V})]} r (1 - \exp(-NE(r)))^L - \frac{1}{N},$$

where $E(r)$ is the random coding error exponent for the channel \mathbf{V} as a function of the rate r . \square

The bound in Theorem II.2 is tight for $N \rightarrow \infty$ [7].

III. UPPER BOUND

In this section, we will derive an upper bound on $C_{N,L}(\mathbf{V})$ that applies for all values of N and L . We then use this upper bound and the lower bound in Theorem II.2 to show that a processing length of $N = \Theta(\log L)$ is sufficient to achieve

$$C_{N,L}(\mathbf{V}) \geq R(\alpha) \triangleq (1 - \alpha) C_0(\mathbf{V}) + \alpha C(\mathbf{V}) \quad (3)$$

for all $\alpha \in [0, 1]$ and necessary for all $\alpha \in [\beta, 1]$ with $\beta \leq 1$.

Our upper bound for $C_{N,L}(\mathbf{V})$ is expressed as a linear combination of the min-cut capacity and of a term reminiscent of the limiting capacity $\frac{1}{N} \log M_0(\mathbf{V}^{\otimes N})$ derived in Theorem II.1. The basic idea is to decompose the equivalent channel transition matrix $\mathbf{V}^{\otimes N}$ into a linear combination of two stochastic matrices, one of which with rank as close as possible to $M_0(\mathbf{V}^{\otimes N})$. We also discuss efficient algorithms to determine such a decomposition.

Theorem III.1. *For any stochastic matrix \mathbf{V} and any integer N , if there exist two stochastic matrices \mathbf{A}_N and \mathbf{B}_N , and $\delta_N \in (0, 1]$ such that*

$$\mathbf{V}^{\otimes N} = \delta_N \mathbf{A}_N + (1 - \delta_N) \mathbf{B}_N \quad (4)$$

then

$$C_{N,L}(\mathbf{V}) \leq \left(1 - (1 - \delta_N)^{L-1}\right) \frac{\log \text{rank}(\mathbf{A}_N)}{N} + (1 - \delta_N)^{L-1} C(\mathbf{V}). \quad (5)$$

Proof. Assume (4) holds, then,

$$\begin{aligned} C_{N,L}(\mathbf{V}) &= \frac{1}{N} C\left(\mathbf{W} \prod_{i=1}^{L-1} (\mathbf{M}_i \mathbf{W})\right) \\ &= \frac{1}{N} C\left((\delta_N \mathbf{A}_N + (1 - \delta_N) \mathbf{B}_N) \prod_{i=1}^{L-1} (\mathbf{M}_i \mathbf{W})\right) \\ &\stackrel{(a)}{\leq} \delta_N \frac{1}{N} C\left(\mathbf{A}_N \prod_{i=1}^{L-1} (\mathbf{M}_i \mathbf{W})\right) \\ &\quad + (1 - \delta_N) \frac{1}{N} C\left(\mathbf{B}_N \prod_{i=1}^{L-1} (\mathbf{M}_i \mathbf{W})\right) \\ &\stackrel{(b)}{\leq} \delta_N \frac{C(\mathbf{A}_N)}{N} + (1 - \delta_N) \frac{1}{N} C\left(\mathbf{W} \prod_{i=2}^{L-1} (\mathbf{M}_i \mathbf{W})\right) \end{aligned}$$

where the inequality (a) follows from the convexity of mutual information in the channel transition matrix and (b) follows from the data processing inequality. By repeating the same argument we get

$$C_{N,L}(\mathbf{V}) \leq \left(1 - (1 - \delta_N)^{L-1}\right) \frac{C(\mathbf{A}_N)}{N} + (1 - \delta_N)^{L-1} C(\mathbf{V}).$$

We can further upper bound $C(\mathbf{A}_N)$ with the logarithm of the rank of \mathbf{A}_N [8] to yield (5). \square

In order to obtain the best bound for any given N , \mathbf{A}_N should be chosen to have the *smallest rank possible*. The possible choice $\mathbf{A}_N = \mathbf{A}_1^{\otimes N}$ and $\delta_N = \delta_1^N$ would give

$$C_{N,L}(\mathbf{V}) \leq \left(1 - (1 - \delta_1^N)^{L-1}\right) \log \text{rank}(\mathbf{A}_1) + (1 - \delta_1^N)^{L-1} C(\mathbf{V})$$

since $\text{rank}(\mathbf{A}_N) = (\text{rank}(\mathbf{A}_1))^N$. However, this choice does not give the best possible bound in general [7].

Notice that, for any matrix \mathbf{A}_N such that (4) holds, the inequality in (5) and Theorem II.1 imply

$$\frac{1}{N} \log M_0(\mathbf{V}^{\otimes N}) \leq \frac{1}{N} \log \text{rank}(\mathbf{A}_N). \quad (6)$$

It can be shown that if, for some N , we find $\text{rank}(\mathbf{A}_N) = M_0(\mathbf{V}^{\otimes N})$, then $C_0(\mathbf{V}) = \frac{1}{N} \log \text{rank}(\mathbf{A}_N)$. In this case, the bound in (5) is tight in the limit as $L \rightarrow \infty$ and the decay of $C_{N,L}(\mathbf{V})$ to the limiting capacity is exponentially fast in L . In [2] we showed that if we impose that $\mathbf{M}_i = \mathbf{M}$ for all $i \in \{1, \dots, L-1\}$ (identical processing), then the limiting capacity is equal to the logarithm on the number of eigenvalues of modulus one of $\mathbf{Q} = \mathbf{W}\mathbf{M}$ and the limit is approached exponentially fast in L . If $\text{rank}(\mathbf{A}_N) = M_0(\mathbf{V}^{\otimes N})$, then the exponential decay also applies to non identical processing. This implies that even for long, but not infinite, cascades the derived limiting result in Theorem II.1 is meaningful. It also highlights the importance of well chosen intermediate processing. The exponent that can be derived from (5) in the case $\text{rank}(\mathbf{A}_N) = M_0(\mathbf{V}^{\otimes N})$, namely

$$\lim_{L \rightarrow \infty} -\frac{1}{L} \log \left(C_{N,L}(\mathbf{V}) - \frac{1}{N} \log M_0(\mathbf{V}^{\otimes N}) \right) \geq -\log(1 - \delta_N),$$

is however not tight in general [7].

The problem of finding the matrix \mathbf{A}_N with minimum rank is equivalent to the *Set Cover Problem* described as follows [9]. Given a universe \mathcal{U} of n elements, a collection $\mathcal{S} = \{\mathcal{S}_1 \dots \mathcal{S}_m\}$ of subsets of \mathcal{U} , and a cost function for each subset in \mathcal{S} , find a minimum cost subcollection of \mathcal{S} that covers all the elements in \mathcal{U} . This problem can be formulated as an integer program by assigning a variable x_i for each set $\mathcal{S}_i \in \mathcal{S}$, where $x_i = 1$ if set \mathcal{S}_i takes part in the subcollection and $x_i = 0$ otherwise. The constraint is that every element in \mathcal{U} must belong to at least one of the picked sets \mathcal{S}_i . The set cover problem and its Linear Program (LP) relaxation (Primal) are provided in Table I, for the special case where the cost of all sets is one, which is the case of interest here. A variety of approximation algorithms are available in the literature for the set cover problem [9]. Those algorithms run in polynomial time and provide approximations with gap at most $\log n$ from the optimal solution.

In our case, the universe \mathcal{U} is the set of $n = |\mathcal{X}|^N$ inputs of the channel $\mathbf{V}^{\otimes N}$. The set of $m = |\mathcal{Y}|^N$ outputs defines \mathcal{S} , in that the subset \mathcal{S}_i contains the inputs that result with nonzero probability in output i , for $i = 1, \dots, m$. The solution of the set cover problem in our case provides us with a set of outputs that cover all inputs. Indeed, let $\text{OPT}_{\mathbf{A}_N}$ denote the minimum cost of the set cover problem. We can construct a matrix \mathbf{A}_N that has rank $\text{OPT}_{\mathbf{A}_N}$ as follows. Take the binary matrix $\lceil \mathbf{V}^{\otimes N} \rceil$, where the ceiling operation is component-wise. To construct matrix \mathbf{A}_N , replace with the all-zero columns those columns of $\lceil \mathbf{V}^{\otimes N} \rceil$ that correspond to outputs where $x_i = 0$ in the set cover problem. Then normalize all rows so that they sum to one. The matrix \mathbf{A}_N contains $\text{OPT}_{\mathbf{A}_N}$ nonzero columns and $m - \text{OPT}_{\mathbf{A}_N}$ all-zero columns, and thus $\text{rank}(\mathbf{A}_N) = \text{OPT}_{\mathbf{A}_N}$. Moreover, \mathbf{A}_N is a valid stochastic matrix because its nonzero columns “cover” all inputs, i.e., each row has at least one nonzero element, and all rows sum to one. Choose δ_N to be the largest number such that the matrix

$$\mathbf{B}_N = \frac{1}{1 - \delta_N} (\mathbf{V}^{\otimes N} - \delta_N \mathbf{A}_N)$$

TABLE I
MAXIMUM INDEPENDENT SET PROBLEM, SET COVER PROBLEM, AND
THEIR LP RELAXATIONS.

Set Cover Problem $\min \sum_i x_i$ $\sum_{i:u \in \mathcal{S}_i} x_i \geq 1, \forall u \in \mathcal{U}$ $x_i \in \{0, 1\}, \forall i$	LP Relaxation (Primal) $\min \sum_i x_i$ $\sum_{i:u \in \mathcal{S}_i} x_i \geq 1, \forall u \in \mathcal{U}$ $0 \leq x_i \leq 1, \forall i$
Max Independent Set Problem $\max \sum_j y_j$ $\sum_{j \in \mathcal{S}_i} y_j \leq 1 \forall \mathcal{S}_i \in \mathcal{S}$ $y_j \in \{0, 1\}, \forall j$	LP Relaxation (Dual) $\max \sum_j y_j$ $\sum_{j \in \mathcal{S}_i} y_j \leq 1 \forall \mathcal{S}_i \in \mathcal{S}$ $0 \leq y_j \leq 1, \forall j$

is a valid stochastic matrix. It is easy to see that this δ_N is at least as large as the minimum of the entries of $\mathbf{V}^{\otimes N}$ that correspond to a non-zero entry in \mathbf{A}_N . Moreover, by using one of the available approximation algorithms, we can calculate in polynomial time a matrix \mathbf{A}_N that has rank bigger than the minimum by a factor of at most $\log n = N \log |\mathcal{X}|$. As we are interested in $\frac{1}{N} \log(\text{rank}(\mathbf{A}_N))$ this implies that the loss we incur by using these approximation algorithms to find an \mathbf{A}_N matrix vanishes as $N \rightarrow \infty$.

The problem of finding the matrix \mathbf{A}_N with minimum rank is closely related to computing $M_0(\mathbf{V}^{\otimes N})$. To see this, consider the (strong) LP dual of the set cover LP relaxation described in Table I. The dual LP is the LP relaxation of the *Maximum Independent Set Problem*, which takes as input a graph adjacency matrix and calculates the graph independence number, that is, the largest number of vertices in the graph no two of which are connected by an edge. This maximum independent set problem can be formulated as an integer program as follows. Assign a variable y_j for each vertex of the graph, $y_j = 1$ if the vertex takes part in the independent set and $y_j = 0$ otherwise. The constraint is that no two picked vertices are connected with an edge.

For our purposes we use the adjacency matrix corresponding to the graph $G(\mathbf{V}^{\otimes N})$. The solution of the maximum independent set problem for $G(\mathbf{V}^{\otimes N})$ directly gives an optimal zero-error code of length N for the channel \mathbf{V} . Denote by OPT_{M_0} this optimal solution. Obviously, OPT_{A_N} is lower-bounded by OPT_{M_0} . In fact, OPT_{A_N} is the minimum number of outputs such that all inputs are covered. Since all inputs are covered, this implies that any $\text{OPT}_{A_N} + 1$ inputs have at least one output in common. Thus $\text{OPT}_{M_0} \leq \text{OPT}_{A_N}$.

Note that, in the instances where the integrality gap between the maximum independent set problem and its LP relaxation is small, then there exists a matrix \mathbf{A}_N such that $\frac{1}{N} \log \text{rank}(\mathbf{A}_N) \approx \frac{1}{N} \log M_0(\mathbf{V}^{\otimes N})$, i.e., the inequality in (6) becomes an equality with all the discussed consequences that this entails.

IV. SCALING LAWS

In this section, we shall show that logarithmic growth of N with L is sufficient, and in many cases also necessary, to achieve rates above the zero-error capacity.

Recall that any rate strictly below the zero-error capacity can be achieved with finite blocklength processing, because the limit superior in the definition of zero-error capacity in (1)

is a true limit [6]. Here we are interested in rates that indeed need an infinite blocklength. For any $\alpha \in [0, 1]$ and for $R(\alpha)$ as in (3), we define

$$N^*(L, \alpha) \triangleq \min \{N : C_{N,L}(\mathbf{V}) \geq R(\alpha)\}.$$

The next theorem gives an upper bound on $N^*(L, \alpha)$, establishing that logarithmic growth of N with L is sufficient to achieve $R(\alpha)$.

Theorem IV.1. *For every $\varepsilon > 0$ there is a N_0 such that for all $N \geq N_0$*

$$N^*(L, \alpha) \leq \min_{r \in [R(\alpha) + \varepsilon, C(\mathbf{V})]} \frac{1}{E(r)} \log \left(\frac{L}{1 - \frac{R(\alpha) + \varepsilon}{r}} \right) \quad (7)$$

is sufficient to achieve $R(\alpha)$.

Proof. Theorem II.2 asserts that for any $r \in [0, C(\mathbf{V})]$

$$C_{N,L}(\mathbf{V}) \geq r \left(1 - \exp(-N E(r)) \right)^L - \frac{1}{N}. \quad (8)$$

Since the right hand side of (8) is always smaller than r , in order to attain $R(\alpha)$, r must satisfy $r \geq R(\alpha)$. By setting the right hand side of (8) equal to $R(\alpha)$, and by using the fact that $(1-x)^L \geq 1-Lx$ for all $x \in [0, 1]$, we get

$$\log \left(1 - \frac{R(\alpha) + \frac{1}{N}}{r} \right) \leq \log(L) - N E(r).$$

Thus for every $\varepsilon > 0$ there is a N_0 such that for all $N > N_0$

$$N \leq \frac{1}{E(r)} \left(\log(L) - \log \left(1 - \frac{R(\alpha) + \varepsilon}{r} \right) \right). \quad (9)$$

But since this is true for all $r \in [R(\alpha) + \varepsilon, C(\mathbf{V})]$, we can minimize the right hand side of (9) over r to get the tightest bound. Since the right hand side of (9) is a continuous function of r and the optimization set is compact, the result in (7) follows. \square

The next theorem establishes that logarithmic growth of N with L is necessary to achieve $R(\alpha)$ for all $\alpha \geq \beta_m$ where β_m is a non-negative constant.

Theorem IV.2.

$$N^*(L, \alpha) \geq \frac{\log(L-1) - \log \log \frac{1}{\alpha - \beta_m}}{\frac{1}{m} \log \frac{1}{\delta_m}} \quad (10)$$

is necessary to achieve $R(\alpha)$ for all

$$\alpha \geq \beta_m \triangleq \frac{\frac{1}{m} \log \text{rank}(\mathbf{A}_m) - C_0(\mathbf{V})}{C(\mathbf{V}) - C_0(\mathbf{V})},$$

where m is any integer such that the stochastic matrices \mathbf{A}_m , \mathbf{B}_m , and the real-valued constant $\delta_m \in (0, 1]$ in the decomposition $\mathbf{V}^{\otimes m} = \delta_m \mathbf{A}_m + (1 - \delta_m) \mathbf{B}_m$ satisfies $\frac{1}{m} \log \text{rank}(\mathbf{A}_m) \leq C(\mathbf{V})$.

Proof. From Theorem III.1,

$$C_{N,L}(\mathbf{V}) \leq \left(1 - (1 - \delta_m^{N/m})^{L-1} \right) \frac{1}{m} \log \text{rank}(\mathbf{A}_m) + (1 - \delta_m^{N/m})^{L-1} C(\mathbf{V}).$$

In order to achieve $R(\alpha)$ it is hence necessary that

$$(1 - \alpha)C_0(\mathbf{V}) + \alpha C(\mathbf{V}) \leq (1 - \delta_m^{N/m})^{L-1} C(\mathbf{V}) \\ + (1 - (1 - \delta_m^{N/m})^{L-1}) \frac{1}{m} \log \text{rank}(\mathbf{A}_m).$$

By re-arranging the different terms, we get

$$\alpha - \beta_m \leq (1 - \delta_m^{N/m})^{L-1} \underbrace{\frac{C(\mathbf{V}) - \frac{1}{m} \log \text{rank}(\mathbf{A}_m)}{C(\mathbf{V}) - C_0(\mathbf{V})}}_{\triangleq \gamma_m}.$$

If the value of m is such that the hypothesis of the theorem are satisfied we have $\beta_m \in [0, 1]$ and $\gamma_m \in [0, 1]$, and hence

$$(1 - \delta_m^{N/m})^{L-1} \geq \alpha - \beta_m.$$

By using the fact that $-\log(1 - x) \geq x$ for all $x \in [0, 1]$, we get that for all $\alpha \geq \beta_m$

$$\log \frac{1}{\alpha - \beta_m} \geq (L - 1) \log \frac{1}{1 - \delta_m^{N/m}} \geq (L - 1) \delta_m^{N/m},$$

and hence

$$N \frac{1}{m} \log \frac{1}{\delta_m} \geq \log(L - 1) - \log \log \frac{1}{\alpha - \beta_m},$$

from which the result in (10) follows. \square

Notice that, since we are interested in the regime $N \gg 1$, the assumptions of the theorem about the integer m can be relaxed to $\lim_{m \rightarrow \infty} \frac{1}{m} \log \text{rank}(\mathbf{A}_m) \leq C(\mathbf{V})$.

The following examples illustrate the use of Theorems IV.1 and IV.2.

Example IV.1. [Binary Symmetric channel.]

Consider a cascade of L binary symmetric channels with crossover probability p . For this channel $C_0(\mathbf{V}) = 0$ and hence we get from Theorem II.1 that any finite length processing performed at every node in the network will result in a zero end-to-end rate as $L \rightarrow \infty$. From Theorem IV.1, we see that logarithmic growth of N with L is *sufficient* to achieve any fraction of the min-cut capacity. We will now show that $N = \Theta(\log L)$ but also *necessary* to achieve any positive fraction of the min-cut capacity. The equivalent channel matrix between any pairs of nodes is $\mathbf{V}^{\otimes N}$ whose smallest entry is p^N . By collecting p^N from all the entries of $\mathbf{V}^{\otimes N}$, we can write

$$\mathbf{V}^{\otimes N} = p^N \mathbf{1}\mathbf{1}^T + (1 - (2p)^N) \mathbf{B},$$

for some stochastic matrix \mathbf{B} , and where $\mathbf{1}$ is the all one column vector of length 2^N . By using this decomposition we get from Theorem IV.2 that

$$N^*(L, \alpha) \geq \frac{(L - 1) - \log \log \frac{1}{\alpha}}{\log \frac{1}{\delta}},$$

with $\delta \triangleq 2p$. Hence logarithmic growth of N with L is necessary for all positive rates. \diamond

Example IV.2. [Pentagon channel.] Consider the ‘‘pentagon’’ channel whose transition matrix \mathbf{V} , for $p \in (0, 1)$, is

$$\mathbf{V} = \begin{pmatrix} 1-p & p & 0 & 0 & 0 \\ 0 & 1-p & p & 0 & 0 \\ 0 & 0 & 1-p & p & 0 \\ 0 & 0 & 0 & 1-p & p \\ p & 0 & 0 & 0 & 1-p \end{pmatrix}.$$

Here $C(\mathbf{V}) = \log(5) - H(p)$ and $C_0(\mathbf{V}) = \log(5)/2$.

For $m = 1$ we can find a matrix \mathbf{A}_1 with $\text{rank}(\mathbf{A}_1) = 3$. However, for $m = 2$ we can find a matrix \mathbf{A}_2 with $\text{rank}(\mathbf{A}_2) = 8 < \text{rank}(\mathbf{A}_1)^2 = 9$. Using the decomposition for $m = 2$ we obtain from Theorem IV.2

$$\beta_2 = \frac{\frac{1}{2} \log \text{rank}(\mathbf{A}_2) - C_0(\mathbf{V})}{C(\mathbf{V}) - C_0(\mathbf{V})} = \frac{\log 8/5}{\log 5 - 2H(p)},$$

showing that logarithmic growth of N with L is necessary to achieve any $R(\alpha)$ with $\alpha \geq \beta_2$. With $m = 2$, Theorem IV.2 does in this case not allow to state that logarithmic growth is necessary for all $\alpha \in [0, 1]$. \diamond

V. CONCLUSIONS

In this paper, we investigated communication through a cascade of L channels, where intermediate nodes can perform a processing on blocks of N symbols. When N is fixed and L goes to infinity, the network capacity can never exceed the zero-error capacity of the underlying channel. On the other hand, when $N \rightarrow \infty$ the network capacity coincides with the min-cut capacity, i.e., the capacity of the underlying channel. Here we study how N must scale with L in order to achieve rates above the zero-error capacity.

We derived bounds on the capacity of finite length cascades and used them to show that logarithmic growth of N with L is sufficient to achieve any rate below the min-cut capacity and necessary to achieve certain rates above the zero-error capacity. We conjecture that logarithmic growth is actually necessary to achieve *any* rate above the zero error capacity. Proving this conjecture and extending the work to more general networks, and more general traffic configurations and resource constraints is part of ongoing work.

REFERENCES

- [1] T. Cover, J. Thomas, ‘‘Elements of Information Theory,’’ *Wiley Series*, 1991.
- [2] U. Niesen, C. Fragouli, D. Tuninetti, ‘‘On the Capacity of an Infinite Cascade of Channels,’’ *Allerton 2005*, Monticello, IL (USA) Sept. 2005.
- [3] D. Tuninetti, C. Fragouli, ‘‘Processing along the way: forwarding vs. coding,’’ *ISITA 2004*, Parma (Italy) Oct. 2004.
- [4] P. Gupta, P. R. Kumar, ‘‘The Capacity of Wireless Networks,’’ *IEEE Trans. on Info. Theory*, Vol. 46(2), March 2000.
- [5] C. E. Shannon, ‘‘The Zero Error Capacity of a Noisy Channel,’’ *IEEE Trans. on Info. Theory*, Vol. 2(3), Sept. 1956.
- [6] J. Körner, A. Orłitsky, ‘‘Zero-Error Information Theory,’’ *IEEE Trans. on Info. Theory*, Vol. 44(6), Oct. 1998.
- [7] U. Niesen, C. Fragouli, D. Tuninetti, ‘‘On Capacity of Line Networks,’’ *submitted to IEEE Trans. on Info. Theory on Dec. 2005*.
- [8] C. E. Shannon, ‘‘Some Geometrical Results in Channel Capacity,’’ *Nachrichtentechnische Zeitschrift*, Vol.10, 1957.
- [9] V. Vazirani, ‘‘Approximation Algorithms,’’ *Springer*, 2001.