



香港中文大學(深圳)
The Chinese University of Hong Kong, Shenzhen

我在港中深的学习体验

王捷

2020年7月11日



个人介绍

- 2016年从红岭中学毕业（高考分数621分）
- 选择理论数学作为专业
- 课余生活主要参与科研工作，连续三年的学术论文被信息论顶会（ISIT2018-2020）接收

On the Tightness of a Cut-Set Bound on Network Function Computation

Jie Wang, Shenghao Yang, and Congdian Li

Abstract—The following model of network function computation in directed acyclic networks is considered: A sink node desires to compute correctly a target function with all possible inputs of the function generated at multiple source nodes. The network links have limited capacity and are error-free. The intermediate network nodes perform network coding without any computation bound. The computing rate is measured by the average number of times that the target function can be computed for one use of the network. Guang, Yang and Li recently proposed a general upper bound on the computing capacity that is tight for all the instances of the problem with known computing capacity in literature. In this paper, we show that their upper bound is not tight in general by explicitly characterizing the computing capacity of an example. Our technique can be extended to characterize upper bounds on the computing capacity of a general instance of the network function computing problem.

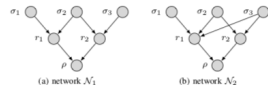


Fig. 1: Network N_1 and N_2 have three binary source nodes $\sigma_1, \sigma_2, \sigma_3$ and one sink node ρ . Each source node may generate 0 or 1, and each edge can transmit one bit per use. The arithmetic sum f_{sum} of the source messages is required to be computed at the sink node ρ .

for general network topologies and the target functions. The bound of Guang, Yang and Li (GYL bound) [6] is in general better than the one in [5]. In particular, for the instance in Fig. 1a, the GYL bound is tight, while the one in [5] not. In general, it is unclear that whether the GYL bound is tight. For all instances of the network function computation problem with known computing capacity that have been studied in literature, the GYL bound is tight. In [8], the computation of the max function is studied in the butterfly network, where the GYL bound is 2, but no rate 2 codes exist. However, the argument in [8] does not exclude the existence of a sequence of codes with rates converging to 2. The tightness of GYL bound is still an open problem by far.

In this paper, we show that the GYL bound is not tight using an explicit instance of the network function computation problem given in Fig. 1b. For this instance, it can be evaluated that the GYL bound is 1, while we show that the computing capacity is $\frac{1}{2} \log_2 6$. Our technique used to characterize the computing capacity of this instance is related to graph coloring used in [9], where the inputs follow certain probability distribution. We also show that this technique can be extended to characterize upper bounds on the computing capacity of a general instance of the network function computing problem.

II. NETWORK FUNCTION-COMPUTING CODES

In this section, we present the model of network function computation as in [5], [6] and discuss some previous upper bounds.

A. Problem Setup

Let $G = (V, E)$ be a directed acyclic graph (DAG) with a finite node set V and an edge set E , where multi-edges between a pair of nodes are allowed. For an edge $e = (u, v)$, we call u the tail of e and v the head of e . A network over G is denoted as $\mathcal{N} = (G, S, \rho)$, where $S \subset V$ is the set of the source nodes, say $S = \{\sigma_1, \sigma_2, \dots, \sigma_s\}$ with $|S| = s$, and

On the Capacity Scalability of Line Networks with Buffer Size Constraints

Shenghao Yang, Jie Wang, Yanyan Dong and Yiheng Zhang

Abstract—The communication capacity of a network of line topology is studied, where only two adjacent nodes are connected by communication channels, and the intermediate network nodes have a buffer size constraint. Let L be the number of hops from the source node to the destination node. For general channels, we provide schemes to achieve $\Omega(1/\ln L)$ rates using a buffer of size $B_1 + B_2$ bits, where B_1 does not change with L and $B_2 = O(\ln \ln L)$. In particular, B_1 bits of the buffer are used to store the data generated from the communication messages, and the other B_2 bits of the buffer are used to store the status of counters with the maximum value $O(\ln L)$.

I. INTRODUCTION

The communication in a network from a source node to a destination node usually needs to go through multiple hops, each of which may introduce errors. In this paper, we are interested in the problem that when the intermediate nodes have a buffer size constraint, how fast the communication rate decreases with the number of hops. The buffer here refers to the storage space for the content used between different intermediate node processing steps, but not the space for storing the processing program itself, where the latter is assumed to be constant.

In particular, we consider a line network of L hops formed by a sequence of nodes, where discrete memoryless channels (DMCs) with the same transition matrix Q exist only between two adjacent nodes. We call the first node the source node and last node the destination node. Except for the source and destination nodes, all the other nodes, called intermediate nodes, have one incoming channel and one outgoing channel. Each intermediate node has a buffer of B bits to keep the content used between different intermediate processing steps. There are no other storage and computation constraints on the network nodes.

Existing works have studied various special cases of the problem. When the buffer size B is allowed to increase with the block length at the source node, the min-cut capacity can be achieved using hop-by-hop decoding and re-encoding [1]. When the zero-error capacity of the channel Q is nonzero, using a constant B can achieve the zero-error capacity for any value of L [2].

S. Yang, J. Wang, Y. Dong and Y. Zhang are with the School of Science and Engineering, The Chinese University of Hong Kong, Shenzhen, Shenzhen, China. S. Yang is also with the Shenzhen Key Laboratory of IoT Intelligent Systems and Wireless Network Technology, The Chinese University of Hong Kong, Shenzhen, Shenzhen, China. S. Yang and J. Wang are also with the Shenzhen Research Institute of Big Data, Shenzhen, China. This work was funded in part by the Shenzhen Science and Technology Innovation Committee (Grant JCY20170817150000267, ZDSYS20170725140921348) and NSFC (Grant 61471215).

Henceforth in this paper, we focus on channel Q with zero-zero-error capacity. Niesen, Fragouli and Tuninetti [2] showed that a class of codes with constant buffer size B can achieve rates $\Omega(e^{-cL})$, where c is a constant. They also showed that if the buffer size B is a log function of L , any rate below the capacity of channel Q can be achieved.

In [3], the problem was studied when the channel between two consecutive nodes is packet erasure, and it was shown that the capacity of the network with a constant buffer size B is preserved if perfect hop-by-hop feedbacks are allowed. The capacity of the network was then characterized using the stationary distribution of an underlying irreducible Markov chain, which however is computational tedious even for networks of small lengths and buffer sizes, and provides little analytical results that could help us to understand the capacity scalability.

In this paper, we study a general DMC Q and show that rate $O(1/\log L)$ can be achieved when $B = B_1 + B_2$, where B_1 does not change with L and $B_2 = O(\ln \ln L)$. In particular, B_1 bits of the buffer are used to store the data generated from the communication messages, and the other B_2 bits of the buffer are used to store the status of couple counters with the maximum value $O(\ln L)$. For most practical channel Q and network length L , $B_1 \gg B_2$, though B_2 tends infinity when $L \rightarrow \infty$.

The coding scheme used in this paper is called batched codes, where all the intermediate nodes are restricted to processing symbols belonging to the same batch, which is a set of coded symbols generated at the source node. This class of codes generalize the one used in [2] by decoupling the buffer size B and the number of channel uses of a batch (called batch block-length), both of which are the same in [2]. In our scheme, we use a batch block-length $O(\ln L)$, and a buffer size $O(\ln \ln L)$, where the latter is minimal when an intermediate node needs to synchronize different batches.

II. PROBLEM FORMULATION

Consider a line network of length L formed by a sequence of nodes $v_i, i = 0, 1, \dots, L$, where communication channels exist only between nodes v_{i-1} and v_i , denoted by (v_{i-1}, v_i) , for $i = 1, \dots, L$. We assume that all the channels are identical discrete memoryless channels (DMCs) with the transition matrix Q .

¹In this paper, we say that $f(n) = O(g(n))$ if there exists a real constant $c > 0$ and there exists an integer constant $n_0 \geq 1$ such that $f(n) \leq c \cdot g(n)$ for every integer $n \geq n_0$; $f(n) = O(g(n))$ if there exists a real constant $c > 0$ and there exists an integer constant $n_0 \geq 1$ such that $f(n) \leq c \cdot g(n)$ for every integer $n \geq n_0$; and $f(n) = \Theta(g(n))$ if both $f(n) = O(g(n))$ and $f(n) = O(g(n))$ are satisfied.

Upper Bound Scalability on Achievable Rates of Batched Codes for Line Networks

Shenghao Yang and Jie Wang

Abstract—The capacity of line networks with buffer size constraints is an open, but practically important problem. In this paper, the upper bound on the achievable rate of a class of codes, called batched codes, is studied for line networks where the channels have 0 zero-error capacity. Batched codes enable a range of buffer size constraints, and are general enough to include special coding schemes studied in the literature for line networks. Existing works have characterized the achievable rates of batched codes for several classes of parameter sets, but leave the cut-set bound as the best existing general upper bound. In this paper, we provide upper bounds on the achievable rates of batched codes as functions of line network length for these parameter sets. Our upper bounds in order of the network length match with the existing achievability results.

I. INTRODUCTION

The communication in a network from a source node to a destination node may go through multiple hops, each of which introduces errors. In this paper, we are interested in the problem that when the intermediate nodes have buffer size constraints, how the communication rate scales with the number of hops.

In particular, we consider a line network of L hops formed by a sequence of nodes, where discrete memoryless channels (DMCs) exist only between two adjacent nodes. We call the first node source node and the last node destination node. Except for the source and destination nodes, all the other nodes, called intermediate nodes, have one incoming channel and one outgoing channel. Each intermediate node has a buffer of B bits to keep the content used between different intermediate processing steps. There are no other storage and computation constraints on the network nodes.

For some cases of the problem, the answers are known. When the buffer size B is allowed to increase with the block length at the source node, the min-cut capacity can be achieved using hop-by-hop decoding and re-encoding [1]. When the zero-error capacity of each channel is nonzero, using a constant buffer size B can achieve the zero-error capacity for any value of L [2].

In this paper, we focus on the DMCs in the line network with finite input and output alphabets and 0 zero-error capacity. Note that for most common channel models, e.g.,

S. Yang and J. Wang are with the School of Science and Engineering, and the Shenzhen Key Laboratory of IoT Intelligent Systems and Wireless Network Technology, The Chinese University of Hong Kong, Shenzhen, Shenzhen, China. S. Yang and J. Wang are also with the Shenzhen Research Institute of Big Data, Shenzhen, China. This work was funded in part by the Shenzhen Science and Technology Innovation Committee (Grant JCY20170817150000267, ZDSYS20170725140921348, JCY2018080816206331) and NSFC (Grant 61471215).

binary symmetric channels and erasure channels, the zero-error capacities are zero. When all cascaded channels are identical, Niesen, Fragouli, and Tuninetti [2] showed that a class of codes with a constant buffer size B can achieve rates $\Omega(e^{-cL})$, where c is a constant. They also showed that if the buffer size B is of order $\ln L$, any rate below the capacity of the channel Q can be achieved. Recently, Yang et al. [3], [4] showed that the end-to-end throughput can be lower bounded by $\Omega(1/\ln L)$ using an intermediate node buffer size $O(\ln \ln L)$.

In contrast to these achievability results, min-cut is still the best upper bound. Characterizing a non-trivial, general upper bound for a line network with buffer size constraints could be difficult as hinted in [5]. We relax the difficulty of the problem by asking the scalability of the upper bound with the network length L for a class of codes, called batched codes.

Batched codes provide a general coding framework for line networks with buffer size constraints, and include the codes studied in the previous works [2]–[4] to show the achievability results as special cases. A batched code has an outer code and an inner code. The outer code encodes the information messages into batches, each of which is a sequence of coded symbols, while the inner code performs a general network coding for the symbols belonging to the same batch. The inner code, comprising of recoding at network nodes on each batch separately, should be designed for specific channels. Batched codes have been studied for designing efficient network coding for packet erasure channels (see, for example, [6], [7]), and practical designs have been provided [8], [9].

A batched code has three basic parameters: batch size M , buffer size B and inner block-length N . The upper bound scalability on the achievable rates of batched codes provides important guidance for us to design batched codes for large networks. For example, we want to know whether the exponential decay of the achievable rate with L is necessary for $B = O(1)$, and whether we can do better than $\Omega(1/\ln L)$ when $B = O(\ln \ln L)$. These questions are answered in this paper (see Table I). In particular, we show that when $N = O(1)$, which implies $M, B = O(1)$, the achievable rates must be exponential decay with L . When $N = O(1/\ln L)$ and $M = O(1)$, which implies $B = O(1/\ln L)$, the achievable rate is $O(1/\ln L)$. These upper bounds have the same order of L as the previous achievability results.


Our results are proved in a general setting of line networks where the DMC channels in the line network can be arbitrarily different except for a mild technical condition. The main technique of our converse is to separate the end-to-end transition

J. Wang and S. Yang is with the School of Science and Engineering, The Chinese University of Hong Kong, Shenzhen, Shenzhen, China. S. Yang is also with the Shenzhen Key Laboratory of IoT Intelligent Systems and Wireless Network Technology, The Chinese University of Hong Kong, Shenzhen, Shenzhen, China, and with the Shenzhen Research Institute of Big Data, Shenzhen, China. C. Li is with the Department of Computer Science, City University of Hong Kong, Hong Kong. This work was funded in part by the NSFC under Grant 61471215, and the Shenzhen Science and Technology Innovation Committee under Grant JCY20170817150000267, ZDSYS20170725140921348, JCY2018080816206331 and JCY20160229165220746.

本科的学习体验：机会多

➤ 开设一系列学术讲座，涉及领域非常广泛



运筹OR帷幄 
已认证的官方帐号

7 人赞同了该回答

@运筹OR帷幄编者按：

5月31日—6月2日，Mostly OM 2019 Workshop 于香港中文大学（深圳）召开。2019年是MostlyOM Workshop十周年，自2009年MostlyOM Workshop开幕，**每年都在清华大学召开，今年也是第一年在对外校举办。**2019年MostlyOM 延续传统，全员大会就运作管理及其相关领域热点话题做出报告。专题研讨会旨在为研究者们提供交互意见和深入合作的平台。此次MostlyOM Workshop 10周年大会由POMS联合主办，会议还包括多场平行报告会。

本科的学习体验：机会多

- 本科生有很多机会从事科研活动



第二个体验：同学强



左上：佐治亚理工学院
左下：卡耐基梅隆大学
右上：哥伦比亚大学
右下：斯坦福大学

第三个体验： 自主学习能力

- 高中的学习：比较功利，在老师的督促下学习
- 港中深的学习：比较自由，倡导自主学习
 - **学时短**，每学期十四周，每门课包含了三小时的正课和一小时的习题课；
 - 上课不会涉及课本的所有方面，**点到为止**
 - 更需要学生**自主学习，主动查漏补缺**