Expander Graph Based Overlapped Chunked Codes

Bin Tang†, Shenghao Yang‡, Yitong Yin†, Baoliu Ye†, and Sanglu Lu†
† National Key Laboratory for Novel Software Technology, Nanjing University, China
‡ ITCS, Institute for Interdisciplinary Information Sciences, Tsinghua University, China
Email: tb@dislab.nju.edu.cn, shyang@tsinghua.edu.cn, {yinyt, yebl, sanglu}@nju.edu.cn

***Abstract*—Chunked codes are a variation of random linear network codes with low computational complexities. In chunked codes, the packets in a file are grouped into small (non-overlapped or overlapped) chunks, and random linear encoding operations are performed within each chunk. Previous studies show that when the chunk size is lower bounded by some increasing function of the file length, chunked codes asymptotically achieve the min-cut capacity. However, in most real applications, the chunk size is required to be a small constant due to the computational constraints of network devices. In this case, it remains unknown which rates can be achieved by chunked codes. In this paper, we address the analysis and design of chunked codes with fixed constant chunk sizes. We first highlight the importance of precoding for chunked codes to achieve constant rates, and then present an analysis of non-overlapped chunked (NOC) codes with precoding. We further introduce a new class of chunked codes, called EOC codes, which are based on expander graphs to form overlapped chunks. Numerical and simulation results show that EOC codes achieve significantly higher rates than NOC codes, and also outperform other state-of-the-art overlapped chunked codes.**

I. INTRODUCTION

Random linear network coding (RLNC) [1], [2] has become a simple but powerful tool for data dissemination over communication networks. In RLNC, participating nodes generate coded packets by linear combinations of all the packets received so far with coefficients randomly chosen from a finite field and forward the coded packets. Due to its random nature, RLNC can be easily implemented in a distributed fashion. It is also capacity-achieving for networks with packet loss in a wide range of scenarios [3], [4].

A major issue in applying RLNC is its high computational complexities. Consider the transmission of a file with K packets. For encoding, RLNC requires $O(K)$ operations to generate a coded packet. And for decoding, it has to invert a $K \times K$ dense matrix and use the inverse to recover the whole file. Usually, the time for decoding is dominated by the file recovery process, which costs $O(K)$ operations for each original packet. When the file is very large, the cost of both encoding and decoding operations becomes prohibitive, making RLNC hardly to be implemented in real systems.

As a variation of RLNCs, chunked codes incur lower computational complexities by grouping the packets of a file into multiple small chunks (a.k.a. generations, classes, etc.), and performing random linear encoding operations within each chunk [5]. One issue in applying chunked codes is scheduling, *i.e.*, when a transmission chance is available, which chunk should be chosen for generating a coded packet. One scheme is *sequential scheduling*, in which the sender keeps on transmitting packets generated using a chunk, and when receiving positive feedback(s) from receiver(s), it starts the transmission of the next chunk. While some efficient feedback protocols for specific applications have been developed [11], [7], in general, such feedbacks incur a non-negligible delay and may consume network resources such as bandwidth, resulting in degraded system performance. An alternative approach without feedbacks is *random scheduling*, where the chunk for generating a coded packet is always randomly picked. Random scheduling is also known to be resilient to any channel erasure patterns.

In the context of random scheduling, Maymounkov *et al.* [8] showed that chunked codes with disjoint chunks, referred to as *non-overlapped chunked (NOC) codes*, can achieve the capacity if the chunk size is on or higher than the logarithm order of the file length. However, in most real applications, the chunk size is required to be a small constant due to the computational constraints of network devices. In this case, it remains unknown which rates (to be formally defined) can be achieved by NOC codes. On the other hand, two research groups have independently shown by simulations that better performance could be achieved for practical chunk sizes by allowing different chunks to share same packets [9], [10], referred to as *overlapped chunked codes*. Although several overlapped chunking schemes have been proposed in the literature [9], [10], [11], the analyzes of them are somewhat heuristic [10], [11], or assume variable chunk sizes [12], [12].

In this paper we address the analysis and design of chunked codes with constant chunk sizes under the strategy of random scheduling. By noting that the random scheduling strategy requires a large number of coded packets for decoding the last chunks, we prove that for any chunked code with a fixed constant chunk size, if applied directly, the rate vanishes inevitably as the file length goes to infinity. Thus, precode, which allows only a fraction of packets to be decoded, plays a vital role to achieve a positive rate. We then present a tight analysis for the NOC codes with precoding, which can help us choose appropriate precodes such that the rates are maximized. It is observed that the maximum rates achieved by NOC codes are far away from 1 (e.g., in a moderate setting with a chunk size of 32 and a finite field size of 16, the maximum rate of

‡Part of the work was done when S. Yang was with the Institute of Network Coding, The Chinese University of Hong Kong.

978-1-4673-2579-0/12/$31.00 ©2012 IEEE

an NOC code is only about 73%).

Towards the improvement of rates, we propose a novel expander graph based overlapped chunking (EOC) scheme, which uses only a small number of overlaps such that the number of chunks is increased slightly, and simultaneously guarantees that whenever any subset of chunks is decoded, it can provide substantial help for decoding the remaining chunks. We also establish a lower bound for the maximum rate of an EOC code with appropriate parameter and precoding, which reveals that EOC codes can achieve significantly higher rates than NOC codes (e.g., in the same setting as mentioned earlier, the maximum rate of an EOC code is at least 93%). Also, simulation results show that EOC outperforms other state-of-the-art overlapped chunking schemes.

II. Preliminaries

We formally describe the model of chunked codes. Consider the distribution of a file with K packets, $\mathcal{P}_1, \mathcal{P}_2, \dots, \mathcal{P}_K$, each of which is composed by a number of symbols from a finite field F_q. The packets are grouped into n chunks, $\mathcal{C}_1, \mathcal{C}_2, \dots, \mathcal{C}_n$, where each chunk is a set of M packets. Throughout the paper we assume that M is a fixed constant which is independent of K. We call a coding scheme non-overlapping if all chunks are disjoint, *i.e.*, $\mathcal{C}_i \cap \mathcal{C}_j = \emptyset$ for all $i \neq j$, and overlapping if otherwise.

Similar to [11], we assume that the file is distributed over a unicast link modeled by an erasure channel using a chunked code under the strategy of random scheduling. In each transmission, the source: (1) selects a chunk uniformly at random, say $\mathcal{C}_j$, from the n chunks; (2) randomly combines packets in $\mathcal{C}_j$ into a new packet $\mathcal{P}_{new} = \sum_i c_i \mathcal{P}_{j_i}$, where $\mathcal{P}_{j_i}$'s are packets in $\mathcal{C}_j$, and $\mathbf{c} = (c_1, c_2, \dots, c_M)$ is a random vector whose entries are picked uniformly and independently from F_q; (3) sends the newly generated packet $\mathcal{P}_{new}$ to the destination over the communication link. The transmission of the source is rateless, *i.e.*, the source does not stop transmitting until the whole file has been decoded by the destination.

Once the destination has collected M coded packets with linearly independent encoding vectors for some chunk, all packets in the chunk can be decoded by the Gauss elimination. The process is completed when all chunks have been decoded. Furthermore, if the chunks are allowed to be overlapped, some packets may appear in more than one chunks, thus a successfully decoded chunk may help decoding other chunks. This leads to an iterative decoding process: whenever a chunk is successfully decoded, the linear system is updated accordingly so that the decoded packets are substituted back helping decode other chunks.

Since each transmitted coded packet is erased randomly and independently, all the received packets are statistically the same. Thus, we can define N as the minimum number of coded packets to be collected such that the whole file can be reliably decoded, *i.e.*, the file fails to be recovered with probability at most inverse polynomial in K, and evaluate the efficiency of a chunked code by the ratio between K and N. Formally, we define the (normalized) rate $R(M)$ of a chunked code with chunk size M as

$$R(M) = \lim_{K \to \infty} \frac{K}{N}. \tag{1}$$

With some abuse of notation, we also refer to K/N as its rate. Since the rate of any chunked code is upper bounded by 1, our main objective is to design efficient chunked codes with rates close to 1.

III. Precodes

As mentioned earlier, for chunked codes, the random scheduling strategy will incur a long tail of the "coupon collector's curve", *i.e.*, a large number of packets have to be collected for decoding the last chunks, which reminds us not to apply chunked codes directly. Formally, we have the following result.

***Theorem** 3.1:* For any chunked code with a constant chunk size, if applied directly, then its rate decays at a rate of $\Omega(\frac{1}{\log K})$ as the file length K grows.

The proof of the above result is similar to the one in [14] for a result for LT codes, and is thus omitted to save the space.

***Example** 3.2:* Consider an NOC code with n chunks, each with size M. To decode the file completely, the destination has to collect M packets of each chunk.[1] It is shown in [15] that, when $K \to \infty$, $N = n \log n + (M-1) n \log \log n + O(n)$. Noting $n = \frac{K}{M}$, $\frac{K}{N} = \Theta(\frac{1}{\log K})$ vanishes.

The idea to achieve a positive constant rate is to apply precoding before using chunked codes for transmission: the source first encodes the original K packets into $K' > K$ *precoded packets* using some precoding scheme, and then transmits these precoded packets using a chunked code. In this case, only a fraction of precoded packets is required to be decoded at the destination so that the whole file can be recovered, thus avoiding the long tail, and achieving a non-vanishing rate.

We make the following assumption about precoding: for any $0 < \eta < 1$ and $\varepsilon > 0$, there exists some precode PC(η), which encodes K original packets into $K' = K/\eta$ precoded packets such that all K original packets can be recovered reliably from any $(1+\varepsilon)\eta K' = (1+\varepsilon)K$ precoded packets. The assumption holds for various codes, including Tornado codes [16], right-regular codes [17], etc.

IV. Analysis for Non-overlapped Chunked Codes

Let ζ_M^j be the probability that an $M \times j$ matrix with entries uniformly chosen from F_q at random has rank M. We have (see a derivation in [18])

$$\zeta_M^j = \begin{cases} \prod_{l=0}^{M-1} (1 - \frac{1}{q^{j-l}}) & j \geq M \\ 0 & j < M. \end{cases}$$

According to the random linear coding scheme, ζ_M^j is just the probability that a chunk is decodable when j coded packets of this chunk are collected.

[1] To make the illustration simple, here q is assumed to be sufficiently large, such that the encoding vectors of any collected packets of a chunk are linearly independent.

Let $p(x) = \sum_{j=M}^{\infty} \frac{e^{-x}x^j}{j!}\zeta_M^j$.

The following result establishes the relationship between the number of decodable original packets and the number of received coded packets when an NOC code without precoding is applied.

***Lemma** 4.1:* Consider the transmission of K packets using a precode-free NOC code with chunk size M. Let η be the fraction of decodable original packets when receiving N coded packets, and $\mu = \frac{NM}{K}$ be the average number of received packets of each chunk. Then for any $0 < \varepsilon < 1$, the event

$$(1-\varepsilon)p(\mu) < \eta < (1+\varepsilon)p(\mu) \tag{2}$$

fails to hold with probability exponentially small in K.

Proof: (Sketch) We employ the technique of Poisson approximation [19]. First, approximate the number of received packets of each chunk by i.i.d. Poisson distribution with mean value μ. Then it is easy to see that $p(\mu)$ gives the probability that a chunk can be decoded, thus the expected fraction of decodable chunks. By Chernoff bounds, the fraction of decodable chunks in the Poisson approximation is tightly concentrated around $p(\mu)$. By applying the same argument as in [19], this implies the tight concentration in the real case, where the number of received packets of each chunk follows a Binomial distribution with mean value μ. Therefore, η is sharply concentrated around $p(\mu)$. ■

According to the above result, one can choose an appropriate precode for NOC codes such that the number of coded packets required for decoding the whole file is minimized, and thus the rate is maximized.

***Theorem** 4.2:* Let $\mu^* = \arg\max_{\mu>0} \frac{p(\mu)}{\mu}$ and $\eta^* = p(\mu^*)$. Then by precoding with PC(η^*), an NOC code can achieve its maximum rate $R_{max}(M)$, which satisfies

$$\frac{p(\mu^*)}{\mu^*}M - \varepsilon \le R_{max}(M) \le \frac{p(\mu^*)}{\mu^*}M + \varepsilon. \tag{3}$$

for any arbitrarily small constant $\varepsilon > 0$.

As the theorem implies, NOC codes with precoding can achieve positive rates. However, such rates are a bit low. For example, when $M = 32$ and $q = 16$, the rate is only about 73%. In the next section, we propose an expander graph based overlapped chunking scheme, which can improve the rates of chunked codes significantly.

V. Expander Graph Based Overlapped Chunked Codes

A. Scheme Description

An expander graph based overlapped chunked (EOC) code with K original packets, chunk size M and a parameter d can be constructed as followings. Here d is an integer with $3 \le d \le M$. Let $n = \frac{K}{M-\frac{d}{2}}$. To keep the analysis simple, we assume that n is an integer and dn is even.

1) Generate a random d-regular graph[2] $G = (V, E)$ with n nodes, where $V = \{1, 2, \ldots, n\}$. A random regular graph is an expander graph with high probability.
2) First associate each edge $e \in E$ with a distinct original packet, denoted by $\mathcal{P}_e$, and then assign the rest original packets evenly over the nodes. Denote the set of packets assigned to node v as $\mathcal{PS}_v$.
3) Chunk $\mathcal{C}_j$, $1 \le j \le n$, consists of packets in $\mathcal{PS}_j$ and packets associated with edges incident to j, *i.e.*,

$$\mathcal{C}_j = \mathcal{PS}_j \cup \{\mathcal{P}_e : e \text{ is incident to node } j\}.$$

Due to the one-to-one correspondence between chunks and nodes, we use the names interchangeably in this section.

From the above description, it is straightforward to see that every chunk overlaps with d chunks, each on a distinct packet. Therefore, when decoding, a chunk can enjoy different help from different decoded neighboring chunks.

B. Analysis

To state the main theorem of this section, we need to introduce some notations. For any $x > 0$, we define a function $f_x(y)$ as

$$f_x(y) = \sum_{w=0}^{d-1} \binom{d-1}{w} y^w (1-y)^{d-1-w} \sum_{j=0}^{\infty} \frac{e^{-x}x^j}{j!}\zeta_{M-w}^j. \tag{4}$$

With this function and its functional powers, we introduce a sequence as $f_x(0), f_x^2(0), \ldots, f_x^l(0)$, where l is an integer, and $f_x^{i+1}(0) = f_x(f_x^i(0))$ for all $i \ge 0$. It can be verified that the sequence is strictly increasing and is upper bounded by 1. Therefore, the sequence converges to a limit, denoted by $f_x^*(0)$, when l goes to infinity, *i.e.*, $f_x^*(0) = \lim_{l\to\infty} f_x^l(0)$.

We further define a function $g(d, x)$ as

$$g(d,x) = \frac{(M - \sqrt{d-1})f_x^*(0) - (\frac{d}{2} - \sqrt{d-1})(f_x^*(0))^2}{x}.$$

The following theorem provides a lower bound on the maximum rate of an EOC code with precoding.

***Theorem** 5.1:* Let $(d^*, \mu^*) = \arg\max_{3\le d\le M, \mu>0} g(d, \mu)$ and $\eta^* = \frac{g(d^*,\mu^*)\mu^*}{M-d^*/2}$. Then by precoding with PC(η^*), an EOC code with degree d^* can achieve a rate of at least $g(d^*, \mu^*) - \varepsilon$, where $\varepsilon > 0$ is an arbitrarily small constant.

According to Theorem 5.1, it can be shown that EOC codes achieve higher rates than NOC codes for different chunk sizes. For example, when $M = 32$ and $q = 16$, an EOC code can achieve a rate of at least 93%, while the rate of an optimal NOC code is only about 73%.

Now we proceed to prove Theorem 5.1.

***Definition** 5.2:* The *l-neighborhood* of a node v, denoted by $G_l(v)$, is defined as the subgraph of G induced by v and all its neighbors within distance l. We say that v is *l-decodable* if $G_l(v)$ is a tree and v can be decoded when the decoding process is restricted within the tree.

In the rest of the section, we set $l = \frac{1}{3}\log_{d-1} n$.

We first derive the probability that a node v is l-decodable in the case that the number of received packets of each chunk follows an i.i.d. Poisson distribution.

[2]A random d-regular graph can be generated using different models. In this paper, we adopt the uniform model, *i.e.*, the d-regular graph is uniformly chosen from all d-regular graphs with node set $\{1, 2, \ldots, n\}$. However, one can obtain the same result for many other models (e.g., permutation model, perfect matching model [20]) by conducting a similar analysis.

Lemma 5.3: Assume that the number of received packets of each chunk follows an i.i.d. Poisson distribution with mean value μ. Then for any node v and $\varepsilon > 0$, the probability that v is l-decodable is at least $(1-\varepsilon)f^*_\mu(0)$ for all sufficiently large n.

Proof: For a random d-regular graph G with n nodes, let $\tau(n)$ be the number of nodes whose l-neighborhood is a tree. Due to [21], for any $\varepsilon > 0$, we have that

$$\Pr(\tau(n) < (1-\varepsilon)n) \leq \frac{1}{\varepsilon^2} O\Big(\frac{(d-1)^{4l}}{n^2}\Big) = \frac{1}{\varepsilon^2} O(n^{-\frac{2}{3}}).$$

So, for sufficiently large n, it can be shown that

$$\Pr(G_l(v) \text{ is a tree}) = \frac{\mathrm{E}[\tau(n)]}{n} \geq 1 - \varepsilon. \tag{5}$$

Now assume that $G_l(v)$ is a tree rooted at v. After deleting a subtree rooted at a child of v, we obtain a $(d-1)$-ary tree. We abuse the notation and denote the resulting tree also by $G_l(v)$. Number the level of root by l and level of leaves by 0. Define $p_i(\mu)$ as the probability that a node at level i is decodable when the decoding process is restricted within the subtree of $G_l(v)$ rooted at the node. In the following, we calculate $p_i(\mu)$ in a bottom-up fashion.

For any leaf, it cannot get any help from other chunk. So,

$$p_0(\mu) = \sum_{j=0}^{\infty} \frac{e^{-\mu}\mu^j}{j!} \zeta^j_M = f_\mu(0).$$

For an internal node at level $i > 0$, assume that its w children have been decoded. Thus, each received packet of this node can be seen as a random linear combination of all packets of this node except for the w overlapping packets. Therefore, when j packets are collected for this node, the node can be decoded with probability ζ^j_{M-w}. By the definition of function $f_\mu(\cdot)$, it can be verified that

$$p_i(\mu) = f_\mu(p_{i-1}(\mu)) = f^i_\mu(0).$$

Noting that $f^*_\mu(0) = \lim_{l\to\infty} f^l_\mu(0)$, we have

$$p_l(\mu) > (1-\varepsilon) f^*_\mu(0) \tag{6}$$

for any $\varepsilon > 0$ when n is sufficiently large.

Combining (5) and (6), the lemma follows. ■

The following lemma gives the number of l-decodable chunks when receiving a number of coded packets.

Lemma 5.4: Let μ be the average number of received coded packets of each chunk in the actual case. Then for any $\varepsilon > 0$, the probability that the number of l-decodable chunks is less than $(1-\varepsilon)f^*_\mu(0)n$ is exponentially small in n.

Proof: (Sketch) The proof is by combining the Poisson approximation with Martingale concentration, where the later benefits from the locality of l-decoding process. In the Poisson case, consider a standard vertex exposure martingale on the random regular graph G. Note that the exposure on one node can affect the number of l-decodable nodes by at most the number of nodes in its l-neighborhood, which is on the order of $n^{\frac{1}{3}}$. Therefore, we can apply the Azuma-Hoeffding Inequality [19] to show that in the Poisson case, the number of l-decodable chunks is tightly concentrated around its expectation, which is at least $(1-\varepsilon)f^*_\mu(0)n$ according to Lemma 5.3. This property holds in the actual Binomial case, which can be shown by applying the argument in [19] to remove the Poisson assumption. ■

Since each packet can appear in at most two chunks, the above result can provide a lower bound for the number of decodable packets. However, such bound is a bit loose. In the following, we provide a much tighter analysis based on some expander arguments.

Consider a d-regular graph G with n nodes. Let $A_{n\times n}$ be its adjacency matrix, where $A(u,v)$ is the number of edges in G between nodes u and v. Clearly, A has n real eigenvalues $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n$. Also, $\lambda_1 = d$, and for all $1 \leq i \leq n$, $|\lambda_i| \leq d$. Denote that $\lambda_{max} = \max\{|\lambda_2|, |\lambda_n|\}$. The following theorem lists some known results about expansion and spectrum of random regular graphs.

Lemma 5.5: Let G be a random d-regular graph on n nodes.

- ([22]) For any subset S of δn nodes, the number of edges between nodes in S is at most $\frac{dn}{2}\Big(\delta^2 + \frac{\lambda_{max}}{d}\delta(1-\delta)\Big)$.
- ([23]) $\lambda_{max} \geq 2\sqrt{d-1} \cdot (1 - O(1/\log^2 n))$.
- ([24]) For any $\varepsilon > 0$, there exists some constant $c > 0$, such that $\Pr[\lambda_{max} \leq 2\sqrt{d-1} + \varepsilon] = 1 - O(n^{-c})$.

Now we are ready to prove Theorem 5.1.

Proof of Theorem 5.1: Let μ be the average number of received packets of each chunk, and η be the fraction of decodable chunks. According to Lemma 5.4, $\eta \geq (1-\varepsilon)f^*_\mu(0)$ for any $\varepsilon > 0$ with high probability. So, according to Lemma 5.5, the number of l-decodable packets is at least

$$\begin{aligned}
&\eta n M - \frac{dn}{2}\Big(\eta^2 + \frac{\lambda_{max}}{d}\eta(1-\eta)\Big)\\
\geq&((M - \sqrt{d-1})\eta - (\frac{d}{2} - \sqrt{d-1})\eta^2)n\\
\geq&((M - \sqrt{d-1})f^*_\mu(0) - (\frac{d}{2} - \sqrt{d-1})(f^*_\mu(0))^2 - \varepsilon)n\\
=&(g(d,\mu) - \varepsilon)n.
\end{aligned}$$

This provides a lower bound for the number of decodable packets, which implies that the fraction of decodable packets is at least $\frac{(g(d,\mu)-\varepsilon)n}{K} \geq \frac{g(d,\mu)}{M-d/2} - \varepsilon$. Now it is straightforward to show the theorem. ■

VI. Performance Evaluation

We evaluate the performance of our EOC scheme, and compare it with the NOC scheme, the head-to-toe overlapped chunking (H2T) scheme [9], and the random annex coding (RAC) scheme [6].

In Fig. 1, we evaluate numerically the achievable rates of both NOC codes and EOC codes provided in Theorem 4.2 and Theorem 5.1, respectively. We see that EOC codes achieve significantly higher rates than NOC codes with the same parameters. In particular, when the chunk size is 128, and the size of finite field is 16, an EOC code can achieve a rate of at least 0.96.

We also conduct simulations to compare EOC with NOC, H2T and RAC. Without precode, the relationships between the number of received packets and the fraction of decodable packets for all mentioned schemes are plotted in Fig. 2. In all simulations, the chunk size is 32 and the finite field size q is set to be infinity to see the extreme performance. (The results in the figure can be well approximated by using sufficiently large finite fields, e.g., $q = 2^8$.) Since it is not easy to get the optimal chunk numbers in EOC, H2T and RAC analytically, for each scheme we perform the simulation for all the valid chunk numbers and choose the one such that the number of received coded packets for decoding the whole file is minimized. The results of EOC* are obtained by using EOC with degree d optimized for using precode (see Theorem 5.1).

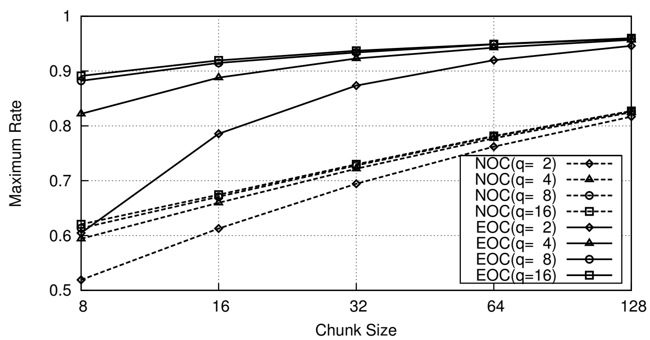

Fig. 1. Comparison of rates achieved by NOC and EOC codes with precoding.

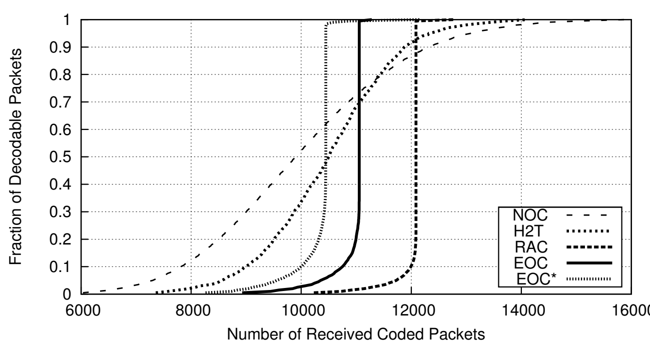

Fig. 2. Comparison between NOC, H2T, RAC, and EOC in terms of the number of received coded packets for decoding a file with 10000 packets. For each scheme, the values in the figure are the average of 100 transmissions.

We observe that both RAC and EOC show an avalanche of decoding after certain number of packets have been received. EOC* is the most efficient scheme for recovering a large fraction (e.g., 95%) of the original packets, but it incurs a longer tail than EOC for decoding the whole file.

VII. Conclusion

In this paper we studied the performance of chunked codes with constant chunk sizes. We highlighted the importance of precoding, and presented a tight analysis for NOC codes. We proposed and analyzed a novel expander graph based overlapped chunking scheme EOC, which outperforms NOC and all state-of-the-art overlapped chunking schemes. In the future, we would like to analyze the performance of EOC codes of finite lengths and in more generalized network models.

Acknowledgment

The authors would like to thank Raymond W. Yeung for many useful discussions. This work is partially supported by the National Natural Science Foundation of China under Grant No. 61170069, 61073028, 61021062 and 60903025; the National Basic Research Program of China under Grant No. 2009CB320705, 2011CBA00300 and 2011CBA00302; the Key Technology Research and Development Program of Jiangsu Province under Grant No. BE2010179; the University Grants Committee of the Hong Kong Special Administrative Region, China under Project No. AoE/E-02/08.

References

[1] T. Ho, R. Koetter, M. Médard, D. R. Karger, and M. Effros, "The benefits of coding over routing in a randomized setting," in *Proc. ISIT*, 2003.

[2] T. Ho, M. Médard, R. Koetter, D. R. Karger, M. Effros, J. Shi, and B. Leong, "A random linear network coding approach to multicast," *IEEE Trans. Inform. Theory*, 52(10), pp. 4413–4430, 2006.

[3] Y. Wu, "A trellis connectivity analysis of random linear network coding with buffering," in *Proc. ISIT*, 2006.

[4] D. S. Lun, M. Médard, R. Koetter, and M. Effros, "On coding for reliable communication over packet networks," *Physical Communication*, 1(1), pp. 3–20, 2008.

[5] P. A. Chou, Y. Wu, and K. Jain, "Practical network coding," in *Proc. Allerton Conf. Comm., Control, and Computing*, 2003.

[6] Y. Lin, B. Li, and B. Liang, "CodeOR: opportunistic routing in wireless mesh networks with segmented network coding," in *Proc. ICNP*, 2008.

[7] D. Koutsonikolas, C.-C. Wang, and Y. C. Hu, "Efficient network-coding-based opportunistic routing through cumulative coded acknowledgments, " *IEEE/ACM Trans. Netw.*, 19(5), pp. 1368–1381, 2011.

[8] P. Maymounkov, N.J.A. Harvey, and D.S. Lun, "Methods for efficient network coding," in *Proc. Allerton Conf. Comm., Control, and Computing*, 2006.

[9] A. Heidarzadeh and A.H. Banihashemi, "Overlapped chunked network coding," in *Proc. ITW*, 2010.

[10] D. Silva, W. Zeng, and F.R. Kschischang, "Sparse network coding with overlapping classes, "in *Proc. NetCod*, 2009.

[11] Y. Li, E. Soljanin, and P. Spasojević, "Effects of the generation size and overlap on throughput and complexity in randomized linear network coding, " *IEEE Trans. Inform. Theory*, 57(2), pp. 1111–1123, 2011.

[12] A. Heidarzadeh and A.H. Banihashemi, "Analysis of overlapped chunked codes with small chunks over line networks," in *Proc. ISIT*, 2011.

[13] A. Heidarzadeh and A.H. Banihashemi, "Network codes with overlapping chunks over line networks: a case for linear-time codes," 2011. [Online]. Available: http://arxiv.org/abs/1105.5736

[14] A. Shokrollahi, "Raptor codes," *IEEE Trans. Inform. Theory*, 52(6), pp. 2551–2567, 2006.

[15] L. Flatto, "Limit theorems for some random variables associated with urn models," *Ann. Probabil.*, 10(4), pp. 927–934, 1982.

[16] M. Luby, M. Mitzenmacher, A. Shokrollahi, and D. Spielman, "Efficient erasure correcting codes," *IEEE Trans. Inform. Theory*, 47(2), pp. 569–584, 2001.

[17] P. Oswald, and A.Shokrollahi, "Capacity-achieving sequences for the erasure channel," *IEEE Trans. Inform. Theory*, 48(12), pp. 3017–3028, 2002.

[18] S. Yang, S.-W. Ho, J. Meng, E. Yang, and R. W. Yeung, "On linear operator channels over finite fields," 2009. [Online]. Available: http://arxiv.org/abs/1002.2293

[19] M. Mitzenmacher, and E. Upfal, "Probablity and computing: randomized algorithms and probabilistic analysis," Cambridge University Press, 2005.

[20] N. C. Wormald, "Models of random regular graphs," *Surverys in Combinatorics*, pp. 239-298, 1999.

[21] I. Dumitriu, and S. Pal, "Sparse regular random graphs: spectral density and eigenvalues," Preprint, 2011.

[22] N. Alon, and F. R. K. Chung, "Explicit construction of linear sized tolerant networks," *Discrete Mathemat.*, 72, pp. 15–19, 1988.

[23] A. Nilli, "On the second eigenvalue of a graph," *Discrete Mathemat.*, 91(2), pp. 207–210, 1991.

[24] J. Friedman, "A proof of Alon's second eigenvalue conjecture," in *Proc. STOC*, 2003.