SCIENCE CHINA
Information Sciences

• **Supplementary File** •

Explicit Construction of Minimum Bandwidth Rack-Aware Regenerating Codes

Liyang ZHOU[1,2] & Zhifang ZHANG[1,2*]

[1]*Key Laboratory of Mathematics Mechanization, Academy of Mathematics and Systems Science, Chinese Academy of Sciences, Beijing* 100190, *China* ;
[2]*School of Mathematical Sciences, University of Chinese Academy of Sciences, Beijing* 100049, *China*

Appendix A Proof of Theorem 1

Theorem 1. The matrix M (i.e., the data file) can be recovered from any k columns of C.

Proof. We equivalently prove the $\bar{d}$ polynomials defined in Step 2 can be recovered from any k columns of C because coefficients of these polynomials contain exactly all entries of M.

First, from the construction of M in **Step 1.** one can see the last $\bar{d}-\bar{k}$ polynomials, i.e., $f_i(x)$ for $i \in [\bar{k}, \bar{d}-1]$, have degree at most $k-1$. Thus from any k columns (actually, just the bottom $\bar{d}-\bar{k}$ rows of these columns) of C one can obtain evaluations of these $\bar{d}-\bar{k}$ polynomials at k distinct points, and then recover the polynomials by Lagrange interpolation. After that, by the symmetric structure of M_1, coefficients of the terms of degree greater than $k-1$ of the first $\bar{k}$ polynomials, i.e., $f_i(x)$ for $i \in [0, \bar{k}-1]$, are simultaneously obtained from the last $\bar{d}-\bar{k}$ polynomials. Thus recovery of the first $\bar{k}$ polynomials reduces to interpolating $\bar{k}$ polynomials of degree at most $k-1$. Therefore, one can recover the remaining k coefficients of each of the first $\bar{k}$ polynomials from the k columns of C.

Appendix B Proof of Theorem 2

Denote $u_0 = k - \bar{k}u$, then $0 \leqslant u_0 < u$. For $e \in [0, \bar{n}-1]$ and $i \in [0, \bar{d}-1]$, define a polynomial $h_i^{(e)}(x) = \sum_{j=0}^{u-1} h_{i,j}^{(e)} x^j$ where

$$h_{i,j}^{(e)} = \begin{cases} \sum_{t=0}^{\bar{k}} m_{i,tu+j} \cdot \xi^{etu} & \text{if } 0 \leqslant j < u_0 \\ \sum_{t=0}^{\bar{k}-1} m_{i,tu+j} \cdot \xi^{etu} & \text{if } u_0 \leqslant j < u-1 \\ \sum_{t=0}^{\bar{d}-1} m_{i,tu+u-1} \cdot \xi^{etu} & \text{if } j = u-1 \end{cases} . \tag{B1}$$

Lemma 1. For all $e \in [0, \bar{n}-1]$ and $i \in [0, \bar{d}-1]$, it has $f_i(\lambda_{(e,g)}) = h_i^{(e)}(\lambda_{(e,g)})$ for all $g \in [0, u-1]$.

Proof. From **Step 2.** it has $f_i(\lambda_{(e,g)}) = \sum_{j \in J} m_{i,j} \lambda_{(e,g)}^j$. Next we rearrange the terms of $f_i(\lambda_{(e,g)})$ according to the values of j mod u for all $j \in J$. Specifically, denote $j = tu + \nu$, where $0 \leqslant \nu < u$. Then

$$f_i(\lambda_{(e,g)}) = \sum_{\nu=0}^{u_0-1} \sum_{t=0}^{\bar{k}} m_{i,tu+\nu} \lambda_{(e,g)}^{tu+\nu} + \sum_{\nu=u_0}^{u-2} \sum_{t=0}^{\bar{k}-1} m_{i,tu+\nu} \lambda_{(e,g)}^{tu+\nu} + \sum_{t=0}^{\bar{d}-1} m_{i,tu+u-1} \lambda_{(e,g)}^{tu+u-1}.$$

Because $\lambda_{(e,g)} = \xi^e \eta^g$ and η has multiplicative order u, it follows $\lambda_{(e,g)}^{tu} = \xi^{etu}$. By the definition in (B1), one can easily verify $f_i(\lambda_{(e,g)}) = \sum_{\nu=0}^{u-1} h_{i,\nu}^{(e)} \lambda_{(e,g)}^{\nu} = h_i^{(e)}(\lambda_{(e,g)})$.

Lemma 2. Consider the leading coefficients of the polynomials $h_i^{(e)}(x)$'s defined in (B1). For $e \in [0, \bar{n}-1]$ denote

$$\boldsymbol{h}_e = (h_{0,u-1}^{(e)}, h_{1,u-1}^{(e)}, ..., h_{\bar{d}-1,u-1}^{(e)})^\tau \in F^{\bar{d}} .$$

Then $(\boldsymbol{h}_0, \boldsymbol{h}_1, ..., \boldsymbol{h}_{\bar{n}-1}) = M_1 \Phi$ where M_1 is the symmetric matrix defined in **Step 1.** and

$$\Phi = \begin{pmatrix} 1 & 1 & \cdots & 1 \\ 1 & \xi^u & \cdots & \xi^{(\bar{n}-1)u} \\ \vdots & \vdots & \vdots & \vdots \\ 1 & (\xi^u)^{\bar{d}-1} & \cdots & (\xi^{(\bar{n}-1)u})^{\bar{d}-1} \end{pmatrix} . \tag{B2}$$

Thus $(\boldsymbol{h}_0, \boldsymbol{h}_1, ..., \boldsymbol{h}_{\bar{n}-1})$ is a codeword of an $(\bar{n}, \bar{k}, \bar{d})$ MBR code.

Proof. The proof is straightforward from the product-matrix construction of MBR codes in [1] and the expression in (B1).

* Corresponding author (email: zfz@amss.ac.cn)

Theorem 2. Any single node failure (i.e., any column of C) can be recovered by downloading $\beta = 1$ symbol from each of $\bar{d}$ helper racks in addition to the transmission within the rack containing the failed node.

Proof. Suppose the node (e^*, g^*) fails and the $\bar{d}$ helper racks are $e_1, ..., e_{\bar{d}} \in [0, \bar{n}-1] - \{e^*\}$. It suffices to prove the column indexed by (e^*, g^*) in C can be computed from the $u-1$ columns indexed by $\{(e^*, g) : 0 \leqslant g \leqslant u-1, g \neq g^*\}$ and $\bar{d}$ symbols each of which is a linear combination of the entries of the punctured code C_{e_i} for $i \in [d]$. Actually, from the MBR code proved in Lemma 2, $\boldsymbol{h}_{e^*}$ can be recovered from $\bar{d}$ symbols $\boldsymbol{\lambda}_{e^*}^\tau \boldsymbol{h}_{e_i}, 1 \leqslant i \leqslant \bar{d}$, where $\boldsymbol{\lambda}_{e^*}^\tau = (1, \xi^{e^*u}, (\xi^{e^*u})^2, ..., (\xi^{e^*u})^{\bar{d}-1})$. Then by Lemma 1 and Lagrange interpolation, $\boldsymbol{h}_{e_i}$ can be expressed as a linear combination of the u columns of C_{e_i} for $1 \leqslant i \leqslant \bar{d}$. Therefore, $\boldsymbol{h}_{e^*}$ can be computed from $\bar{d}$ symbols each of which is a linear combination of the entries of the punctured code C_{e_i} for $i \in [d]$. Again by Lemma 1 and Lagrange interpolation, the erased column $C_{(e^*,g^*)}$ is a linear combination of $\boldsymbol{h}_{e^*}$ and the remaining $u-1$ columns of C_{e^*}, thus the theorem follows.

Appendix C The systematic form of our MBRR code

Without loss of generality, let the first k nodes be the systematic nodes. Suppose $k = \bar{k}u + u_0$ where $0 \leqslant u_0 < u$, so the systematic nodes are all the nodes from rack 0 to rack $\bar{k}-1$ plus u_0 nodes in rack $\bar{k}$. Denote the B data symbols as $s_1, s_2, ..., s_B$. Next we are to define the MBRR encoding map that maps $(s_1, ..., s_B)$ to a $\bar{d} \times n$ code matrix C such that the first k columns of C contain $s_1, ..., s_B$. For simplicity, let $C_{[k]}$ denote the code matrix C restricted to the first k columns. Note that $B = k\bar{d} - \frac{\bar{k}(\bar{k}-1)}{2}$. It means $C_{[k]}$ contain $\frac{\bar{k}(\bar{k}-1)}{2}$ redundant symbols besides the B data symbols. The idea is to determine the redundant symbols from the B data symbols first and then recover the message matrix $\tilde{M}$ by Theorem 1 such that $\tilde{M}\Lambda = C$. Thus the systematic encoding map is a composition of $(s_1, ..., s_B) \to \tilde{M}$ and $\tilde{M}\Lambda$. Since $\tilde{M}$ has the same structure as displayed in Fig. 1, the resulting code is still an MBRR code. The details are given below.

First we place the B data symbols properly into $C_{[k]}$ except $\frac{\bar{k}(\bar{k}-1)}{2}$ entries which are for the redundant symbols. Specifically, label the columns of C by $(e, g) \in [0, \bar{n}-1] \times [0, u-1]$ and rows by $i \in [0, \bar{d}-1]$, then the column indexed by $(e, u-1)$ for $e \in [0, \bar{k}-2]$ has redundant symbols in its ith row as $i \in [e+1, \bar{k}-1]$. The remaining positions are filled up with the data symbols in order. We illustrate the placement in Fig. C1.

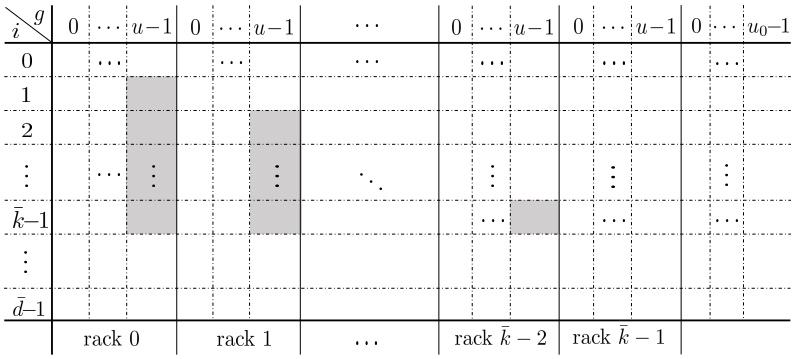

Figure C1 An illustration of $C_{[k]}$. The shadowed positions are redundant symbols and the remaining positions are filled up with B data symbols in order.

Next we show the first k columns of C excluding the undetermined redundant symbols are sufficient to recover a message matrix $\tilde{M}$ such that $\tilde{M}$ has the same structure as displayed in Fig. 1 and $\tilde{M}\Lambda = C$. The key step is to recover the symmetric matrix $\tilde{M}_1$. By Lemma 1 we know the code symbols in each row within each rack actually coincides with a local polynomial of degree at most $u-1$. Here we denote the local polynomial by $\tilde{h}_i^{(e)}(x)$. From Fig. C1 one can see that for $e \in [0, \bar{k}-1]$, rack e has no redundant symbols in its ith row for $i \in [0, e] \cup [\bar{k}, \bar{d}-1]$, namely, these rows are already known from the data symbols. As a result, one can interpolate the local polynomials $\tilde{h}_i^{(e)}(x)$ and obtain the leading coefficients $\tilde{h}_{i,u-1}^{(e)}$. These recovered $\tilde{h}_{i,u-1}^{(e)}$'s are listed in the right side of (C2). Moreover, suppose

$$\tilde{M}_1 = \begin{pmatrix} \tilde{S} & \tilde{T} \\ \tilde{T}^\tau & 0 \end{pmatrix}, \tag{C1}$$

where $\tilde{S}$ is a $\bar{k} \times \bar{k}$ symmetric matrix and $\tilde{T}$ is a $\bar{k} \times (\bar{d} - \bar{k})$ matrix. Then by Lemma 2 it has

$$\begin{pmatrix} \tilde{S} & \tilde{T} \\ \tilde{T}^\tau & 0 \end{pmatrix} \begin{pmatrix} 1 & 1 & \cdots & 1 \\ 1 & \xi^u & \cdots & \xi^{(\bar{k}-1)u} \\ \vdots & \vdots & \vdots & \vdots \\ 1 & (\xi^u)^{\bar{d}-1} & \cdots & (\xi^{(\bar{k}-1)u})^{\bar{d}-1} \end{pmatrix} = \begin{pmatrix} \tilde{h}_{0,u-1}^{(0)} & \tilde{h}_{0,u-1}^{(1)} & \cdots & \tilde{h}_{0,u-1}^{(\bar{k}-2)} & \tilde{h}_{0,u-1}^{(\bar{k}-1)} \\ * & \tilde{h}_{1,u-1}^{(1)} & \cdots & \tilde{h}_{1,u-1}^{(\bar{k}-2)} & \tilde{h}_{1,u-1}^{(\bar{k}-1)} \\ & & & \ddots & \vdots \\ * & * & \cdots & & \\ * & * & \cdots & * & \tilde{h}_{\bar{k}-1,u-1}^{(\bar{k}-1)} \\ \tilde{h}_{\bar{k},u-1}^{(0)} & \tilde{h}_{\bar{k},u-1}^{(1)} & \cdots & \tilde{h}_{\bar{k},u-1}^{(\bar{k}-2)} & \tilde{h}_{\bar{k},u-1}^{(\bar{k}-1)} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \tilde{h}_{\bar{d}-1,u-1}^{(0)} & \tilde{h}_{\bar{d}-1,u-1}^{(1)} & \cdots & \tilde{h}_{\bar{d}-1,u-1}^{(\bar{k}-2)} & \tilde{h}_{\bar{d}-1,u-1}^{(\bar{k}-1)} \end{pmatrix}, \tag{C2}$$

where on the right side of (C2) only the leading coefficients that can be derived from the data symbols by now are written out and entries in the $*$ positions are viewed as unknowns. However, it is enough to recover $\tilde{T}$ and $\tilde{S}$ from the currently known leading

coefficients. First, from the last $\bar{d}-\bar{k}$ rows in (C2) one can recover $\tilde{T}$ by multiplying the inverse of a $\bar{k}\times\bar{k}$ Vandermonde matrix. Then substituting the recovered entries of $\tilde{T}$ into the first row in (C2), one can obtain a linear system of equations of the first row entries of $\tilde{S}$. The coefficient matrix is again a $\bar{k}\times\bar{k}$ Vandermonde matrix. Thus one can recover the first row of $\tilde{S}$. Then go to the second row of (C2). Since $\tilde{S}$ is symmetric and its first row has been recovered, there are only $\bar{k}-1$ unknowns in the second row of $\tilde{S}$. Accordingly, the known entries in the second row of the right side of (C2) are enough to recover these unknowns. Continue this process and one can finally recover $\tilde{S}$ row by row. Thus we have proved the following theorem.

Theorem 3. The matrix $\tilde{M}_1$ can be uniquely determined by $\{\tilde{h}^{(e)}_{i,u-1} \mid e\in[0,\bar{k}-1], i\in[0,e]\cup[\bar{k},\bar{d}-1]\}$. Furthermore, each entry of $\tilde{M}_1$ can be expressed as a linear combination of the B data symbols.

After recovery of $\tilde{M}_1$, we can fill up the matrix on the right side of (C2). Thus for rack $e\in[0,\bar{k}-1]$ and for each row $i\in[0,\bar{d}-1]$, we have obtained the leading coefficients of the local polynomials $h^{(e)}_i(x)$. Since all data symbols already cover evaluations of each of these polynomials at $u-1$ points, one can easily derive the values at the uth points provided the leading coefficients are known. Thus all entries of $C_{[k]}$ have been recovered. Then by the data reconstruction property proved in Theorem 1, one can derive from $C_{[k]}$ the desired message matrix $\tilde{M}$. Through the process, we know each entry of $\tilde{M}$ can be expressed as a linear combination of the B data symbols which defines a preprocess before the product-matrix encoding map and finally leads to a systematic MBRR code.

We give an example to illustrate the transformation to systematic MBRR codes.

Example 1. Suppose $n=12, k=7, u=3, \bar{d}=3$ and $\beta=1$. Then the MBRR code has $\alpha=3, B=20$. Denote the B data symbols as $s_1,...,s_{20}$. We need to determine a 3×8 message matrix $\tilde{M}$ each entry of which is a linear combination of the B data symbols and the code matrix $C=\tilde{M}\Lambda$ has a systematic form. Without loss of generality, we may assume

$$C_{[k]}=\begin{pmatrix} s_1 & s_4 & s_7 & s_9 & s_{12} & s_{15} & s_{18}\\ s_2 & s_5 & * & s_{10} & s_{13} & s_{16} & s_{19}\\ s_3 & s_6 & s_8 & s_{11} & s_{14} & s_{17} & s_{20}\end{pmatrix},\quad \tilde{M}=\begin{pmatrix} m_{0,0} & m_{0,1} & \boldsymbol{m_{0,2}} & m_{0,3} & m_{0,4} & \boldsymbol{m_{0,5}} & m_{0,6} & \boldsymbol{m_{0,8}}\\ m_{1,0} & m_{1,1} & \boldsymbol{m_{1,2}} & m_{1,3} & m_{1,4} & \boldsymbol{m_{1,5}} & m_{1,6} & \boldsymbol{m_{1,8}}\\ m_{2,0} & m_{2,1} & \boldsymbol{m_{2,2}} & m_{2,3} & m_{2,4} & \boldsymbol{m_{2,5}} & m_{2,6} & \mathbf{0}\end{pmatrix},$$

where the $*$ in $C_{[k]}$ means the redundant symbol as displayed in Fig. C1, and the symbols in bold face in $\tilde{M}$ form the symmetric matrix $\tilde{M}_1$ in (C1). Note according to our construction, the set $J=[0,6]\cup\{8\}$.

According to (B1), it has

$$\tilde{M}_1\begin{pmatrix}1 & 1\\ 1 & \xi^u\\ 1 & (\xi^u)^2\end{pmatrix}=\begin{pmatrix}m_{0,2} & m_{0,5} & m_{0,8}\\ m_{1,2} & m_{1,5} & m_{1,8}\\ m_{2,2} & m_{2,5} & 0\end{pmatrix}\begin{pmatrix}1 & 1\\ 1 & \xi^u\\ 1 & (\xi^u)^2\end{pmatrix}=\begin{pmatrix}\tilde{h}^{(0)}_{0,u-1} & \tilde{h}^{(1)}_{0,u-1}\\ * & \tilde{h}^{(1)}_{1,u-1}\\ \tilde{h}^{(0)}_{2,u-1} & \tilde{h}^{(1)}_{2,u-1}\end{pmatrix}. \tag{C3}$$

Note the $\tilde{h}^{(i)}_{j,u-1}$'s on the right side of (C3) can be derived from $s_1,...,s_{20}$. For example, $\tilde{h}^{(0)}_{0,u-1}$ is determined as the leading coefficient of a degree 2 polynomial with three evaluations s_1,s_4,s_7, and $\tilde{h}^{(1)}_{0,u-1}$ is determined by s_9,s_{12},s_{15}. Then from the last row in (C3), one can recover $(m_{2,2},m_{2,5})$ as

$$(m_{2,2},m_{2,5})=(\tilde{h}^{(0)}_{2,u-1},\tilde{h}^{(1)}_{2,u-1})\begin{pmatrix}1 & 1\\ 1 & \xi^u\end{pmatrix}^{-1}.$$

Then the symmetry of $\tilde{M}_1$ implies $m_{0,8}=m_{2,2}$ and $m_{1,8}=m_{2,5}$. In a similar way, one can derive $(m_{0,2},m_{0,5})$, $m_{1,2}=m_{0,5}$, and finally derive $m_{1,5}$. Therefore, $\tilde{M}_1$ is completely determined by the known entries of $C_{[k]}$.

After recovery of $\tilde{M}_1$, one can fill up the matrix on the right side of (C3) by computing the $*$ entry which is supposed to be $\tilde{h}^{(0)}_{1,u-1}$. Then with the leading coefficient $\tilde{h}^{(0)}_{1,u-1}$ and two evaluations s_2,s_5, one can recover the $*$ entry in C_k as the third evaluation of a degree 2 polynomial. Thus $C_{[k]}$ is completely determined which means the data stored in the first k nodes are known. Then by the data reconstruction process described in the proof of Theorem 1, one can finally obtain the desired $\tilde{M}$.

References

1 Rashmi K V, Shah N B, Vijay Kumar P. Optimal exact-regenerating codes for distributed storage at the MSR and MBR points via a product-matrix construction. IEEE Trans. Inform. Theory, 2011, 57: 5227-5239