Rate-compatible ARACA codes

K.J. Jeon and K.S. Kim✉

A design method for rate-compatible (RC) accumulate repeat accumulate check accumulate (ARACA) codes as protograph-based RC low-density parity-check codes is presented. The proposed RC-ARACA codes provide not only good iterative decoding thresholds with the linear minimum distance growth property but also efficient encoder/decoder structures with low complexities, so that they are appropriate to ultra-reliable and low latency communication services for fifth generation.

Introduction: Low-density parity-check (LDPC) code is considered the most promising candidate for fifth generation (5G) to support new emerging services {e.g. enhanced mobile broadband and ultra-reliable and low latency communication (URLLC) [1]} due to near-optimal performance and high-speed encoder/decoder structure. In practical systems, rate-compatible (RC) codes are preferred because they can both provide various code rates efficiently and support incremental redundancy hybrid automatic repeat request (HARQ). To support the above services for 5G, RC codes need to have good waterfall performance, no error floor and low encoding and decoding complexity with a linear order. Thus, for the design of RC-LDPC codes, we need to consider the above three properties. In [2–4], several RC-LDPC codes covering a wide range of code rates with good iterative decoding thresholds have appeared recently. These codes are classified as RC protograph-based raptor-like (PBRL) codes which are typically composed of a highest-rate code (HRC) and its lower-RCs by adding single PC (SPC) incremental redundancy codes (IRCs). Although such RC-LDPC codes can provide some desired properties, they fail to simultaneously satisfy the above three criteria by the following reasons. (i) Samsung codes [2] suffer from error floors due to their lack of the linear minimum distance growth (LMDG) property. (ii) PBRL-1 [3] and PBRL-2 [4] codes suffer from the lack of an efficient encoder structure and require employing a dense block-circulant systematic generator matrix (BCSGM) with a high complexity for encoding an HRC [5]. Recently in [5], Jeon and Kim proposed accumulate repeat accumulate check accumulate (ARACA) codes, which can simultaneously provide the above three criteria. However, the ARACA codes in [5] are not yet enough to support various 5G applications due to the lack of the nested code structure. Thus, in this Letter, we propose a design methodology for a family of RC-ARACA codes satisfying the above three desired properties.

Notations: Calligraphic and Boldface characters represent sets, matrices or vectors, respectively. $\mathbf{0}_n$ and $\mathbf{0}_{m\times n}$ denote the all zero row vector of length n and the all zero $m\times n$ matrix, respectively, $[\boldsymbol{a}; \boldsymbol{b}] = [\boldsymbol{a}^{\mathrm{T}}, \boldsymbol{b}^{\mathrm{T}}]^{\mathrm{T}}$ where $(\cdot)^{\mathrm{T}}$ denotes the matrix transpose for vectors $\boldsymbol{a}$ and $\boldsymbol{b}$, $\mathrm{diag}\{\boldsymbol{a}\}$ denotes the diagonal matrix with its diagonal vector $\boldsymbol{a}$ and $||\boldsymbol{a}||_1$ stands for the L_1-norm of a vector $\boldsymbol{a}$. In addition, $f(\boldsymbol{a}, b) = \{i|a_i \geq b,\ i = 1, \ldots, L\}$ for a given vector $\boldsymbol{a} = [a_1, \ldots, a_L]$ and $|\mathcal{A}|$ stands for the cardinality of set $\mathcal{A}$.

RC-ARACA code: We design a family of RC-ARACA codes by considering: (i) an HRC meeting the above three criteria and (ii) IRCs using SPC codes or RA codes to provide additional coding gains while maintaining efficient encoder/decoder structures. Let $\mathcal{R} = \left\{((n-m)/(n-1+i))\right\}_{i=0}^{I}$ be the set of code rates for designing a family of RC-ARACA codes, where n and m are positive integers and $\mathcal{H} = \left\{\boldsymbol{H}^{(R)}|R \in \mathcal{R}\right\}$ be the set of the corresponding protomatrices (representing the protographs). In addition, $\boldsymbol{H}^{((n-m/n-1))}$ denotes the $m\times n$ HRC protomatrix, in which the $(n-m+1)$th column denotes a punctured precoding variable node (VN) having the highest degree. On the basis of the HRC protomatrix $\boldsymbol{H}^{((n-m/n-1))}$, the $(m+i)\times(n+i)$ protomatrices $\boldsymbol{H}^{((n-m)/(n-1+i))} = [\tilde{\boldsymbol{H}}^{((n-m)/(n-2+i))};\ \tilde{\boldsymbol{h}}_{\mathrm{IR}}^{(i)}] = [[\boldsymbol{H}^{((n-m)/(n-2+i))},\ 0];\ [\boldsymbol{h}_{\mathrm{IR}}^{(i)},\ p^{(i)}]],\ 1 \leq i \leq I$, are built by sequentially adding a column (VN) and a row [check node (CN)] to $\boldsymbol{H}^{((n-m)/(n-2+i))}$ where $\boldsymbol{h}_{\mathrm{IR}}^{(i)} = [\bar{\boldsymbol{h}}_{\mathrm{IR}}^{(i)},\ \mathbf{0}_{i-1}]$, $\bar{\boldsymbol{h}}_{\mathrm{IR}}^{(i)} = [h_{\mathrm{IR}}^{(i)}[1], \ldots, h_{\mathrm{IR}}^{(i)}[n]]$ and $p^{(i)} \in \{1, 2\}$.

To limit the search space to a reasonable size for the protograph design of a good HRC $\boldsymbol{H}^{((n-m/n-1))}$, we impose some constraints as follows. In inner code part 1, only outer connected degree-2 zigzag close loops are allowed since additional coding gain from degree-1 open loops is negligible for a high-RC (see [5]). Also, in outer code part 1, the degree of outer connections up to 3 is allowed. In outer code part 2, the degree of information VN (except the information VN used for precoding) is set to either 3 or 4 because at least 3 repetitions of each information VN is required to guarantee the LMDG property [5] and it has been reported that repetitions more than four may degrade the iterative decoding threshold [3, 4]. Also, in outer code part 2, the $(n-m+1)$th column (i.e. the punctured VN) should include at least one single edge to operate the message-passing decoder well [6] and the sum of the entries of the corresponding column, except the first one (i.e. precoder) should be >3 in order to achieve a good error floor performance [5]. Note that the above constraints reduce the search space significantly compared with the full search space while avoiding meaningful performance degradation. From among the possible candidates, a suboptimal HRC protomatrix is found using the reciprocal channel approximation-based density evolution, in which the search is terminated when the binary-input (BI) additive white Gaussian noise channel (AWGNC) iterative decoding threshold is below the capacity threshold (i.e. Shannon limit) plus a given threshold T_{c}.

In (1), $\bar{\boldsymbol{h}}_{\mathrm{IR}}^{(i)}$ and $p^{(i)}$, for $i = 1, \ldots, I$ are determined sequentially for rates $R = (n-m/n), \ldots, (n-m/(n+I-1))$. To limit the search space to a reasonable size, the entries of $\bar{\boldsymbol{h}}_{\mathrm{IR}}^{(i)}$ are constrained as follows: (i) $h_{\mathrm{IR}}^{(i)}[n-m+1] = 2$ if i is odd and 1 otherwise; (ii) $|f(\bar{\boldsymbol{h}}_{\mathrm{IR}}^{(i)}, 1)| \leq N_{\mathrm{e}}$, $q_{\min} \leq ||\bar{\boldsymbol{h}}_{\mathrm{IR}}^{(i)}||_1 \leq q_{\max}^{(i)}$ and $h_{\mathrm{IR}}^{(i)}[j] \leq 1$ for $1 \leq j \leq n$, where N_{e}, $q_{\min}$ and $q_{\max}^{(i)}$ are small positive integers; and (iii) $|f(\sum_{i'=1}^{i} \bar{\boldsymbol{h}}_{\mathrm{IR}}^{(i')},\ \alpha_1\gamma_i)| \leq \beta_1 n$, $|f(\sum_{i'=1}^{i} \bar{\boldsymbol{h}}_{\mathrm{IR}}^{(i')},\ \alpha_2\gamma_i)| \geq (1-\beta_2)n$ and $|f(\bar{\boldsymbol{h}}_{\mathrm{IR}}^{(i)}, 1) \bigcap f(\bar{\boldsymbol{h}}_{\mathrm{IR}}^{(i')}, 1)| \leq N_{\mathrm{o}}$, $i' < i$, for $2 \leq i \leq I$, where N_{o} is a small positive integer, $\alpha_1 > 1$, $0 < \alpha_2, \beta_1, \beta_2 < 1$ and $\gamma_i = (\sum_{i'=1}^{i} ||\bar{\boldsymbol{h}}_{\mathrm{IR}}^{(i')}||_1)/n$. Here, the first constraint means that an edge connecting a high-degree VN to each CN has a degree of either 1 or 2, which reflects the facts that (i) the high-degree VN connected to all of the CNs is punctured for better iterative decoding threshold as reported in [3, 4]; (ii) a substantial number of CNs need to be connected to the punctured VN with a single edge because a punctured VN can be recovered with reliable messages when it has more neighbouring survived CNs connected to transmitted VNs, except the punctured one [6]; and (iii) it is desired not to use higher-degree edges to reduce short cycles. Also, the second constraint is set to limit the search space because, inspired from the results in [2–4], it is possible to design good RC-LDPC codes even if the degree of each edge and the number of edges of each CN are limited to some small integers. In addition, the third constraint aims that there are appropriate numbers of relatively high- and low-degree VNs and that the edge connections among IRCs are well distributed for better iterative decoding threshold as noted in [2–4]. Owing to the above constraints, the search space for finding $\bar{\boldsymbol{h}}_{\mathrm{IR}}^{(i)}$ is roughly reduced to $O(n!)$ from $O(\mathrm{e}^n)$. Note that the candidates of the reduced search space by the above three constraints have similar degree distributions, so that they have fairly comparable iterative decoding thresholds. Thus, we can design $\bar{\boldsymbol{h}}_{\mathrm{IR}}^{(i)}$ as follows: (i) Randomly choose the candidate among the reduced search space; (ii) update the survived candidate if the chosen one has better iterative decoding threshold; (iii) stop if the iterative decoding threshold of survived candidate is below the capacity threshold plus T_{c}; (iv) update $T_{\mathrm{c}} = T_{\mathrm{c}} + \epsilon$ for $\exists\epsilon > 0$ if the current iteration index is an integer multiple of $L_{\max}$. Then, go to step i.

Now, we present the design example of an RC-ARACA code. Set $n = 11$, $m = 3$ and $\mathcal{R} = \{(8/10+i)|i = 0, \ldots, 22\}$. In addition, set $q_{\min} = 3$ and $q_{\max}^{(i)} = 7$ for $1 \leq i \leq 6$ (i.e. $8/11 \leq R \leq 1/2$) and $q_{\max}^{(i)} = 5$ for $7 \leq i \leq 14$ (i.e. $1/2 < R \leq 1/3$) and $q_{\max}^{(i)} = 4$ for $15 \leq i \leq 22$ (i.e. $1/3 < R \leq 1/4$). Then, set $N_{\mathrm{o}} = 5$, $\alpha_1 = 2$, $\alpha_2 = 0.75$, $\beta_1 = 0.2$, $\beta_2 = 0.6$, $T_{\mathrm{c}} = 0.36\,\mathrm{dB}$ and $L_{\max} = 30$. The average number of iterations required to obtain $\bar{\boldsymbol{h}}_{\mathrm{IR}}^{(i)}$ is roughly 32 for $1 \leq i \leq 6$, 24 for $11 \leq i \leq 14$ and 9 for $15 \leq i \leq 22$. The rate−1/4 protomatrix found by the proposed method is

$$\boldsymbol{H}^{\left(\frac{1}{4}\right)} = \begin{bmatrix} \left[\boldsymbol{H}^{\left(\frac{4}{5}\right)},\ \mathbf{0}_{3\times 22}\right] \\ \left[\boldsymbol{h}_{\mathrm{IR}}^{(1)},\ p^{(1)},\ \mathbf{0}_{21}\right] \\ \vdots \\ \left[\boldsymbol{h}_{\mathrm{IR}}^{(22)},\ p^{(22)}\right] \end{bmatrix} = \begin{bmatrix} \boldsymbol{H}^{\left(\frac{4}{5}\right)} & \mathbf{0}_{3\times 22} \\ h_{\mathrm{IR}}^{(1)} & \left[p^{(1)},\ \mathbf{0}_{21}\right] \\ \vdots & \vdots \\ h_{\mathrm{IR}}^{(22)} & \left[\mathbf{0}_{21},\ p^{(22)}\right] \end{bmatrix} = \begin{bmatrix} \boldsymbol{H}^{\left(\frac{4}{5}\right)} & \mathbf{0}_{3\times 22} \\ \boldsymbol{H}_{\mathrm{IR}} & \boldsymbol{P} \end{bmatrix}, \quad (1)$$

where

$$\boldsymbol{H}\left(\frac{4}{5}\right)=\begin{bmatrix}1&0&0&0&0&0&0&0&2&0&0\\0&2&2&2&2&2&2&3&1&2&0\\0&2&2&2&2&2&2&0&3&1&2\end{bmatrix},\qquad(2)$$

$$\boldsymbol{H}_{\mathrm{I}R}=\begin{bmatrix}0&1&1&1&1&0&0&0&2&0&0\\0&1&1&1&1&1&0&0&1&0&0\\0&1&1&1&1&0&1&0&2&0&0\\0&1&1&1&1&0&0&1&1&0&0\\0&1&1&0&0&1&1&0&2&0&0\\0&1&1&0&0&1&0&0&1&0&0\\0&1&1&0&0&0&0&0&2&0&1\\0&1&0&1&1&0&0&0&1&0&0\\0&1&1&0&0&1&0&0&2&0&0\\0&0&1&0&0&1&0&1&1&0&0\\0&1&0&0&0&0&0&0&2&1&0\\0&0&0&0&0&0&0&1&1&0&1\\0&1&0&0&0&0&0&0&2&0&0\\0&0&1&0&1&0&0&0&1&1&0\\0&0&0&0&0&0&0&1&2&0&0\\0&0&0&1&1&1&0&0&1&0&0\\0&0&0&0&0&0&0&0&2&0&1\\0&1&1&0&0&0&0&0&1&0&0\\0&0&0&0&0&0&0&0&2&1&1\\0&0&0&1&1&0&0&0&1&0&0\\0&0&0&0&0&1&1&0&2&0&0\\0&0&0&1&1&0&0&0&1&0&0\end{bmatrix},\qquad(3)$$

and $\boldsymbol{P}=\mathrm{diag}\{[1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 2\ 1\ 1\ 1\ 1\ 2\ 1\ 1\ 1\ 1]\}$, the ninth column (highest-degree VN) denotes the punctured VN. The iterative decoding thresholds for the BI-AWGNC of the above RC-ARACA codes are between those of PBRL-2 codes and those of PBRL-1 codes.

The PC matrix (PCM) is obtained by two-step lifting as in [5], in which the first lifting uses a small expansion $B_1 = 3$ and the second lifting uses the circulants of the identity matrix with size $B_2 = 43$. The PCMs of PBRL-1 and PBRL-2 codes are constructed from the protomatrices corresponding to (2) in [3] and (13) and (14) in [4] using the above same methodology, respectively. Also, the PCMs of Samsung codes are constructed from the adjacent matrices in [2] by the above second lifting with $B_2 = 41$.

The encoding of an RC-ARACA code is as follows. (i) An HRC is encoded in four steps [5] with a linear encoding complexity. (ii) Each IRC is encoded with either only block SPC (BSPC) operations for degree-1 parity VNs or a BSPC operation and the combination of the bit interleaver, accumulator and deinterleaver for degree-2 parity VNs. In the encoding of a PBRL-1 (or a PBRL-2) code, an HRC is encoded by the dense BCSGM and each IRC is encoded by only BSPC operation. In the encoding of a Samsung code, an HRC is encoded by the Richardson–Urbanke encoding method [7] with a linear encoding complexity and each IRC encoding is similar to that in a PBRL-1 (or PBRL-2) code. Note that the required numbers of binary additions for code rate of $R = 1/2$ are 10,707 for the RC-ARACA codes, 9840 for the Samsung codes, 219,470 for the PBRL-1 codes and 221,916 for the PBRL-2 codes, respectively. Also, we assume that the decoder for each code uses a sum–product algorithm (SPA)-based-layered decoding (LD) algorithm [8], where each layer denotes a bundle of rows of the same size as the circulant submatrix (B_2). Note that the corresponding decoder complexity per iteration is given by $2d_v^*N^* + 2(d_c^* - 1)M + 6 \times 2d_c^*M$ [2], where N^*, M, d_v^* and d_c^* denote the number of VNs excluding degree-1 VNs, the number of CNs, the average VN degree without degree-1 VNs and the average CN degree, respectively. Then, the required decoder complexity (counting the number of operations with complexity similar to an addition) for code rate of $R = 1/2$ is given by 146,286 for the RC-ARACA codes, 146,206 for the Samsung codes, 149,898 for the PBRL-1 codes or 135,708 for the PBRL-2 codes, respectively. All code families show comparable decoder complexities since their PCMs are similarly dense.

Performance evaluation: The frame error rate (FER) performance results are obtained over the AWGNC using the SPA-LD algorithm with 50 iterations until 200 frame errors occur. Fig. 1*a* compares the FER performance of the proposed RC-ARACA codes with those of the PBRL-1 codes, PBRL-2 codes and Samsung codes with $K = 1032$ (or 1025 for the Samsung codes) for code rates of $R = 1/2$, 1/3 and 1/4. Compared to the Samsung codes, the proposed RC-ARACA codes show similar waterfall performance and outperform the Samsung codes in the UR regime (e.g. FER of 10^{-5}) because the proposed RC-ARACA codes do not suffer from error floors due to their LMDG property. Compared to the PBRL-1 codes, both code families show excellent performances in the UR regime due to their LMDG property and the proposed RC-ARACA codes show slightly better waterfall performance than the PBRL-1 codes due to the better iterative decoding thresholds. Compared to the PBRL-2 codes, even if the proposed RC-ARACA codes show slightly worst waterfall performance than the PBRL-2 codes, they show better performance than the PBRL-2 codes in the UR regime because the PBRL-2 codes suffer from degraded error floor performance caused by their smaller degrees of repetitions in their inner codes. Fig. 1*b* compares the required signal-to-noise ratio (SNR) to achieve the target FERs (10^{-2} and 10^{-5}) for the above code families according to the code rate. For the target FER of 10^{-2}, the four RC-LDPC codes show similar performance. However, for the target FER of 10^{-5}, the proposed RC-ARACA codes outperform the Samsung codes and the PBRL-2 codes and show similar performance of the PBRL-1 codes over a wide range of code rates (Note that the PBRL codes suffer from high encoding complexity.).

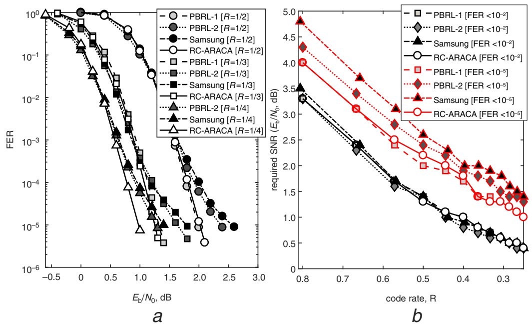

Fig. 1 *Performance comparison for each code family*

a FER performance comparison for $R = 1/2$, 1/3 and 1/4
b Required SNR to achieve target FERs of 10^{-2} and 10^{-5} when $K = 1032$

Conclusion: This Letter presented a design method for an RC-ARACA code having good iterative decoding thresholds with the LMDG property and efficient encoder/decoder structures from a reasonably reduced search space, which can be considered as a promising solution for 5G.

Acknowledgment: This research was supported by IITP grant funded by the Korea government (MSIT) [2015-0-00300 & 2016-0-00208].

Submitted: *13 December 2017* E-first: *31 January 2018*
doi: 10.1049/el.2017.4653
One or more of the Figures in this Letter are available in colour online.
K.J. Jeon and K.S. Kim (*Department of Electrical and Electronic Engineering, Yonsei University, 50 Yonsei-ro, Seodaemun-gu, Seoul 120–749, Republic of Korea*)

✉ E-mail: ks.kim@yonsei.ac.kr

References

1 ITU-R Rec. M.2083-0.: 'IMT vision-framework and overall objectives of the future development of IMT for 2020 and beyond,' 2015
2 3GPP.: 'Preliminary evaluation results on new channel coding scheme for NR,' 2016, R1-164008
3 Nguyen, T.V., and Nosratinia, A.: 'Rate-compatible short-length protograph LDPC codes', *Commun. Lett.*, 2013, **17**, (5), pp. 948–951
4 Chen, T., Vakilinia, K., Divsalar, D., *et al.*: 'Protograph-based raptor-like LDPC codes', *Trans. Commun.*, 2015, **63**, (5), pp. 1522–1532
5 Jeon, K.J., and Kim, K.S.: 'Accumulate repeat accumulate check accumulate codes', *Trans. Commun.*, 2017, **65**, (11), pp. 4585–4599
6 Ha, J., Kim, J., Klinc, D., *et al.*: 'Rate-compatible punctured low-density parity-check codes with short block lengths', *Trans. Inf. Theory*, 2006, **52**, (2), pp. 728–738
7 Richardson, T.J., and Urbanke, R.L.: 'Efficient encoding of low-density parity-check codes', *Trans. Inf. Theory*, 2001, **47**, (2), pp. 638–656
8 Mansour, M., and Shanbhag, N.: 'High-throughput LDPC decoders', *Trans. Very Large Scale Integr. (VLSI) Syst.*, 2003, **11**, (6), pp. 976–996