Information Reconciliation for QKD with Rate-Compatible Non-Binary LDPC Codes

Kenta KASAI, Ryutaroh MATSUMOTO, and Kohichi SAKANIWA
Dept. of Communications and Integrated Systems
Tokyo Institute of Technology
152-8550 Tokyo, JAPAN
{kenta@comm., ryutaroh@it., sakaniwa}@ss.titech.ac.jp

Abstract—We study information reconciliation (IR) scheme for the quantum key distribution (QKD) protocols. The IR for the QKD can be considered as the asymmetric Slepian-Wolf problem, which low-density parity-check (LDPC) codes can solve with efficient algorithms, i.e., the belief propagation. However, the LDPC codes are needed to be chosen properly from a collection of codes optimized for multiple key rates, which leads to complex decoder devices and performance degradation for unoptimized key rates. Therefore, it is desired to establish an IR scheme with a single LDPC code which supports multiple rates. To this end, in this paper, we propose an IR scheme with a rate-compatible non-binary LDPC code. Numerical results show the proposed scheme achieves IR efficiency comparable to the best known conventional IR schemes with lower decoding error rates.

Index Terms—quantum key distribution, information reconciliation, low-density parity-check code, iterative decoding

I. INTRODUCTION

The quantum key distribution (QKD) protocol invented in [1] is one of technologies nearest to practical realization among various quantum information processing technologies. The goal of a QKD protocol is to share a common random string, called key, between two legitimate users Alice and Bob secretly from the eavesdropper Eve. Alice and Bob can use an authenticated public classical channel between them to achieve the goal, but Eve can see all the contents in the public channel. In addition to this classical channel, there is a quantum channel between Alice and Bob over which quantum objects are transmitted from Alice to Bob. Observe that this is a quantum extension of the model CW introduced by Ahlswede and Csiszár [2, Section 9.2] with the classical noisy channel replaced by the quantum noisy one.

As categorized in [3], [4], a QKD protocol can usually be divided into four steps:

1) Quantum transmission and reception
2) Channel parameter estimation
3) Information reconciliation
4) Privacy amplification

Note that the third and fourth steps are essentially the same as the information theoretically secure key agreement introduced by Maurer, Ahlswede, and Csiszár [2, Chapter 9]. Thus, many parts of this paper are also relevant to the information theoretically secure key agreement.

Traditional security proofs for QKD protocols, for example [5], combines the information reconciliation (IR) and the privacy amplification. Because of that, we could not study the IR in QKD protocols separately from the privacy amplification, for example, we could not investigate what kind of the information reconciliation was suitable without considering the privacy amplification. This situation was reversed by the several new security proofs [6], [7], [8], [9], [10], [11], [12], [3], [13], [14], which enabled us to study the information reconciliation in QKD protocols without considering other steps in QKD protocols.

The rest of this paper is organized as follows. Section II reviews the Slepian-Wolf coding [15, Section 15.4] and its relation to the IR and conventional scheme. Section III proposes an IR scheme with a rate-compatible non-binary LDPC code. Section IV explains the decoding algorithm for the proposed IR scheme. Section V shows the numerical results. We conclude the paper in Section VI.

II. PROBLEM STATEMENT AND RELATED WORKS

We assume that physical objects with two-dimensional state spaces are transmitted in the QKD protocols. This assumption is valid in one of several common realization of QKD protocols. Another common realization of QKD protocols uses infinite-dimensional objects [16]. Information reconciliation in such a case is discussed in [17], [18], [19], [20]. The IR in the infinite-dimensional case seems more challenging than the two-dimensional case.

After the channel parameter estimation, Alice has an n-bit binary string \( \zeta_n = (\zeta_1, \ldots, \zeta_n)^T \), Bob has \( \nu_n = (\nu_1, \ldots, \nu_n)^T \), and they know an estimate of the joint probability distribution \( \overline{P}_{XY} \) assuming that \( \zeta_i, \nu_i \) are i.i.d. for all \( i = 1, \ldots, n \). The goal of the IR is for Bob to produce a string \( \zeta_n \) by (possibly two-way) conversation with Alice over the public channel. The entire content of their conversation depends on \( \zeta_n \) and \( \nu_n \), and \( \langle \zeta_n, \nu_n \rangle \) denotes the entire conversation. The desirable properties of the IR are

- Make \( \Pr(\zeta_n \neq \zeta_n) \) sufficiently close to zero.
- Make the mutual information \( \langle \zeta_n; \nu_n \rangle \) as small as possible.

The reason behind the second property is that we must subtract \( \langle \zeta_n; \nu_n \rangle \) bits from the length of the final
secret key [7], [13], because \((n; (n; n))\) is the amount of information leaked to Eve during the conversation over the public channel. Note that decreasing \((n; (n; n))\) is totally different from decreasing the number of bits in the conversation \((n, n)\). For example, the celebrated IR protocol Cascade [21], [22] exchanges many bits between Alice and Bob bit by bit, while keeping \((n; (n; n))\) relatively small.

In the one-way conversation, \((n, n)\) is a function of \(n\), denoted by \((n)\). We have that \((n; (n)) \leq (n, n)\) \((\leq n)\) \(\leq \) the number of bits in \((n, n)\). We can find a good IR method by saving the number of bits in \((n, n)\) while enabling Bob to decode \(n\) from \((n)\) and \(n\). This is a kind of data compression problem, called the Slepian-Wolf (SW) problem.

A simplified version of the general Slepian-Wolf problem [15, Section 15.4] is given in Figure 1. The main information \(n\) is statistically correlated with the side information \(n\). The encoder (data compressor) can only use \(n\) for generating the codeword (compressed data) \(SW(n)\) of some fixed length. On the other hand, the decoder (decompressor) can use both \(SW(n)\) and \(n\).

If \(n\) is unavailable by the decoder, the compression rate \(R\) must be \(> (n)\), the entropy of \(n\), in order for the decoding error probability \(Pr[n, \neq n]\) to be negligible. The availability of \(n\) improves the optimal compression rate to \((n)\) from \((n)\). The encoder and the decoder are assumed to know (a good estimate of) the joint probability distribution \(XY\), and they are usually optimized for a particular \(XY\). This special form of the Slepian-Wolf coding is called an asymmetric Slepian-Wolf coding [23], because the roles of \(\cdot\) and \(\cdot\) are asymmetric at the decoder.

We return to the IR. Recall that Alice has \(n\) and Bob has \(n\). If Alice sends the codeword \(SW(n)\), then Bob can recover \(n\) with high probability by the Slepian-Wolf decoder and \(n\). The amount of information leaked to Eve is estimated as \((n; SW(n)) \leq (SW(n)) \leq \) the number of bits in \(SW(n)\). Thus, if the compression rate is better\(^1\), then the upper bound on the leaked information is smaller.

In [24], LDPC codes are used for the SW-coding. At arbitrary but fixed rate, the information bits are encoded as a syndrome of an irregular LDPC code. With a help from the syndrome and the correlated side information, the information bits are efficiently estimated by an efficient decoding algorithm, belief propagation (BP) decoding. The irregular LDPC code is optimized [24] by density evolution and used with very large information bits. The encoder and decoder have to be equipped with memory devices for multiple LDPC codes of large code length \(\sim 10^6\). Each irregular LDPC code of channel coding rate \(e\) is designed to have lower decoding bit error rate (BER) over channel with \((n, n)\) which is as close to SW-bound \(1 - e\) as possible.

In [25], Elkouss et al. proposed a rate-adaptive IR scheme with a set of optimized irregular LDPC codes for multiple channel coding rates, e.g., \(e = 0.1, 0.2, \ldots, 0.9\). For simplicity, assume those codes are highly optimized and capacity achieving over the BSC with conditional entropy \((n) = 1 - e\). In other words, each code of rate \(e\) attains vanishing decoding BER over any BSC with conditional entropy \((n) < 1 - e\). For the side information correlated through the BSC with conditional entropy \((n, n)\) such that \(1 - (n) \in \{0.1, \ldots, 0.9\}\), one can use the LDPC code of channel coding rate \(e = 1 - (n)\). The SW-bound is achieved in this case. Otherwise, one has to use the LDPC code of lower channel coding rate \(e < 1 - (n)\) such that \(e \in \{0.1, 0.2, \ldots, 0.9\}\), instead. Since this LDPC code is not optimized for the BSC with \((n)\), the performance is degraded. Such degradation leads to so called saw effects [25].

This issue of the saw effects can be solved by introducing rate-compatible SW-coding [26]. The rate-compatible SW-coding uses only a single mother LDPC code and is easily applied to IR as follows. First, Alice encodes the information bits \(n\) into encoded bits by the mother code. In response to Bob’s request, Alice exposes gradually \(n\) bits of the Slepian-Wolf decoding graphs but use only the single Tanner graph of the mother code to decode the information bits \(n\).

III. RATE-COMPATIBLE IR SCHEME WITH NON-BINARY LDPC CODES

In this section, we propose an IR scheme in conjunction with rate-compatible non-binary LDPC codes. Non-binary LDPC codes were originally invented by Gallager [27]. Davey and MacKay [28] found non-binary LDPC codes can outperform binary LDPC codes. The \((2, 2)\)-regular non-binary LDPC codes on \(GF(2^p)\) with \(2^p \geq 64\) are empirically known as the best performing error-correcting codes. Pouliat et al. [29] optimized \((2, 2)\)-regular non-binary LDPC codes by considering binary images of \(GF(2^p)\) symbols. The main shortcoming of non-binary LDPC codes is decoding complexity. However, reduced complexity algorithms for decoding non-binary LDPC codes have recently been proposed [30].

The SW-coding using non-binary LDPC codes was initially proposed in [31]. The non-binary LDPC codes are \((2, 2)\)-regular. The structure of the codes is based on [26] and the

\(^1\)Strictly speaking, the use of the Slepian-Wolf coding and the simple minimization of the number of bits in \(f_{SW}(X^n)\) neglect the optimization of the auxiliary random variables \(U\) and \(V\) in [3], which are the quantum counterparts of \(U\) and \(Q\) in [2, Theorem 9.2].
extension is straightforward. The numerical results revealed that the scheme faces a difficulty upon the use of weakly correlated side information. That degradation is due to the poor performance of (2,2)-regular non-binary LDPC codes.

After the channel estimation, Alice and Bob know \( (\_, \_) \). The maximum and minimum number of exposing parity bits \( \bar{n} \) and \( \underline{n} \) are set according to \( (\_, | \_) \). For example, as \( \bar{n} = 1 \) and \( \underline{n} = 0.95 \) \( (\_, | \_) \), respectively. We denote the number of exposing bits per conversation by \( E \), for example, \( E = 1 \) or \( E = 8 \). We consider the following IR scenario. First, Alice encodes the first \( \underline{n} \) encoded parity bits to Bob over the public channel. Bob has the exposed parity bits and the side information \( \bar{n} \). Bob tries to estimate \( \bar{n} \) with a help of the exposed parity bits and \( \underline{n} \) by the decoder of the mother code. If Bob detects the situation that the decoder can not recover \( \bar{n} \), Bob asks Alice to expose more parity bits. This procedure continues until Bob recovers \( \bar{n} \) or the number of exposed bits reaches \( \bar{n} \). The number of exposing request from Bob to Alice is bounded above by \( \frac{(\bar{n} - \underline{n})}{E} + 1 \).

Recall that the conventional IR scheme [25] requires preparing a set of optimized LDPC codes of multiple coding rates. In the proposed scheme, Alice and Bob use only the single mother non-binary LDPC code for any number of exposed parity bits.

A. Mother non-binary LDPC codes

Varodayan et al. realized the rate-compatibility using a binary low-density parity-check accumulator (LDPCA) code as a mother code. The LDPCA code can be considered as a rate-one LDGM code concatenated with an inner accumulator. The mother code used in this paper is motivated by the construction of the LDPCA codes.

We employ a non-binary LDPC codes defined on GF(2p) with \( p = 2^r \) and rate \( 3/8 \). For each \( r \), we fix a Galois field GF(2p) with a primitive element \( \alpha \) and its primitive polynomial \( \pi \). Once a primitive element \( \alpha \) of GF(2p) is fixed, each symbol is given a 3-bit representation [32, pp. 110]. For example, with a primitive element \( \alpha \in GF(2^3) \) such that \( \pi(\alpha) = \alpha^3 + \alpha + 1 = 0 \), each symbol is represented as \( 0 = (0, 0, 0), 1 = (1, 0, 0), \alpha = (0, 1, 0), \alpha^2 = (0, 0, 1), \alpha^3 = (1, 1, 0), \alpha^4 = (0, 1, 1), \alpha^5 = (1, 1, 1) \) and \( \alpha^6 = (1, 0, 1) \). With such correspondence, we deal with bits and the corresponding symbol in GF(p) interchangeably. We refer to elements in GF(2p) and GF(p) as non-binary symbols or simply symbols.

Let \( n = |X| \) denote the information symbol length of the mother non-binary LDPC code. Alice encodes the information symbols in GF(2p) into coded 8 /3 symbols. The number of the parity bits is 5 /3. This is larger than the mother non-binary LDPC code \( n \). However, Alice does not expose all the parity bits. The maximum number of exposed parity bits is set as \( \bar{n} \leq \underline{n} \).

The mother non-binary LDPC code is defined as follows. Let \( A \) be a 2 /3 by random matrix of column weight 2 and row weight 3. Each non-zero entry in \( A \) is chosen uniformly at random from GF(2p) \( \{0\} \). And we use a non-singular 2 /3 by 2 /3 matrix \( B \) and an by diagonal matrix \( \Lambda \) which can be written as

\[
B = \begin{pmatrix}
1 & 0 & 0 & 0 & 0 & 1 \\
2 & 1 & 0 & 0 & 0 & 0 \\
0 & 3 & 1 & 0 & 0 & 0 \\
\vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\
0 & 0 & 0 & 2^n/3-1 & 1 & 0 \\
0 & 0 & 0 & 0 & 2^n/3 & 1
\end{pmatrix},
\]

\[\Lambda = \text{diag}(\lambda_1, \ldots, \lambda_N),\]

where \( i \) for \( i = 1, \ldots, 2^n/3 \) and \( \lambda_i \) for \( i = 1, \ldots, \) are chosen uniformly at random from GF(2p) \( \{0\} \).

From the assumption of non-singularity, it holds that \( \prod_{i=1}^{2^n/3} \lambda_i \neq 1 \). The information symbol sequence \( X_N = (X_1, \ldots, X_N)^T \) is in GF(2p) \( N \), or equivalently can be viewed in GF(2) \( N \).

The mother code that we use in this paper is defined by a collection of codewords

\[(X_N)^T, (A^{2^n/3})^T, (R_N)^T\]^T

such that

\[A^{2^n/3} = B^{-1}X_N,\]

\[R_N = \Lambda^{-1}X_N.\]

We refer to \( A^{2^n/3} = (A_1, \ldots, A_{2^n/3})^T \) and \( R_N = (R_1, \ldots, R_N) \) as accumulated symbols and repeated symbols, respectively. The equation (1) is equivalently written as

\[AX_N + BA^{2^n/3} = 0,\]

\[\Lambda R_N + X_N = 0.\]

It follows that the code has 8 /3 symbols in GF(2p) and they are constrained by 5 /3 parity-check equations over GF(2p). The parity-check matrix of the mother code is given as

\[
\begin{pmatrix}
A & B & 0 \\
0 & 0 & \Lambda
\end{pmatrix}.
\]

It follows that \( (X_N)^T, (A^{2^n/3})^T, (R_N)^T \) = 0. The \( i \)-th symbol in the codeword \( (X_N)^T, (A^{2^n/3})^T, (R_N)^T \) is corresponding to an information symbol if \( i \leq \underline{n} \), an accumulated symbol if \( i + 1 \leq \underline{n} + 1 \leq 5 /3 \) and a repeated symbol if \( 5 /3 + 1 \leq \underline{n} \).

B. Exposing Order

After the information symbols \( X_N \in GF(2^p)^N \) are encoded with the mother code, in response to Bob’s request Alice gradually exposes the coded bits in a fixed order. First, Alice gradually exposes 2 /3 parity bits by bits from \( A^{2^n/3} \in GF(2^6)^{2^n/3} \), where \( = \). After that, Alice gradually exposes /3 parity bits by bits. The /3 bits are chosen uniformly at random from \( R_N \in GF(2^p)^N \) and exposed in an arbitrary but fixed order.

The following explains the order of exposing the first 2 /3 bits in \( A^{2^n/3} \). For simplicity, we focus on symbol-wise exposing. This can be easily interpolated by random bit-wise
exposing. We define $E_j$ as a set of the indices of the first $\gamma_i$ parity symbols to be exposed. $E_j$ is referred to as an exposing set. From $E_j$ for $\gamma_i = 1, \ldots, 2/3$, one can know the order of exposing symbols in $A^{2N/3}$. The exposing set $E_2$ is designed so that indices of exposed symbols are uniformly separated in the sense of Lee distance. For example, with $2/3 = 16$, $E_2$ are defined as follow.

$$
\begin{align*}
E_0 &= \{ \} \\
E_1 &= \{ 1 \} \\
E_2 &= \{ 1, 9 \} \\
E_4 &= \{ 1, 5, 9, 13 \} \\
E_8 &= \{ 1, 3, 5, 7, 9, 11, 13, 15 \} \\
E_{16} &= \{ 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16 \}
\end{align*}
$$

It can be seen that when all the parity symbols in $E_{2i}$ are exposed, the decoding on the Tanner graph behaves as if it ran on a $(2, 3 \cdot 16 / 2)$-regular non-binary LDPC code.

The exposing sets $E_j$ of $\gamma_i$ symbols ($2i \leq \gamma_i < 2i+1 \leq 2/3$) are defined by choosing randomly so that $E_{2i} \subset E_j \subset E_{2i+1}$, and $E_{2i-1} \subset E_j$. Thus we obtain $E_j$ for $\gamma_i = 0, \ldots, 2/3$. Then, Alice exposes the only symbol in $E_j \setminus E_{j-1}$ as the $\gamma_i$-th exposed parity symbol for $\gamma_i = 1, \ldots, 2/3$. For example, the order of the first 16 exposed parity symbols is given as follows.

$$A_1, A_9, A_5, A_{13}, A_7, A_{11}, A_3, A_{15}, A_4, A_{10}, A_8, A_{14}, A_2, A_{16}, A_{12}, A_6.$$ 

After, Alice exposes $2/3$ accumulated symbols $A^{2N/3}$, if Bob still requests to expose more parity bits, she gradually exposes more $\gamma_i/3$ parity bits randomly chosen from repeated symbols $R^{N}$ in an arbitrary but fixed order by $\gamma_i$ bits.

The Tanner graph of the non-binary LDPC code is depicted in Fig. 2. The variable nodes from left to right represent the repeated symbols $R^N$, the information symbols $X^N$, and the accumulated symbols $X^{2N/3}$.

**IV. Decoding Algorithm**

For $0 < \gamma_i \leq 1$, after Bob receives the exposed bits of a codeword of the mother non-binary LDPC, he estimates $\gamma_i$ by decoding the mother code with a help of $\gamma_i$. The non-binary LDPC codes are represented by Tanner graphs. Every variable node is associated with one symbol in a codeword. And every check node is associated one parity-check equation in Eq. (2). As observed in Eq. (2), the graph has $8/3$ variable nodes and $5/3$ check nodes. With some abuse of notation, we denote by $\gamma_i$ the variable node corresponding to the $\gamma_i$-th symbol of the code for $1 \leq \gamma_i \leq 8/3$. Similarly, we denote by $\gamma_i$ the check node corresponding to the $\gamma_i$ parity-check equation for $1 \leq \gamma_i \leq 5/3$. Denote the $(\gamma_i, \gamma_j)$ entry of $H$ by $w_{\gamma_i, \gamma_j} \in \text{GF}(2^p)$.

The decoding of LDPC codes can be nicely represented by a message passing algorithm on the graph. Each message is represented by a probability vector of length $2^p$. The $\gamma_i$-th variable node is corresponding to an information symbol if $1 \leq \gamma_i \leq 8/3$, to an accumulated symbol if $1 \leq \gamma_i \leq 5/3$, or to a repeated symbol if $1 \leq \gamma_i \leq 5/3 + 1 \leq \gamma_i \leq 8/3$. The decoder exchanges these messages between the variable and check nodes. We extend the definition of $X^N$ by defining $X_\gamma$ for $1 \leq \gamma_i \leq 8/3$ as a random variable of the $\gamma_i$-th encoded symbol. Let $W_\gamma$ and $w_\gamma$ be the decoder’s knowledge on the $\gamma_i$-th encoded symbol and its realization, respectively. If $1 \leq \gamma_i \leq 8/3$, the knowledge $w_\gamma$ is brought about by the side information $p_{\gamma_i+1}, \ldots, p_{\gamma_i+1}$. If $1 \leq \gamma_i \leq 5/3$, the knowledge $w_\gamma$ is brought about by the exposed bits that stem from the $\gamma_i$-th encoded symbol.

**Initialization:** The initial messages from a variable node to an adjacent check node is given by the conditional probability of $X_\gamma$ given $W_\gamma$.

$$v^{(0)}(\ ) \equiv v^{(0)}(\ ) = \Pr(X_\gamma = W_\gamma = w_\gamma) \quad \forall v \in \text{GF}(2^p).$$

And the initial message from a check node to an adjacent variable node is given by a uniform distribution

$$v^{(0)}(\ ) = 1/2 \quad \forall v \in \text{GF}(2^p).$$

Iterate the following message update rules.

**Check to Variable:** For a check node $c$, define $\partial_c \gamma_i$ as a set of variable nodes adjacent to $\gamma_i$. In this setting, the decoder iterates the following update rules for all the variable nodes and check nodes.

$$v_{\gamma_i}(\ ) = \frac{1}{\partial_c} \sum_{\gamma_j \in \partial_c} c_{\gamma_j, \gamma_i}(\ ) = c_{w_{\gamma_j, \gamma_i}}(\ )$$

$$\forall \gamma_i, \{ c_{\gamma_j, \gamma_i}(\ ) = 1 \quad \forall c \in \text{GF}(2^p).$$

$$v_{\gamma_i}(\ ) = \frac{1}{\partial_c} \sum_{\gamma_j \in \partial_c} c_{\gamma_j, \gamma_i}(\ ) = c_{w_{\gamma_j, \gamma_i}}(\ )$$

$$\forall \gamma_i, \{ c_{\gamma_j, \gamma_i}(\ ) = 1 \quad \forall c \in \text{GF}(2^p).$$
and minimum number of exposed parity bits are set as $N = 24576$.

The information length $n$ are known to perform well, and 8-bit symbols are easy to handle. The information length is $24576$ bits or equivalently $24576/8 = 3072$ symbols, i.e., 3K byte. The maximum and minimum number of exposed parity bits are set as $8 = 3072$ symbols, i.e., 3K byte. The maximum number of exposing bits per conversation is set as $8 = 3072$ symbols, i.e., 3K byte. The number of exposing request from Bob to Alice is bounded above by $(\overline{r} - \overline{\pi})/\overline{p} + 1 = 24.04$.

In Fig. 3, the blue curve shows the IR efficiency of Cascade [21] with $\epsilon = 10^{5}$. The the green curve plots the IR efficiency for the best known conventional binary LDPC coded IR scheme [25] with 9 optimized irregular LDPC codes with $= 10^{6}$ of channel coding rate $\epsilon = 0.90, 0.85, \ldots, 0.50$. The number of exposed bits is given as $\overline{p}(\epsilon) = (1 - \epsilon)$. Each point below the curve is plotted at $(\overline{p}(\epsilon), 1 - \epsilon)$ at the optimized irregular LDPC code of channel coding rate $\epsilon$ attains decoding bit error rate (BER) below $1.5 \cdot 10^{-6}$. Each irregular LDPC code can be considered to have the optimal channel for the best IR efficiency. If the estimated channel does not coincide with any of such optimized channels, Alice and Bob need to use the second best regular LDPC code for the estimated channel. In other words, Alice and Bob need to use irregular LDPC code with lower channel coding rate, i.e., with larger number of exposed bits. This sub-optimality leads to the degradation of IR efficiency and the saw effect.

The red curve shows the IR efficiency for the proposed IR scheme. Each point is attained with BER blow $2.0 \cdot 10^{-7}$. Due to the rate-compatibility, any points on the red curve are expected to be attained by the proposed IR scheme. Taking into account the fact the proposed IR scheme are free of saw effects, we conclude the proposed IR scheme is comparable to the best known IR scheme [25] in IR efficiency with lower decoding BER and with much smaller information length.

VI. CONCLUSIONS

In this paper, we proposed rate-compatible the IR for the QKD protocol with non-binary LDPC codes. Numerical results show the proposed scheme achieves IR efficiency comparable to the best known conventional IR scheme with lower decoding error rates.

REFERENCES

Information Reconciliation Efficiency $f = m / (nH(X|Y))$

Conditional Entropy $H(X|Y)$

Non-binary LDPC $n = 24576$

Binary LDPC $n = 10^6$

Cascade $n = 10^5$

$m/n = 1.0$

Fig. 3. IR efficiency comparison: Due to the rate-compatibility of the mother non-binary LDPC code, the proposed IR scheme is free of saw effects. The proposed scheme with $n = 24576$ information bits exhibit comparable IR efficiency to the best known binary LDPC coded IR scheme [25] with $n = 10^6$ and Cascade [21] with $n = 10^5$. 


