Preprints of the 21st IFAC World Congress (Virtual)
Berlin, Germany, July 12-17, 2020

On the Secret Sharing Scheme Based on Supercodes Decoding

Fedor Ivanov * **Alexey Kreschuk** ** **Eugenii Krouk** ***

* *National Research University Higher School of Economics & Insitute for Information Transmission Problems, Moscow, Russia (e-mail: fivanov@hse.ru).*
** *National Research University Higher School of Economics & Insitute for Information Transmission Problems, Moscow, Russia (e-mail: krsch@iitp.ru).*
*** *National Research University Higher School of Economics, Moscow, Russia (e-mail: ekrouk@hse.ru).*

Abstract: Secret sharing schemes have been studied intensively for the last 20 years, and these schemes have a number of real-world applications. There are a number of approaches to the construction of secret sharing schemes. One of them is based on codes of forward error correction (FEC). In fact, every linear code can be used to construct secret sharing schemes. For instance original Shamir secret sharing scheme is based on erasure decoding of Reed-Solomon codes. One of the main drawbacks of secret sharing schemes based on FEC is a dependence between number of users (participants) and field size of FEC. In this paper we propose a new scheme of secret sharing based on iterative decoding of LDPC codes in terms of supercodes decoding concept. In this scheme a field size can be made arbitrary and independent on the number of participants.

Keywords: LDPC codes, iterative decoding, supercodes decoding, secret sharing, quasi-cyclic LDPC codes, signal-noise ratio.

1. INTRODUCTION

Secret-sharing schemes are a technique which is used in many modern cryptographic protocols. A secret-sharing scheme consists of a dealer D who has some secret $\mathcal{S}$, a set of n parties (users), and a collection $\mathcal{A}$ of subsets of parties called the access structure. A secret-sharing scheme for $\mathcal{A}$ is a method by which the dealer distributes shares to the parties such that:

- Any subset in $\mathcal{A}$ can reconstruct $\mathcal{S}$
- Any subset not in $\mathcal{A}$ cannot reveal any partial information on the $\mathcal{S}$.

Originally motivated by the problem of secure information storage, secret-sharing schemes have found a number of other applications in cryptography and distributed computing: Byzantine agreement M. Ben-Or, et al. [1988], secure computations D. Chaum, et al. [1988],R. Cramer, et al. [2000], threshold cryptography Y. Desmedt et al. [1992], access control M. Naor. [1998] and attribute-based encryption V. Goyal, et al. [2006]–B. Waters. [2008].

Secret-sharing schemes were introduced by BlakleyG. R. Blakley. [1979] and Shamir A. Shamir. [1979] for the threshold case, i.e. for the case where the subsets that can reconstruct the secret are all the sets whose cardinality is at least a certain threshold: if n is a number of parties, $k < n$ is a threshold and secret $\mathcal{S}$ is shared into the n subsets of $\mathcal{A}$: $\mathcal{A} = \{\mathcal{A}_1, \mathcal{A}_2, \dots, \mathcal{A}_n\}$, then in order to reconstruct $\mathcal{S}$ any subset of $\mathcal{A}$ with cardinality at least k is required and any subset of $\mathcal{A}$ with smaller cardinality: $\{\mathcal{A}_{i_1}, \mathcal{A}_{i_2}, \dots, \mathcal{A}_{i_t}\}$, $t < k$, $0 < i_j \le n$ does not allow reconstruct $\mathcal{S}$.

Secret-sharing schemes for general access structures were introduced and constructed by Ito, Saito, and Nishizeki in paper M. Ito et al. [1993]. More efficient schemes were presented in J. Benaloh et al. [1990]. In this paper Benaloh and Leichter proved that if an access structure can be described by a small monotone formula then it has an efficient perfect secret-sharing scheme. This was generalized by Karchmer and Wigderson M. Karchmer et al. [1993] who showed that if an access structure can be described by a small monotone span program then it has an efficient scheme (a special case of this construction appeared before in E. F. Brickell. [1989]).

There are several approaches of constructing secret sharing schemes. One of them is based on FEC. In fact the first Shamir's scheme is FEC based. The relationship between Shamir's secret sharing scheme and the Reed-Solomon codes was pointed out by McEliece and Sarwate in 1981 R.J. McEliece et al. [1981]. Ather this paper was published, several authors have considered the construction of secret sharing schemes using linear FEC. On of the most important papers among them was written by Massey where he utilised linear codes for secret sharing and pointed out the relationship between the access structure and the code-

⋆ The research was supported by grant of President of Russian Federation, No. MK-1248.2020.9

words of minimal weight of the dual code of the underlying code J.L. Massey [1993], J.L. Massey [1995].

In this paper we consider another approach of secret sharing. It will be based on Quasi-Cyclic Low-Density Parity-Check Codes (QC-LDPC) with iterative decoding algorithm. Both codes and decoding rule were suggested by Gallager in Gallager [1963]. These linear block codes are defined by their parity-check matrice $\mathbf{H}$ characterized by a relatively small number of ones in their rows and columns.

Some classes of LDPC codes are used in cryptography, e. g. in McEliece codes-based asymmetric key cryptosystem McEliece [1978]. Usually these codes is applied to reduce key size in public-key cryptosystem M. Baldi et al. [2007].

In this paper we use QC-LDPC codes with decoding based on supercodes to construct threshold secret-sharing scheme. The main idea of this scheme is to apply decoding based on supercodes to recover common secret.

The paper is organized as follows: in 2 we introduce most common scheme of secret sharing and describe the main idea of threshold scheme. In section 3 we consider main definitions and notation refered to error-correction codes, that will be used later. In 4 we present the most common scheme of supercodes decoding that was first considered in Abramov et al. [2014]. In section 5 we consider the most common design of QC-LDPC Codes and in 6 we describe the decoding of these codes based on the main principles of supercodes decoding. Finally, in 7 we present our new secret-sharing scheme based on QC-LDPC codes with supercodes decoding.

2. SECRET SHARING SCHEMES

In this section we will give the most general description of secret sharing schemes.

Suppose that there are n participants in the sharing of a secret.

We will say that set (coalition) $\mathcal{A}_o \in \{1, 2, \dots, n\} = [n]$ of participants is *permitted* if these participants, having united, can gain access to the secret. All other coalitions that are not permitted are called *forbidden*.

The access structure of the secret sharing scheme will be called the pair (Δ, Γ), where the set of allowed sets is Γ, and Δ is the set of forbidden coalitions.

One of the main participants in the secret sharing scheme is the dealer. The dealer's task is to calculate the shares of the secret and distribute them among the participants.

Let $\mathcal{S}_0$ denote the finite nonempty set of all possible secret values with the corresponding random variable η taking the value on the Cartesian product of the sets $\mathcal{S}_1 \times \mathcal{S}_2 \times \dots \times \mathcal{S}_n$ and with the distribution function P on it, where the sets $\mathcal{S}_i$ are finite, η_i are the corresponding random variables on $\mathcal{S}_i$, and $s_i \in \mathcal{S}_i$ is the value of η_i . Dealer uses $(\eta_1, \dots, \eta_n)$ as a set of fractions of the secret $s_0 \in \mathcal{S}_0$. After choosing the secret s_0 with probability $p(s_0)$, the dealer sends the participants the secret fractions $s_1, s_2, \dots, s_n$ with the probability $P_{s_0}(s_1, s_2, \dots, s_n)$, namely, for the i-th participant, the secret fractions will be s_i. Then the coalition of participants $\mathcal{A}_o$ receives a collection $(s_i, i \in \mathcal{A}_o)$. In order for the secret sharing scheme to implement the access structure (Δ, Γ), we must ensure that all allowed coalitions can restore the secret. Formally, this can be written as follows:

$$P(\eta = s_0 | \eta_i = s_i, i \in \mathcal{A}_o) \in \{0, 1\}, \forall \mathcal{A}_o \in \Gamma$$

Let us note that each of the participants receives his share s_i and does not have information about the values of other shares, but he knows all the sets $\mathcal{S}_i$, as well as both probability distributions $p(s_0)$ and $P_{s_0}(s_1, \dots, s_n)$.

Let us introduce the concepts of *perfect* secret-sharing schemes. A *perfect* secret-sharing scheme is such a scheme in which forbidden sets do not receive any additional information to the available a priori about the possible value of the secret. This can be formalized as follows:

$$P(\eta = s_0 | \eta_i = s_i, i \in \mathcal{B}) = P(\eta = s_0), \forall \mathcal{B} \in \Gamma.$$

Let us consider a class of perfect schemes, namely threshold schemes. We will call scheme (Δ, Γ) a (k, n)-threshold scheme if any $\mathcal{A} \in [n]$, $|\mathcal{A}| > k - 1$ is in Γ and any $\mathcal{B} \in [n]$, $|\mathcal{B}| < k$ is in Δ.

Such schemes include, for example, the Shamir scheme and the Blackley scheme. Such secret sharing schemes are used to construct threshold cryptosystems. In a threshold cryptosystem, a message can be decrypted by a specific coalition of participants, between which the secret is shared. The group of participants has a common public encryption key, and the decryption key is divided between them using a scheme. A particular case of such a system is a threshold signature scheme. Threshold cryptography is used to store a secret key, for example, in the governmentatl and military areas, and it is also used in cloud environments and electronic voting schemes.

But in practice, threshold schemes are not enough in some cases, since the permitted sets can be arbitrary. One solution is to issue several keys to one participant, but such a solution is inefficient. In 2010, A. Abramov proposed the construction of a general-purpose secret sharing system, based on error-correcting codes in which the access structure can be arbitrary, with only one key being given to each participant.

3. FEC - PRELIMINARIES

Let us introduce some notation and definitions devoted to error-correcting codes that we will use in the paper.

Let us consider field F_2 of two elements 0 and 1 and modulo 2 operation. If V is a vector space of length n-tuples over F_2 ($V = F_2^n$), then any k-dimensional subspace $C \subset V$ is called linear (n, k) code. Each code C can be described either by it's generator matrix $\mathbf{G}$ (with size $k \times n$) constructed by any basis of C:

$$\mathbf{G} = \left(\mathbf{g}_1^T, \mathbf{g}_2^T, \dots, \mathbf{g}_k^T\right)^T,$$

where $\mathbf{g}_i$, $i = 1..k$ form basis of C, or by it's parity-check matrix $\mathbf{H}$ (with size $(n - k) \times n$) constructed by basis $\mathbf{h}_i$, $i = 1..n - k$ of orthogonal to C space $C^\perp$:

$$\mathbf{H} = \left(\mathbf{h}_1^T, \mathbf{h}_2^T, \dots, \mathbf{h}_{n-k}^T\right)^T.$$

In terms of either generator or parity-check matrix we can give to equivalent definitions of code C:

$$C = \{\mathbf{c} \in V : \mathbf{c} = \mathbf{u}\mathbf{G}, \forall \mathbf{u} \in F_2^k\} = \{\mathbf{c} \in V : \mathbf{c}\mathbf{H}^T = \mathbf{0}\}.$$

Let us consider arbitraty (n, k) code $A \subset V$. If $A' \subset V$: $A \subset A'$ then code A' is called *supercode* of code A. It is obvious that A' is a linear (n, k') code where $k' > k$.

The concept of error-correcting (n, k) code is inextricably linked to two functions: encoding and decoding.

The encoding $\psi(u)$ maps all possible vectors $\mathbf{u} = (u_1, u_2, \ldots u_k) \in F_2^k$ to elements of A: $\psi : F_2^k \mapsto A$ by the rule: $\psi(\mathbf{u}) = \mathbf{uG}$.

The decoding function (algorithm) $\xi = \psi^{-1}$ is an invertion of ψ in some set of vectors $R = \{\mathbf{y} = f(\mathbf{c}, \mathbf{e}) : \mathbf{c} \in A, \mathbf{e} \in E\}$ which is called correctable vectors: $\xi(\mathbf{y}) = \mathbf{u}$ for $\mathbf{y} \in R$. In this notation R may not be in V. E is some subset of R which is called a set of correctable patterns of errors and $f(\mathbf{x}, \mathbf{e})$ is a some function which depends on channel of information transmission, for instance $f(\mathbf{x}, \mathbf{e}) = -2\mathbf{x} + \mathbf{1} + \mathbf{e}$, where $\mathbf{e}$ is a random variable distributed according to the normal distribution law $N(0, \sigma^2)$. This function $f(\mathbf{x}, \mathbf{e})$ is corresponded to channel with Additive White Gaussian Noise (AWGN) with BPSK manipulation.

Any channel is decribed by two sets: a set $\mathcal{X}$ of inputs (transmitted codewords), a set $\mathcal{Y}$ of outputs (received words) and conditional probability function $p(\mathbf{x} \in \mathcal{X} | \mathbf{y} \in \mathcal{Y})$ to have input $\mathbf{x}$ for given output $\mathbf{y}$. The simpliest decoding function $\xi = \psi^{-1}$ can be described as follows:

$$\xi(\mathbf{y}) = \underset{\mathbf{x} \in \mathcal{X}}{\operatorname{argmax}}\, p(\mathbf{x}|\mathbf{y}).$$

This decoding rule is known as maximum-likelihood (ML) and it gives an optimal solution $\mathbf{x}$ but has an exponential complexity $O(|A|)$. In the next section we describe an idea of supercodes decoding of any code A which has almost the same performance (in terms of cardinality of E) but the complexity is significantly smaller (but in general still remains exponential).

4. DECODING BASED ON SUPERCODES

The first paper where supercode decoding was considered is Barg et al. [1999]. The complexity and the performance of this algorithm were also studied. It was also shown that asymptotic complexity of supercode decoding is exponentially smaller than the complexity of all other methods known. At the same time this algorithm performs complete minimum-distance decoding for almost all long linear codes. This algorithm develops the ideas of covering-set decoding and split syndrome decoding. Here we only describe the main idea of this algorithm that will be sufficient to obtain LDPC decoding based on supercodes.

Let us consider linear (n, k) code $A \subset V$. Let us also consider a set of supercodes $A_i \subset V$: $A \subset A_i$. We will assume that any code A_i has simplier decoding ξ_{A_i} than decoding ξ_A of code A. This assumption is rather natural, since each code A_i have less redundant symbols (smaller syndrome size) thus can be decoded by Viterbi algorithm on code trellis with smaller number of states than in original code A. Moreover, we will also suppose that each decoder ξ_{A_i} produces a list of codewords L_i.

Now let us describe supercodes decoding itself. The input of the algorithm is as follows:

- System of supercodes A_i, $i = 1..s$
- Received from channel vector $\mathbf{y}$

The output of the algorithm is a either such $\mathbf{x}$: $\mathbf{H}\mathbf{x}^T = \mathbf{0}$ or denial of decoding.

The decoding steps are as follows:

- Form a list L_i, $L_i = \xi_{A_i}(\mathbf{y})$, $i = 1..s$.
- Find an intersection $L = \cap L_i$
- If $L = \emptyset$, then return denial of decoding
- Else find $\mathbf{x} = \operatorname{argmax}_{\mathbf{x} \in L} p(\mathbf{x}|\mathbf{y})$ and return $\mathbf{x}$.

In fact lists L_i are not required to include codewords of A_i themselves. Instead of codewords of supercode A_i list L_i can include only some distribution on $\mathcal{X}$ that was calculated from initial distribution obtained from channel (in the case of soft values of $\mathbf{y}$) by decoding algorithm ξ_{A_i}. For instance well-known belief-propagation (BP) decoder of LDPC codes or BCJR decoder for codes on trellis can produce output distribution on $\mathcal{X}$ after several decoding iterations. The main issue in this case is to calculate $L = \cap L_i$. We will show that for LDPC codes this calculation is equivalent to vertical step of BP decoder for generalized LDPC codes.

5. QUASI-CYCLIC LDPC CODES

In this section we will give a brief introduction to quasi-cyclic LDPC codes that will be the main part of our secret sharing scheme. First of all let us define arbitrary quasi-cyclic codes.

Definition 1. Linear code A of length n is a quasi-cyclic (QC) if there is some integer n_0 such that every right/left cyclic shift of any codeword $\mathbf{c} \in A$ in n_0 places is again a codeword of A: $x^{n_0 m}\mathbf{c} \mod (x^n - 1) \in A$ for any $m \in \mathbb{N}$.

If $n = n_0 p$ then both basis matrices $\mathbf{G}$ and $\mathbf{H}$ can be constructed by $p \times p$ circulant blocks.

Definition 2. Square matrix $\mathbf{D}$ is called circulant matrix in all their rows (columns) $\mathbf{d}_i$, $i > 1$ are distinct cyclic shifts of first row (column) $\mathbf{d}_1$. Thus, this matrix are completely defined by it's first row (column).

Now let us define quasi-cyclic LDPC codes.

Definition 3. A linear code A of length n is called (regular) Quasi-Cyclic Low-Density Parity-Check Code (QC-LDPC) if:

- A is a quasi-cyclic.
- $\mathbf{H}$ can be represented as follows:

$$\mathbf{H} = \begin{pmatrix} \mathbf{D}_{11} & \mathbf{D}_{12} & \ldots \mathbf{D}_{1n_0} \\ \mathbf{D}_{21} & \mathbf{D}_{22} & \ldots \mathbf{D}_{2n_0} \\ \ldots & \ldots & \ldots \\ \mathbf{D}_{l1} & \mathbf{D}_{l2} & \ldots \mathbf{D}_{ln_0} \end{pmatrix},$$

  where $\mathbf{D}_{ij}$ are $p \times p$ circulant matrices with row (column) weights w_{ij}, $w_{ij} << p$, $1 < l < n_0 << p$.

The main feature of QC-LDPC codes is that the total number of ones $p \sum_{i,j} w_{ij}$ in $\mathbf{H}$ must be significantly smaller than the total number of elements $l n_0 p^2$ in $\mathbf{H}$. In the most common cases $1 \leq w_{ij} \leq 3$.

If numbers of unities in each column and row of $\mathbf{H}$ are constants: l and n_0 then QC-LDPC code is called (l, n_0)-regular.

"Sparseness" of $\mathbf{H}$ allows to implement low-complexity iterative decoding for recovering codewords of A from received noisy data. In the next section we will describe a main idea of well-known BP decoding of QC-LDPC codes.

6. SOFT DECODING OF LDPC CODES BASED ON SUPERCODES

Let us consider some arbitrary LDPC code $\mathbf{A}$ of length n with parity-check matrix $\mathbf{H}$ with size $m \times n$. Each row of $\mathbf{H}$ will be denoted by $\mathbf{c}_i$ and will be considered by a set of indices j, $1 \leq j \leq n$ such that $h_{ij} = 1$. In fact, each row $\mathbf{c}_i$ is a single parity-check code (SPC) of length $wt(\mathbf{c}_i) = n_i$ such that $(\mathbf{c}_i, \mathbf{v}) = 0 \mod 2$ for any $\mathbf{v} \in A$. We will call $\mathbf{c}_i$ as a *check node.*

With each code symbol (we also call it *variable node*) v_i, $i \in [n]$ we will assign set $C_i = \{(\mathbf{c}_{i_1}, \ldots, \mathbf{c}_{i_t}) : h_{i_1 i}, h_{i_2 i}, \ldots h_{i_t i} = 1\}$ of SPC codes, connected with symbol v_i. At the same manner we denote a set $V_j = \{(i_1, \ldots, i_k) : h_{ji_1}, h_{ji_2}, \ldots, h_{ji_1} = 1\}$ of symbols that are connected with j-th check-node $\mathbf{c}_j$, $j \in [m]$.

Let us describe a general class of decoding algorithms for LDPC codes. These algorithms are called message passing algorithms, and are iterative algorithms. The reason for their name is that at each round of the algorithms messages are passed from variable nodes to check nodes, and from check nodes back to variable nodes. The messages from variable nodes to check nodes are computed based on the observed value of the message node and some of the messages passed from the neighboring check nodes to that variable node. An important aspect is that the message that is sent from a variable node v_i to a check node $\mathbf{c}_j \in C_i$ must not take into account the message sent in the previous round from $\mathbf{c}_j$ to $v_i \in V_j$. The same is true for messages passed from check nodes to variable nodes.

One important subclass of message passing algorithms is the belief propagation algorithm. This algorithm is present in Gallager's work Gallager [1963]. The messages passed between variable and check nodes in this algorithm are probabilities, or beliefs. More precisely, the message $m_{v_i \mapsto \mathbf{c}_j}$ passed from a variable node v_i to a check node $\mathbf{c}_j \in C_i$ in l-th iteration is the probability that v_i has a certain value given the observed value y_i of that variable node, and all the values connected to v_i in the prior round $l-1$ from check nodes from $C_i/\mathbf{c}_j$: $m_{v_i \mapsto \mathbf{c}_j} = Pr(v_i | y_i, C_i, l-1)$. On the other hand, the message $m_{\mathbf{c}_j \mapsto v_i}$ passed from $\mathbf{c}_j$ to $v_i \in V_j$ is the probability that $\mathbf{c}_j$ has a certain value given all the messages passed to $\mathbf{c}_j$ in the previous round $l-1$ from V_j/i.

It is easy to derive formulas for these probabilities under independence assumption. It is sometimes advantageous to work with likelihoods, or sometimes even log-likelihoods $\ln \frac{Pr(x_i=0)}{Pr(x_i=1)}$ instead of probabilities. In this case the decoding algorithm is as follows:

(1) If $l = 0$ then $m^{(l)}_{v_i \mapsto \mathbf{c}_j} = y_i$, where y_i are log-likelihoods received from channel, $i \in [n]$, $j \in [m]$, goto 3.

(2) If $l > 0$, then $m^{(l)}_{v_i \mapsto \mathbf{c}_j} = y_i + \sum\limits_{\mathbf{c}_{j'} \in C_i/\mathbf{c}_j} m^{(l-1)}_{\mathbf{c}_{j'} \mapsto v_i}$, $i \in [n]$, $j \in [m]$

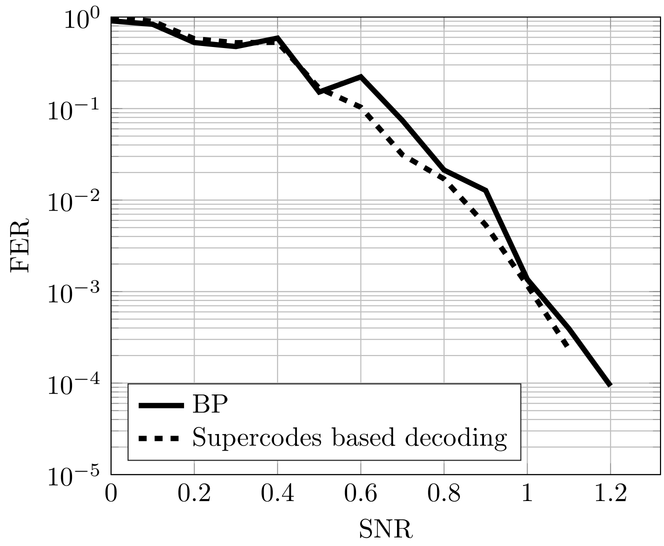

Fig. 1. Decoding of $(9, 15)$-regular QC-LDPC codes of length $n = 1920$

(3) $m^{(l)}_{\mathbf{c}_j \mapsto v_i} = \ln \dfrac{1 + \prod\limits_{v_{i'} \in V_j/i} \tanh \frac{m^{(l)}_{v_{i'} \mapsto \mathbf{c}_j}}{2}}{1 - \prod\limits_{v_{i'} \in V_j/i} \tanh \frac{m^{(l)}_{v_{i'} \mapsto \mathbf{c}_j}}{2}}$, $i \in [n]$, $j \in [m]$

(4) $r^{(l)}_i = y_i + \sum\limits_{\mathbf{c}_{j'} \in C_i} m^{(l-1)}_{\mathbf{c}_{j'} \mapsto v_i}$, $i \in [n]$.

(5) If $r^{(l)}_i < 0$ then $x_i = 1$ else $x_i = 0$, $i \in [n]$.

(6) If $\mathbf{H}\mathbf{x}^T = \mathbf{0}$ then return $\mathbf{x}$ and exit. Else goto 7.

(7) $l := l + 1$.

(8) If $l > l_{max}$ (predefined maximal number of iterations), return denial of decoding. Else goto 2.

This decoding algorithm can be obviously represented in terms of supercodes decoding under assumption that instead of decoding of SPC codes $\mathbf{c}_j$ in stage (3) and updating information from variable nodes to check nodes in stages (1) and (2) algorithm decodes a sequence of supercodes A_j such that $V_{A_j} = [n]$, i. e. supercode A_j consists of such SPC codes $\mathbf{c}_t$ that $\cup V_{\mathbf{c}_t} = [n]$. In this case at stage (3) any message $m_{A_j \mapsto \mathbf{v}}$ updates all log-likelihoods of received word $\mathbf{y}$.

Let us assume that parity-check matrix $\mathbf{H}$ of LDPC code is represented as:

$$\mathbf{H} = \begin{pmatrix} \mathbf{H}_1 \\ \mathbf{H}_2 \\ \ldots \\ \mathbf{H}_t \end{pmatrix}$$

and each $\mathbf{H}_i$ is a parity-check matrix of supercode A_i. For instance, if we consider QC-LDPC code then each block row $(\mathbf{D}_{i1}\mathbf{D}_{i2} \ldots \mathbf{D}_{in_0})$ can be suggested as parity-check matrix $\mathbf{H}_i$ of supercode A_i. In this case the supercodes-based decoding of QC-LDPC codes can be described as in Alg. 1.

Decoder 1 threats LDPC code as a generalized LDPC with constituent codes themselves being LDPC. Simulation results for $(9, 15)$-regular QC-LDPC codes for both original and proposed decoders are presented in Fig. 1.

Input: $A_1, A_2, \ldots, A_t$ — supercodes
$\mathbf{y} = (y_1, y_2, \ldots, y_n)$ — log-likelihoods, received from the channel,
$\mathbf{r}' = \mathcal{A}(A, \mathbf{r}, I_{\text{in}})$ — iterative decoding algorithm of code A where, $\mathbf{r}$ — input log-likelihoods , $\mathbf{r}'$ — output log-likelihoods, I_{in} is the number of iterations, $I_{max} = I_{\text{in}} I_{\text{out}}$ — total number of decoding iterations
Output: $\mathbf{L}$ — output log-likelihoods after decoding.

```
/* Initialization                                    */
L ← y
for i = 1,t do
 | L_i ← 0
end
for j = 1,I_out do
  for i = 1,t do
    r ← L − L_i
    r' ← A(A_i, r, I_in)
    L'_i ← r' − r
  end
  for i = 1,t do
   | L_i ← L'_i
  end
  L ← Σ_{i=1}^{t} L_i + y
end
return L
```

Algorithm 1: Proposed Decoder

In this picture solid line corresponds to traditional BP decoding of LDPC code, and dashed one corresponds to supercodes decoding. The total number of iterations is 50 ($I_{in} = 5$ and $I_{out} = 10$ for supercodes decoding). The communication channel is AWGN with BPSK manipulation. Simulation shows the same performance as usual BP decoding.

7. SECRET-SHARING SCHEME BASED ON LDPC CODES

Before we are going to describe secret sharing scheme based on LDPC codes, we now can give a main idea of one. In the supercodes decoding scheme a number s of supercodes A_i, $= 1..s$ can be made arbitrary. Let us suppose that codeword $\mathbf{x}$ of code $\mathbf{A}$ is a secret. If we assume that both communication channels between dealer and participants, and between participants are noiseless, then dealer can generate such noise vector $\mathbf{e}$ that in order to decode received sequence $\mathbf{y} = f(\mathbf{x}, \mathbf{e})$ any subset of $l < s$ supercodes A_{i_j}, $j = 1..l$ is necessary and sufficient: it means that $\mathbf{x}$ can be recovered from any set of $l_1 \geq l$ supercodes and can not be recovered from any set of supercodes with cardinality smaller than l. Thus each of participants have a pair $(\mathbf{y}, A_i)$ and only coalition of l or more participants can recover $\mathbf{x}$ from $\mathbf{y}$.

Let us suppose some LDPC code A with parity-check matrix $\mathbf{H}$ in the form:

$$\mathbf{H} = \begin{pmatrix} \mathbf{H}_1 \\ \mathbf{H}_2 \\ \ldots \\ \mathbf{H}_t \end{pmatrix}.$$

Let us also assume that there are $l \leq U \leq t$ users in the secret sharing scheme. The common secret, distributed between these users is a codeword $\mathbf{x}$ of LDPC code with parity-check matrix $\mathbf{H}$. Since all codes A_i with parity-check matrices $\mathbf{H}_i$ are supercodes of A then $\mathbf{x}\mathbf{H}_i^T = \mathbf{0}$ for all $i = 1..t$. But since codes A_i have rates higher than rate of A then these codes can correct less errors than code A. This fact is a basis of the secret sharing scheme.

Let us suppose that dealer can generate Additive White Gaussian Noise (AWGN) with arbitrary variance σ and zero mean: $N(0, \sigma)$. Let us also suppose that for a given code A dealer knows variance σ_{crit} that allows to decode any coalition of supercodes A_1, A_2, …, A_l of cardinality l with error probability smaller than P_e, where P_e is small enough. Moreover, let us also assume that dealer also knows noise variance $\sigma_s > \sigma_{crit}$ such that for any coalition of supercodes A_1, A_2, …, $A_{l'}$, $l' < l$ probability of error close to $1 - P_e$. The values σ_s, σ_{crit} can be obtained, from instance, using Density Evolution (DE) technique, described in Luby et al. [2001]. In this case the secret sharing scheme can be described as follows:

- *Secret generation.*
  - Encode vector $\mathbf{u}$ by generator matrix $\mathbf{G}$ of QC-LDPC code: $\mathbf{x} = \mathbf{uG}$
  - Add noise to modulated $\mathbf{x}$: $\mathbf{y} = \mathbf{2x} - \mathbf{1} + \eta$, where $\eta \sim N(0, \sigma_{crit})$.
  - Calculate log-likelihoods: $\mathbf{L} = \frac{2\mathbf{y}}{\sigma_{crit}^2}$
- *Secret sharing*
  - Dealer sends to each user $1 \leq i \leq U$ a pair $(\mathbf{H}_i, \mathbf{L})$. In the case of QC-LDPC codes instead of sending whole matrix $\mathbf{H}_i$ dealer can only send a sequence $\mathbf{d}_{i1}$, …, $\mathbf{d}_{in_0}$ of the first rows of circulants $\mathbf{D}_{i1}, \mathbf{D}_{i2}, \ldots, \mathbf{D}_{in_0}$ thus reducing lengths of keys.
- *Secret recovering*
  - If there are any coalition of $k \geq l$ users $i_1, i_2, \ldots i_k$, then construct a parity-check matrix $\mathbf{H}_c$:
  $$\mathbf{H}_c = \begin{pmatrix} \mathbf{H}_{i_1} \\ \mathbf{H}_{i_2} \\ \ldots \\ \mathbf{H}_{i_k} \end{pmatrix}$$
  and decode the corresponding code by Alg. 1 to recover $\mathbf{x}$ from $\mathbf{L}$

This scheme guarantees that codeword $\mathbf{x}$ will be recovered from $\mathbf{L}$ by any coalition of users that includes not less than l members with probability not less than $1 - P_e$ but if the number of users in coalition smaller than l then the probability of $\mathbf{x}$ to be recovered is at most P_e for any predefined P_e.

8. CONCLUSION

In this paper we propose a new scheme of secret sharing based on iterative decoding of LDPC codes in terms of supercodes decoding concept. This scheme can be generalized for an arbitrary number of users in the case when we allow to supercodes being intersected. This scheme is field size and secret length independent.

REFERENCES

M. Ben-Or, S. Goldwasser, and A. Wigderson. Completeness theorems for non-cryptographic fault-tolerant distributed computations. In Proc. of the 20th ACM Symp. on the Theory of Computing, pages 1–10, 1988.

D. Chaum, C. Crepeau, and I. Damgard. Multiparty unconditionally secure protocols. In Proc. of the 20th ACM Symp. on the Theory of Computing, pages 11—19, 1988.

R. Cramer, I. Damg åard, and U. Maurer. General secure multi-party computation from any linear secret-sharing scheme. In B. Preneel, editor, Advances in Cryptology – EUROCRYPT 2000, volume 1807 of *Lecture Notes in Computer Science,* pages 316—334. Springer-Verlag, 2000.

Y. Desmedt and Y. Frankel. Shared generation of authenticators and signatures. In J. Feigenbaum, editor, Advances in Cryptology – CRYPTO '91, volume 576 of *Lecture Notes in Computer Science,* pages 457-–469. Springer-Verlag, 1992.

M. Naor and A. Wool. Access control and signatures via quorum secret sharing. *IEEE Transactions on Parallel and Distributed Systems,* volume 9, no. 1, pages 909–922, 1998.

V. Goyal, O. Pandey, A. Sahai, and B. Waters. Attribute-based encryption for fine-grained access control of encrypted data. In Proc. of the 13th ACM conference on Computer and communications security, pages 89–98, 2006.

B. Waters. Ciphertext-policy attribute-based encryption: An expressive, efficient, and provably secure realization. Technical Report 2008/290, Cryptology ePrint Archive, 2008. *http://eprint.iacr.org/.*

G. R. Blakley. Safeguarding cryptographic keys. In R. E. Merwin, J. T. Zanca, and M. Smith, editors, Proc. of the 1979 AFIPS National Computer Conference, volume 48 of AFIPS Conference proceedings, pages 313—317. AFIPS Press, 1979.

A. Shamir. How to share a secret. *Communications of the ACM,* volume 22, pages 612–613, 1979.

M. Ito, A. Saito, and T. Nishizeki. Secret sharing schemes realizing general access structure. In Proc. of the IEEE Global Telecommunication Conf., Globecom 87, pages 99–102, 1987. Journal version: Multiple assignment scheme for sharing secret. *J. of Cryptology,* volume 6, no. 1, pages 15–20, 1993.

J. Benaloh and J. Leichter. Generalized secret sharing and monotone functions. In S. Goldwasser, editor, Advances in Cryptology – CRYPTO '88, volume 403 of *Lecture Notes in Computer Science*, pages 27–35. Springer-Verlag, 1990.

M. Karchmer and A. Wigderson. On span programs. In Proc. of the 8th IEEE Structure in Complexity Theory, pages 102–111, 1993.

E. F. Brickell. Some ideal secret sharing schemes. *Journal of Combin. Math. and Combin. Comput.*, vol. 6 pages 105–113, 1989.

R.J. McEliece and D.V. Sarwate On sharing secrets and Reed-Solomon codes *Comm. ACM 24*, pages 583–584, 1981.

J.L. Massey Minimal codewords and secret sharing Proc. 6th Joint Swedish-Russian Workshop on Information Theory, pages 276–279, 1993.

J.L. Massey Some applications of coding theory in cryptography *Codes and Ciphers: Cryptography and Coding IV*, Formara Ltd, Esses, England, pages 33–47, 1995.

R. G. Gallager Low-Density Parity-Check Codes *MIT Press*, Cambridge, MA, 1963.

R. J. McEliece A public-key cryptosystem based on algebraic Coding Theory *DSN Progress Report*, pages 114—116, 1978.

M. Baldi, F. Chiaraluce, R. Garello, and F. Mininni Quasi-cyclic low-density parity-check codes in the McEliece cryptosystem In 2007 IEEE International Conference on Communications, pages 951—956, 2007.

A. Abramov, E. Krouk Generalized supercodes decoding 2014 XIV International Symposium on Problems of Redundancy in Information and Control Systems. *IEEE*, pages. 1–5, 2014.

A. Barg, E. Krouk and H. C. van Tilborg On the complexity of minimum distance decoding of long linear codes. *IEEE Transactions on Information Theory*, volume 45, no. 5, pages 1392–1405, 1999.

M. G. Luby, M. Mitzenmacher, M. A. Shokrollahi, and D. A. Spielman Improved low-density parity-check codes using irregular graphs *IEEE Trans. Inform. Theory*, volume 47, no. 2, pages 585–598, 2001.