On Iterative Decoding of HDPC Codes Using Weight-Bounding Graph Operations

Joakim Grahl Knudsen, Constanza Riera*, Lars Eirik Danielsen, Matthew G. Parker, and Eirik Rosnes
Dept. of Informatics, University of Bergen, Thormøhlensgt. 55, 5008 Bergen, Norway
email: {joakimk, larsed, matthew, eirik}@ii.uib.no
*Bergen University College, Nygårdsgt. 112, 5008 Bergen, Norway, email: csr@hib.no

***Abstract*—In this paper, we extend our work on iterative soft-input soft-output (SISO) decoding of high density parity check (HDPC) codes. Edge-local complementation (ELC) is a graph operation which can be used to give structural diversity during decoding with the sum-product algorithm (SPA). We describe the specific subgraphs required for ELC to not increase the weight of the Tanner graph beyond a specified upper bound. We call this controlled operation weight-bounding ELC (WBELC). A generalized iterative SISO HDPC decoder based on SPA decoding is described, which can be configured to employ our SPA-ELC decoders, or iterative permutation decoding (SPA-PD). The latter is a state-of-the-art decoding algorithm for HDPC codes, using permutations from the automorphism group of the code. We observe performance improvements over SPA-PD when the SISO HDPC decoder is configured to use SPA-ELC in conjunction with WBELC.**

I. Introduction

Iterative soft decision decoding algorithms are known to give results which approach the theoretical limits postulated by Shannon [1]. Specifically, the use of such algorithms for the decoding of random, sparse linear codes yields near-optimum error-rate performance when the blocklength goes to infinity. The best known instance is low density parity check codes, decoded with the sum-product algorithm (SPA). Inspired by these results, the aim of much research has been to develop practical (non-asymptotic) codes and decoders exhibiting comparable performance. Recently, iterative decoding techniques have been adapted to classical linear codes, which have strong structural properties (large minimum distance, and small description complexity in hardware implementation), but are non-sparse. One state-of-the-art decoder for such *high density parity check* (HDPC) codes [2] is the iterative permutation decoder (SPA-PD) [3], which performs very well on Bose-Chaudhuri-Hocquenghem codes, as well as on quadratic residue codes [4, 5], over the additive white Gaussian noise (AWGN) channel. Our paper is an extension of our previous work on iterative, graph-local decoding of HDPC codes using a graph operation known as edge-local complementation (ELC) [5, 6]. The contribution of this work is the description of subgraphs on which ELC will not increase the number of edges in the graph beyond a desired threshold–a trait we call weight-bounding ELC (WBELC). We describe an SPA-WBELC algorithm – an instance of a generalized soft-input soft-output (SISO) HDPC decoder – which gives an improvement over our previous algorithm, SPA-ELC [5].

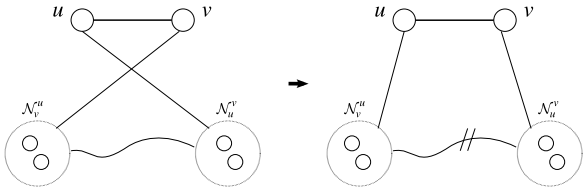
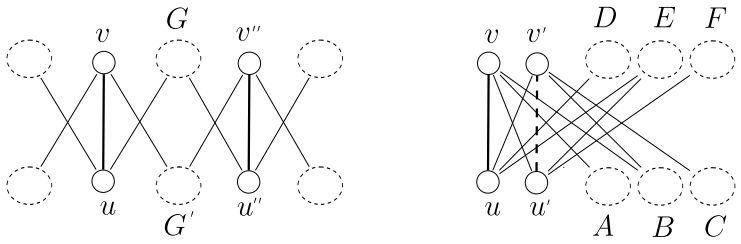


Fig. 1. ELC on edge (u, v) of a bipartite simple graph. Doubly slashed links mean that the edges connecting the two sets have been complemented; edges are replaced by non-edges, and vice versa. This graph may be a subgraph of a larger graph.

We also extend our scope towards less structured HDPC codes (i.e., smaller automorphism group), for which we also observe an improvement over SPA-PD. Most significantly, we show a gain when the size of the automorphism group is one–moving towards random codes–in which case SPA-PD 'reduces' to SPA.

A binary linear code $\mathcal{C}$ of length n and dimension k is denoted by $[n, k, d_{\mathbf{min}}]$, and $\mathcal{C}^\perp$ is its dual. The automporhism group is denoted by $\mathrm{Aut}(\mathcal{C})$, and if it consists of the identity permutation alone, we say that $\mathrm{Aut}(\mathcal{C})$ is trivial. The $(n-k) \times n$ parity check matrix and the corresponding Tanner graph are denoted by H and $\mathbf{TG}(H)$, respectively. All definitions regarding H have obvious equivalents for $\mathbf{TG}(H)$, and vice versa, so we will use these representations interchangeably. H is said to be systematic if its columns can be reordered into the form $[I\, P]$, where I is the identity matrix of size $n-k$. The transpose of H is written H^T. The weight of H, denoted by $|H|$, is the number of non-zero entries in H, and the minimum weight of H is lower-bounded by $\max(k(d_{\mathbf{min}}(\mathcal{C})-1)+n-k,\ (n-k)d_{\mathbf{min}}(\mathcal{C}^\perp))$. Accordingly, the number of edges of $\mathbf{TG}(H)$ is $|H|$. The local neighborhood of a node v is the set of nodes adjacent to v, and is denoted by $\mathcal{N}_v$, while $\mathcal{N}_v^u$ is shorthand notation for $\mathcal{N}_v \backslash \{u\}$. $|\mathcal{E}_{A,B}|$ denotes the number of edges in the subgraph induced by the nodes in $A \cup B$. $\mathcal{E}_{u,v}$ is shorthand notation for $\mathcal{E}_{\mathcal{N}_u^v, \mathcal{N}_v^u}$, the local neighborhood of the edge (u, v). ELC requires that H is systematic, so, as a simplification, we may describe the subgraphs on which ELC is WBELC using a simple bipartite graph (undirected, no double edges) $G = \left(\begin{smallmatrix} 0 & P \\ P^T & 0 \end{smallmatrix}\right)$. By taking the P-part as one of the two partitions, G is equivalent to $\mathbf{TG}(H)$, and straight-forward mappings exist to implement ELC operations directly on $\mathbf{TG}(H)$ [5]. The operation of ELC on an edge (u, v) is to complement the edges of $\mathcal{E}_{u,v}$, followed by swapping the nodes u and v–see Fig. 1. In the following,



(a) Theorem 4 (b) Theorem 5

Fig. 2. Depth-2 WBELC. The dashed edge in Theorem 5 is a non-edge. The edges between sets in the (bipartite) subgraphs are not shown.

we will use boldface notation for vectors.

The following section describes WBELC. The remainder of this article details the application of this controlled ELC operation in a SISO HDPC decoding algorithm, in an extension of our previous work on the SPA-ELC decoder. Finally, we present simulations results, and compare the decoding algorithms.

II. WBELC

The effect of repeated random ELC is that the average weight of H tends to $\frac{k(n-k)}{2} + (n-k)$. In this section, we introduce a restriction on the ELC operation, being that an ELC on a certain edge in the graph is only allowed if $|H|$ remains below a given threshold, T. We give a complete description of the conditions that are necessary and sufficient in order to achieve this bound, both for a single ELC and for two consecutive ELCs. Using these conditions, we improve the perormance of the SISO HDPC decoder.

We begin by formalizing the notion of WBELC. If the weight change due to the complementation caused by ELC is bounded, then the weight of the entire graph is bounded, and we say that the ELC is WBELC.

A. Depth 1

There is a simple condition for one ELC to be WBELC.

Theorem 1: ELC on (u, v) does not increase the weight of the graph by more than a threshold T *iff*

$$|\mathcal{N}_u^v||\mathcal{N}_v^u| - 2|\mathcal{E}_{u,v}| \leq T.$$

B. Depth 2

For many graphs, it is simply not possible to bound the weight increase by any reasonable threshold using only a single ELC. The notion of WBELC can be extended to the case of consecutive ELC operations. In this work, we will completely characterize WBELC to within depth 2, where we use the compact notation $\{(u, v), (u', v')\}$ for an ordered pair of edges. Incidentally, the search space can be significantly reduced from checking all pairs of edges in the graph.

Theorem 2: ELC on $\{(u, v), (v, v')\}$, where $v' \in \mathcal{N}_u^v$, gives the same graph as ELC on (u, v'). Consequently, depth-2 WBELC reduces in this case to depth-1 WBELC.

Note that, due to the swap of ELC on (u, v), (v, v') and (u, v') refer to the same edge–see Fig. 1. From this theorem, we see that we need only consider pairs of non-adjacent edges, i.e., at a distance of at least one edge apart. However, it can be shown that the search space can be further reduced by noting that the distance can also not be greater than two edges.

Theorem 3: Let $T \geq -1$. Any depth-2 WBELC where the pair of edges are at a distance greater than two edges apart, will always reduce to either one or two separate instances of depth-1 WBELC.

One implication of Theorem 3 is that depth-2 WBELC, like depth-1 WBELC, only acts locally on a graph. For $T < -1$, there is an additional case (not discussed in this paper), not covered by Theorem 3. Thus, the following three theorems describe all possible depth-2 WBELC cases for $T \geq -1$.

Let us first consider the case where the pair of edges are at a distance of exactly two edges apart, Fig. 2(a). Given an edge (u, v), let $u'', v'' \notin \mathcal{N}_u \cup \mathcal{N}_v$ be such that $G = \mathcal{N}_u^v \cap \mathcal{N}_{u''}^{v''} \neq \emptyset$, and, similarily, $G' = \mathcal{N}_{v''}^{u''} \cap \mathcal{N}_v^u \neq \emptyset$.

Theorem 4: ELC on $\{(u, v), (u'', v'')\}$ does not increase the weight of the graph by more than a threshold T *iff*

$$\begin{aligned}&|\mathcal{N}_u^v||\mathcal{N}_v^u| + |\mathcal{N}_{u''}^{v''}||\mathcal{N}_{v''}^{u''}| - 2|\mathcal{E}_{u,v}| \ + \\ &4|\mathcal{E}_{G,G'}| - 2|\mathcal{E}_{\mathcal{N}_{u''}^{v''},\mathcal{N}_{v''}^{u''}}| - 2|G||G'| \ \leq T.\end{aligned}$$

For the next theorem, given an edge (u, v) and two nodes u' and v', we denote by $B = N_v^{u,u'} \cap N_{v'}^{u,u'}$, $A = N_v^{u,u'} \setminus B$, $C = N_{v'}^{v,v'} \setminus B$, $E = N_u^{v,v'} \cap N_{u'}^{v,v'}$, $D = N_u^{v,v'} \setminus E$, and $F = N_{u'}^{v,v'} \setminus E$, see Fig. 2(b).

We now consider the case where both u' and v' are in the neighborhood of (u, v).

Theorem 5: ELC on $\{(u, v), (u', v')\}$ does not increase the weight of the graph by more than a threshold T *iff*

$$\begin{aligned}&|F| - |E| - |B| - 2|\mathcal{E}_{A,E\cup F}| - 2|\mathcal{E}_{B,D\cup E}| - 2|\mathcal{E}_{C,D\cup F}| \ + \\ &|C| + |A|(|E| + |F|) + |B|(|D| + |E|) + |C|(|D| + |F|) \leq T.\end{aligned}$$

Note that the edge (u', v') is created by the first ELC. Last, we consider the case where either u' or v' belong to $\mathcal{N}_u^v \cup \mathcal{N}_v^u$, but not both. Without loss of generality, let $v' \in \mathcal{N}_u^v$ be connected to $u' \notin \mathcal{N}_v^u$.

Theorem 6: ELC on $\{(u, v), (u', v')\}$ gives the same graph as ELC on $\{(u, v'), (u', v)\}$.

Note that $\{(u, v'), (u', v)\}$ is covered by Theorem 5.

III. Iterative SISO HDPC Decoding

We have previously described the SPA-ELC decoder, which, essentially, consists of SPA iterations interspersed with random ELC operations [5]. Since ELC complements edges, we avoid

loss of extrinsic information (on edges) by executing a *flooding* scheduling SPA iteration in the order 'functions, then variables.' At this point, all messages, μ, have been accumulated in variable nodes, making it safe to change the graph. A generalized SISO HDPC decoder is listed in Algorithm 1, which can be configured to perform the decoding algorithms compared in this work–see Section IV.

Both SPA-PD and SPA-ELC suffer a performance loss if the extrinsic contribution of the soft input vector, $\mathbf{L}$, is not scaled down (damped) in between iterations. For each variable node, v, the SPA produces a decision based on two pieces of information; the extrinsic information produced by the decoder, and the input to iteration j, L_j^v. $\mathbf{L}_0$ is the received noisy channel vector and τ is the maximum number of decoder iterations. The damping coefficient, $\alpha_0 \leq \alpha \leq 1$, represents the amount of 'trust' in the extrinsic information versus the input after the current iteration [2], $L_{j+1}^v := L_j^v + \alpha(\Sigma_{u\in\mathcal{N}_v}\, \mu_j^{v\leftarrow u})$, $\forall\, u \in \mathcal{N}_v$. As the decoder converges, the information produced by the graph is assumed to become more reliable (hopefully converging towards the maximum-likelihood codeword), so our trust in the decoder state may be increased accordingly. This is normally reflected by incrementing α with iteration number j. A *global* damping rule (GD) scales down all variable nodes, and re-initializes all edges, $\mu_{j+1}^{v\rightarrow u} := L_{j+1}^v$, $\forall\, v \in \mathbf{TG}(H)$. We propose an *edge-local* damping rule (LD), which restricts the application of the damping-and-initialization rule to new edges due to ELC on (u, v), $\mu_{j+1}^{v'\rightarrow u'} := L_{j+1}^v$, $\forall\, (u', v') \in \mathcal{E}_{u,v}$. All other edges retain messages computed in iteration j.

SPA-PD applies a random permutation (PD) $\mathbf{L}_j := \sigma(\mathbf{L}_j)$, $\sigma \in \mathrm{Aut}(\mathcal{C})$, before re-initializing $\mathbf{TG}(H)$ with global damping. SPA decoding on a fixed graph suffers a performance loss when global damping is applied, which suggests that the benefit of damping is to moderate the effects of modifications (e.g., permutations, Gaussian elimination, ELC) to $\mathbf{TG}(H)$. Note that damping is disabled by configuring $\alpha_0 := 1$.

A. SPA-WBELC

The SPA-WBELC algorithm uses the theorems in Section II to determine a random WBELC operation on the current $\mathbf{TG}(H)$, and applies the corresponding one or two ELC operations, with edge-local damping. Let H_j denote the matrix after j iterations of the SISO HDPC decoder. It is helpful to reduce the weight of the initial matrix, H_0, in a preprocessing stage, as this has a positive effect on SPA decoding. This can be done using repeated random WBELC with $T = -1$, for non-increasing weight. A simple but effective heuristic, if the preprocessing gets stuck, is to allow one random (i.e., unbounded) ELC. Then, for SPA-WBELC decoding, a threshold $T \geq -1$ must be determined, such that WBELC yields a sufficient number of distinct matrices of weight $|H| \leq |H_0| + T$, to give structural diversity during decoding.

IV. Results

The aim of this paper is to explore the effects of ELC decoding, while maintaining a bound on the weight of $\mathbf{TG}(H)$. We will show that the SPA-WBELC decoder outperforms SPA-PD when $\mathrm{Aut}(\mathcal{C})$ is small. For this work, we chose the best codes we could find at practical blocklengths: two extremal (in terms of minimum distance) self-dual $[36, 18, 8]$ and $[38, 19, 8]$ codes from [7], and an extremal double circulant self-dual $[68, 34, 12]$ code from [8]. We use the notation $\mathcal{C}^n$ to refer to these codes, and we have that $|\mathrm{Aut}(\mathcal{C}^n)| \approx n$, except $\mathcal{C}^{38}$ which has a trivial $\mathrm{Aut}(\mathcal{C})$.

Algorithm 1 SISO-HDPC$(p, I_1, I_2, I_3, \alpha_0, \mathrm{OP}, \mathrm{DR})$

```
α = α_0
for I_3 times do
    Restart decoder from channel vector
    for I_2 times do
        Stop if syndrome check is satisfied
        Apply damping rule, DR, with coefficient α
        Apply at random p operations, OP
        for I_1 times do
            Apply SPA iteration ('flooding' scheduling)
        end for
    end for
    Increment α towards 1
end for
```

The matrices used were optimized on weight, both in non-systematic form (for SPA and SPA-PD), as well as systematic form (for SPA-ELC and SPA-WBELC). For $\mathcal{C}^{36}$ and $\mathcal{C}^{38}$, we were able to compute the entire ELC orbit of the codes, to find optimal-weight matrices in systematic form to be $|H_0^{36}| = 156$ and $|H_0^{38}| = 166$. For $\mathcal{C}^{68}$, the orbit is infeasibly large, yet, using WBELC preprocessing, we were able to find a systematic matrix of weight $|H_0^{68}| = 488$. For non-systematic form, minimum-weight codewords of $\mathcal{C}^{\perp}$ were combined to assemble matrices of weight 152, 154, and 492, respectively, which is very close to the lower bound based on $d_{\mathbf{min}}(\mathcal{C}^{\perp})$.

The simulation results compare the proposed SPA-WBELC$(p, I_1, I_2, I_3, \alpha_0, T)$ = SISO-HDPC$(p, I_1, I_2, I_3, \alpha_0$,WBELC$(T)$, LD) decoder against standard SPA(τ) = SISO-HDPC$(0, 1, \tau, 1, 1, -, -)$, where we ensure that $\tau = I_1 I_2 I_3$; SPA-PD$(I_1, I_2, I_3, \alpha_0)$ = SISO-HDPC$(1, I_1, I_2, I_3, \alpha_0, \mathrm{PD}, \mathrm{GD})$; and our previous ELC decoder, SPA-ELC$(p, I_1, I_2, I_3, \alpha_0)$ = SISO-HDPC$(p, I_1, I_2, I_3, \alpha_0, \mathrm{ELC}, \mathrm{LD})$. We compare frame-error rate (FER) when signalling over the AWGN channel, and measure complexity in SPA messages, $\frac{1}{F}\Sigma_F \Sigma_{j=0}^{J\leq\tau} |H_j|$, where J is the number of iterations used for a frame, and F the total number of frames simulated. For comparisons between SPA-ELC and SPA-WBELC, we use a comparative number, p, of ELC operations (one WBELC is one or two ELC operations). The most significant result is that SPA-WBELC outperforms SPA-PD in FER on $\mathcal{C}^{38}$ and $\mathcal{C}^{36}$, even when $\mathrm{Aut}(\mathcal{C})$ is non-trivial. For $\mathcal{C}^{68}$, we approach the performance of SPA-PD quite closely. In addition, we see that SPA-WBELC will generally result in an improvement over SPA-ELC. This gain is consistent for all codes attempted, and is most significant at low signal-to-noise ratio (SNR). At

high SNR, the performance of SPA-WBELC will, in general, approach that of SPA-ELC. This is assumed to be linked to the average number of iterations per frame approaching zero, such that the number of operations (ELC or WBELC) also goes down, diminishing the difference between the respective decoders. The point at which the performance of SPA-WBELC 'breaks off' towards SPA-ELC is influenced by the choice of T. By increasing T, the break occurs at higher SNR. Yet, this is obviously at the expense of increased average weight, such that, for some T sufficiently high, SPA-WBELC equals SPA-ELC also at low SNR.

For $\mathcal{C}^{38}$, $\mathrm{Aut}(\mathcal{C})$ is trivial, such that SPA-PD 'reduces' to SPA. In this extreme setting, ELC-based decoding has its most interesting gain. The SISO HDPC decoder is sensitive to choice of parameters, so various configurations (of T, I_1, I_2, I_3, and p) were systematically attempted in order to arrive at the presented data.

For complexity, we observe the desired effect of bounding the weight increase due to ELC. For SPA-ELC, the average weight of H quickly settles around $k(k+2)/2$ (the codes are self-dual), whereas for SPA-WBELC, the average weight is $|H_0| + T$. The SPA-WBELC decoder has a uniform improvement in complexity over both SPA and SPA-ELC, and can also be pushed down quite close to SPA-PD. We have also simulated SPA and SPA-PD on systematic matrices (not shown), to verify that FER performance is not significantly sensitive to this.

V. Conclusion

We have developed a new algorithm for the decoding of linear codes on graphs, which is particularly suited for HDPC codes. The main idea of this work is to use a graph operation, ELC, in a controlled manner. We described the necessary and sufficient conditions for this operation to be weight-bounding, and discuss its application in SPA decoding. The results show a significant improvement over standard (flooding) SPA, our previous algorithm SPA-ELC, as well as over SPA-PD in codes with limited structure.

References

[1] C. E. Shannon, "A mathematical theory of communication," *Bell Systems Technical Journal*, vol. 27, pp. 379–423, 623–656, 1948.

[2] J. Jiang and K. R. Narayanan, "Iterative soft decision decoding of Reed-Solomon codes," *IEEE Commun. Lett.*, vol. 8, no. 4, pp. 244–246, Apr. 2004.

[3] T. R. Halford and K. M. Chugg, "Random redundant iterative soft-in soft-out decoding," *IEEE Trans. Commun.*, vol. 56, no. 4, pp. 513–517, Apr. 2008.

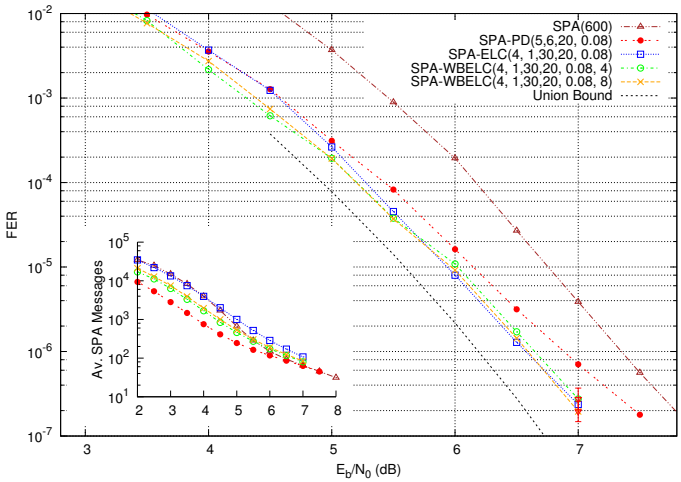
[4] I. Dimnik and Y. Be'ery, "Improved random redundant iterative HDPC decoding," *IEEE Trans. Commun.*, vol. 57, no. 7, pp. 1982–1985, Jul. 2009.

[5] J. G. Knudsen, C. Riera, L. E. Danielsen, M. G. Parker, and E. Rosnes, "Random edge local complementation and iterative soft-decision decoding," in *Proc. Int. Symp. Inform. Theory*, Seoul, Korea, Jul. 2009, pp. 899–903.

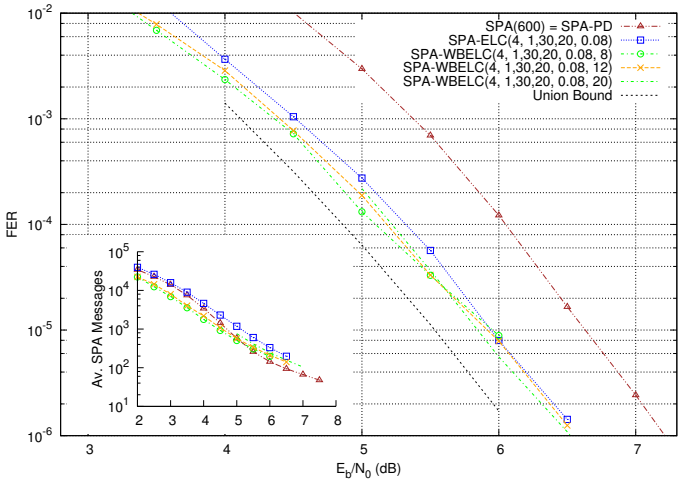
[6] A. Bouchet, "Isotropic systems," *European J. Comb.*, vol. 8, pp. 231–244, Jul. 1987.

[7] M. Harada, "New extremal self-dual codes of lengths 36 and 38," *IEEE Trans. Inform. Theory*, vol. 45, no. 7, pp. 2541–2543, Nov. 1999.

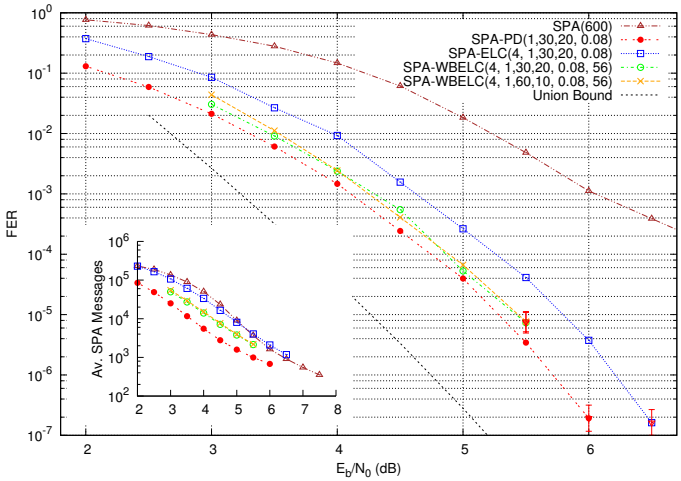
[8] T. A. Gulliver and M. Harada, "Classification of extremal double circulant self-dual codes of lengths 64 to 72," *Des. Codes Cryptogr.*, vol. 13, pp. 257–269, 1998.



(a) $\mathcal{C}^{36} = [36, 18, 8]$, with $|\mathrm{Aut}(\mathcal{C})| = 32$



(b) $\mathcal{C}^{38} = [38, 19, 8]$, with $|\mathrm{Aut}(\mathcal{C})| = 1$



(c) $\mathcal{C}^{68} = [68, 34, 12]$, with $|\mathrm{Aut}(\mathcal{C})| = 68$

Fig. 3. Simulations results. Each SNR point is simulated until at least 100 frame-error events were observed (otherwise, error bars indicate significance). The union bound is calculated based on the full weight enumerator of the code.