Refined Upper Bounds on Stopping Redundancy of Binary Linear Codes

Yauhen Yakimenka, Vitaly Skachek
Institute of Computer Science, University of Tartu, Estonia

yauhen@ut.ee, vitaly@ut.ee

Abstract (read only this if you are lazy)

The l-th stopping redundancy $\rho_l(\mathcal{C})$ of the binary $[n,k,d]$ code $\mathcal{C}$, $1 \le l \le d$, is defined as the minimum number of rows in the parity-check matrix of $\mathcal{C}$, such that the smallest stopping set is of size at least l. The stopping redundancy $\rho(\mathcal{C})$ is defined as $\rho_d(\mathcal{C})$. In this work, we improve on the probabilistic analysis of stopping redundancy, proposed by Han, Siegel and Vardy, which yields the best bounds known today. In our approach, we judiciously select the first few rows in the parity-check matrix, and then continue with the probabilistic method. By using similar techniques, we improve also on the best known bounds on $\rho_l(\mathcal{C})$, for $1 \le l \le d$. Our approach is compared to the existing methods by numerical computations.

Motivation: MP decoding failures

Settings:

- $\mathcal{C} \subset \mathbb{F}_2^n$ is a linear $[n,k,d]$ code, $\mathcal{C}^\perp$ is dual to $\mathcal{C}$
- BEC, Message-passing (MP) decoding

Some facts:

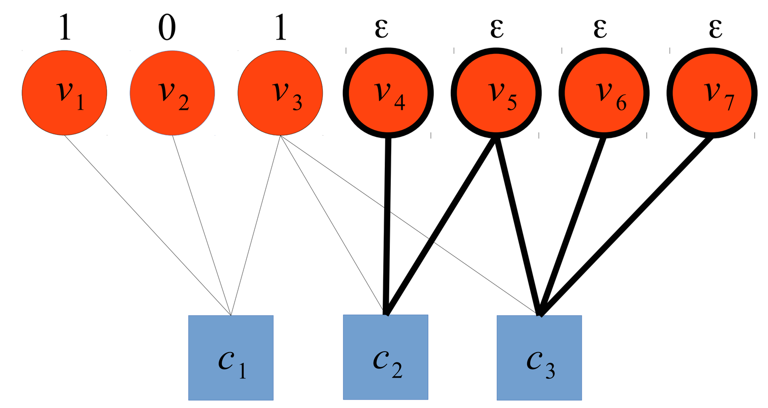
- Decoder fails ⇔ stopping set is erased
- Stopping sets undesirable
- Erasure of stopping sets of small size is more probable
- (!) Stopping sets are defined for H, not for $\mathcal{C}$
- Additional (redundant) rows could eliminate stopping sets
- Every codeword of $\mathcal{C}$ induces stopping set ⇒ those not possible to eliminate

Idea: use redundant H which eliminates all stopping sets of size $< d$

- **Stopping redundancy** $\rho(\mathcal{C})$ is the minimum number of rows in H s.t. there are no stopping of size $< d$
- Always achievable: $\rho(\mathcal{C}) \le 2^{n-k} - 1$

Stopping sets

Definition 1. A *stopping set* in Tanner graph is a subset S of variable nodes such that all the check nodes that are neighbours of a node in S are connected to *at least two nodes* in S.



E.g. $\{v_4, v_5, v_6, v_7\}$ is a stopping set

Definition 2. A *stopping set* is a set of columns of H with the property that the projection of H onto these columns does not contain a row of weight one.

$$H = \begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 \end{bmatrix}$$

Stopping redundancy: example

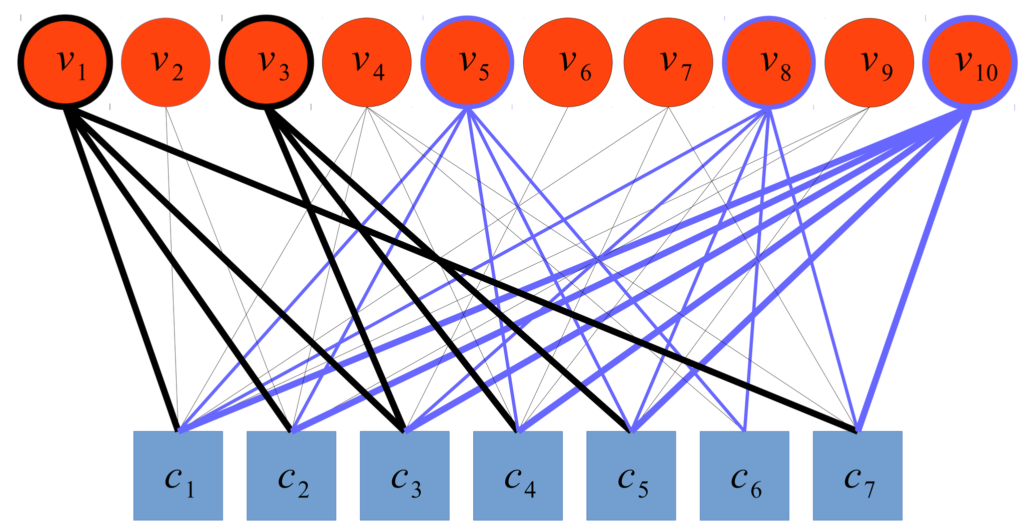
$\mathcal{C} = [10, 3, 4]$ with parity-check matrix H:

$$H = \begin{matrix} c_1 \\ c_2 \\ c_3 \\ c_4 \\ c_5 \\ c_6 \\ c_7 \end{matrix} \begin{pmatrix} 1 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 1 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 1 \end{pmatrix}$$

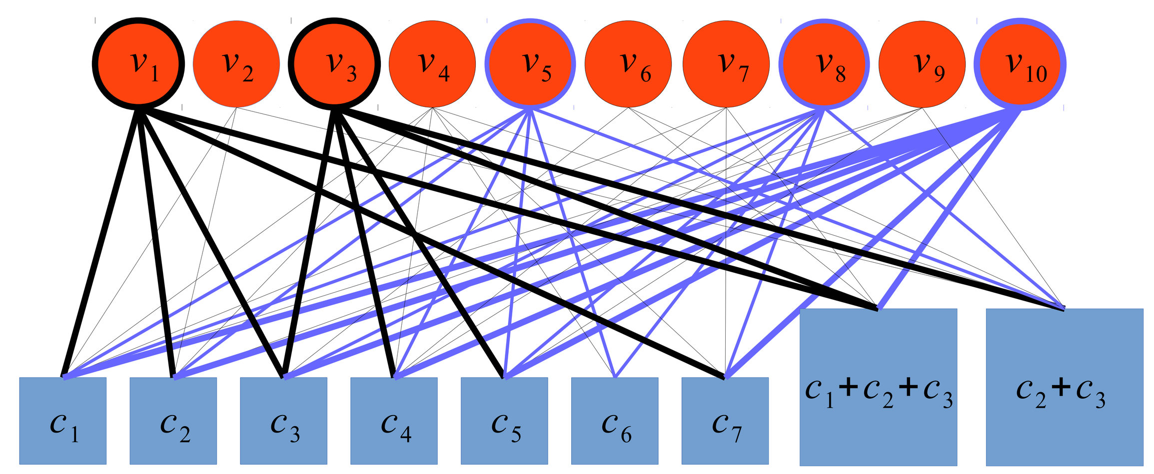
Stopping sets of size less than 4: $\{1,3,10\}$, $\{1,5,8\}$, $\{4,8,10\}$, $\{5,8,10\}$ (bold and blue – examples).
We could add redundant rows to this matrix in order to eliminate these stopping sets.

$$H' = \begin{matrix} c_1 \\ c_2 \\ c_3 \\ c_4 \\ c_5 \\ c_6 \\ c_7 \\ c_1+c_2+c_3 \\ c_2+c_3 \end{matrix} \begin{pmatrix} 1 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 1 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 1 & 1 & 1 & 0 & 1 & 1 & 0 \end{pmatrix}$$

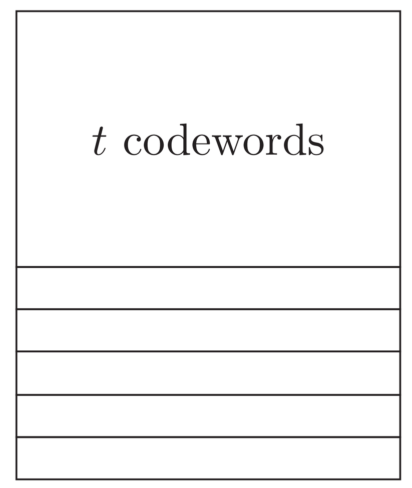
The same in terms of Tanner graph:



Adding redundant parity checks corresponds to introducing check nodes:



Probabilistic approach (Han-Siegel-Vardy'08)



Step 1

1. Choose t random codewords of $\mathcal{C}^\perp$ (no repetition)
2. Find expectations of
   - number of stopping sets left
   - rank deficiency
3. Guaranteed existence

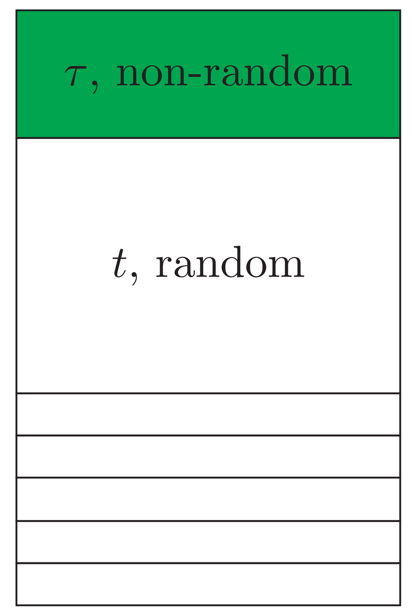
Step 2

1. Add one more random codeword of $\mathcal{C}^\perp$ (no repetition)
2. Find expectations of
   - *decrease* of number of stopping sets left
   - *increase* of rank
3. Guaranteed existence

Iterate...

Stop when there are no small stopping sets left and rank is $n-k$

Our improvements

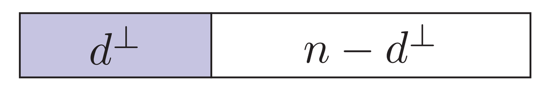


Main trick:

- Choose some first row(s) non-randomly and carefully
- ...so that we know how many stopping sets left
- ...or can bound their number
- And then apply technique of Han-Siegel-Vardy

Candidates for non-random rows:

- One or couple of rows of weight $d^\perp$



- Just take some codewords of $\mathcal{C}^\perp$, calculate straightforward (might be slow and not optimal)

Numerical results

We compare the bounds on the stopping redundancy obtained in [2] and [1] with our results. We consider two codes: the extended [24, 12, 8] binary Golay code and the extended [48, 24, 12] binary Quadratic Residue (QR) code. Both of them are known to be self-dual.

Upper bounds on the stopping redundancy

	[24, 12, 8] Golay	[48, 24, 12] QR
[2, Thm 4]	2509	4540385
[1, Thm 1]	198	3655
[1, Thm 3]	194	3655
[1, Thm 4]	187	3577
[1, Thm 7]	182	3564
One row of weight $d^\perp$	180	3538
Two rows of weight $d^\perp$	177	3515

References

[1] J. Han, P. H. Siegel, and A. Vardy. Improved probabilistic bounds on stopping redundancy. *Information Theory, IEEE Transactions on*, 54(4):1749–1753, 2008.

[2] M. Schwartz and A. Vardy. On the stopping distance and the stopping redundancy of codes. *Information Theory, IEEE Transactions on*, 52(3):922–932, 2006.