

RESEARCH STATEMENT

DEANNA DREHER

Digital data transmission is vital in today's world, but inherently prone to corruption due to flaws in the transmission medium. Shannon [10] proved that it is possible to efficiently transmit data across virtually any channel in such a way that errors can almost always be detected and corrected. The cost of this detection and correction is in the amount of redundant information transmitted: greater detection and correction requires more redundancy, which is costly in terms of time and physical resources. Thus a major goal of classical coding theory has been to find codes that maximize the number of errors that can occur and still be corrected while minimizing the amount of redundant information transmitted.

Unfortunately, knowing that an error *can* be corrected does not provide a practical means of actually correcting it. Indeed, many straight-forward approaches to correcting errors cannot be implemented in reasonable time. Modern coding theory is concerned with finding good codes with good decoders; many of these decoders are sub-optimal in the sense that they may not correct every possible error, but their simplicity and speed make them attractive from an implementation standpoint. In the last twenty years, *iterative message-passing decoding algorithms* [9], [12] have become a prominent class of sub-optimal decoders, due to the fact that they can be applied to certain classes of codes, such as *low-density parity-check codes* [3], [7] and *turbo codes* [1], to provide near-optimal error correction quickly and easily. As these algorithms are sub-optimal, there has been much work [2], [4], [8] done towards identifying and characterizing when these decoders fail, or at least fail to correct errors that are known to be correctable, in hopes of improving the decoders and the codes for which these algorithms are used. My research focuses on understanding and characterizing these so-called *pseudocodewords*.

I. BACKGROUND

Iterative message-passing decoders operate on the *Tanner graph* G of a code, which is a bipartite graph with *check nodes* and *variable nodes*. A codeword can be viewed as a *configuration* on G , i.e., an assignment of 0's and 1's to the variable nodes in G so that each check node has an even number of neighbors with an assignment of 1, and this notion of configuration generalizes to any bipartite graph with variable nodes and check nodes. The local nature of iterative message-passing decoding algorithms leads to the consideration of *graph covers*, which are graphs that look locally like a given base graph. An M -*cover* of G is a graph that, loosely speaking, looks like M copies of G that have been twisted together. Some graph cover configurations are induced by configurations on G , while others are not, which leads to the notion of pseudocodewords: if $\tilde{\mathbf{c}}$ is a configuration on an M -cover \tilde{G} of G , the corresponding *normalized graph cover pseudocodeword* is $\mathbf{f} = (f_x)_{x \in X}$, where X is the set of variable nodes in G and f_x is the proportion of copies of x in \tilde{G} that are assigned a 1.

Graph cover pseudocodewords have been elegantly characterized by the *fundamental cone* [6] and the *fundamental polytope* of a Tanner graph [11]. While much progress has been made in the realm of graph cover pseudocodewords, Wiberg [12] showed that iterative message-passing decoding algorithms are precisely modeled by *computation trees*, and not graph covers. A computation tree looks locally, except at the leaf nodes, like G , and precisely models the computations made by iterative message-passing decoders.

II. REALIZATIONS OF GRAPH COVER PSEUDOCODEWORDS

In using graph covers to analyze iterative message-passing decoding, we are only concerned with graph cover configurations that induce computation tree configurations. Since computation trees are necessarily connected, it follows that only graph cover configurations on connected covers are of interest. We show that for the majority of interesting or practical codes, every normalized graph cover pseudocodeword does in fact have a connected realization.

Theorem 1. *Let G be a Tanner graph. Then any normalized graph cover pseudocodeword that can be realized on an M -cover can be realized on a connected M -cover.*

Notice that the cover degree given in Theorem 1 assumes that the normalized graph cover pseudocodeword \mathbf{f} under inspection is realizable on an M -cover. This leads us to examine for which M there exists a realization of \mathbf{f} on an M -cover.

Theorem 2. *Let G be a Tanner graph, let \mathcal{P} be the fundamental polytope of G , and let $\mathbf{f} \in \mathcal{P}$. Let $M \in \mathbb{N}$ be such that $M\mathbf{f}$ is an integer vector that reduces to a codeword modulo 2. Then \mathbf{f} is realizable on an M -cover of G .*

III. CYCLE CODES AND PSEUDOWEIGHT

Koetter and Vontobel show that the performance of graph cover decoding is largely determined by the minimum *pseudoweight*, over all graph cover pseudocodewords, of the Tanner graph of the code. The minimum pseudoweight is always achieved at a *minimal vertex* of the fundamental polytope of the Tanner graph of the code and is generally accepted as a good predictor of performance in iterative decoding [5]. For the class of cycle codes, we show that there is an elegant graphical characterization of these minimal vertices of the fundamental polytope.

Theorem 3. *Let G be the Tanner graph of a cycle code, let \mathcal{P} be its fundamental polytope and let $\mathbf{f} \in \mathcal{P}$. Then \mathbf{f} is a minimal vertex of \mathcal{P} if and only if \mathbf{f} is the normalized graph cover pseudocodeword corresponding to a minimal realization of some cycle or dumbbell (i.e. two cycles connected by a path).*

The asymptotic performance of graph cover decoding is determined not only by the minimum pseudoweight of a code, but also by the number of vertices of \mathcal{P} with that weight [11], [13]. We show that the minimum pseudoweight of a cycle code is the minimum distance of the code, and is achieved only by codewords, from which it follows that graph cover decoding is optimal asymptotically.

Theorem 4. *If C is a cycle code with parity check matrix H , then the minimum pseudoweight of a H is the minimum distance of C . Moreover, if \mathbf{f} is a minimal vertex of the fundamental polytope \mathcal{P} , and \mathbf{f} is not a codeword, then $w(\mathbf{f}) \geq 2d_{\min}$.*

IV. REALIZATIONS OF COMPUTATION TREE PSEUDOCODEWORDS ON GRAPH COVERS

Returning to the foundational work of Wiberg [12], we know that iterative message-passing decoding algorithms are precisely modeled by computation trees. Thus, if we are to use graph cover configuration in our analysis of iterative message-passing decoding algorithms, we need to know somehow compare the set of graph cover configurations with the set of computation tree configurations. We prove that every computation tree configuration is induced by graph cover configuration.

Theorem 5. *Let G be the Tanner graph of a cycle code. Let R be a computation tree of G , and let M be the maximum number of times any node in G appears in R . Then there exists a connected $2M$ -cover \tilde{G} of G such that for any configuration \mathbf{c} on R , there exists a graph cover configuration $(\tilde{\mathbf{c}}, \tilde{G})$ containing a copy of (\mathbf{c}, R) .*

Theorem 6. *Let G be a Tanner graph and suppose (\mathbf{c}, R) is a computation tree configuration of G . Let M be the maximum number of times any check node in G appears in R .*

Then there exists a connected graph cover configuration on a $4(M + 1)$ -cover that induces the configuration (\mathbf{c}, R) and has $(\frac{1}{2}, \dots, \frac{1}{2})$ as its normalized graph cover pseudocodeword.

V. FUTURE WORK

Traditionally, the cost of a graph cover pseudocodeword is an inner product, but this notion of cost provides no insight into the cost of computation tree configurations induced by a certain graph cover pseudocodeword, as we have seen that certain graph cover pseudocodewords induce every possible computation tree configuration, regardless of cost. One question of interest is whether there is a better notion of cost, or one that captures more information about a graph cover configuration, that might relate to the cost of induced computation tree pseudocodewords. Additionally, since we generally see relatively quick convergence of iterative message-passing algorithms when they do converge, one might even bound the degree of the cover necessary to realize a graph cover pseudocodeword by the depth of the computation tree. Thus the assumption that the algorithm is taking into consideration the full graph cover configuration may not be valid. We also gave a graphical characterization of the most likely errors the minimal vertices of the fundamental polytope of code under graph cover decoding of cycle codes. A major area of open research is whether there is a clean graphical characterization of these minimal vertices for more general low-density parity-check codes. Another open question is whether we can graphically classify other interesting subsets of the vertices of the fundamental polytope, e.g. is there a graphical characterization of *all* vertices of the fundamental polytope?

I am eager to working with undergraduate students and possibly even high school students on extensions of this research. Although this topic is deep and has far-reaching practical implications on error correction, it contains many pieces that would be accessible to a student with a basic graph theory primer. As my research is motivated by decoding algorithms often studied by electrical engineers, it also lends itself well to interdisciplinary work, and I am interested in pursuing and developing the kind of interdisciplinary collaboration that I have experienced here at UNL.

REFERENCES

- [1] C. Berrou, A. Glavieux, and P. Thitimajshima. Near Shannon limit error-correcting coding and decoding. In *Proceedings of the 1993 IEEE International Conference on Communications*, pages 1064–1070, Geneva, Switzerland, 1993.
- [2] G. D. Forney, Jr., R. Koetter, F. R. Kschischang, and A. Reznik. On the effective weights of pseudocodewords for codes defined on graphs with cycles. In *Codes, systems, and graphical models (Minneapolis, MN, 1999)*, volume 123 of *IMA Vol. Math. Appl.*, pages 101–112. Springer, New York, 2001.
- [3] R. G. Gallager. *Low-Density Parity Check Codes*. MIT Press, Cambridge, MA, 1963.
- [4] G.A. Horn. *Iterative Decoding and Pseudocodewords*. PhD thesis, California Institute of Technology, Pasadena, CA, 1999.
- [5] C. Kelley and D. Sridhara. Pseudocodewords of Tanner graphs. *IEEE Transactions on Information Theory*, 53:4013–4038, November 2007.
- [6] R Koetter, W.-C. W. Li, P. O. Vontobel, and J. L. Walker. Characterizations of pseudo-codewords of LDPC codes. *Advances in Mathematics*, 213:205–229, 2007.
- [7] D. J. C. MacKay and R. M. Neal. Near Shannon limit performance of low-density parity check codes. *IEE Electronic Letters*, 32(18):1645–1646, August 1996.
- [8] T. Richardson, A. Shokrollahi, and R. Urbanke. Design of capacity-approaching irregular low-density parity check codes. *IEEE Transactions on Information Theory*, February 2001.
- [9] T. Richardson and R. Urbanke. The capacity of low-density parity check codes under message-passing decoding. *IEEE Transactions on Information Theory*, February 2001.
- [10] C. E. Shannon. A mathematical theory of communication. *Bell System Technical Journal*, 27:379–423 and 623–656, July and October 1948.
- [11] P. Vontobel and R. Koetter. Graph-cover decoding and finite-length analysis of message-passing iterative decoding of LDPC codes. To appear in *IEEE Transactions on Information Theory*.
- [12] N. Wiberg. *Codes and Decoding on General Graphs*. PhD thesis, Linköping University, Linköping, Sweden, 1996.
- [13] Shu-Tao Xia and Fang-Wei Fu. Minimum pseudo-codewords of LDPC codes. pages 109–113, Oct. 2006.