2008

Towards Universal Cover Decoding

Nathan Axvig
University of Nebraska - Lincoln, s-naxvig1@math.unl.edu

Deanna Dreher
University of Nebraska - Lincoln

Katherine Morrison
University of Nebraska - Lincoln, kmorris2@gmail.com

Eric T. Psota
University of Nebraska-Lincoln, epsota@unl.edu

Lance C. Pérez
University of Nebraska-Lincoln, lperez@unl.edu

See next page for additional authors

Citation: Axvig, N., Dreher, D., Morrison, K., Psota, E., Perez, L. C., & Walker, J. L. (2008).
Towards universal cover decoding. In 2008 International Symposium on Information Theory
and its Applications, ISITA 2008
doi 10.1109/ISITA.2008.4895386

International Symposium on Information Theory and its Applications, ISITA2008
Auckland, New Zealand, 7-10, December, 2008

Towards Universal Cover Decoding

Nathan Axvig[†], Deanna Dreher[†], Katherine Morrison[†], Eric Psota[‡],
Lance C. Pérez[‡] and Judy L. Walker[†]

[†] Department of Mathematics
University of Nebraska-Lincoln
Lincoln, Nebraska, USA
E-mail: {s-naxvig1, s-dturk1}@math.unl.edu
{s-kmorri11, jwalker}@math.unl.edu

[‡] Department of Electrical Engineering
University of Nebraska-Lincoln
Lincoln, Nebraska, USA
E-mail: epsota24@bigred.unl.edu, lperez@unl.edu

1. Introduction

Low complexity decoding of low-density parity-check (LDPC) codes may be obtained from the application of iterative message-passing decoding algorithms to the bipartite Tanner graph of the code. Arguably, the two most important decoding algorithms for LDPC codes are the sum-product decoder and the min-sum (MS) decoder. On a bipartite graph without cycles (a tree), the sum-product decoder minimizes the probability of bit error, while the min-sum decoder minimizes the probability of word error [9].

While the behavior of sum-product and min-sum is easily understood when operating on trees, their behavior becomes much more difficult to characterize when the Tanner graph has cycles. Wiberg [9] showed that decoding can be modeled by finding minimal cost configurations on computation trees that are formed at successive iterations of sum-product/min-sum, and returning the value assigned to the root nodes of these trees. Additionally, he proved that for an error to occur at a particular variable node, there must exist a deviation of non-positive cost on the computation tree rooted at this node.

In this paper, we are interested in analyzing the non-codeword errors that occur during parallel, iterative decoding with the min-sum decoder. Recently, work has been done relating the min-sum decoder to the linear programming (LP) decoder via graph covers [8]. The LP decoder, as defined by Feldman [3], recasts the problem of decoding as an optimization problem whose feasible set is a polytope defined by the parity-check matrix of a code. In [8], it is shown that LP decoding can be realized as a decoder operating on graph covers. The notion that non-codeword outputs of LP decoding are related to non-codeword outputs of min-sum decoding is attractive from an analytical perspective. However, the performance of LP and min-sum are not consistently related [2]. Therefore, a different theoretical model is needed to explore the relationship between decoding on graph covers and decoding on computation trees. To bridge this gap, we will turn to the notion of decoding on the universal cover. Universal covers can be thought of as both infinite computation trees and infinite graph covers. For this reason, decoding on universal covers provides an intuitive link between LP decoding and min-sum decoding of LDPC codes.

This paper is an extension of previous work done by the authors in [2]; thus, much of the requisite background material is drawn from [2]. Section 2 introduces the definition of universal covers. Properties related to configurations on universal covers and their corresponding costs are established in Section 3. Finally, in Section 4 a preliminary definition of the universal cover decoder is given, and it is shown that under certain conditions the universal cover decoder agrees with the LP decoder.

2. Universal Covers

Finite covers of Tanner graphs and their applications to the decoding of low-density parity-check codes have been studied extensively (see, e.g., [4, 5, 8]). As such, we do not provide rigorous definitions and discussion of graph covers, though we do include in Figure 1 a small example to illustrate the concept. In this section we turn our attention to the *universal cover*, a well-studied object in topology that we now define in the context of graph theory.

Definition 2.1. *Let G be a finite connected graph and suppose the cover $\widehat{\pi} : \widehat{G} \to G$ enjoys the following uni-*

This work was supported in part by NSF grant DMS-0735099.

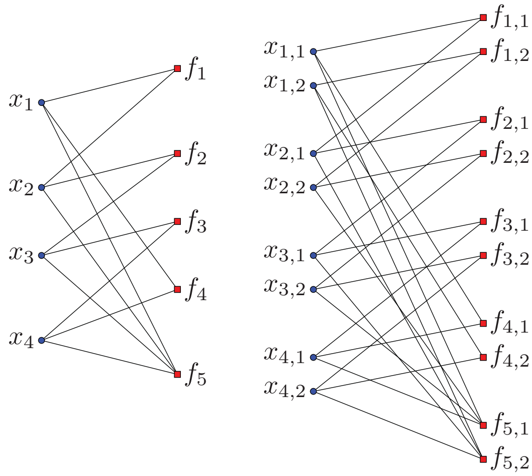

Figure 1: A Tanner graph T for the [4,1,4] repetition code (left) and a 2-cover of T (right).

versal property: For any connected cover $\pi : \widetilde{G} \to G$ of G, there is a covering map $\widetilde{\pi} : \widehat{G} \to \widetilde{G}$ such that $\pi \circ \widetilde{\pi} = \widehat{\pi}$. Then $\widehat{\pi} : \widehat{G} \to G$ is called a universal cover *of G.*

If G is a tree, then G is its own universal cover. When G is not a tree, a practical way of constructing a universal cover of G is to build the computation tree $\widehat{G}_v$ of infinite depth rooted at a vertex v of G. It is clear that $\widehat{G}_v$ is a cover; that it is a universal cover follows from Theorem 1.24 in [7], noting that trees have trivial fundamental group. For example, when this construction is applied to an n-cycle, the universal cover is simply a path extending infinitely in both directions.

In light of Definition 2.1, for the remainder of this paper all Tanner graphs are assumed to be finite and connected. Given a Tanner graph T, the universal cover of T can also be thought of as an infinite Tanner graph. This relationship is given by Definition 2.2.

Definition 2.2 ([2], Definition 6.2)**.** *Let $T = (X \cup F, E)$ be a Tanner graph, and let $\widehat{\pi} : \widehat{T} \to T$ be the universal cover of T. Set*

$$\widehat{X} := X(\widehat{T}) = \bigcup_{x \in X} \widehat{\pi}^{-1}(x)$$

and

$$\widehat{F} := F(\widehat{T}) = \bigcup_{f \in F} \widehat{\pi}^{-1}(f).$$

We call $\widehat{X}$ the set of variable nodes *of $\widehat{T}$ and $\widehat{F}$ the set of* check nodes *of $\widehat{T}$. A* configuration *on $\widehat{T}$ is an assignment $\widehat{\mathbf{c}} = (\widehat{c}_{\widehat{x}})_{\widehat{x} \in \widehat{X}}$ of 0's and 1's to the variable nodes of $\widehat{T}$ such that the binary sum of the neighbors of each check node in $\widehat{T}$ is 0. A* universal cover pseudocodeword *for T is a configuration on $\widehat{T}$.*

3. Configurations and Cost on Universal Covers

Some basic relationships between universal cover pseudocodewords, graph cover pseudocodewords and computation tree pseudocodewords are established in Proposition 6.3 of [2]. This proposition first states that every computation tree psuedocodeword can be extended to a configuration on the universal cover and that any universal cover pseudocodeword can be truncated to a computation tree pseudocodeword. Additionally, every graph cover pseudocodeword $\boldsymbol{\omega}$ that has a *connected realization* induces a universal cover pseudocodeword. Here, we define a connected realization of $\boldsymbol{\omega}$ to be a pair $(\widetilde{T}, \widetilde{\mathbf{c}})$ such that $\widetilde{T}$ is a connected cover of T and $\widetilde{\mathbf{c}}$ is a codeword in the code defined by $\widetilde{T}$ such that the normalized graph cover pseudocodeword of $\widetilde{\mathbf{c}}$ (see, e.g., [8]) is $\boldsymbol{\omega}$. If $\boldsymbol{\omega}$ has a connected realization, we often say that $\boldsymbol{\omega}$ is a *connected* graph cover pseudocodeword. With this definition, it is clear how $\boldsymbol{\omega}$ will induce a universal cover pseudocodeword: first, let $(\widetilde{T}, \widetilde{\mathbf{c}})$ be a connected realization of $\boldsymbol{\omega}$ and let $\widehat{T}$ be the universal cover of T. By Definition 2.1, $\widehat{T}$ also covers $\widetilde{T}$ via a map $\widetilde{\pi}$. The configuration $\widetilde{\mathbf{c}}$ can then be lifted through $\widetilde{\pi}$, much as we lift codewords onto finite computation trees. Similarly, any configuration on a connected cover of T induces a universal cover pseudocodeword. An important implication of this proposition is that the universal cover is an environment in which it is natural to consider both computation tree pseudocodewords and connected graph cover pseudocodewords.

With the ultimate goal of defining a universal cover decoder as motivation, the authors propose a cost function on infinite computation trees [2]. The cost function presented in this paper differs slightly from its original form in [2] but is still designed to capture a limiting value of normalized versions of Wiberg's cost function [9].

Definition 3.1 (See also [2], Definition 6.5)**.** *Let $T = (X \cup F, E)$ be a Tanner graph and let $\widehat{T}_v$ be the universal cover of T, realized as an infinite computation tree rooted at the variable node v of T. For any positive integer m, let $R_v^{(m)}$ be the computation tree of depth m rooted at v, so that $R_v^{(m)}$ is formed by truncating $\widehat{T}_v$ after the $2m^{th}$ level. For any configuration $\widehat{\mathbf{c}}_v$ on $\widehat{T}_v$, let $\widehat{\mathbf{c}}_v^{(m)}$, $m \geq 1$, be the truncation of $\widehat{\mathbf{c}}_v$ to $R_v^{(m)}$, and let $G(\widehat{\mathbf{c}}_v^{(m)})$ be the cost of $\widehat{\mathbf{c}}_v^{(m)}$ as given by Wiberg [9]. The* rooted cost *of the universal cover configuration $\widehat{\mathbf{c}}_v$ on the infinite computation tree $\widehat{T}_v$ is defined to be*

$$G_v(\widehat{\mathbf{c}}_v) := \limsup_{m \to \infty} \left(\frac{|X|}{\left| X_v^{(m)} \right|} G(\widehat{\mathbf{c}}_v^{(m)}) \right),$$

where $X_v^{(m)}$ is the set of variable nodes of $R_v^{(m)}$.

The normalization factor of $\frac{|X|}{\left|X_v^{(m)}\right|}$ above ensures that for a given log-likelihood vector the limit supremum is applied to a bounded sequence, hence guaranteeing that the rooted cost exists and is finite. Though the use of the limit supremum guarantees convergence, it is useful to ask for which universal cover pseudocodewords the rooted cost can be computed with a limit rather than a limit supremum, It is also interesting to explore the relationships between a universal cover pseudocodeword's structure (i.e., how it assigns binary values to variable nodes) and the corresponding rooted cost. Before these questions are addressed, we begin with discussion on the necessary background material and tools that will go into the proofs. We begin this discussion with Theorem 3.2, which states that for a particular class of codes the distribution of variable nodes in finite computation trees approaches uniformity as the number of iterations goes to infinity.

Theorem 3.2. *Let $T = (X \cup F, E)$ be the Tanner graph of a (d_X, d_F)-regular LDPC code with $d_X, d_F \geq 3$. For any positive integer m, let $X_v^{(m)}(x)$ be the set of copies of variable node x in $X_v^{(m)}$. For any $x \in X$, we have*

$$\lim_{m\to\infty} \frac{\left|X_v^{(m)}(x)\right|}{\left|X_v^{(m)}\right|} = \frac{1}{|X|}.$$

The notion of *non-backtracking random walks* plays a key role in subsequent arguments; therefore, we provide a brief introduction to this concept before giving the proof of Theorem 3.2. A non-backtracking random walk is a random process on an arbitrary graph G, which is not assumed to be bipartite. It is described as follows: select a vertex v_0 of G from which to begin the walk, and select uniformly at random an edge e_1 incident to v_0. Let v_1 be the other endpoint of e_1 and select uniformly at random an edge $e_2 \neq e_1$ incident to v_1. Repeat this process some predetermined finite number of times.

More formally (and restricting to walks of even length for reasons that will become clear shortly), let $W_v^{(m)}$ be the set of all non-backtracking walks of length $2m$ in G whose initial vertex is v, and define the probability of the walk $w \in W_v^{(m)}$ with vertices $v = v_0, v_1, \ldots, v_{2m}$ to be

$$P(w) := \frac{1}{\deg(v)(\deg(v_1) - 1)\ldots(\deg(v_{2m-1}) - 1)}.$$

A non-backtracking random walk of length $2m$ with initial vertex v is then a pair $(w, P(w))$ where $w \in W_v^{(m)}$ and $P(w)$ is its probability. One can see that this is a probability measure on $W_v^{(m)}$ by using induction on m, and it is clear that this measure agrees with the intuitive description in the previous paragraph.

In our situation, the graph G is a Tanner graph T, and the walks of interest to us must start and end at variable nodes. Since any such walk must have even length because T is bipartite, we focus exclusively on walks of even length.

Let $q_v^{(m)}(x) = \sum_{w\in W} P(w)$, where $W = W_v^{(m)}(x)$ is the set of non-backtracking walks in T of length $2m$ that start at variable node v and end at variable node x. With this definition, we see that $q_v^{(m)}(x)$ is the probability that a non-backtracking random walk in T of length $2m$ that starts at vertex v will have terminal vertex x [6]. With the following result on non-backtracking random walks from [6], we have the tools necessary to proceed with the proof of Theorem 3.2.

Theorem 3.3 (See [6], Theorem 1.2(ii))**.** *Let $T = (X \cup F, E)$ be a Tanner graph with minimum degree at least 3. Using the notation established above, we have*

$$\lim_{m\to\infty} q_v^{(m)}(x) = \frac{\deg(x)}{|E|}$$

for all $v, x \in X$.

Proof of Theorem 3.2. Recall that $X_v^{(m)}$ is the set of variable nodes in the computation tree $R_v^{(m)}$ and that $X_v^{(m)}(x)$ is the set of copies of the variable node x of T in $X_v^{(m)}$. For $m \geq 1$, the biregularity of T forces the number of non-backtracking walks in T of length $2m$ that start at any given variable node to be $\tau^{(m)} := \frac{d_X}{d_X-1}(d_X - 1)^m(d_F - 1)^m$, with each walk equally probable. Let $\widehat{T}_v$ be the universal cover of T, realized as an infinite computation tree rooted at v, so that $R_v^{(m)}$ is the truncation of $\widehat{T}_v$ to depth m. Let $\eta_v^{(m)}(x)$ be the number of copies of variable node x in the $2m^{\text{th}}$ level of $\widehat{T}_v$. There is a natural bijection between non-backtracking walks in T that start at v and paths in $\widehat{T}_v$ that start at the root node; thus, $\eta_v^{(m)}(x)$ is precisely the number of non-backtracking walks in T of length $2m$ that start at v and end at x. Therefore $q_v^{(m)}(x) = \frac{\eta_v^{(m)}(x)}{\tau^{(m)}}$.

Let $p_v^{(m)}(k)$ be the probability of picking uniformly at random a variable node in the $2k^{\text{th}}$ level from all variable nodes in $R_v^{(m)}$. Since the probability of selecting uniformly at random a copy of variable node x from all variable nodes of $R_v^{(m)}$ is $\frac{\left|X_v^{(m)}(x)\right|}{\left|X_v^{(m)}\right|}$, we have

$$\frac{\left|X_v^{(m)}(x)\right|}{\left|X_v^{(m)}\right|} = \sum_{i=0}^{m} q_v^{(i)}(x) p_v^{(m)}(i). \qquad (3.1)$$

Let $\epsilon > 0$ be given and set $L = \frac{\deg(x)}{|E|} = \frac{d_X}{d_X|X|} = \frac{1}{|X|}$. By Theorem 3.3, there is a positive integer M_1 such that $|q_v^{(m)}(x) - L| < \frac{\epsilon}{3}$ for all $m \geq M_1$. Since the number of variable nodes from one level to the next in the computation tree grows exponentially by a factor of $(d_X - 1)(d_F - 1) \geq 4$, the probability of selecting a variable node from the first $2M_1$ levels of a computation tree diminishes to zero as the depth of the tree increases. Thus, we can find $M_2 > M_1$ such that for all $m \geq M_2$,

$$\sum_{i=0}^{M_1} p_v^{(m)}(i) < \frac{\epsilon}{3}.$$

By writing L as $\sum_{i=1}^{m} p_v^{(m)}(i)L$ and using Equation 3.1, one can use standard triangle inequality arguments to show that

$$\left| \frac{\left|X_v^{(m)}(x)\right|}{\left|X_v^{(m)}\right|} - L \right| \leq \epsilon$$

for all $m \geq M_2$, which concludes the proof. ∎

Theorem 3.4 below gives our first result on how structure is related to rooted cost. One implication of Theorem 3.4 is of particular importance: for a universal cover pseudocodeword induced by a connected graph cover pseudocodeword $\boldsymbol{\omega}$ the rooted cost, which is derived from normalized versions of Wiberg's cost function, is equal to the cost of $\boldsymbol{\omega}$ in linear programming decoding [3]. This fact plays a key role in Section 4.

Theorem 3.4 (see also [2], Theorem 6.6)**.** *Let $T = (X \cup F, E)$ be the Tanner graph of a (d_X, d_F)-regular LDPC code, with $d_X, d_F \geq 3$. Let $\widetilde{T}$ be a connected cover of T, let $\widetilde{\mathbf{c}}$ be a codeword in the code defined by $\widetilde{T}$, and let $\boldsymbol{\omega} = \boldsymbol{\omega}(\widetilde{\mathbf{c}})$ be the normalized pseudocodeword associated to $\widetilde{\mathbf{c}}$. Suppose that, on the universal cover $\widehat{T}_v$ of T realized as an infinite computation tree rooted at the variable node v of T, the configuration $\widehat{\mathbf{c}}_v$ is induced by $\widetilde{\mathbf{c}}$. Then*

$$G_v(\widehat{\mathbf{c}}_v) = \lim_{m\to\infty} \left(\frac{|X|}{\left|X_v^{(m)}\right|} G(\widehat{\mathbf{c}}_v^{(m)}) \right) = \boldsymbol{\lambda} \cdot \boldsymbol{\omega},$$

where $\boldsymbol{\lambda}$ is the vector of log-likelihood ratios.

Proof. Let M be the degree of the cover $\widetilde{T} = (\widetilde{X} \cup \widetilde{F}, \widetilde{E})$ of T. Note that $\widehat{T}_v$ is a universal cover of $\widetilde{T}$ and that $\widetilde{T}$ is finite and connected with variable node degree $d_{\widetilde{X}} = d_X \geq 3$ and check node degree $d_{\widetilde{F}} = d_F \geq 3$, and let $\widehat{\mathbf{c}}_v$ be a configuration on $\widehat{T}_v$ induced by $\widetilde{\mathbf{c}}$. Then we have

$$\begin{aligned} G_v(\widehat{\mathbf{c}}_v) &= \limsup_{m\to\infty} \frac{|X|}{\left|X_v^{(m)}\right|} \sum_{\widetilde{x}\in\widetilde{X}} \lambda_{\widetilde{x}} \left| X_v^{(m)}(\widetilde{x}) \cap \mathrm{supp}(\widehat{\mathbf{c}}_v^{(m)}) \right| \\ &= \limsup_{m\to\infty} \frac{|X|}{\left|X_v^{(m)}\right|} \sum_{\widetilde{x}\in\mathrm{supp}(\widetilde{\mathbf{c}})} \lambda_{\widetilde{x}} \left| X_v^{(m)}(\widetilde{x}) \right| \\ &= |X| \sum_{\widetilde{x}\in\mathrm{supp}(\widetilde{\mathbf{c}})} \lambda_{\widetilde{x}} \lim_{m\to\infty} \frac{\left|X_v^{(m)}(\widetilde{x})\right|}{\left|X_v^{(m)}\right|} \\ &= |X| \sum_{\widetilde{x}\in\mathrm{supp}(\widetilde{\mathbf{c}})} \lambda_{\widetilde{x}} \frac{1}{\left|\widetilde{X}\right|} \end{aligned}$$

by Theorem 3.2, where $\lambda_{\widetilde{x}}$ is the Wiberg min-sum cost function assigned to node $\widetilde{x}$. To continue the string of equalities, we use that $\left|\widetilde{X}\right| = M\,|X|$, $\lambda_{\widetilde{x}} = \lambda_x$ for each $\widetilde{x} \in \widetilde{X}$ in the inverse image of x under the covering map, and the number of such $\widetilde{x}$ in the support of $\widetilde{\mathbf{c}}$ is precisely $M\omega_x$. We then have:

$$\begin{aligned} G_v(\widehat{\mathbf{c}}_v) &= |X| \sum_{x\in\mathrm{supp}(\boldsymbol{\omega})} M\omega_x \lambda_x \frac{1}{M\,|X|} \\ &= \sum_{x\in\mathrm{supp}(\boldsymbol{\omega})} \lambda_x \omega_x \\ &= \boldsymbol{\lambda} \cdot \boldsymbol{\omega}, \end{aligned}$$

as desired. ∎

We conclude this section by examining the costs of another class of universal cover pseudocodewords, defined below.

Definition 3.5. *Let T be a Tanner graph and let $\widehat{T}$ be the universal cover of T. A* minimal universal cover pseudocodeword *is a configuration on $\widehat{T}$ whose support does not properly contain the support of any non-zero universal cover pseudocodeword.*

If a minimal universal cover pseudocodeword assigns a value of 1 to a particular output node, this corresponds precisely to the notion of a deviation, as defined by Wiberg [9]. Proposition 3.6 describes these configurations more precisely.

Proposition 3.6. *Let T be a Tanner graph such that each check node has degree at least 2, and let $\widehat{T}$ be its universal cover. Let $\widehat{\mathbf{c}}$ be a configuration on $\widehat{T}$, let A be* *the neighborhood of* $\text{supp}(\widehat{\mathbf{c}})$*, and let* S *be the subgraph of* $\widehat{T}$ *induced by* $\text{supp}(\widehat{\mathbf{c}}) \cup A$*. Then* $\widehat{\mathbf{c}}$ *is minimal if and only if* S *is connected and each check node in* A *is adjacent to exactly two variable nodes in* $\text{supp}(\widehat{\mathbf{c}})$*.*

The proof of Proposition 3.6 may be found in [1]. Proposition 3.7 below shows that the class of minimal universal cover pseudocodewords is, in fact, disjoint from the class of universal cover pseudocodewords induced by connected graph cover pseudocodewords. Intuitively, this is plausible since minimal universal cover pseudocodewords seem to have significantly fewer variables set to 1 than do the others. With the aid of the characterization given by Proposition 3.6 we make this intuitive justification rigorous.

Proposition 3.7. *Let* $T = (X \cup F, E)$ *be the Tanner graph of a* (d_X, d_F)*-regular LDPC code of length* n *with* $d_X, d_F \geq 3$*. Let* $\widehat{T}_v$ *be the universal cover of* T*, realized as the infinite computation tree rooted at the variable node* v *of* T*. Let* $\widehat{\mathbf{c}}_v$ *be a minimal universal cover pseudocodeword on* $\widehat{T}_v$ *and assume that the coordinates of the log-likelihood vector are all finite. Then*

$$G_v(\widehat{\mathbf{c}}_v) = \lim_{m\to\infty} \left(\frac{|X|}{\left|X_v^{(m)}\right|} G(\widehat{\mathbf{c}}_v^{(m)}) \right) = 0.$$

Moreover, $\widehat{\mathbf{c}}_v$ *is not induced by any graph cover pseudocodeword.*

Proof. By Proposition 3.6, we have that $\left|X_v^{(m)} \cap \text{supp}(\widehat{\mathbf{c}}_v^{(m)})\right|$ grows on the order of $(d_X - 1)^m$, but the size of $X_v^{(m)}$ grows on the order of $(d_X - 1)^m (d_F - 1)^m$. Since $(d_X - 1) < (d_X - 1)(d_F - 1)$, it follows that

$$\lim_{m\to\infty} \frac{\left|X_v^{(m)} \cap \text{supp}(\widehat{\mathbf{c}}_v^{(m)})\right|}{\left|X_v^{(m)}\right|} = 0. \tag{3.2}$$

Using the assumption that the log-likelihoods are finite and Equation 3.2, we have

$$G_v(\widehat{\mathbf{c}}) = \lim_{m\to\infty} \left(\frac{|X|}{\left|X_v^{(m)}\right|} G(\widehat{\mathbf{c}}_v^{(m)}) \right) = 0.$$

It remains to be shown that $\widehat{\mathbf{c}}$ is not induced by any graph cover pseudocodeword. If it were, by Theorem 3.4 the rooted cost $G_v(\widehat{\mathbf{c}})$ would be equal to $\boldsymbol{\lambda} \cdot \boldsymbol{\omega}$, where $\boldsymbol{\omega}$ is some normalized, connected, graph cover pseudocodeword. But we have shown that $G_v(\widehat{\mathbf{c}}) = 0$ for all vectors $\boldsymbol{\lambda}$ in which all coordinates are finite, so it must be that $\boldsymbol{\omega} = 0$. Since $\widehat{\mathbf{c}}$ is not the all-zeros configuration, it is not induced by the all-zeros graph cover pseudocodeword. ∎

4. Decoding on Universal Covers

Definition 4.1 gives a working definition of universal cover decoding.

Definition 4.1. *Let* T *be a Tanner graph with variable nodes* $x_1, \ldots, x_n$ *and, for* $1 \leq i \leq n$*, let* $\widehat{T}_{x_i}$ *be the universal cover of* T *realized as an infinite computation tree rooted at* x_i*. For a given received vector* $\mathbf{y}$*, let* θ_i *be the probability that a randomly chosen configuration of minimal rooted cost on* $\widehat{T}_{x_i}$ *has assigned a 1 to the root node* x_i*.* Universal cover (UC) decoding *is defined as the decoder that returns the vector*

$$\text{UC}(\mathbf{y}) = (\theta_1, \ldots, \theta_n).$$

The motivation for Definition 4.1 is two-fold. First, we wish to mimic Wiberg's model of min-sum [9] by making a bit-wise decision that is based on the binary value the root node receives from a minimal cost computation tree configuration. We then modify this approach by returning the probability that the root node x_i receives a 1 from a minimal rooted cost configuration, since the universal cover does not, *a priori*, have a root node. In particular, for a given universal cover pseudocodeword (e.g., those induced by connected graph cover pseudocodewords of regular LDPC codes), it is possible for the configuration to look different from various potential root nodes (copies of x_i), yet still have the same rooted cost.

To make the probabilities $\theta_1, \ldots, \theta_n$ of Definition 4.1 well-defined, one needs a probability measure on the set of universal cover configurations. The search for a meaningful probability measure is an area of current study. One particular property that this probability measure should display is given in the next definition.

Definition 4.2. *Let* $T = (X \cup F, E)$ *be a Tanner graph and let* $\widehat{T}_v$ *be the universal cover of* T *realized as an infinite computation tree rooted at the variable node* v *of* T*. A probability measure on the set of configurations on* $\widehat{T}_v$ *is called* admissible *if, for every normalized, connected graph cover pseudocodeword* $\boldsymbol{\omega} = (\omega_x)_{x \in X}$*, the probability that an arbitrarily chosen configuration on* $\widehat{T}_v$ *that is induced by* $\boldsymbol{\omega}$ *assigns a 1 to the root node of* $\widehat{T}_v$ *is* ω_v*.*

If an admissible measure exists for the Tanner graph $T = (X \cup F, E)$ with $X = \{x_1, \ldots, x_n\}$, we can relate the output of the universal cover decoder to that of LP decoding. First, restrict the universal cover decoder in the following manner. For each $i = 1, 2, \ldots, n$, consider only the set of configurations on $\widehat{T}_{x_i}$ induced by connected graph cover pseudocodewords. From this set, find the set of minimal cost configurations. Let θ_i be the probability that a randomly chosen minimal cost configuration has assigned a 1 to the root node x_i, as in Definition 4.1. Define

$$\mathrm{UC}\,|_{\mathrm{GC}}(\mathbf{y}) := (\theta_1, \theta_2, \ldots, \theta_n).$$

From this point on, we will consider only (d_X, d_F)-regular LDPC codes of length n with $d_X, d_F \geq 3$ so as to utilize a number of earlier results. Let $\boldsymbol{\lambda}$ be the log-likelihood vector for the received vector $\mathbf{y}$. Theorem 3.4 shows that the rooted cost of a configuration induced by a configuration $\widetilde{\mathbf{c}}$ on a finite connected cover of T is equal to $\boldsymbol{\lambda} \cdot \boldsymbol{\omega}(\widetilde{\mathbf{c}})$, where $\boldsymbol{\omega}(\widetilde{\mathbf{c}})$ denotes the normalized graph cover pseudocodeword associated with $\widetilde{\mathbf{c}}$, and that this value is independent of the root node of $\widehat{T}$. Thus, a configuration on $\widehat{T}_{x_i}$ induced by a connected graph cover pseudocodeword $\widetilde{\mathbf{c}}$ will have minimal rooted cost if it minimizes $\boldsymbol{\lambda} \cdot \boldsymbol{\omega}$ where $\boldsymbol{\omega}$ ranges over all possible normalized connected graph cover pseudocodewords. In our situation, every graph cover pseudocodeword has a connected realization by Theorem 2.10 of [1]. This implies that the set over which we are minimizing is precisely the set of rational points in the fundamental polytope $\mathcal{P}$ [8], where $\mathcal{P}$ is the feasible set for the LP decoder [3]. Note that the vertices of $\mathcal{P}$ [3] are rational, and are thus included in this set. Given that the cost function of the LP decoder is simply the vector of log-likelihoods, we have the following proposition.

Proposition 4.3. *Let $T = (X \cup F, E)$ be the Tanner graph of a (d_X, d_F)-regular LDPC code with $d_X, d_F \geq 3$. Let $\mathcal{P}$ be the fundamental polytope of the parity-check matrix defined by T. Suppose that some $\mathbf{v} \in \mathcal{P}$ satisfies*

$$\boldsymbol{\lambda} \cdot \mathbf{v} < \boldsymbol{\lambda} \cdot \boldsymbol{\omega}$$

for every $\boldsymbol{\omega} \in \mathcal{P} \setminus \{\mathbf{v}\}$, and that an admissible probability measure exists. Then $\mathbf{v}$ is a vertex of $\mathcal{P}$, and universal cover decoding restricted to graph cover configurations, as described above, agrees with LP decoding; in other words, $\mathrm{UC}\,|_{\mathrm{GC}}(\mathbf{y}) = \mathbf{v}$, where $\mathbf{y}$ is the channel output.

Proof. That $\mathbf{v}$ must be a vertex of $\mathcal{P}$ is clear. Write $X = \{x_1, \ldots, x_n\}$. Since $\mathbf{v}$ is the unique value of $\mathrm{argmin}\{\boldsymbol{\lambda} \cdot \boldsymbol{\omega} \,|\, \boldsymbol{\omega} \in \mathcal{P}\}$, a configuration $\widetilde{\mathbf{c}}$ on a finite connected cover of T induces a configuration on $\widehat{T}_{x_i}$ of minimal rooted cost among all pseudocodewords induced by connected graph cover pseudocodewords if and only if its corresponding normalized graph cover pseudocodeword is $\mathbf{v}$. Since the probability measure used for universal cover decoding is admissible, we have that the probability that an arbitrarily chosen element of the minimal rooted cost configurations on $\widehat{T}_{x_i}$ assigns a binary value of 1 to the root node x_i is v_i. Thus, $\mathrm{UC}\,|_{\mathrm{GC}}(\mathbf{y}) = \mathbf{v}$. ■

The proposed universal cover decoder agrees with linear programming decoding under the conditions described in Proposition 4.3. Further research on universal cover decoding should help to solidify our understanding of both LP decoding and iterative message-passing decoding by providing the missing link between these two sets of decoders.

References

[1] N. Axvig, D. Dreher, K. Morrison, E. Psota, L.C. Pérez, and J. L. Walker. Towards a universal theory of decoding and pseudocodewords. SGER Technical Report 0801, University of Nebraska-Lincoln, http://www.math.unl.edu/∼jwalker7, March 2008.

[2] N. Axvig, E. Price, E. Psota, D. Turk, L.C. Pérez, and J.L. Walker. A universal theory of pseudocodewords. In *Proceedings of the 45th Annual Allerton Conference on Communication, Control, and Computing*, September 2007.

[3] J. Feldman. *Decoding Error-Correcting Codes via Linear Programming*. PhD thesis, Massachusetts Institute of Technology, Cambridge, MA, 2003.

[4] C. Kelley and D. Sridhara. Pseudocodewords of Tanner graphs. *IEEE Transactions on Information Theory*, 53:4013–4038, November 2007.

[5] 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.

[6] R. Ortner and W. Woess. Non-backtracking random walks and cogrowth of graphs. *Canadian Journal of Mathematics*, 59(4):828–844, 2007.

[7] V. V. Prasolov. *Elements of combinatorial and differential topology*, volume 74 of *Graduate Studies in Mathematics*. American Mathematical Society, Providence, RI, 2006. Translated from the 2004 Russian original by Olga Sipacheva.

[8] 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*.

[9] N. Wiberg. *Codes and Decoding on General Graphs*. PhD thesis, Linköping University, Linköping, Sweden, 1996.