SIAM J. DISCRETE MATH.
Vol. 11, No. 4, pp. 664–671, November 1998
© 1998 Society for Industrial and Applied Mathematics
011

A NEW DECODING ALGORITHM FOR COMPLETE DECODING OF LINEAR BLOCK CODES*

YUNGHSIANG S. HAN†

Abstract. In this paper we present and describe an improved version of the Zero-Neighbors algorithm, which we call the Zero-Coverings algorithm. We also present a method for finding a smallest subset of codewords (Zero-Coverings) which need to be stored to perform the Zero-Coverings algorithm. For some short codes, the sizes of Zero-Coverings are obtained by computer searches; for long codes, an asymptotic bound on the sizes of such subsets is also given.

Key words. coding, decoding, linear codes, block codes

AMS subject classifications. 94B35, 94B05

PII. S0895480197323974

1. Introduction. In general, complete decoding [11] for a linear block code has proved to be an NP-hard computational problem [1]. That is, it is unlikely that a polynomial time (space) complete decoding algorithm for a linear block code can be found. A new decoding algorithm, the Zero-Neighbors algorithm (ZNA) [9], using the concept of a Zero-Neighbors, was proposed. Only the codewords in a Zero-Neighbors need to be stored and used in the decoding procedure. The size of a Zero-Neighbors is very small compared to $\min(2^k, 2^{n-k})$ for $n >> 1$ and a wide range of code rates $R = k/n$. An improvement of the Zero-Neighbors algorithm, the Zero-Guards algorithm (ZGA), was recently presented [7, 10]. The ZGA further reduces the number of codewords to be stored. The special set of these codewords is called *Zero-Guards*. The time and space complexity of the ZNA and ZGA are determined by the sizes of the Zero-Neighbors and the Zero-Guards used, respectively. The problem here is how to find the smallest subset of codewords that can be used to perform the ZNA-like decoding procedure. We call all the decoding algorithms that perform a ZNA-like decoding procedure "ZNA-like" algorithms. Similarly, we call any subset of codewords that can be used to perform a ZNA-like algorithm procedure a "ZN-like" subset of codewords. The ZN-like subset of codewords with the smallest size is called an "optimal ZN-like set." Furthermore, a ZNA-like algorithm using an optimal ZN-like set is denoted as an "optimal ZNA-like" algorithm.

In this paper we present an optimal ZNA-like algorithm, the Zero-Coverings algorithm, and give a systematic way in which to find an optimal ZN-like set, a Zero-Coverings. Furthermore, an asymptotic bound on the size of an optimal ZN-like set is derived for long codes. In section 2 we briefly review the Zero-Neighbors and the Zero-Guards algorithms. In section 3 we give a description of the Zero-Coverings algorithm and, in the next section, properties of Zero-Coverings are presented. We also give a systematic way to find Zero-Coverings. Simulation results and an asymptotic bound on the size of a Zero-Coverings are given in section 5. Remarks and conclusions are given in section 6.

*Received by the editors July 7, 1997; accepted for publication (in revised form) January 29, 1998; published electronically September 1, 1998. This work was supported by National Science Council ROC grant NSC 87-2218-E-260-002. Portions of this research were presented at the IEEE International Symposium on Information Theory, Ulm, Germany, June 1997.

http://www.siam.org/journals/sidma/11-4/32397.html

†Department of Computer Science and Information Engineering, National Chi Nan University, Puli NanTou, Taiwan, 545 R.O.C. (yshan@csie.ncnu.edu.tw).

2. The Zero-Neighbors and the Zero-Guards algorithms. In this section we briefly describe the ZNA and an improved version of it, the ZGA. First, we give some definitions.

Let $\boldsymbol{Z}$ be the set of all binary vectors of length n, and let $\boldsymbol{C} \subset \boldsymbol{Z}$ be a binary linear block code. Let $d(\boldsymbol{x}_1, \boldsymbol{x}_2)$ denote the Hamming distance between $\boldsymbol{x}_1, \boldsymbol{x}_2 \in \boldsymbol{Z}$. Let $w(\boldsymbol{x}) = d(\boldsymbol{x}, \boldsymbol{0})$ denote the Hamming weight of $\boldsymbol{x}$ and let $\oplus$ denote the modulo-2 addition. Furthermore, let $d_{\min}$ be the nonzero minimum weight of codewords in $\boldsymbol{C}$. In this paper we will assume that $d_{\min} \geq 2$.

DEFINITION 2.1. *The domain $D(\boldsymbol{c})$ of a codeword $\boldsymbol{c} \in \boldsymbol{C}$ is the set of all $\boldsymbol{x} \in \boldsymbol{Z}$ such that $d(\boldsymbol{x}, \boldsymbol{c}) \leq d(\boldsymbol{x}, \boldsymbol{c}')$, for all $\boldsymbol{c}' \in \boldsymbol{C}$.*

DEFINITION 2.2. *The vicinity $B(\boldsymbol{x})$ of $\boldsymbol{x} \in Z$ is the set of all $\boldsymbol{y} \in Z$ such that $d(\boldsymbol{x}, \boldsymbol{y}) = 1$. The domain frame $G(\boldsymbol{c})$ of a codeword $\boldsymbol{c} \in \boldsymbol{C}$ is the set $G(\boldsymbol{c}) = \bigcup_{\boldsymbol{x} \in D(\boldsymbol{c})} B(\boldsymbol{x}) - D(\boldsymbol{c})$.*

DEFINITION 2.3. *A set of Zero-Neighbors (ZN) is a set N_0 of codewords such that*

$$G(\boldsymbol{0}) \subset \bigcup_{\boldsymbol{c} \in N_0} D(\boldsymbol{c}), \text{ where}$$

$$|N_0| = \min \left\{ |N| \mid N \subset \boldsymbol{C}, G(\boldsymbol{0}) \subset \bigcup_{\boldsymbol{c} \in N} D(\boldsymbol{c}) \right\}.$$

It can be shown that if $\boldsymbol{x} \notin D(\boldsymbol{0})$, there exists a $\boldsymbol{c} \in N_0$ such that $w(\boldsymbol{x} \oplus \boldsymbol{c}) < w(\boldsymbol{x})$. Thus, the Zero-Neighbors algorithm is as follows.

Algorithm. Let $\boldsymbol{y} = \boldsymbol{y}_0 \in \boldsymbol{Z}$ be the received vector to be decoded. At the ith step of the algorithm we calculate $w(\boldsymbol{y}_{i-1} \oplus \boldsymbol{c})$ for all $\boldsymbol{c} \in N_0$. If there exists a $\boldsymbol{c}_i \in N_0$ such that $w(\boldsymbol{y}_{i-1} \oplus \boldsymbol{c}_i) < w(\boldsymbol{y}_{i-1})$, we set $\boldsymbol{y}_i = \boldsymbol{y}_{i-1} \oplus \boldsymbol{c}_i$ and go to the next step; otherwise, the algorithm terminates. If the algorithm terminates at the $(m+1)$th step, then $\boldsymbol{y}_m = \boldsymbol{y} \oplus \sum_{i=1}^{m} \boldsymbol{c}_i \in D(\boldsymbol{0})$ and can be taken as a coset leader, while $\boldsymbol{c} = \sum_{i=1}^{m} \boldsymbol{c}_i \in \boldsymbol{C}$ is a codeword that is one of the closest to $\boldsymbol{y}$.

We need only to store a ZN to accomplish this algorithm. It can be shown that the number of steps m mentioned above is less than or equal to $n - \lfloor \frac{d_{\min}}{2} \rfloor$. Furthermore, if $\boldsymbol{1}$ is in $\boldsymbol{C}$, then $m \leq \lfloor \frac{n+1}{2} \rfloor$. Another improved version of the ZNA, the ZGA, is described next.

DEFINITION 2.4. *The frontier $F(\boldsymbol{0})$ of $\boldsymbol{0}$ is the set of all $\boldsymbol{x} \in \boldsymbol{Z}$ such that all its proper descendants* [12] *belong to $D(\boldsymbol{0})$ and $\boldsymbol{x} \notin D(\boldsymbol{0})$.*

DEFINITION 2.5. *A Zero-Guards (ZG) is a set RN_0 of codewords such that*

$$F(\boldsymbol{0}) \subset \bigcup_{\boldsymbol{c} \in RN_0} D(\boldsymbol{c}), \text{ where}$$

$$|RN_0| = \min \left\{ |N| \mid N \subset \boldsymbol{C}, F(\boldsymbol{0}) \subset \bigcup_{\boldsymbol{c} \in N} D(\boldsymbol{c}) \right\}.$$

In other words, the set of domains of codewords in RN_0 forms a minimum covering of $F(\boldsymbol{0})$. It is not difficult to see that $F(\boldsymbol{0}) \subset G(\boldsymbol{0})$. Consequently, the number of codewords in a ZG is less than or equal to that in a ZN. The decoding procedure of the Zero-Neighbors algorithm described above can be applied to the Zero-Guards algorithm while we use a ZG instead of a ZN in the procedure.

3. An optimal ZN-like set. In this section we will give a systematic way to find an optimal ZN-like set, a Zero-Coverings (ZC), which is related to a Zero-Guards. First, we give a formal definition of a ZN-like subset of codewords.

DEFINITION 3.1. *A ZN-like subset of codewords, C_{ZN}, is a subset of $\boldsymbol{C}$ with the following property: for every received vector $\boldsymbol{y}$, if $\boldsymbol{y} \notin D(\mathbf{0})$, then there exists a $\boldsymbol{c} \in C_{ZN}$ such that $w(\boldsymbol{y} \oplus \boldsymbol{c}) < w(\boldsymbol{y})$.*

It has been shown that a ZN and a ZG are ZN-like subsets of codewords in [9] and [6], respectively. It is not difficult to see that if N_0 in the algorithm given in section 2 is replaced with C_{ZN}, the algorithm will still perform complete decoding. That is, the algorithm is a ZNA-like algorithm. Since the time and space complexity of any ZNA-like algorithm grow with the size of C_{ZN}, in order to reduce the complexity we need to find the smallest C_{ZN}.

DEFINITION 3.2. *The covering domain $D_c(\boldsymbol{c})$ of a codeword $\boldsymbol{c} \in \boldsymbol{C}$ is the set of all $\boldsymbol{x} \in F(\mathbf{0})$ such that $d(\boldsymbol{x}, \boldsymbol{c}) < d(\boldsymbol{x}, \mathbf{0})$.*

That is, $D_c(\boldsymbol{c})$ contains all vectors in the frontier $F(\mathbf{0})$ such that they are closer to $\boldsymbol{c}$ than to $\mathbf{0}$. Furthermore, if $\boldsymbol{x} \in D(\boldsymbol{c})$, then $\boldsymbol{x} \in D_c(\boldsymbol{c})$ for any $\boldsymbol{x} \in F(\mathbf{0})$.

DEFINITION 3.3. *A set of Zero-Coverings (ZC) is a subset of $\boldsymbol{C}$ such that*

$$F(\mathbf{0}) = \bigcup_{\boldsymbol{c} \in ZC} D_c(\boldsymbol{c}), \text{ where} \tag{1}$$

$$|ZC| = \min \left\{ |N| \mid N \subset \boldsymbol{C}, F(\mathbf{0}) = \bigcup_{\boldsymbol{c} \in N} D_c(\boldsymbol{c}) \right\}. \tag{2}$$

In other words, the set of covering domains of a Zero-Coverings forms a minimum covering of the frontier $F(\mathbf{0})$. The algorithm for solving general minimum covering problems can be found in [5].

There are many properties of the frontier $F(\mathbf{0})$, derived in [6], that can help us to find $F(\mathbf{0})$. We state these properties here without proofs. The details of these properties can be found in [6].

LEMMA 3.4. *Let $S(\boldsymbol{x}, a) = \{\boldsymbol{v} \mid \boldsymbol{v} \in \boldsymbol{Z}, w(\boldsymbol{v}) = a$ and $\boldsymbol{v}$ be a descendant of $\boldsymbol{x}\}$. Then $\boldsymbol{x} \in F(\mathbf{0})$ iff $\boldsymbol{x} \notin D(\mathbf{0})$ and $S(\boldsymbol{x}, w(\boldsymbol{x}) - 1) \subset D(\mathbf{0})$.*

LEMMA 3.5. *If $\boldsymbol{x} \in F(\mathbf{0})$, then there exists at least one $\boldsymbol{c} \in \boldsymbol{C}$ such that $\boldsymbol{x} \in D(\boldsymbol{c})$ and $\boldsymbol{x}$ is a descendant of $\boldsymbol{c}$.*

LEMMA 3.6. *Let $\boldsymbol{x} \in F(\mathbf{0})$. If $d(\boldsymbol{x}, \boldsymbol{c}) < w(\boldsymbol{x})$, then $\boldsymbol{x}$ is a descendant of $\boldsymbol{c}$.*

LEMMA 3.7. *Let $\boldsymbol{y} \in \boldsymbol{Z}$ and $\boldsymbol{y} \notin D(\mathbf{0})$. Then there exists a descendant $\boldsymbol{x}$ of $\boldsymbol{y}$ such that $\boldsymbol{x} \in F(\mathbf{0})$.*

LEMMA 3.8. *For every $\boldsymbol{c} \in \boldsymbol{C}$ and $\boldsymbol{c} \neq \mathbf{0}$ there exists a descendant $\boldsymbol{x}$ of $\boldsymbol{c}$ such that $\boldsymbol{x} \in F(\mathbf{0})$.*

The following are some new results that are related to covering domains.

LEMMA 3.9. *If $\boldsymbol{x} \in F(\mathbf{0})$, then there exists at least one $\boldsymbol{c} \in \boldsymbol{C}$ such that $\boldsymbol{x} \in D_c(\boldsymbol{c})$ and $\boldsymbol{x}$ is a descendant of $\boldsymbol{c}$.*

Proof. Since $\boldsymbol{x} \in F(\mathbf{0})$ and $\boldsymbol{x} \in D(\boldsymbol{c})$ imply that $\boldsymbol{x} \in D_c(\boldsymbol{c})$, by Lemma 3.5, the result holds. □

LEMMA 3.10. *If $\boldsymbol{x} \in D_c(\boldsymbol{c})$, then $\boldsymbol{x}$ is a descendant of $\boldsymbol{c}$.*

Proof. The result follows directly from Lemma 3.6. □

LEMMA 3.11. *If $\boldsymbol{c} \in ZC$, then there exists one $\boldsymbol{x} \in F(\mathbf{0})$ such that $\boldsymbol{x} \in D_c(\boldsymbol{c})$ and $\boldsymbol{x} \notin D_c(\boldsymbol{c}')$, $\boldsymbol{c}' \neq \boldsymbol{c}$, $\boldsymbol{c}' \in ZC$.*

Proof. Assume that there is no $\boldsymbol{x} \in F(\mathbf{0})$ such that $\boldsymbol{x} \in D_c(\boldsymbol{c})$ and $\boldsymbol{x} \notin D_c(\boldsymbol{c}'), \boldsymbol{c}' \neq \boldsymbol{c}$, $\boldsymbol{c}' \in ZC$. Then for every $\boldsymbol{x} \in D_c(\boldsymbol{c})$ and $\boldsymbol{x} \in F(\mathbf{0})$, there exists at least one $\boldsymbol{c}' \in ZC$, $\boldsymbol{c}' \neq \boldsymbol{c}$ such that $\boldsymbol{x} \in D_c(\boldsymbol{c}')$. Therefore, if we remove $\boldsymbol{c}$ from ZC we also have $F(\mathbf{0}) = \bigcup_{\boldsymbol{c} \in ZC} D_c(\boldsymbol{c})$. The above result contradicts the statement that ZC is a minimum set such that $F(\mathbf{0}) = \bigcup_{\boldsymbol{c} \in ZC} D_c(\boldsymbol{c})$. □

Next we need to prove that a ZC is a ZN-like subset of codewords. In order to show this, it is sufficient to prove the following theorem.

THEOREM 3.12. *$\boldsymbol{y} \notin D(\mathbf{0})$ iff there exists one $\boldsymbol{c} \in ZC$ such that $w(\boldsymbol{y} \oplus \boldsymbol{c}) < w(\boldsymbol{y})$.*

Proof. Assume that $\boldsymbol{y} \notin D(\mathbf{0})$. From Lemma 3.7, there exists a descendant $\boldsymbol{x}$ of $\boldsymbol{y}$ such that $\boldsymbol{x} \in F(\mathbf{0})$. Consider a $\boldsymbol{c} \in ZC$ such that $\boldsymbol{x} \in D_c(\boldsymbol{c})$. Hence, $w(\boldsymbol{y} \oplus \boldsymbol{c}) = d(\boldsymbol{y}, \boldsymbol{c}) \leq d(\boldsymbol{y}, \boldsymbol{x}) + d(\boldsymbol{x}, \boldsymbol{c}) < d(\boldsymbol{y}, \boldsymbol{x}) + d(\boldsymbol{x}, \mathbf{0}) = w(\boldsymbol{y})$. Assume that $\boldsymbol{y} \in D(\mathbf{0})$. Then $d(\boldsymbol{y}, \mathbf{0}) \leq d(\boldsymbol{y}, \boldsymbol{c})$ for all $\boldsymbol{c} \in \boldsymbol{C}$. Thus, $w(\boldsymbol{y}) \leq w(\boldsymbol{y} \oplus \boldsymbol{c})$ and no $\boldsymbol{c} \in ZC$, such that $d(\boldsymbol{y} \oplus \boldsymbol{c}) < w(\boldsymbol{y})$. □

Now we prove that ZC is an optimal ZN-like set.

THEOREM 3.13. *A Zero-Coverings is an optimal ZN-like set.*

Proof. Assume that we have a ZN-like subset of codewords, C_{ZN}. Let $\boldsymbol{x} \in F(\mathbf{0})$. Since $\boldsymbol{x} \notin D(\mathbf{0})$, by the properties of C_{ZN}, there exists one $\boldsymbol{c} \in C_{ZN}$ such that $d(\boldsymbol{x}, \boldsymbol{c}) < d(\boldsymbol{x}, \mathbf{0})$. Therefore, $\boldsymbol{x} \in D_c(\boldsymbol{c})$. If we run through all of the elements in $F(\mathbf{0})$, we have a subset of C_{ZN}, denoted as C'_{ZN}, such that

$$F(\mathbf{0}) = \bigcup_{\boldsymbol{c} \in C'_{ZN}} D_c(\boldsymbol{c}).$$

Consequently, any ZN-like subset of codewords will contain a subset that satisfies the above equality. Therefore, by Definition 3.3, a ZC is a ZN-like subset of codewords with the smallest size that satisfies the above equality. □

In general, the ZGA is not an optimal ZNA-like algorithm. One example to illustrate this fact is given in the appendix.

4. Properties of the frontier of 0 and a Zero-Coverings. In this section we give some theorems describing the properties of the frontier of $\mathbf{0}$ and a ZC that can be used to find the ZC.

DEFINITION 4.1. *Let $\boldsymbol{xC}$ be the coset containing $\boldsymbol{x}$. Furthermore, let $w(\boldsymbol{xC})$ be the Hamming weight of a coset leader in $\boldsymbol{xC}$.*

THEOREM 4.2. *$\boldsymbol{x} \in F(\mathbf{0})$ iff $w(\boldsymbol{x}) - 2 \leq w(\boldsymbol{xC}) \leq w(\boldsymbol{x}) - 1$ and for every vector $\boldsymbol{v}$ in $\boldsymbol{xC}$ with $w(\boldsymbol{v}) < w(\boldsymbol{x})$, $w(\boldsymbol{x} \oplus \boldsymbol{v}) = w(\boldsymbol{x}) + w(\boldsymbol{v})$.*

Proof. Assume that $\boldsymbol{x} \in F(\mathbf{0})$. Since $w(\boldsymbol{xC}) < w(\boldsymbol{x})$, then $w(\boldsymbol{xC}) \leq w(\boldsymbol{x}) - 1$. Furthermore, assume that $w(\boldsymbol{xC}) < w(\boldsymbol{x}) - 2$. Let $\boldsymbol{u}$ be a coset leader in $\boldsymbol{xC}$ and Let $\boldsymbol{v}_1$ be an immediate descendant of $\boldsymbol{x}$ which differs from $\boldsymbol{x}$ in the ith position. Furthermore, let $\boldsymbol{v}_2$ be a vector that differs from $\boldsymbol{u}$ only in the ith position. Then $w(\boldsymbol{v}_2) \leq w(\boldsymbol{x}) - 2$ and $w(\boldsymbol{v}_1) = w(\boldsymbol{x}) - 1$. Since $\boldsymbol{u}$ and $\boldsymbol{x}$ are in the same coset, $\boldsymbol{v}_1$ and $\boldsymbol{v}_2$ are also in the same coset. Thus, $\boldsymbol{v}_1 \notin D(\mathbf{0})$. This contradicts the statement that $\boldsymbol{v}_1 \in D(\mathbf{0})$.

Assume that $w(\boldsymbol{x} \oplus \boldsymbol{v}) \neq w(\boldsymbol{x}) + w(\boldsymbol{v})$ for a vector $\boldsymbol{v}$ in $\boldsymbol{xC}$, where $w(\boldsymbol{v}) < w(\boldsymbol{x})$. Then there are two cases to consider:

1. $w(\boldsymbol{v}) = w(\boldsymbol{xC})$. Since $w(\boldsymbol{x} \oplus \boldsymbol{v}) \neq w(\boldsymbol{x}) + w(\boldsymbol{v})$, there exists a position such that $\boldsymbol{x}$ and $\boldsymbol{v}$ are one in that position. Let $\boldsymbol{v}_3$ and $\boldsymbol{v}_4$ be descendants of $\boldsymbol{x}$ and $\boldsymbol{v}$, which differ from them in the position just mentioned, respectively. Since $\boldsymbol{x}$ and $\boldsymbol{v}$ are in the same coset, then $\boldsymbol{v}_3$ and $\boldsymbol{v}_4$ are in the same coset, also. Obviously, $w(\boldsymbol{v}_3) > w(\boldsymbol{v}_4)$. This contradicts the statement that $\boldsymbol{v}_3 \in D(\mathbf{0})$.

2. $w(\boldsymbol{v}) \neq w(\boldsymbol{xC})$. Then $w(\boldsymbol{v}) = w(\boldsymbol{x}) - 1$ and $w(\boldsymbol{xC}) = w(\boldsymbol{x}) - 2$. In this case, the argument is similar to that above.

Now assume that, for every vector $\boldsymbol{v}$ in $\boldsymbol{xC}$ with $w(\boldsymbol{v}) < w(\boldsymbol{x})$, $w(\boldsymbol{x} \oplus \boldsymbol{v}) = w(\boldsymbol{x}) + w(\boldsymbol{v})$ and $w(\boldsymbol{x}) - 2 \leq w(\boldsymbol{xC}) \leq w(\boldsymbol{x}) - 1$. We want to prove that $\boldsymbol{x} \in F(\boldsymbol{0})$. That is, we need to prove that every immediate descendant of $\boldsymbol{x}$ belongs to $D(\boldsymbol{0})$. Let $\boldsymbol{v}$ be any vector in $\boldsymbol{xC}$ such that $w(\boldsymbol{v}) < w(\boldsymbol{x})$. Let $\boldsymbol{v}_5$ be an immediate descendant of $\boldsymbol{x}$ that differs from $\boldsymbol{x}$ in the ith position. Let $\boldsymbol{v}_6$ be a vector that is one in the ith position and that differs from $\boldsymbol{v}$ only in that position. Therefore, $\boldsymbol{v}_5$ and $\boldsymbol{v}_6$ are in the same coset. $\boldsymbol{v}_6$ has a weight of at least $w(\boldsymbol{x}) - 1$ since $w(\boldsymbol{x} \oplus \boldsymbol{v}) = w(\boldsymbol{x}) + w(\boldsymbol{v})$. Therefore, $\boldsymbol{v}_5 \in D(\boldsymbol{0})$. □

Base on the above theorem, we can design an efficient algorithm to find $F(\boldsymbol{0})$ from a standard array. Furthermore, we can find the $D_c(\boldsymbol{c})$ from a standard array by the following theorems. Since the proofs of the theorems are simple, we omit them here.

THEOREM 4.3. *Let $\boldsymbol{x} \in F(\boldsymbol{0})$; $\boldsymbol{x} \in D_c(\boldsymbol{c})$ iff there exists a vector $\boldsymbol{v}$ in $\boldsymbol{xC}$ such that $w(\boldsymbol{v}) < w(\boldsymbol{x})$ and $\boldsymbol{c} = \boldsymbol{v} \oplus \boldsymbol{x}$. Furthermore, if $\boldsymbol{x} \in D_c(\boldsymbol{c})$, then $w(\boldsymbol{c}) = 2w(\boldsymbol{x}) - 2$ or $w(\boldsymbol{c}) = 2w(\boldsymbol{x}) - 1$.*

THEOREM 4.4. *Let $\boldsymbol{x} \in F(\boldsymbol{0})$ and $\boldsymbol{x} \in D_c(\boldsymbol{c})$; then $w(\boldsymbol{xC}) \leq d(\boldsymbol{x}, \boldsymbol{c}) \leq w(\boldsymbol{xC})+1$.*

The following result can be used to derive an upper bound on the size of a ZC.

THEOREM 4.5. *Let r be the covering radius of the code $\boldsymbol{C}$. If $\boldsymbol{c} \in \boldsymbol{C}$ and $w(\boldsymbol{c}) > 2r + 1$, then $\boldsymbol{c} \notin ZC$.*

Proof. Assume that $\boldsymbol{c} \in ZC$. From Lemma 3.11 there exists an $\boldsymbol{x} \in F(\boldsymbol{0})$, $\boldsymbol{x} \in D_c(\boldsymbol{c})$, and $\boldsymbol{x} \notin D_c(\boldsymbol{c}'), \boldsymbol{c}' \neq \boldsymbol{c}$. Since $\boldsymbol{x} \in F(\boldsymbol{0}), w(\boldsymbol{x}) \leq r + 1$ and $d(\boldsymbol{x}, \boldsymbol{c}) \leq r$. Hence, $w(\boldsymbol{c}) = w(\boldsymbol{x}) + d(\boldsymbol{x}, \boldsymbol{c}) \leq 2r + 1$. Therefore, if $w(\boldsymbol{c}) > 2r + 1$, then $\boldsymbol{c} \notin ZC$. □

THEOREM 4.6. *Let $\boldsymbol{c}_1$,$\boldsymbol{c}_2 \in \boldsymbol{C}$ and $\boldsymbol{c}_1$ be a proper descendant of $\boldsymbol{c}_2$. Then, $\boldsymbol{c}_2 \notin ZC$.*

Proof. Assume that $\boldsymbol{c}_2 \in ZC$ and $\boldsymbol{c}_3 = \boldsymbol{c}_1 \oplus \boldsymbol{c}_2$. Then, by Lemma 3.11, there exists an $\boldsymbol{x} \in F(\boldsymbol{0})$ such that $\boldsymbol{x} \in D_c(\boldsymbol{c}_2)$ and $\boldsymbol{x} \notin D_c(\boldsymbol{c}')$, $\boldsymbol{c}' \neq \boldsymbol{c}_2$, $\boldsymbol{c}' \in ZC$. Furthermore, by Lemma 3.6, $\boldsymbol{x}$ is a descendant of $\boldsymbol{c}_2$. By Lemma 3.6, if $d(\boldsymbol{x}, \boldsymbol{c}_1) < w(\boldsymbol{x})$, then $\boldsymbol{x}$ is a descendant of $\boldsymbol{c}_1$. In this case, $d(\boldsymbol{x}, \boldsymbol{c}_2) = d(\boldsymbol{x}, \boldsymbol{c}_1) + w(\boldsymbol{c}_3)$. Since $w(\boldsymbol{c}_3) \geq 2$, by Theorem 4.4, $\boldsymbol{x} \notin D_c(\boldsymbol{c}_2)$, which contradicts the statement that $\boldsymbol{x} \in D_c(\boldsymbol{c}_2)$. Therefore, $d(\boldsymbol{x}, \boldsymbol{c}_1) \geq w(\boldsymbol{x})$. Similarly, we have $d(\boldsymbol{x}, \boldsymbol{c}_3) \geq w(\boldsymbol{x})$. Therefore, $d(\boldsymbol{x}, \boldsymbol{c}_2) = d(\boldsymbol{x}, \boldsymbol{c}_1) + d(\boldsymbol{x}, \boldsymbol{c}_3) - w(\boldsymbol{x}) \geq w(\boldsymbol{x})$. This contradicts the statement that $\boldsymbol{x} \in D_c(\boldsymbol{c}_2)$. □

The above theorem is much less restrictive than Theorem 3 in [9] which states that if $\boldsymbol{c}_1$ and $\boldsymbol{c}_3$ are in N_0, then $\boldsymbol{c}_2 \notin N_0$. The following result gives a low bound on the number of codewords in ZC.

THEOREM 4.7. *All codewords of minimum weight belong to a ZC.*

Proof. Let $\boldsymbol{c}$ be a codeword of minimum weight. From Lemma 3.8, there exists one $\boldsymbol{x} \in F(\boldsymbol{0})$ and $\boldsymbol{x}$ is a descendant of $\boldsymbol{c}$. Thus, $d(\boldsymbol{c}, \boldsymbol{c}') \leq d(\boldsymbol{c}, \boldsymbol{x}) + d(\boldsymbol{x}, \boldsymbol{c}') = w(\boldsymbol{c}) - w(\boldsymbol{x}) + d(\boldsymbol{x}, \boldsymbol{c}')$, where $\boldsymbol{c}' \neq \boldsymbol{c}$ and $\boldsymbol{c}' \in \boldsymbol{C}$. Hence, $d(\boldsymbol{x}, \boldsymbol{c}') \geq w(\boldsymbol{x}) + [d(\boldsymbol{c}, \boldsymbol{c}') - w(\boldsymbol{c})]$. Since $\boldsymbol{c}$ is of minimum weight, $d(\boldsymbol{c}, \boldsymbol{c}') - w(\boldsymbol{c}) \geq 0$. Thus, $d(\boldsymbol{x}, \boldsymbol{c}') \geq w(\boldsymbol{x})$. But since $\boldsymbol{x} \notin D(\boldsymbol{0})$, then $\boldsymbol{x} \in D_c(\boldsymbol{c})$, and $\boldsymbol{x} \notin D_c(\boldsymbol{c}')$. Therefore, $\boldsymbol{c} \in ZC$. □

5. Analysis of the size of a Zero-Coverings. In this section we give sizes of Zero-Coverings for some short codes that are obtained by computer searches. For long codes, an asymptotic bound on the size of a Zero-Coverings is given. As pointed out in section 2, the space and time complexity of the ZCA are determined by the size of a ZC. Therefore, we will focus on the discussion of the size of a ZC.

In Table 1 we give the sizes of the Zero-Coverings for some linear block codes. We also indicate, for comparison, the numbers of codewords and coset leaders for those codes. Since finding a ZC is an NP-hard computational problem (the minimum covering problem), for some codes we can obtain only upper bounds on the sizes of a ZC. The algorithm for solving the minimum covering problem used here is modified from the approximation algorithm given in [5].

TABLE 1
The sizes of Zero-Coverings for some linear block codes.

$code(n, k, d_{\min})$	2^k	2^{n-k}	$\|ZC\|$
$BCH(15, 7, 5)$	128	256	63
$QR(17, 9, 5)$	512	256	≤ 76
$BCH(21, 12, 5)$	4096	512	≤ 189
$QR(23, 11, 8)$	2048	4096	506
$QR(31, 16, 7)$	65536	32768	≤ 2271
$QR(47, 24, 11)$	16777216	8388608	≤ 17296

Now we give an asymptotic bound on the size of a ZC for long codes. The asymptotic bound will be characterized by the function

$$F_{ZC}(R) = \lim_{n\to\infty} 1/n \log_2 |ZC|,$$

where $R = k/n$ is the code rate [9].

THEOREM 5.1.

$$\begin{aligned} F_{ZC}(R) &\leq H_2(2H_2^{-1}(1-R)) - (1-R) \text{ when } R > 0.1887, \\ &\leq R \text{ otherwise,} \end{aligned}$$

where $H_2(x)$ is the binary entropy function of x and H_2^{-1} is the inverse of $H_2(x)$ for $0 \leq x \leq 1/2$.

Proof. For large n, the size of a ZC can be estimated by using the following facts:

1. The number of codewords with weight j, a_j can be estimated by $a_j = \binom{n}{j}/2^{n-k}$ for $j \geq d_{\min}$ [11].
2. For virtually all linear (n, k) codes,

$$r = nH_2^{-1}(1-R) + o(n),$$

where $o(n)$ denotes a function satisfying $\lim_{n\to\infty} o(n)/n = 0$ [4, 2, 8, 3].

3. For virtually all linear (n, k) codes, $d_{\min} \geq nH_2^{-1}(1-R) + o(n)$ [11, 3].

By Theorem 4.5 and fact 1, we have

$$|ZC| \leq \sum_{j=d_{\min}}^{2r+1} a_j.$$

By facts 2 and 3, the above inequality will be

$$|ZC| \leq (r+2)B,$$

where B is the largest value among $a_{d_{\min}}, a_{d_{\min}+1}, \ldots,$ and a_{2r+1}.

If $2r + 1 \leq \lfloor n/2 \rfloor$, then

$$B = a_{2r+1};$$

otherwise

$$B = \binom{n}{\lfloor n/2 \rfloor} / 2^{n-k}.$$

By calculation, when $R > 0.1887$, $2r+1 \le \lfloor n/2 \rfloor$, where $r = nH_2^{-1}(1-R)+o(n)$. Furthermore, by the relation

$$2^{nH_2(\lambda)-o(n)} \le \binom{n}{\lambda n} \le 2^{nH_2(\lambda)},$$

we have

$$\begin{aligned} B &= 2^{n[H_2(2H_2^{-1}(1-R))-(1-R)]} \text{ when } R > 0.1887, \\ &= 2^k \text{ otherwise.} \end{aligned}$$

Since $r + 2 = nH_2^{-1}(1-R) + o(n) + 2 \ll 2^k$ or $2^{n[H_2(2H_2^{-1}(1-R))-(1-R)]}$ when n is large, then

$$\begin{aligned} |ZC| &\le 2^{n[H_2(2H_2^{-1}(1-R))-(1-R)]} \text{ when } R > 0.1887, \\ &\le 2^k \text{ otherwise.} \end{aligned}$$

Therefore,

$$\begin{aligned} F_{ZC}(R) &\le H_2(2H_2^{-1}(1-R)) - (1-R) \text{ when } R > 0.1887, \\ &\le R \text{ otherwise.} \quad \square \end{aligned}$$

We remark here that the asymptotic bound turns out to be the same as that for the size of a Zero-Neighbors presented in [9] that is based on a geometric argument. However, the argument used here is simpler and more direct than that used in [9].

6. Conclusions. In this paper we have presented an improved ZNA-like decoding algorithm, the Zero-Coverings algorithm. The time and space complexity analysis of this algorithm are also given. Although the asymptotic bound given here indicates that the complexity of this algorithm is growing exponentially with code length n, from the computer simulation, a good computation gain can be obtained. For example, by the results in Table 1, the computation gain for code $(47, 24, 11)$ is at least $(2^{23}/17296)/24 = 20$. However, due to limitation of the memory and computation power of the computer, we can obtain simulation results only for short codes.

The decoding procedure presented here is a complete decoding procedure [11]. That is, the procedure always finds the codeword that is closest to the received vector. The procedure can be modified to an incomplete decoding (bounded-distance decoding) procedure in order to further reduce the decoding computation needed. Furthermore, although the decoding algorithm presented in this paper is designed for binary linear block codes, it can be generalized to nonbinary linear block codes.

Appendix. In this appendix we give an example to show that ZGA is not an optimal ZNA-like algorithm. Let code $(12, 5, 3)$ be a linear code generated by the following generating matrix:

$$\boldsymbol{G} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix}.$$

From computer simulation we have

$$ZG = ZC \cup \{111100000000\},$$

where ZC is a set containing 12 codewords. Thus, $|ZC|$ is less than $|ZG|$.

Acknowledgments. I would like to thank the reviewer for his invaluable suggestions, which I believe have helped me to improve the presentation of my paper. In addition, I would like to thank Elaine Weinman for her invaluable help on the language check.

REFERENCES

[1] E. R. Berlekamp, R. J. McEliece, and H. C. A. van Tilborg, *On the inherent intractability of certain coding problems*, IEEE Trans. Inform. Theory, IT-24 (1978), pp. 384–386.

[2] V. M. Blinovskii, *Lower asymptotic bound on the number of linear code words in a sphere of given radius in f_q^n*, Problemy Peredachi Informatsii, 23 (1987), pp. 50–53.

[3] J. T. Coffey and R. M. Goodman, *The complexity of information set decoding*, IEEE Trans. Inform. Theory, 36 (1990), pp. 1031–1037.

[4] G. Cohen, *A nonconstructive upper bound on covering radius*, IEEE Trans. Inform. Theory, 29 (1983), pp. 352–353.

[5] T. H. Cormen, C. E. Leiserson, and R. L. Rivest, *Introduction to Algorithms*, MIT Press, Cambridge, MA, 1991.

[6] Y. S. Han and C. R. P. Hartmann, *The zero-guards algorithm for general minimum distance decoding problem*, IEEE Trans. Inform. Theory, 43 (1997), pp. 1655–1658.

[7] C. R. P. Hartmann and L. B. Levitin, *An improvement of the zero-neighbors minimum distance decoding algorithm: The zero-guard algorithm*, IEEE Internat. Symp. on Information Theory, Kobe, Japan, 1988.

[8] L. B. Levitin, *Covering radius of almost all linear codes satisfies the Goblick bound*, IEEE Internat. Symp. on Information Theory, Kobe, Japan, 1988.

[9] L. B. Levitin and C. R. P. Hartmann, *A new approach to the general minimum distance decoding problem: The zero-neighbors algorithm*, IEEE Trans. Inform. Theory, 31 (1985), pp. 378–384.

[10] L. B. Levitin, M. Naidjate, and C. R. P. Hartmann, *Generalized identity-guards algorithm for minimum distance decoding of group codes in metric space*, IEEE Internat. Symp. on Information Theory, San Diego, CA, 1990.

[11] F. J. MacWilliams and N. J. A. Sloane, *The Theory of Error-Correcting Codes*, Elsevier, New York, 1977.

[12] W. W. Peterson, *Error-Correcting Codes*, 2nd ed., MIT Press, Cambridge, MA, 1972.