Some Results on Bit/Byte Bounded Codes and Proximity Detecting Codes (A Brief Note)*

Nitin Vaidya
Department of Computer Science
Texas A&M University
College Station, TX 77843-3112
vaidya@cs.tamu.edu

Technical Report 99-013

May 26, 1999†

Abstract

This report presents some new results on two classes of codes: (t, u)-bit/byte bounded codes that can handle up to u bytes in error, provided that each byte contains at most t erroneous bits, and t-proximity detecting codes that can detect if a received word is within distance t of the transmitted codeword.

1 Introduction

In this report, we consider two types of codes, and present some new results. First class of codes is referred to as the bit/byte-bounded error control codes. (t, u)-codes considered here can handle errors in up to u bytes provided that at most t bits in each byte are in error.

*This research is supported in part by grant MIP 9423735 from the National Science Foundation.
†Revised July 1999 to add acknowledgement of NSF support, omitted by oversight in the original version.

The (t, u)-error model is similar to a symmetric error model in [6, 5], but somewhat different from the t/u-error model used in our previous work on bit/byte bounded errors [4, 3]. The second class of codes is proximity detecting codes [1].

2 (t, u)-Unidirectional Error Detecting (UED) Codes

We first derive a lower bound on the number of checkbits needed in a systematic (t, u)-UED code.

Theorem 1 *A systematic (t, u)-UED code must use at least $\lceil u \log_2(t + 1) \rceil$ checkbits.*

Proof: Consider a data word D that includes u non-zero words such that t bits in each non-zero byte are 1. Also, consider all the data words that are *covered* by D (a word W is covered by D if D has a 1 in each bit position where W has a 1). There are $(t + 1)^u$ such data words, including D itself. Let the set of these $(t+1)^u$ data words be called S. Since any data word in set S can be changed into any other data word S, due to a t/u-error, distinct checkbits must be associated with each data word in set S. Therefore, at least $\lceil u \log_2(t+1) \rceil$ are needed. □

For $u = 2$, the table below lists the lower bound for various values of t.

t	1	2,3	4	$5 \leq t \leq 7$	$8 \leq t \leq 10$	$11 \leq t \leq 15$	$16 \leq t \leq 21$
lower bound	2	4	5	6	7	8	9

Optimal $(t, 2)$-UED codes

References [2, 3] present the design of a $t/1$-unidirectional error correcting code, using a single digit error correcting code, say C_1. The basic idea behind this design is to encode the *number* of non zero bits in each byte, modulo $t + 1$, as the data. If C_1 is used as an error correcting code, then it can pinpoint the actual byte (at most one such byte is allowed) in which the number of non-zero bits has changed – this results in the $(t, 1)$-UEL capability[1] On the other hand, if C_1 is used for error detection, then it can be used to detect up to 2 bytes in which the number of non-zero bits has changed. This yields a $(t, 1)$-UED capability.

[1]Actually, the codes in [2, 3] are defined to be $t/1$-UEC. In a t/u-error, at most t bits in up to u bytes are erroneous. However, note that, t/u-errors and (t, u)-errors are identical for $u = 1$.

The above discussion implies that the $(t,1)$-UEC codes in [2, 3] are also $(t,2)$-UED. Additionally, it turns out that for several values of t, the $(t,2)$-UED codes thus obtained match the lower bound on the number of checkbits in the table above. The above design can be easily extended to obtain (t,u)-UED codes by choosing C_1 to be u-error detecting code, for the given u.

3 Non-Binary Proximity Detecting Codes

Proximity detecting codes [1] are useful to detect when a received word is within a specified distance of the transmitted codeword. In our previous work, we considered design of binary proximity detecting codes. Now, we consider non-binary proximity detecting codes. In this case, each digit in a codeword is non-binary. We assume that the sender and the receiver are connected by a bus, which is initialized to all-0. The sender sends all the digits of the codeword together. At the receiver, all non-zero bits representing a single digit arrive together, however, bits in different bytes take different amounts of time to arrive. (Alternatively, the bus may use multi-valued logic that can carry non-binary digits.)

A t-digit proximity detecting code (t-DPD) will let the receiver determine whether it has received all but, at most, t non-zero digits of the codeword.

Let X_i denote the i-th digit of X.

Definition 1 *For X and Y, $X \subseteq Y$ if and only if, for all i, either $X_i = 0$ or $X_i = Y_i$.*

If $X \subseteq Y$, we say that Y covers X.

Definition 2 *If (i) $A \subseteq X$ and $A \subseteq Y$, and (ii) for any $B \leq A$, if $B \subseteq X$ and $B \subseteq Y$, then $B \leq A$, then A is said to be the maximum common subset of X and Y, and denoted as $M(X,Y)$.*

Definition 3 *Weight of a word X is the number of non-zero digits in X.*

Theorem 2 *A code C with minimum codeword weight t is t-digit proximity detecting (t-DPD) if and only if for any $X,Y \in C$, such that $X \neq Y$, one of the following conditions is true: (a) $N(X,M(X,Y)) = N(Y,M(X,Y)) \leq t$ or (b) $N(X,M(X,Y)) > t$ and $N(X,M(X,Y)) > t$.*

Proof: This theorem generalizes a result previously obtained for binary codes [1]. The proof for the theorem is obtained by generalizing a proof in [1]. □

Corollary 1 *Constant weight codes are t-DPD for all values of t.*

Proof: In a constant weight code, weight of each codeword is identical, say W. Since weight(X) = weight$(Y) = W$, if X and Y belong to the constant weight code, it follows that $N(X, M(X,Y)) = N(Y, M(X,Y))$. Therefore, by Theorem 2, the code is t-DPD for any $t \leq W$. Also, if $t > W$, them t exceeds weight of every codeword in the constant weight code – in this case, the code is trivially t-DPD. □

4 Summary

This report presents some new results on bit/byte codes and proximity detecting codes.

References

[1] N. H. Vaidya and S. Perisetty, "Systematic proximity-detecting codes," *IEEE Trans. Info. Theory*, pp. 1852–1863, November 1997.

[2] N. H. Vaidya, "Unidirectional error control codes," in *Digest of papers: The* 23^{rd} *Int. Symp. Fault-Tolerant Comp.*, pp. 120–129, June 1993.

[3] N. H. Vaidya, "Unidirectional bit/byte error control," *IEEE Trans. Computers*, vol. 44, pp. 710–714, May 1995.

[4] N. H. Vaidya and D. K. Pradhan, "A new class of bit- and byte-error control codes," *IEEE Trans. Inform. Theory*, vol. 38, pp. 1617–1623, September 1992.

[5] J. K. Wolf, "On an extended class of error–locating codes," *Inform. and Control*, vol. 8, pp. 163–169, April 1965.

[6] J. K. Wolf and B. Elspas, "Error–locating codes – A new concept in error control," *IEEE Trans. Inform. Theory*, vol. 9, pp. 113–117, April 1963.