Can the storage capacity of memories built from unreliable components be determined?

Shashi Kiran Chilappagari, *Student Member, IEEE,* Bane Vasic, *Senior Member, IEEE* and Michael Marcellin, *Fellow, IEEE*

I. Introduction

A memory is a device in which information is stored at some time and retrieved at a later time [1] . Let the information be stored in form of bits in registers (memory elements) each of which can store a single bit. The information storage capability of a memory is the number of information bits it stores [1]. Building a memory with information storage capability of k bits with reliable memory elements requires k registers. Such a memory is termed as an irredundant memory. Now, consider the problem of information storage with unreliable memory elements. Due to the component failures, the information read out of the memory may not be identical to the information stored originally. Hence, to ensure reliable storage, the information needs to be stored in coded form (see [1] for an excellent discussion on the importance of coded form). Initially, a codeword $\underline{c}$ from some error correcting code $\mathcal{C}$ is stored in the memory. The unreliable nature of the memory elements introduces errors in the registers and the contents of the memory differ from the initial state. To ensure reliability, a correcting circuit is employed which performs error correction and updates the contents of the registers with an estimate of the original codeword. Hence, a fault-tolerant memory system (referred to as memory system or simply memory henceforth) consists of memory elements (referred to as storage circuit) and a correcting circuit. Let the correcting circuit be also built of unreliable components. The coding of information along with the correcting circuit introduce redundancy into the memory system.

II. Problem Description

Definition 1: [1] The *complexity* of a memory is the number of components within the memory. A component is a device which either performs an elementary operation or stores a single bit where an elementary operation is any Boolean function of two binary operands.

Note that there can be many memory architectures with different complexities but the same information storage capability.

Definition 2: [1] The *redundancy* of a memory is the ratio of the complexity of the memory to the complexity of an irredundant memory which has the same information storage capability .

Definition 3: [1] Let $\underline{c}$ be the stored codeword. The decoding equivalence class of a codeword $\underline{c}$ is the set of words which decode to $\underline{c}$ when decoded with a decoder built of reliable components. A *memory failure* is said to have occurred if the contents of the memory do not belong to the decoding equivalence class of $\underline{c}$. Arbitrarily reliable information storage is possible in a memory if the probability of memory failure can be made arbitrarily small.

Definition 4: [1] The *storage capacity*, C, of a memory is a number such that for all memory redundancies greater than $1/C$, arbitrarily reliable information storage is possible.

Problem: Consider the problem of building a memory with memory elements and logic gates which fail according to a known random mechanism. What is the minimum redundancy memory in which arbitrarily reliable information is possible?

The above problem can be reformulated into the following question: given n memory cells and m universal logic gates which fail following a known random mechanism, what is the optimal memory and logic gate architecture which stores/processes the maximum number of information bits with arbitrary low probability of error? This complex problem can be divided and reformulated in many ways, but, interestingly, even some of the most fundamental questions related to this problem are still unanswered. For example, it is not known for what range of failure rates reliable information storage is possible. The answers depend on the failure mechanism and we address a few relevant issues in subsequent sections.

III. Motivation

During the past four decades, the decrease in transistor size and the increase in integration factor have led to very small, fast, and power efficient chips. As the demand for power efficiency continues, a wide range of new nano-scale technologies (see [2] for a complete list of references) is being actively investigated for processing and storage of digital data. Although it is difficult to discern which of these approaches will become a technological basis for computers in the future, it is widely recognized that due to their miniature size and variations in technological process, the nano-components will be inherently unreliable. Even in more traditional semiconductor technologies, reducing transistor size has already started affecting circuit reliability, and it is widely believed that transistor failures (both transient and permanent) will

This work is funded by NSF under Grant CCF-0634969, ITR-0325979 and INSIC-EHDR program.

S. K. Chilappagari, B. Vasic and M. Marcellin are with the Department of Electrical and Computer Engineering, University of Arizona, Tucson, AZ, 85721 USA (e-mails: shashic@ece.arizona.edu, vasic@ece.arizona.edu, marcellin@ece.arizona.edu).

become one of the main technological obstacles as the trend of increasing the integration factor continues. Transient failures are also observed in satellite and deep space communication systems due to single event upsets caused by neutrons and alpha particles that strike the silicon [3] as well as by electromagnetic interference and thermal fluctuations. These can cause a glitch that corrupts bits in memory cells or the outputs of a logic gate [4]. The frequency of transient failures per second due to cosmic ray ions is known to be as large as 0.033 [5]. Traditional architectures that ensure fault tolerance are incapable of handling such increased unreliability, and with alternative nano-storage technologies on the industry roadmaps, the development of novel reliable computers is of critical importance.

In traditional models of memory and communication systems with error correction coding, it is assumed that the operations of an error correction encoder and decoder are deterministic and that the randomness (in the form of noise and/or errors) exists only in the channel. However, if digital logic in the encoder and decoder is built of faulty components (devices), then the errors and noise do effect the operations performed, and reduce the reliability of the whole system. Making error correcting codes stronger and transmitters and receivers more complex will not necessarily improve the performance of a system. It is likely that for a given failure mechanism, there is a trade off between receiver complexity and its performance.

IV. Previous Work

Von Neumann [6] was the first to study computation using faulty gates. In [6], he showed that, under certain conditions, increased gate redundancy can lead to increased reliability of a circuit. However, it was shown that, in general, computation by faulty gates with non-zero computational capacity is not possible (see [7], [8]). The study of storage circuits made of unreliable components led to much more optimistic results. Taylor in [1] proved that a memory has an associated information storage capacity, C, such that arbitrarily reliable information storage is possible for all memory redundancies greater than $1/C$. The methodology of the proof, however, does not allow one to explicitly calculate the storage capacity. This construction was further studied by Kuznetsov in [9] and we will refer to it as the Taylor-Kuznetsov (TK) scheme. Hadjicostis [10] was able to generalize Taylor's scheme to fault tolerant linear finite state machines. Spielman [11] obtained the best result for a general model of computation, by marrying the ideas of von Neumann with Reed-Solomon (RS) codes.

In both von Neumann and Taylor-Kuznetsov models, a faulty component, generally a logic gate or a memory element is subject to transient faults, i.e., faults that manifest themselves at particular time steps but do not necessarily persist for later times [10]. It is also assumed that gates fail independently of each other, and that the defects are not permanent, i.e., a gate that malfunctioned at some point in time may give correct output subsequently and that failure occurs by flipping the correct result with some probability, i.e., if the correct result is "1", the gate gives "0" and vice versa (analogous to a binary symmetric channel). Such failure mechanism is referred to as von Neumann type of error. We note that the probability of "1" flipping to "0" and the probability of "0" flipping to "1" do not need to be the same, i.e., the failure can be input dependent. We also note that this model can readily be extended to non-binary logic gates [12], [13] and memory elements [14], [15], [16]. From a theoretical point of view, both von Neumann type of error as well as the transient error models assuming dependent gate failures [17] fall into the class of Pippenger's ϵ-admissible failure model [18]. Taylor also considered a model where components fail permanently but are bad components are periodically replaced. Most of our discussion pertains to von Neumann type of error though results for one model can be generalized to other model.

V. Challenges, Recent Work and Insights

The problem of computing capacity in classical coding theory does not consider the complexity of the decoding algorithm employed. However, from the definition of storage capacity, it is clear that, to compute the complexity and redundancy of a memory, a fixed implementation of the decoder needs to be considered and only decoders whose complexity is linear in codelength can have nonzero storage capacity. In fact, Taylor claims that only iterative decoding of LDPC codes can result is such nonzero capacity. Even in the restricted case of LDPC codes the rate, redundancy and thresholds are interrelated in a complex manner making the problem intractable. The bounds deducible from the work of Taylor and Kuznetsov are approximately in the order of 10^{-6} (those reported in the papers are in the orders of 10^{-8}) and it is not known if these are the best possible bounds. The current state of theoretical research in systems made of unreliable components can be compared to that in the area of communications before the renaissance of LDPC codes and iterative decoding algorithms [19]. The spirit and methodology of Taylor and Kuznetsov's work [1], [9] is similar to Gallager's results [20] on LDPC codes. The large body of knowledge in iterative decoding gained in past decade, especially techniques developed for code construction and optimization as well as techniques for analysis of iterative decoding algorithms have not been exploited so far to improve reliability of memory systems.

Recently, in [21], we gave an analytical characterization of faulty decoders for one step majority logic decoders. Varshney [22] studied effects of decoder noise in message passing algorithms and [22] can be seen as an attempt to apply density evolution to decoders made of unreliable gates. In [23], we showed how expander graph arguments [24] can be applied to show that decoders based on bit flipping algorithms have nonzero storage capacity. The methodology of [23] can be applied to any class of decoders which have the following properties: (a) The decoder is iterative and (b) The decoder is monotonic i.e., the number of errors (or wrong messages) decreases with each iteration.

One observation is that results from adversarial channel can be extended to binary symmetric channel model by Chernoff bounds [25]. The work of Sipser and Spielman on expander

codes [24] and subsequent work by Guruswami and Indyk [26] exhibit a powerful class of linear time encodable and decodable codes for the adversarial channel model. The main challenge in analyzing faulty memories for such codes is to find a suitable way to characterize the errors due to the decoder. How do we characterize a decoder built entirely of unreliable components? Can it modeled as a black box which takes in a corrupted codeword and outputs another corrupted codeword possibly with less number of errors? What happens if the input cannot be decoded by a perfect decoder? Does the faulty decoder return the input? We are not aware of an information theoretic framework to answer such questions.

Apart from proving asymptotic results, an important problem to solve is to determine the performance of finite size memories. From the definition of memory failure, it can be seen that, finite length analysis of iterative decoders plays an important role in determining the performance of such memories. It is well known that we are far from having a complete understanding of iterative decoders for finite lengths with the exception of the binary erasure channel. Techniques for error floor analysis such as trapping sets and pseudocodewords need to be investigated for faulty memories as well.

REFERENCES

[1] M. Taylor, "Reliable information storage in memories designed from unreliable components," *Bell System Technical Journal*, vol. 47, pp. 2299–2337, 1968.

[2] B. Vasic and S. K. Chilappagari, "An information theoretical framework for analysis and design of nano-scale fault-tolerant memories based on low-density parity-check codes," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 54, no. 11, pp. 2438–2446, Nov. 2007.

[3] R. M. Goodman and M. Sayano, "The reliability of semiconductor ram memories with on-chip error-correction coding," *IEEE Trans. Inform. Theory*, vol. 37, pp. 884–896, May 1991.

[4] J. F. Ziegler, "Terrestrial cosmic ray intensities," *IBM Journal of Research and Development*, vol. 42, pp. 117–139, Jan. 1998.

[5] A. Campbell, P. McDonald, and K. Ray, "Single event upset rates in space," *IEEE Trans. Nucl. Sci.*, vol. 39, pp. 1828–1835, Dec. 1992.

[6] J. V. Neumann, *Probabilistic Logics and the Synthesis of Reliable Organisms from Unreliable Components*, ser. Automata Studies. Princeton: Princeton University Press, 1956, pp. 43–98.

[7] R. L. Dobrushin and S. I. Ortyukov, "Lower bound for the redundancy of self-correcting arrangements of unreliable functional elements," *Probl. Inform. Transm.*, vol. 13, pp. 59–65, 1977.

[8] N. Pippenger, "Developments in 'the synthesis of reliable organisms from unreliable gates'," in *Symposia in Pure Mathematics*, 1990, pp. 311–324.

[9] A. Kuznetsov, "Information storage in a memory assembled from unreliable components," *Problems of Information Transmission*, vol. 9, pp. 254–264, 1973.

[10] C. N. Hadjicostis and G. C. Verghese, "Coding approaches to fault tolerance in linear dynamic systems," *IEEE Trans. Inform. Theory*, vol. 51, no. 1, pp. 210–228, Jan. 2005.

[11] M. Sipser and D. Spielman, "Expander codes," *IEEE Trans. Inform. Theory*, vol. 42, no. 6, pp. 1710–1722, Nov. 1996.

[12] H. Kimura, T. Hanyu, and M. Michitaka, "Multiple-valued logic-in-memory VLSI using MFSFETs and its applications," *Journal of Multiple-Valued Logic and Soft Computing*, vol. 9, no. 1, pp. 23–42, 2003. [Online]. Available: http://www.oldcitypublishing.com/MVLSC/MVLSC.htm

[13] D. Olson, "Multiple-valued logic circuit architecture; supplementary symmetrical logic circuit structure (SUS-LOC)," October 17 2000. [Online]. Available: http://www.uspto.gov/patft/

[14] P. Cappelletti, C. Golla, P. Olivo, and E. Zano, *Flash Memories*. Boston, MA: Kluwer, 1999.

[15] B. Eitan, R. Kazerounian, A. Roy, G. Crisenza, P. Cappelletti, and A. Modelli, "Multilevel flash cells and their trade-offs," in *Int. Electron Device Meeting Tech. Dig.*, Dec. 8-11 1996, pp. 169–172.

[16] B. Ricco, G. Torelli, M. Lanzoni, A. Manstretta, H. E. Maes, D. Montanari, and A. Modelli, "Nonvolatile multilevel memories for digital applications," *Proc. IEEE*, vol. 86, no. 12, pp. 2399–2423, Dec. 1998.

[17] R. Bahar, J. Mundy, and J. Chen, "A probability-based design methodology for nanoscale computation," in *Intl. Conf. Computer Aided Design (ICCAD)*. IEEE Press, Nov. 9-13 2003, pp. 480–486.

[18] N. Pippenger, "Invariance of complexity measures for networks with unreliable gates," *J. Assoc. Comput. Mach.*, vol. 36, pp. 531–539, 1989.

[19] D. J. C. MacKay, "Good error-correcting codes based on very sparse matrices," *IEEE Trans. Inform. Theory*, vol. 45, no. 2, pp. 399–431, Mar. 1999.

[20] R. G. Gallager, *Low Density Parity Check Codes*. Cambridge, MA: M.I.T. Press, 1963.

[21] S. K. Chilappagari, M. Ivkovic, and B. Vasic, "Analysis of one step majority logic decoders constructed from faulty gates," in *IEEE International Symposium on Information Theory*, July 9-14 2006, pp. 469–473.

[22] L. R. Varshney, "Performance of ldpc codes under noisy message-passing decoding," in *Information Theory Workshop, 2007. ITW '07. IEEE*, 2-6 Sept. 2007, pp. 178–183.

[23] S. K. Chilappagari and B. Vasic, "Fault tolerant memories based on expander graphs," in *Information Theory Workshop, 2007. ITW '07. IEEE*, 2-6 Sept. 2007, pp. 126–131.

[24] M. Sipser and D. Spielman, "Expander codes," *IEEE Trans. Inform. Theory*, vol. 42, no. 6, pp. 1710–1722, Nov. 1996.

[25] H. Chernoff, "A measure of asymptotic efficiency for tests of a hypothesis based on the sum of observations," *Annals of Mathematical Statistics*, vol. 23, pp. 493–507, 1952.

[26] V. Guruswami and P. Indyk, "Linear-time encodable/decodable codes with near-optimal rate," *IEEE Trans. Inform. Theory*, vol. 51, no. 10, pp. 3393–3400, Oct. 2005.