Schrödinger's CRCs

(Fast Abstract)

Lee Pike
Galois, Inc.
Portland, USA
Email: leepike@galois.com

***Abstract*—I revisit the fault-tolerance of cyclic redundancy checks (CRCs), expanding on the work of Driscoll et al [1]. I introduce the concepts of Schrödinger-Hamming weight and Schrödinger-Hamming distance, and I argue that under a fault model in which stuck-at-one-half or slightly-out-of-spec faults dominate, current methods for computing the fault detection of CRCs may be over-optimistic.**

***Keywords*-cyclic redundancy check (CRC), slightly-out-of-spec, fault-tolerance, reliability**

I. Introduction

CRCs sometimes fail. The rate at which CRCs fail to detect errors is usually computed assuming the probability of bit errors is independent or that bit errors arrive in short bursts. In some cases, this probability calculation may be overly-optimistic. The purpose of this note is to motivate a new way to measure the fault-tolerance of CRCs.

Some faults randomly "flip bits"—e.g., electromagnetic disturbances. Other faults, however, tend to be correlated "in the same direction" for a specific receiver, causing a receiver to interpret 0-bits as 1-bits or vice versa. For example, in a "stuck-at-$\frac{1}{2}$" fault, a transmitter with a weak driver may fail to drive a signal sufficiently high while sending 1-bits or fails to drive a signal sufficiently low while sending 0-bits. Another example in the dimension of time is slightly-out-of-spec faults in which a transmitter's and receiver's timing are slightly off, and so the receiver samples the transmitter's signal either too early or too late. Slightly-out-of-spec faults can have similar effects by causing a receiver to consistently sample high-to-low or low-to-high signal transitions too early or late.

Such faults can be particularly nefarious at the system level when a single sender broadcasts to multiple receivers. Such scenarios can lead to Byzantine (or asymmetric) faults, which are often assumed to be impossible or highly improbable in system design [1], [2].

II. Schrödinger's CRCs

Paulitsch *et al.* describe circumstances in which CRCs are not as effective at detecting faults as designers might assume they are, particularly for ultra-dependable embedded systems [2]. The faults mentioned in the previous section are cited as cases in which the reliability afforded by CRCs can be overestimated. The term "Schrödinger's CRC" was coined by Driscoll *et al.* to describe cases resulting from these sort of faults [1]. (The term is a tribute to the "Schrödinger's cat" thought experiment in quantum mechanics.) The purpose of this note is to expand on Driscoll's brief treatment.

Let all bit errors be *exclusively* one of

- 0s may randomly be flipped to 1s.
- 1s may randomly be flipped to 0s.

Call these *Schrödinger bit errors*. Instances in which a CRC fails to detect a fault due to Schrödinger bit errors are called *Schrödinger CRCs*.

For example, consider the following frame-check sequences (FCSs) computed from the USB-5 polynomial ($x^5 + x^2 + 1$) for 11-bit words [3]:[1]

	11-Bit Message	FCS
Original	`10110110011`	`01001`
Corrupted	`1`**`1`**`11011`**`1`**`011`	**`1`**`1001`

Both FCSs are valid for their respective data-words. Notice that only 0s in the original message are interpreted as 1s—no 1s are interpreted as 0s. One receiver might interpret the original message correctly while the other interprets it in the corrupted manner resulting in a Byzantine fault. (By the way, other corruptions with valid FCSs are possible for this message—e.g., computing the CRC of `1`**`1`**`11011`**`11`**`11` results in `01`**`1`**`01`).

Of course, if enough bit errors of any kind are present in a message and its FCS, a CRC may erroneously pass. The fault-tolerance of CRCs is usually measured by computing their Hamming weights and Hamming distances. The *Hamming weight* (HW) is a function on a data-word width w, a CRC polynomial, and a fixed number of bit errors e, and returns the total number of possible undetected corruptions of the data-words of width w and their FCSs together resulting from e bit errors. The *Hamming distance* (HD) is smallest number of bit errors resulting in a non-zero HW. Koopman and Chakravarty analyze the HWs and HDs of common CRCs [3].

[1] I follow the convention of computing the CRC by first appending the 6-bit word (the length of the polynomial) '000000' onto the lower-order of the data-word and performing polynomial division (in Galois Field 2) over the resultant 16-bit word and returning the remainder as the FCS.

While the concept of a HD might be appropriate for measuring the resilience of a CRC to random bit errors, is not be a good metric for calculating the probability of Schrödinger's CRCs. If the probability of faults like stuck-at-$\frac{1}{2}$ faults or slightly-out-of-spec faults dominate the probability of random bit errors, then a metric for Schrödinger's CRCs better measures the fault-tolerance of CRCs.

Fix a data-word width w, a CRC polynomial, and a number of bit errors e. Then the *Schrödinger-Hamming weight* (SHW) is the total number of possible undetected corruptions of data-words of width w and their FCSs together resulting from e Schrödinger bit errors. The *Schrödinger-Hamming distance* (SHD) is smallest number of Schrödinger bit errors resulting in a non-zero SHW. For our example of USB-5 on 11-bit words, the HD is three (from Koopman and Chakravarty [3]) and our example shows the SHD is three, too.

SHWs and SHDs are bound below by their respective HWs and HDs, since there are strictly fewer combinations of Schödinger bit errors possible than arbitrary errors. However, it appears to be an open question whether for any data-word size and polynomial, the SHD is *strictly* greater than the HD.

It is possible to detect *all* Schrödinger bit errors by encoding a 0 as a 01 bit pattern and a 1 as a 10 bit pattern—a Manchester encoding [4].[2] The ability to detect Schrödinger bit errors further supports the use of Manchester physical-layer encodings, known to have superior fault-detection capabilities, in ultra-critical systems. On the other hand, Schrödinger CRCs must be dealt with if the encoding used is a less-tolerant (but possibly more efficient—Manchester encodings double the message length) encoding in legacy systems. More generally, the interplay between physical-layer encoding and CRCs has not be fully explored, as noted by Paulitsch *et al.* [2].

III. Schrödinger CRC Probability Calculations

Let us informally calculate the probability of a Schrödinger CRC for a particular data-word width and polynomial under a concrete fault model. These calculations are not meant to be a definitive analysis but to provide an intuition about how Schrödinger bit errors can affect the system-level fault-tolerance.

First, a small simulation is used.[3] Each simulation randomly generates a data-word and computes its FCS. Then without loss of generality, it randomly flips from 0 to i of the 0-bits to 1-bits, where i is the total number of 0s in the data-word and FCS together. The simulator counts the percentage of runs in which the Schrödinger bit errors are not caught by the CRC—i.e., a Schrödinger CRC occurs. Carrying out this simulation for the USB-5 polynomial mentioned earlier on 11-bit data-words [3], we get approximately 1.65% of one million generated runs resulting in Schrödinger CRCs.

Suppose that once a component fails, it causes any number of Schrödinger bit errors with equal probability, as in our simulation. This assumption contrasts with the usual bit-error rate assumption of uncorrelated bit errors, or correlated errors over short bursts [2]. However, the assumption is not unreasonable for persistent stuck-at-$\frac{1}{2}$ or slightly-out-of-spec faults.

If the throughput is a modest 48 bits/second, then that's $10,800$ 16-bit messages/hour, so the cumulative probability (applying the cumulative distribution function) of at least one Schrödinger CRC per hour given a 1.65% probability from the simulation is approximately one (i.e., greater than $1 - 1\,e{-}78$). So the probability of consistently observing Schrödinger CRCs *provided* a component fails in a manner resulting in Schrödinger bit errors is nearly one: as noted by Paulitsch *et al.*, "The probability of a Schrödinger's CRC is hard to evaluate. A worst-case estimate of its occurrence due to a single device is the device failure rate" [2].

This probability is *in addition* to the probability of observing undetected bit errors from statistically-independent transient bit errors. For example, assuming the bit-error probability is $1\,e{-}5$, then the cumulative probability of at least three bit errors in a 16-bit message (recall that three is the HD for USB-5) is just less than $5.6\,e{-}13$. And $5.6\,e{-}13$ is a gross upper-bound on the probability of an undetected CRC per message, as the vast majority of the bit errors would still be caught with three or more bit errors. So the cumulative probability of an undetected message corruption is just over $6\,e{-}9$/hour assuming $10,800$ messages/hour.

If the component failure-rate causing stuck-at-$\frac{1}{2}$ or slightly-out-of-spec faults is significantly greater than $6\,e{-}9$/hour, then the probability of consistently observing Schrödinger CRCs dominates the probability of rarely observing uncaught bit errors due to transient faults.

[2] As pointed out to this author by Twan van Laarhoven (private communication, February, 2010).

[3] The simulation code is available at http://leepike.wordpress.com/source-code/crc-hs/.

References

[1] K. Driscoll, B. Hall, H. Sivencrona, and P. Zumsteg, "Byzantine fault tolerance, from theory to reality," in *Computer Safety, Reliability, and Security, 22nd International Conference (SAFECOMP)*, ser. Lecture Notes in Computer Science, 2003, pp. 235–248.

[2] M. Paulitsch, J. Morris, B. Hall, and E. Latronico, "Coverage and the use of cyclic redundancy codes in ultra-dependable systems," in *DSN '05: Dependable Systems and Networks*. IEEE Computer Society, 2005, pp. 346–355.

[3] P. Koopman and T. Chakravarty, "Cyclic redundancy code (CRC) polynomial selection for embedded networks," in *DSN '04: Dependable Systems and Networks*. IEEE Computer Society, 2004, p. 145.

[4] ——, "Analysis of the train communication network protocol error detection capabilities," Carnegie Mellon University, Tech. Rep., 2001.