University of California

Los Angeles

Approximation in Synchronization and Computation

by

Amirhossein Reisizadehmobarakeh

A thesis submitted in partial satisfaction

of the requirements for the degree

Master of Science in Electrical Engineering

2016

Abstract of the Thesis

Approximation in Synchronization and Computation

by

Amirhossein Reisizadehmobarakeh

Master of Science in Electrical Engineering
University of California, Los Angeles, 2016
Professor Lara Dolecek, Chair

Approximate solutions result in lower complexity and expense compared to exact solutions, by tolerating a limited distortion. This thesis in centered on two primary problems: synchronization and computation. We will seek approximate solutions for these two problems throughout the thesis.

The first part of the thesis in concerned with approximation in file synchronization. File synchronization plays an important role in data sharing applications where several users own edited versions of an original file and they need to synch their files with the original one. Previous works have studied bounds and algorithms for exact reconstruction, where the goal is to exactly synchronize the copies of the original file. In contrast, a more challenging scenario is where the copies may not need to be perfectly synched, i.e. it suffices to reconstruct them within a pre-defined distance, in some notion. In this part, we address approximate synchronization from an information-theoretic viewpoint. The model we employ for edition is via a binary deletion channel. Transmitter owns a binary file, which can be the representation of any type of data including text, image, video, etc., and feeds it to the deletion channel. Receiver obtains an edited version of the string

and approximately reconstructs the main sequence along with extra information receives from the transmitter.

In this thesis, we study the approximate synchronization problem, a more relaxed scenario in which the final reconstructed file does not need to be identical to the original file. We study the case when a binary file undergoes deletion errors with some small deletion rate (so that the total number of deletions is linear in file length). We derive an upper bound on the optimal rate of information that the transmitter (owner of the original file) needs to provide to the receiver (owner of the edited file) to allow the receiver to reconstruct the original file to within a predefined target distortion.

The second part of the thesis focuses on approximate in computation, and Hamming distance calculation as a specific type of computation. Performing computation inside the memory unit (and not fetching data to the processing unit) introduces several benefits, e.g. energy and time saving, avoiding bottleneck congestion. Memristors are introduced as the memory units storing data in the resistive arrays. Computation in the memory in performed by measuring the resistance of the resistive elements, each representing one 0/1 bit. However, noisy measurements challenge the addressed scheme proposed before. We explicitly take the effect of noise into the consideration. Confidence bounds quantitatively show how accurate one can perform the computation in the noisy setup compared to the noise-free scenario. With respect to the context of the problem, we model the noise in two different approaches. One is bit-flipping noise, in which resistive components are read in a way similar to a binary symmetric channel with certain error probability. Secondly, a Gaussian model is considered for the noise in which the output of the measurement will be a continuous random variable. We provide confidence bounds for this two noise models and two single and multiple measurements settings.

Lastly, I would like to thank my advisor, Professor Lara Dolecek for guiding and supporting me over the years. I also would like to thank my other thesis committee members, Professor Suhas Diggavi and Professor Christina Fragouli for their guidance through this process.

The thesis of Amirhossein Reisizadehmobarakeh is approved.

Suhas N. Diggavi

Christina Panagio Fragouli

Lara Dolecek, Committee Chair

University of California, Los Angeles

2016

"We know the past but cannot control it. We control the future but cannot know it."

Claude Shannon

Contents

List of Figures

To my mom...

Chapter 1

Introduction

This thesis is concerned with two common operations occur on data files, *synchronization* and *computation*. We open with reviewing some known facts on these two concepts and the new fields of study they introduce.

1.1 Background on Synchronization and Computation

Recently, we have been hearing the term *big data* frequently. Where does this term originate? Every day, we take photos and store them on our devices. We may post our photos on Facebook. We store our files on Dropbox. We share them with our friends, etc. One can imagine how huge is the amount of data being produced/transferred/stored every day. Various operations are applied on data. In this work, we specifically study two of them: synchronization and computation.

File Synchronization

As pointed out before, we transfer our files to our friends on a shared medium. We can consider a two-party communication medium which the two ends are linked to each other. These two parties may share a text file. One party slightly changes

the file at his/her end. How can he/she inform the other party about these changes such that the other party can reconstruct the edited file? Obviously, the first party has to provide the second party some more information about the changes. This procedure is called *file synchronization* or briefly *synchronization* (Figure 1.1). In this context, file may refer to any collection of data, e.g. a binary sequence. The problem of interest here is to study the required supplementary information one party has to provide for the other party for synchronization and algorithms performing reconstruction.

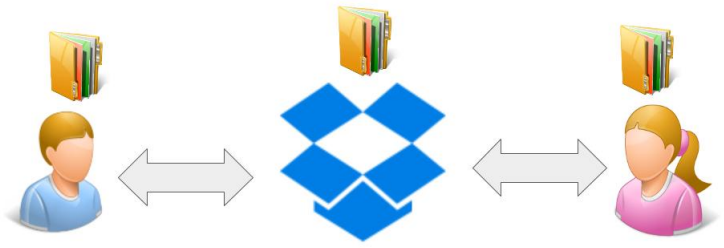

FIGURE 1.1: file synchronization

Several synchronization protocol have been introduces; however, most of them perform exact synchronization, i.e. the copies at the two ends are synchronized perfectly. But, depending on the type of the files, it may not be required to synch them perfectly. Approximate synchronization arises in this situation, allowing the two copies of the file to have a limited distortion. One can easily observe that we can perform the synchronization with less extra information from the first party, in the approximate setup.

In-memory computation

As pointed out before, we deal with huge volume of data nowadays. Computation is one of the canonical operations performed on data. Data is stored in memory unit and computation occurs in precessing unit. Therefore, for any

computation on the data, it firstly needs to be transferred from memory unit to the processing unit. Transfer of data between units has several downsides; it consumes energy and occupies the linking bottlenecks. Therefore, finding efficient scheme for computation is vital in energy/time saving. *In-memory computation* is one solution for this challenge. One may not need to fetch data from memory unit to processing unit for computation. Instead, one may be able to perform computation in the memory unit and as the result, much energy and time would be saved.

We study one specific operation on binary vectors which is Hamming distance calculation, i.e. Hamming distance of two binary vectors stored in the memory is calculated by some measurements inside the memory. Several coding schemes have been proposed to calculate the distance for non-ideal memory components. However, all of these works are along with noise-free measurements assumption. We take the noisy measurements assumption into our consideration in this thesis.

1.2 Outline of Contributions

Below, we present a brief outline containing the contributions of this thesis. The first part of the work is centered on the previously-described concept of approximate synchronization. The second part is concerned with Hamming distance calculation with noisy measurements. Our contributions and future directions for our work are summarized in Chapter 4.

Chapter 2 Contributions

In this chapter, after reviewing some known results from exact synchronization, we formulate the approximate synchronization problem from an information-theoretic point of view. Then, we provide an upper bound on the optimal rate of approximate problem in therms of the optimal one of the exact problem. Firstly,

i.i.d. uniform and non-uniform binary sources are studied. Afrewards, we extend the result to arbitrary M-ary i.i.d. sources.

Chapter 3 Contributions

In Chapter 3, we introduce the notion of noisy measurements in the memory units and study the effect of noise in the accuracy of Hamming distance calculation. Two models will be considered for noise in this context, bit-flipping and Gaussian noise. We provide upper bounds on the deviation of the calculations due to the presence of noise. We then extend our results from single measurement setup to multiple measurement scenario.

Chapter 2

Approximation in Synchronization

2.1 Introduction and motivation

We are living in the era of information where gigantic volumes of data are being produced, transferred, or stored every day. This data expectedly keeps data storage and data links operate constantly which yields in undesirable rise in energy consumption and traffic congestion. However, not the whole volume of data is new, i.e. a significant portion of produced/transferred/stored data is repetitive. Here is the point we can exploit this fact and seek the more efficient schemes for data transmission. One of the most applicable operations between two separate users is data *synchronization*.

The ability to efficiently synchronize large files is critical to the success of sharing resources on the cloud. Since the data being stored in shared mediums grows exponentially every year, it is imperative to use optimized synchronization algorithms and protocols. File synchronization techniques have been developed through a variety of approaches. The popular utility rsync synchronizes files by

combining a strong hash function with a weaker rolling checksum [Tri09]. More recently, there has been a growing body of research from Venkataramanan *et al.* [VZR10, VTR13, VSR15] and Dolecek *et al.* [YD14, BSYD13, SBSD14], that provides synchronization algorithms for the recovery from edit errors for the interactive setting in which the transmitter and the receiver are connected through a two-way communication link. File synchronization has also been studied in interactive communication and coding theory settings [Bra14], and a related problem of set reconciliation in which remote users reconcile sets of unordered objects [MTZ03, MV12].

In this context, edition refers to insertion/deletion/substitution of symbols in a sequence. The insertion/deletion/substitution channel was introduced by Levenshtein [Lev66], and Dobrushin in [Dob67] provided the information coding. This channel was primarily studied by Gallager and Dobrushin in [Gal61] and [Dob67]. Gallager exploited convolutional codes over insertion/deletion/substitution channels to correct synchronization errors and derived lower bounds for achievable rates.

Optimal synchronization under deletion edits is closely associated with the capacity of deletion channel. Although the capacity of deletion channel is still an open problem, different tight bounds have been provided, [DMP07], [KM10], [TKMS10]. Diggavi *et al.* in [DG01] and [DG06] derived lower bounds on the achievable rate for deletion channels, motivated by the transmission of information over finite buffer channels. In [DMP07], Diggavi *et al.* provide two upper bounds deletion channel which one provides an asymptotic upper bound for large deletion probability. [KM10] computes two leading terms of the capacity expansion for small deletion probability and proves that the capacity, up to these two terms, is achieved. A detailed survey on binary deletion channel and related channels with synchronization errors is provided in [Mit09].

In contrast to [VZR10]-[SBSD14] which focus on exact synchronization, in this work we study another interesting case: approximate synchronization. We derive an upper bound on the optimal rate of information that the encoder needs to transmit to allow the decoder to reconstruct the original file to within a predefined target distance. Our elementary derivation adapts the information-theoretic source coding approach from [MRT11] for the approximate synchronization scenario.

Consider two parties A and B communicating in a shared medium. Party A owns a text file and would like B to have a copy of this file. Therefore, party A sends a copy of his own file to other end, B. It is possible that A slightly modifies the text file. But how can he inform B about theses changes? A trivial approach is that A sends out the whole new file to B. One can easily conclude that this approach is much sub-optimal, in this sense that much of the transferred data from A to B already exists on the other end. In other words, A needs not to transfer the whole new file, but it suffices to inform B just about the changes. The procedure in which the two parties inform each other about the changes in their data files is called *file synchronization*. In this context, *file* may refer to any collection of data, e.g. a binary string of a certain length. There are various applications that synchronization plays a critical role in them, e.g. Dropbox, Google drive etc. Sometimes A and B require to reach perfectly synchronized copies of a file. We call this scenario *exact synchronization*. On the other hand, there exists situations where A and B are satisfied with two slightly different copies of a file, similar to each other with respect to a small imperfection. This scenario is named as *approximate synchronization*.

In this thesis, we will consider certain models for file (string) editing. We can point out to three well-studied notions of edition operations on files: deletion, insertion, substitution. In this work, we put our main focus on file synchronization under deletion errors.

2.2 Problem setup

The problem of efficient file synchronization has been studied in [MRT11] and [VTR11] through a source coding approach, where synchronization errors are induced by a deletion channel. Deletion channel and its associated problems are known as hardest ones in information theory.

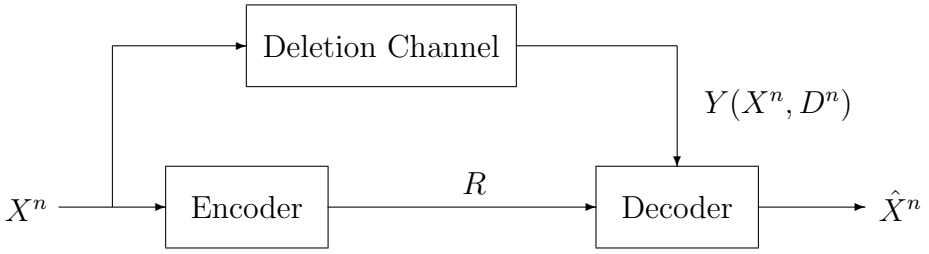

FIGURE 2.1: Sequence synchronization with help of deletion side-information

Throughout the thesis, we use X_i^j to denote the string $(X_i, X_{i+1}, \cdots, X_j)$. We drop the subscript i when $i = 1$. Consider a binary sequence $X^n = (X_1, \cdots, X_n)$, where the X_i's are independently drawn from the $\mathsf{Ber}(1/2)$ distribution. This sequence is fed to a memoryless deletion channel with deletion probability β, $0 < \beta \ll 1$, i.e., every X_i gets deleted independently with this small probability β. The output sequence, $Y(X^n, D^n)$, is a function of the input sequence X^n and the deletion pattern D^n. The deletion pattern is a binary sequence representing the positions of the deleted bits in the input sequence. For instance, if $X^n = (1, 1, 0, 1, 0, 0, 1)$ and $D^n = (0, 1, 0, 0, 1, 1, 0)$, then $Y(X^n, D^n) = (1, 0, 1, 1)$. The goal is to reconstruct sequence X^n at the receiver, provided the transmitter sends additional side-information of rate R. Our set-up is shown in Figure 2.1.

For arbitrary binary sequences of the same length, $X^n = (X_1, \cdots, X_n)$ and $Z^n = (Z_1, \cdots, Z_n)$, we denote the *normalized* Hamming distortion as

$$d_H(X^n, Z^n) = \frac{1}{n} \sum_{i=1}^{n} X_i \oplus Z_i.$$

The set of all binary sequences of any lengths is denoted by $\{0,1\}^*$.

2.3 Exact synchronization

As pointed out in the introduction, there are certain situations in which the two copies of file at the two ends need to be exactly the same. For instance, if the type of file is text file, then even a minor a-synchronization in the two copies would result in corrupting the whole content of the file. Now, we review some results on exact synchronization problem. Figure 2.1 depicts the model for a synchronization scheme. Transmitter (party A) owns string X^n. The other party, owns an edited version of X^n which is $Y(X^n, D^n)$. As the notation explains, the received sequence is the output of a deletion channel, with input X^n and deletion pattern D^n, where $\mathsf{P}(D_i = 1) = \beta$. Therefore, transmitter needs to provide more information about its own sequence for the receiver such that the receiver can exploit this new information along with the edited sequence and reconstruct the sequence $\hat{X}^n$.

We recall the following important results for the exact synchronization case.

Definition 2.1. ([MRT11]) A distributed source code for deletion side-information with parameters $(n, |\mathcal{M}_n|)$ is a tuple (g_n, ψ_n) consisting of an encoding function $g_n \colon \{0,1\}^n \to \mathcal{M}_n$ and a decoding function $\psi_n \colon \mathcal{M}_n \times \{0,1\}^* \to \{0,1\}^n$.

Definition 2.2. ([MRT11]) A real number R_e is called an achievable rate for exact synchronization if there exists a sequence of distributed source codes $\{(g_n, \psi_n)\}_{n \geq 1}$

for deletion side-information with parameters $(n, |\mathcal{M}_n|)$ satisfying

$$\lim_{n\to\infty} \mathbb{P}(X^n \neq \psi_n(g_n(X^n), Y(X^n, D^n))) = 0$$

and $\limsup_{n\to\infty} \frac{1}{n} \log |\mathcal{M}_n| \leq R_e$.

It was shown in [MRT11] that for the exact synchronization problem, the minimum achievable rate is $R_e^* = -\beta \log \beta + \beta(\log 2e - C) + O(\beta^{2-\epsilon})$, for any $\epsilon > 0$ and constant $C = \sum_{l=1}^{\infty} 2^{-l-1} l \log l \approx 1.29$.

Finding the optimal rate for exact synchronization under deletion edits is closely related to the problem of capacity of deletion channel which is known as one of the hardest open problems in information theory ([DMP07]). From information theory, Shannon capacity of a channel is the maximum mutual information between input and output of the channel over all possible input distributions. In [KM10], Kanoria *et al.* showed that the mutual information across the i.i.d. deletion channel with i.i.d. $\mathsf{Ber}(1/2)$ input is

$$\lim_{n\to\infty} \frac{1}{n} I(X^n; Y(X^n, D^n)) = 1 + \beta \log \beta - \beta(\log 2e - C) + O(\beta^{2-\epsilon}),$$

which implies

$$\lim_{n\to\infty} \frac{1}{n} H(Y(X^n, D^n)|X^n) = -\beta \log \beta + \beta(\log 2e - C) + O(\beta^{2-\epsilon}).$$

This result is consistent with the optimal rate R_e^* obtained in [MRT11], in the sense that both have the same leading terms $-\beta \log \beta + \beta(\log 2e - C)$.

2.4 Approximate synchronization

We briefly reviewed some results from exact synchronization problem in the last section. As pointed out before, there are certain applications in which the two copies of the file at the two ends may not need to be perfectly synched. For instance, for two image files, a small distortion between the two images could be tolerated. We call these two copies *approximately* synchronized. Allowing a minor distortion in the synchronization results in saving the bandwidth or smaller rate. In this section we study the problem of approximation from an information theory point of view.

Let us first set up the problem of approximate synchronization, originated from the exact synchronization problem. We define distributed source codes for the approximation problem similar to ones defined for the exact problem (Definition 2.1). Here, encoder provides rate-R_a information to help the decoder reconciling the original file (see Figure 2.1).

Definition 2.3. A real number R_a is called an achievable rate for approximate synchronization if there exists a sequence of distributed source codes $\{(g_n, \psi_n)\}_{n\geq 1}$ for deletion side-information with parameters $(n, |\mathcal{M}_n|)$ satisfying

$$\lim_{n\to\infty} \mathbb{E}[d_H(X^n, \psi_n(g_n(X^n), Y(X^n, D^n)))] \leq d_T$$

and $\limsup_{n\to\infty} \frac{1}{n} \log |\mathcal{M}_n| \leq R_a$, for a pre-defined target distortion $d_T \in [0, 1]$.

Note that the exact synchronization problem is a special case of the approximate problem for which $d_T = 0$. The minimum achievable rate for the approximate synchronization problem is denoted by R_a^*.

2.4.1 Uniform sources

Binary Sources

As the first step, we study the approximate synchronization for i.i.d. $\mathsf{Ber}(1/2)$ sources. Let us first recall some basic facts.

Consider two binary sequences $U^k = (U_1, \cdots, U_k)$ and $V^k = (V_1, \cdots, V_k)$, where U_i's are drawn from the distribution $\mathsf{Ber}(1/2)$ independently. Then,

$$\mathbb{E}[d_H(U^k, V^k)] = \frac{1}{2}.$$

In other words, estimating the input by choosing a sequence of random bits with respect to an arbitrary Bernoulli distribution results in an expected distortion of $\frac{1}{2}$. Therefore, for $d_T \geq \frac{1}{2}$, the minimum achievable rate is $R_a^* = 0$.

Let $X^n = (X_1, \cdots, X_m, X_{m+1}, \cdots, X_n)$ be a binary sequence fed into a deletion channel outputting $Y(X^n, D^n)$, where X_i's are i.i.d. and $\mathsf{Ber}(1/2)$. Assume that $Y(X^m, D^m)$ is the corresponding output for a deletion channel with X^m as input (Figure 2.2). For simplicity, let us denote $[Y(X^n, D^n)]_1^{\lceil m(1-\beta)\rceil}$ as $Y^{(m)}(X^n, D^n)$. Intuitively, we expect sequences $Y(X^m, D^m)$ and $Y^{(m)}(X^n, D^n)$ to convey the same amount of information about input sequence X^m for large enough m. Due to the causality and memoryless assumptions for the channel, $Y(X^n, D^n)$ is a concatenation of corresponding outputs of two input sequences X^m and X_n^{m+1}. Since $\mathbb{E}[|Y(X^m, D^m)|] = m(1-\beta)$, then $Y^{(m)}(X^n, D^n)$ and $Y(X^m, D^m)$ reveal the same information about X^m, asymptotically. More precisely,

$$\lim_{m\to\infty} \frac{1}{m}(I(X^m; Y^{(m)}(X^n, D^n)) - I(X^m; Y(X^m, D^m))) = 0.$$

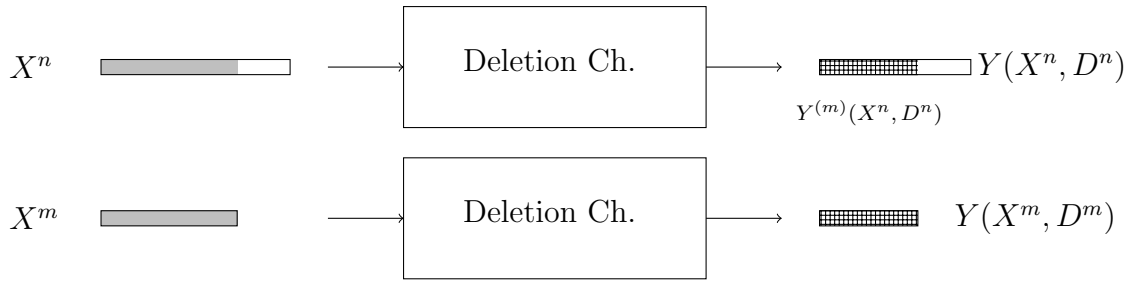

FIGURE 2.2: Relationship between encoded and decoded sequences.

Paper [MRT11] provides explicit form of R_e^* for i.i.d. and non-i.i.d. deletion channels, both with uniform i.i.d. sources. Following theorem relates the rates of the exact and approximate reconstruction for uniform sources.

Lemma 2.4. *Consider two binary sequences* $X^n = (X_1, \cdots, X_n)$ *and* $Y^n = (X_1, \cdots, X_j, Z_{j+1}, \cdots, Z_n)$*, where* X_i*'s and* Z_i*'s are i.i.d and* $\mathsf{Ber}(1/2)$*. Then,* $\mathbb{E}[d_H(X^n, Y^n)] \leq d_T$ *for* $j \geq n(1 - 2d_T)$*.*

Proof. As shown before, $\mathbb{E}[d_H(X_{j+1}^n, Y_{j+1}^n)] = \mathbb{E}[d_H(X_{j+1}^n, Z_{j+1}^n)] = \frac{1}{2}$. Therefore,

$$\begin{aligned}\mathbb{E}[d_H(X^n, Y^n)] &= \mathbb{E}[\frac{1}{n}\sum_{i=j+1}^{n} X_i \oplus Z_i] \\ &= \frac{n-j}{2n} \\ &\leq d_T.\end{aligned}$$

□

Theorem 2.5. *The optimal rate* R_a^* *for approximate synchronization is bounded by* $R_a^* \leq (1 - 2d_T)R_e^*$*, where* d_T *is the target distortion and* R_e^* *is the optimal rate for exact synchronization.*

Proof. It is sufficient to show that rate $R = (1 - 2d_T)R_e^*$ is achievable for approximate synchronization problem. Thus, we need to show that there exists a sequence of distributed source codes $\{(g_n, \psi_n)\}_{n\geq 1}$ for deletion side-information with parameters $(n, |\mathcal{M}_n|)$ satisfying $\lim_{n\to\infty} \mathbb{E}[d_H(X^n, \psi_n(g_n(X^n), Y(X^n, D^n)))] \leq d_T$ and

$\limsup_{n\to\infty} \frac{1}{n} \log |\mathcal{M}_n| \leq (1-2d_T)R_p^*$, for a given target distortion $0 \leq d_T \leq \frac{1}{2}$. As [MRT11] shows, there exists a sequence of source codes $\{(g_m^*, \psi_m^*)\}_{m\geq 1}$ for deletion side-information with parameters $(m, |\mathcal{M}_m^*|)$ satisfying

$$\lim_{m\to\infty} \mathbb{P}(X^m \neq \psi_m^*(g_m^*(X^m), Y(X^m, D^m))) = 0$$

and $\limsup_{m\to\infty} \frac{1}{m} \log |\mathcal{M}_m^*| \leq R_p^*$.

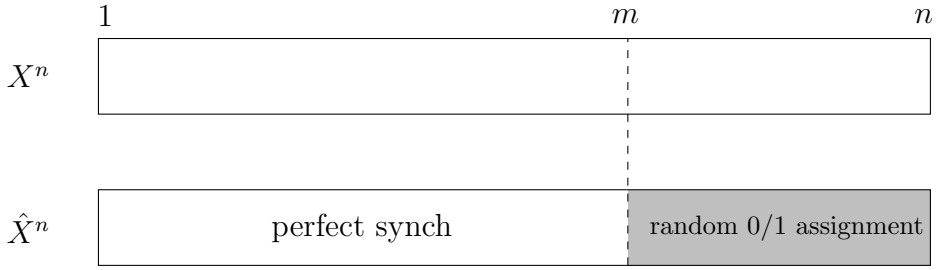

FIGURE 2.3: Approximate reconstruction for uniform source

Now, we need to introduce proper encoding and decoding functions for approximate synchronization problem achieving desired rate. Encoding and decoding functions are denoted by g_n and ψ_n, respectively. For $n \geq 1$, put $m = \lceil n(1-2d_T) \rceil$. We will establish the lossy distributed source code for the approximate problem based on the source code achieving the minimum rate for the perfect problem. For any binary sequence X^n, define the encoding function as

$$g_n(X^n) = g_m^*(X^m),$$

and the decoding function as

$$\mathcal{M}_n = \mathcal{M}_m^*,$$

$$\psi_n(g_n(X^n), Y(X^n, D^n)) = (\psi_m^*(g_m^*(X^m), Y^{(m)}(X^n, D^n)), Z^{n-m}),$$

where Z_i's are randomly drawn from $\mathsf{Ber}(1/2)$ distribution (Figure 2.3). As an

immediate result of Lemma 3.2, the introduced code satisfies the distortion constraint, i.e.,

$$\lim_{n\to\infty} \mathbb{E}[d_H(X^n, \psi_n(g_n(X^n), Y(X^n, D^n)))] \leq d_T.$$

Therefore, we just need to show that this code achieves rate $(1-2d_T)R_p^*$. We can write

$$\begin{aligned}
\lim_{n\to\infty} \sup \frac{1}{n} \log |\mathcal{M}_n| &= \lim_{n\to\infty} \sup \frac{1}{n} \log |\mathcal{M}_m^*| \\
&= \lim_{m\to\infty} \sup \frac{1-2d_T}{m-\delta} \log |\mathcal{M}_m^*| \\
&\leq (1-2d_T)R_p^*,
\end{aligned}$$

note that $m = \lceil n(1-2d_T) \rceil$ yields $m - \delta = n(1-2d_T)$ for some $\delta \in [0, 1)$. □

M-ary Sources

We can extend the results of binary sources to non-binary sources by the similar arguments. Let X^n be a vector such that for every $1 \leq i \leq n$, X_i is uniformly drawn from an M-ary alphabet $\mathcal{X} = \{b_1, \cdots, b_M\}$, independently, i.e. $\mathsf{P}(X_i = b_j) = \frac{1}{M}$ for all $1 \leq j \leq M$. We can define the synchronization problem for M-ary sources just similar to the binary case. The only modification is in the definition of normalized Hamming distance which here is the number of positions in the two vectors normalized by the length of the two sequences.

Following corollary summarizes the the upper bound for M-ary source scenario which can be proved similar to Theorem 2.5.

Corollary 2.6. *The optimal rate R_a^* for approximate synchronization for i.i.d. M-ary sources is bounded by $R_a^* \leq (1 - Md_T)R_e^*$, where d_T is the target distortion and R_e^* is the optimal rate for the exact synchronization.*

To briefly comment on the proof of the above Corollary, notice that two random M-ary sequence of the same length ave the expected distance of $1/M$.

Therefore, to synchronize two sequence of length n within target distance d_T, it suffices to exactly synchronize them in the first $n(1-Md_T)$ positions and randomly assign symbols in remaining positions.

2.4.2 Non-uniform sources

Thus far, we have bounded the optimal rate of the approximation synchronization for i.i.d. $\mathsf{Ber}(1/2)$ sources. A more general problem arises when the source outputs X^n where X_i's are i.i.d. $\mathsf{Ber}(q)$, i.e., $\mathsf{P}(X_i = 1) = 1 - \mathsf{P}(X_i = 0) = q$ for some q between 0 and 1. We define distributed source codes and achievable rates for both exact and approximate synchronization problem in the same manner as we did for uniform sources. The problem of interest is to upper bound the optimum rate of the approximate problem by the one of exact problem.

Let us denote the optimal rate for exact synchronization by $R_e^*(q)$, where q is the source distribution parameter. Similarly, the minimum achievable rate for approximate synchronization is denoted by $R_a^*(q)$. Clearly, the minimum rate is also a function of the deletion probability and the target distortion as well; however, we will omit them from our notation for simplicity. We seek to upper bound $R_a^*(q)$ in terms of $R_e^*(q)$ and target distortion. The key idea is that the decoder reconstructs a portion of the sequence exactly and assigns 0/1 bits to the remaining portion. Notice that finding a closed form expression for R_e^* with arbitrary source distributions is a difficult problem which warrants its own separate study.

Lemma 2.7. *Consider two binary sequences of the same length $U^k = (U_1, \cdots, U_k)$ and $V^k = (V_1, \cdots, V_k)$, where U_i's and V_i's are drawn i.i.d. from the distribution*

$\mathsf{Ber}(q)$ *and* $\mathsf{Ber}(r)$*, respectively. Then,*

$$r^* = \arg\min_r \mathbb{E}[d_H(U^k, V^k)] = \begin{cases} 0 & \text{if } q \leq \frac{1}{2}, \\ 1 & \text{if } q > \frac{1}{2}. \end{cases}$$

As Lemma 2.7 denotes, for i.i.d. distributed input sequence drawn from $\mathsf{Ber}(q)$, if $q \leq \frac{1}{2}$, then estimating the input sequence by assigning an all-one sequence would guarantee the expected distortion to be q. Therefore, for $d_T \geq q$, the minimum achievable rate is $R_e^*(q) = 0$. By the same argument, if $q > \frac{1}{2}$, then the minimum achievable rate is $R_e^*(q) = 0$, for $d_T \geq 1 - q$.

Lemma 2.8. *Consider two binary sequences* $X^n = (X_1, \cdots, X_n)$ *and* $\hat{X}^n = (X_1, \cdots, X_j, Z_{j+1}, \cdots, Z_n)$*, where* X_i*'s are i.i.d.* $\mathsf{Ber}(q)$*. Then,* $\mathbb{E}[d_H(X^n, \hat{X}^n)] \leq d_T$ *for*

- $j \geq n(1 - \frac{1}{1-q} d_T)$*,* $q \leq \frac{1}{2}$*, and* $Z_{j+1}^n = (0, \cdots, 0)$*,*
- $j \geq n(1 - \frac{1}{q} d_T)$*,* $q > \frac{1}{2}$*, and* $Z_{j+1}^n = (1, \cdots, 1)$*.*

Now, we have all the required ingredients to provide an upper bound for the optimal rate of approximate synchronization problem.

Theorem 2.9. *For i.i.d. and non-uniform sources,*

- *if* $q \leq \frac{1}{2}$*,* $R_a^*(q) \leq (1 - \frac{1}{1-q} d_T) R_e^*(q)$*,*
- *if* $q > \frac{1}{2}$*,* $R_a^*(q) \leq (1 - \frac{1}{q} d_T) R_e^*(q)$*.*

Proof. We prove the first case here; the second case can be derived by the same argument. It is sufficient to show that the rate $R = (1 - \frac{1}{1-q} d_T) R_e^*(q)$ is achievable for the approximate synchronization problem. Thus, we need to show that there exists

a sequence of distributed source codes $\{(g_n, \psi_n)\}_{n\geq 1}$ for deletion side-information with parameters $(n, |\mathcal{M}_n|)$ satisfying

$$\lim_{n\to\infty} \mathbb{E}[d_H(X^n, \psi_n(g_n(X^n), y(X^n, D^n)))] \leq d_T$$

and $\limsup_{n\to\infty} \frac{1}{n} \log |\mathcal{M}_n| \leq (1 - \frac{1}{1-q} d_T) R_e^*(q)$, for a given target distortion $0 \leq d_T \leq q$.

From the exact synchronization problem, there exists a sequence of source codes $\{(g_m^*, \psi_m^*)\}_{m\geq 1}$ for deletion side-information with parameters $(m, |\mathcal{M}_m^*|)$ satisfying

$$\lim_{m\to\infty} \mathbb{P}(X^m \neq \psi_m^*(g_m^*(X^m), Y(X^m, D^m))) = 0$$

and $\limsup_{m\to\infty} \frac{1}{m} \log |\mathcal{M}_m^*| \leq R_e^*(q)$.

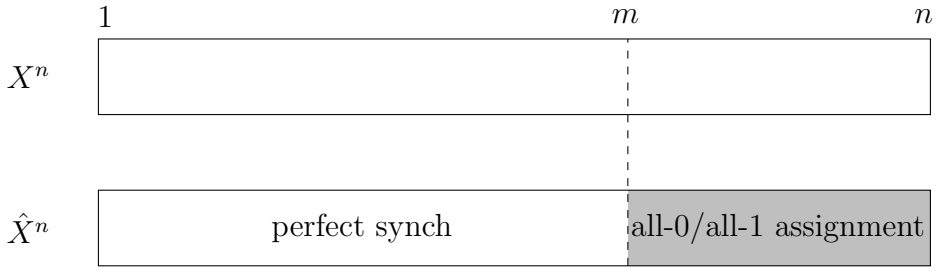

FIGURE 2.4: Approximate reconstruction for non-uniform source

Now, we need to introduce proper encoding and decoding functions for the approximate synchronization problem achieving the desired rate (Figure 2.1). The encoding and decoding functions are denoted by g_n and ψ_n, respectively. For $n \geq 1$, set $m = \left\lceil n(1 - \frac{1}{1-q} d_T) \right\rceil$. We will establish the lossy distributed source code for the approximate problem based on the source code achieving the minimum rate for the exact problem. Define the encoding function as follows: for any binary sequence X^n, $g_n(X^n) = g_m^*(X^m)$, $\mathcal{M}_n = \mathcal{M}_m^*$, and the decoding function

$$\psi_n(g_n(X^n), Y(X^n, D^n)) = (\psi_m^*(g_m^*(X^m), Y^{(m)}(X^n, D^n)), Z^{n-m}),$$

where $Z^{n-m} = (0, \cdots, 0)$, Figure 2.4. As an immediate result of Lemma 2.8, the introduced code satisfies the distortion constraint, i.e.,

$$\lim_{n\to\infty} \mathbb{E}[d_H(X^n, \psi_n(g_n(X^n), Y(X^n, D^n)))] \leq d_T.$$

Therefore, we just need to show that this code achieves rate $(1 - \frac{1}{1-q}d_T)R_e^*(q)$,

$$\begin{aligned}\limsup_{n\to\infty} \frac{1}{n} \log |\mathcal{M}_n| &= \limsup_{m\to\infty} \frac{1 - \frac{1}{1-q}d_T}{m - \varepsilon} \log |\mathcal{M}_m^*| \\ &\leq (1 - \frac{1}{1-q}d_T)R_e^*(q).\end{aligned}$$

Note that $m = \left\lceil n(1 - \frac{1}{1-q}d_T) \right\rceil$ yields $m - \varepsilon = n(1 - \frac{1}{1-q}d_T)$ for some $\varepsilon \in [0, 1)$. □

These results show how efficient the two parties could be in their synchronization scheme, in terms of the optimal exact synchronization scheme. Several exact synchronization schemes have been introduced. In the next section, we will exploit the scheme used for roof of the main theorems of this section to come up with an deterministic algorithm performing approximate synchronization.

2.5 Conclusion

In this chapter, we briefly went over results from exact synchronization problem and then introduced an approximate variation of the file synchronization. We considered the deletion model and formulated the problem through information-theoretic techniques and established an upper bound on the optimal rate for both uniform and non-uniform i.i.d. binary sources. This bound was then extended to M-ary i.i.d. sources. These bounds show that -in the worst case- how much better one can synchronize two files when a limited amount of distortion is allowed, compared to synchronize them perfectly.

Chapter 3

Approximation in Computation

3.1 Introduction and motivation

In recent years, we have been hearing the term "big data" more often. Numerous smart devices with several software applications have been distributed. Unsurprisingly, the volume of data created by this applications are growing faster and faster. This amount of data should be stored in reliable storage devices, leading to establishment of huge data centers. Storage is not the end of the story. Useful data is stored since it most likely will be recalled again later. Any operation on stored data requires the device to recall the information from the memory and bring it to the processing unit. Therefore, transfer of data inside the storage devices is another energy and time consuming characteristics of big data centers. It has been said that in near future, the most energy consuming units will be data centers. Moreover, fetching the data from the memory unit to the processing unit imposes clogging on the bottlenecks. Hence, improving in-memory computations more efficient is vitally critical for energy consuming issues. One applicable operation on binary vectors is calculating the Hamming distance between a pair of vectors stored in the memory. To comment about previous works, Cassuto *et. al*

[CC15] studied the problem of in-memory computation; they first model a memory unit by an array of resistive components which will be called *memristor*. Every resistive has a binary state which can store one bit of information. Then it is shown that how one can calculate the Hamming distance between a pair of row vectors by measuring the equivalent conductance in the memristor. Then the effect of physical non-ideality on the measurement scheme is studied. As expected, non-ideal memristor arrays would require more efforts (in terms of measurements) to calculate the Hamming distance. They proposed several coding constructions such that the distance would be calculated more efficiently even though the resistive elements are not ideal. Memristor arrays have been studied under different issued, as well. As a downside of memristor crossbar arrays, the way these arrays are programmed may undesirably effect the correctness of reads from the memory. *Sneak paths* is the phenomenon causing the accuracy of reads from the memory to depend on the content of the memristor. Cassuto *et. al* [CKY14a], [CKY14b] investigated this issue from an information-theoretic perspective and provided efficient scheme to read the array elements while avoiding sneak paths.

Even though [CC15] solidly models the Hamming distance calculation by array measurements, however, it has been assumed that the reads are noise-free. Therefore, computation under noisy measurements seems a reasonable path for extending their results. We study this problem in this chapter for different scenarios. Reading from the memory is does not occur always perfectly, i.e. the measured value could be a noisy version of the actual stored data. Moreover, the memory device may not be physically ideal, as well. We will investigate the effect of all of these undesired issues on the accuracy of the distance measurement between two row vectors.

3.2 Problem setup

The model we consider for a memory in this context is an array of resistors which will be called *memristor* in this thesis. Every resistive element in the memristor has a binary state, either high-resistance or low-resistance. For the sake of simplicity in calculations, we alternatively use the notion of conductance for resistive elements. We denote the 1-state conductance by G and 0-state conductance by ϵG, where G is a finite conductance and $0 \leq \epsilon < 1$ is the physical parameter denoting the accuracy of the hardware. For instance, $\epsilon = 0$ is translated to physically ideal memristor. In Figure 3.1, high-conductance resistive elements (representing bit "1") and low-conductance resistive elements (representing bit "0") are depicted with black and white elements, respectively.

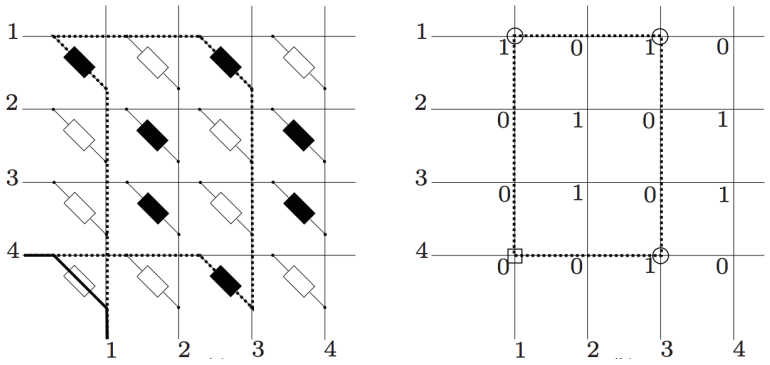

FIGURE 3.1: memristor: arrays of resitances (from [CC15])

For two binary vectors $\mathbf{x} = (x_1, \cdots, x_n)$ and $\mathbf{y} = (y_1, \cdots, y_n)$, we employ the standard definition Hamming weight $W_H(\mathbf{x}) = \sum_{i=1}^{n} x_i$ and Hamming distance $D_H(\mathbf{x}, \mathbf{y}) = \sum_{i=1}^{n} |x_i - y_i|$.

3.2.1 Ideal memristor

We start our model description from the ideal case in which $\epsilon = 0$. Then, the equivalent conductance of the circuit induced by the row pair $\mathbf{x}, \mathbf{y}$ is

$$G_{eq}(\mathbf{x}, \mathbf{y}) = \frac{G}{2} \sum_{i=1}^{n} x_i y_i.$$

For the brevity in the notation, define the normalized equivalent conductance $G_{\mathbf{x},\mathbf{y}} = \sum_{i=1}^{n} x_i y_i$. The following simple derivation expresses the Hamming distance of $\mathbf{x}$ and $\mathbf{y}$ in terms of Hamming weights and eqivalent conductance

$$\begin{aligned} D_H(\mathbf{x}, \mathbf{y}) &= \sum_{i=1}^{n} |x_i - y_i| \\ &= \sum_{i=1}^{n} |x_i - y_i|^2 \\ &= \sum_{i=1}^{n} x_i + \sum_{i=1}^{n} y_i - 2 \sum_{i=1}^{n} x_i y_i \qquad (3.1) \\ &= W_H(\mathbf{x}) + W_H(\mathbf{y}) - 2G_{\mathbf{x},\mathbf{y}}. \qquad (3.2) \end{aligned}$$

The latter expression conveys the fact that having three measurements, $W_H(\mathbf{x})$, $W_H(\mathbf{y})$, and $G_{\mathbf{x},\mathbf{y}}$, one can easily calculate the Hamming distance. Notice that implementation of physically ideal resistive devices may be expensive, therefore, we need to take the effect of non-ideality into our consideration.

3.2.2 Non-ideal memristor

Now, suppose the memristor is not ideal, i.e. $0 \leq \epsilon < 1$. The equivalent conductance of the circuit induced by the row pair $\mathbf{x}, \mathbf{y}$ is

$$G_{eq}(\mathbf{x}, \mathbf{y}) = \frac{G}{2} \sum_{i=1}^{n} x_i y_i + \frac{2\epsilon}{1+\epsilon}(1 - x_i) y_i + \frac{2\epsilon}{1+\epsilon} x_i (1 - y_i) + \epsilon (1 - x_i)(1 - y_i).$$

Similar to the ideal case, define the normalized equivalent conductance $G_{\mathbf{x},\mathbf{y}} = \frac{2}{G} G_{eq}(\mathbf{x}, \mathbf{y})$. We can write

$$\begin{aligned} G_{\mathbf{x},\mathbf{y}} &= \sum_{i=1}^{n} x_i y_i + \frac{2\epsilon}{1+\epsilon}(1-x_i)y_i + \frac{2\epsilon}{1+\epsilon}x_i(1-y_i) + \epsilon(1-x_i)(1-y_i) \\ &= \frac{\epsilon(1-\epsilon)}{1+\epsilon}\sum_{i=1}^{n} x_i + \frac{\epsilon(1-\epsilon)}{1+\epsilon}\sum_{i=1}^{n} y_i + \frac{(1-\epsilon)^2}{1+\epsilon}\sum_{i=1}^{n} x_i y_i + n\epsilon. \end{aligned}$$

Using equation (3.2), Hamming distance can be rephrased as

$$D_H(\mathbf{x}, \mathbf{y}) = \frac{1+\epsilon}{(1-\epsilon)^2}\Big[(1-\epsilon)(W_H(\mathbf{x}) + W_H(\mathbf{y})) + 2n\epsilon - 2G_{\mathbf{x},\mathbf{y}}\Big]. \tag{3.3}$$

This expression denotes that Hamming distance can be calculated by having three measurements, $W_H(\mathbf{x})$, $W_H(\mathbf{y})$, and $G_{\mathbf{x},\mathbf{y}}$, and the accuracy parameter ϵ.

Thus far, we have been implicitly assumed that all the measurements are noise-free; however, this is not usually the case. In the following discussions we will consider the effect of noisy measurements on distance calculations. Before that, we need to mathematically model the noise presented in the measurements.

3.3 Noise modeling: BSC

Different statistical models can be considered for noise in this context. Since we are dealing with binary bits, it seems natural to assume a bit-flipping for the noise like what happens in binary symmetric channel (BSC). More precisely, suppose $\tilde{\mathbf{x}}$ and $\tilde{\mathbf{y}}$ are noisy measurements of $\mathbf{x}$ and $\mathbf{y}$, respectively. We model the noisy measurement by a BSC, i.e. every bit is correctly read from the memory with probability $1-\beta$ and is read as the fllipped bit with probability β. For instance, $\tilde{x}_i$ is the output of a BSC(β) with x_i as input (Figure 3.2). The Hamming distance between vectors $\tilde{\mathbf{x}}$ and $\tilde{\mathbf{y}}$ is denoted by $D_H(\tilde{\mathbf{x}}, \tilde{\mathbf{y}})$.

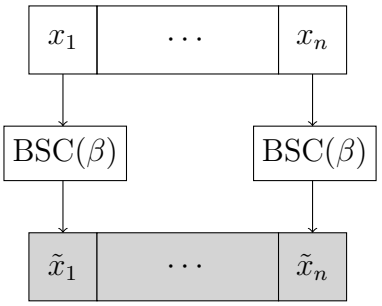

FIGURE 3.2: Noisy measurement: BSC model

3.3.1 Single measurement

Having considered a solid model for the noise, now we can evaluate the Hamming distance perturbed by noise. As the first step, assume every bit is measured once from the ideal memristor. Using equation (3.1), the deviation in the Hamming distance can be written as

$$\begin{aligned} |D_H(\tilde{\mathbf{x}},\tilde{\mathbf{y}}) - D_H(\mathbf{x},\mathbf{y})| &= \left| \sum_{i=1}^{n}(x_i - \tilde{x}_i) + \sum_{i=1}^{n}(y_i - \tilde{y}_i) - 2\sum_{i=1}^{n}(x_i y_i - \tilde{x}_i\tilde{y}_i) \right| \\ &\leq \sum_{i=1}^{n}|x_i - \tilde{x}_i| + \sum_{i=1}^{n}|y_i - \tilde{y}_i| + 2\sum_{i=1}^{n}|x_i y_i - \tilde{x}_i\tilde{y}_i| \,. \end{aligned} \tag{3.4}$$

Taking the expected value from the latter equation and Lemma 3.1 yields

$$\begin{aligned} \mathbb{E}\Big[\, |D_H(\tilde{\mathbf{x}},\tilde{\mathbf{y}}) - D_H(\mathbf{x},\mathbf{y})| \,\Big] &\leq \sum_{i=1}^{n}\mathbb{E}\Big[\, |x_i - \tilde{x}_i| \,\Big] + \sum_{i=1}^{n}\mathbb{E}\Big[\, |y_i - \tilde{y}_i| \,\Big] \\ &\quad + 2\sum_{i=1}^{n}\mathbb{E}\Big[\, |x_i y_i - \tilde{x}_i\tilde{y}_i| \,\Big] \\ &\leq 4n\beta - n\beta^2 \end{aligned}$$

Lemma 3.1. *For independent reads $\tilde{x}_i$ and $\tilde{y}_i$, we have $1-\gamma = P(\tilde{x}_i\tilde{y} = x_i y_i) = 1 - \beta + \frac{\beta^2}{2}$.*

Proof. Since every bit is equally probable 0 or 1, we can write

$$\begin{aligned}
1-\gamma =& P(\tilde{x}_i\tilde{y}_i = x_iy_i)\\
=& \frac{1}{4}p(\tilde{x}_i\tilde{y}_i = 0|x_i = 0, y_i = 0) + \frac{1}{4}p(\tilde{x}_i\tilde{y}_i = 0|x_i = 0, y_i = 1)\\
&+ \frac{1}{4}p(\tilde{x}_i\tilde{y}_i = 0|x_i = 1, y_i = 0) + \frac{1}{4}p(\tilde{x}_i\tilde{y}_i = 1|x_1, y_i = 1)\\
=& \frac{1}{4}\Big\{p(\tilde{x}_i = 0, \tilde{y}_i = 0|x_i = 0, y_i = 0) + p(\tilde{x}_i = 0, \tilde{y}_i = 1|x_i = 0, y_i = 0)\\
&+ p(\tilde{x}_i = 1, \tilde{y}_i = 0|x_i = 0, y_i = 0) + p(\tilde{x}_i = 0, \tilde{y}_i = 0|x_i = 0, y_i = 1)\\
&+ p(\tilde{x}_i = 0, \tilde{y}_i = 1|x_i = 0, y_i = 1) + p(\tilde{x}_i = 1, \tilde{y}_i = 0|x_i = 0, y_i = 1)\\
&+ p(\tilde{x}_i = 0, \tilde{y}_i = 0|x_i = 1, y_i = 0) + p(\tilde{x}_i = 0, \tilde{y}_i = 1|x_i, = 1y_i = 0)\\
&+ p(\tilde{x}_i = 1, \tilde{y}_i = 0|x_i = 1, y_i = 0) + p(\tilde{x}_i = 1, \tilde{y}_i = 1|x_i = 1, y_i = 1)\Big\}\\
=& 1 - \beta + \frac{\beta^2}{2}.
\end{aligned}$$

□

If $W_H(\mathbf{x})$ and $W_H(\mathbf{y})$ are known, by a similar argument,

$$\mathbb{E}\Big[\,|D_H(\tilde{\mathbf{x}}, \tilde{\mathbf{y}}) - D_H(\mathbf{x}, \mathbf{y})|\,\Big] \leq 2n\beta - n\beta^2.$$

As a more general situation, we consider the effect of noisy measurements in the non-ideal memristor array, i.e. $0 < \epsilon < 1$. Cassuto *et. al* [CC15] showed that for enough hardware accuracy, Hamming distance of two vectors can be calculated by only one measurement of equivalent conductance. More explicitly, they proved the following theorem for non-ideal memristors.

Lemma 3.2. *([CC15]) If* $0 < \epsilon < 1/(2n-1)$*, then the Hamming distance* $D_H(\mathbf{x}, \mathbf{y})$ *can be calculated exactly from a single array measurement* $G_{\mathbf{x},\mathbf{y}}$ *by*

$$D_H(\mathbf{x}, \mathbf{y}) = \frac{G_{\mathbf{x},\mathbf{y}} - \lfloor G_{\mathbf{x},\mathbf{y}} \rfloor - \epsilon(n - \lfloor G_{\mathbf{x},\mathbf{y}} \rfloor)}{\epsilon\frac{1-\epsilon}{1+\epsilon}}.$$

Proof. See [CC15]. □

The hardware accuracy in the memristor is translated by parameter ϵ. Smaller ϵ represents higher hardware accuracy. Now, we can bring the hardware accuracy issue into our analyses of Hamming distance deviation.

Let $G_{\tilde{\mathbf{x}},\tilde{\mathbf{y}}}$ be a noisy measurement of the equivalent conductance of the row pair $\mathbf{x}, \mathbf{y}$. For $0 < \epsilon < 1/(2n-1)$, Lemma 3.2 indicates that Hamming distance could be calculated by a single measurement, i.e.

$$D_H(\tilde{\mathbf{x}}, \tilde{\mathbf{y}}) = \frac{G_{\tilde{\mathbf{x}},\tilde{\mathbf{y}}} - \lfloor G_{\tilde{\mathbf{x}},\tilde{\mathbf{y}}} \rfloor - \epsilon(n - \lfloor G_{\tilde{\mathbf{x}},\tilde{\mathbf{y}}} \rfloor)}{\epsilon \frac{1-\epsilon}{1+\epsilon}}.$$

Therefore, we can write the deviation in the Hamming distance as

$$D_H(\tilde{\mathbf{x}}, \tilde{\mathbf{y}}) - D_H(\mathbf{x}, \mathbf{y}) = \frac{1+\epsilon}{\epsilon(1-\epsilon)} \Big\{ G_{\tilde{\mathbf{x}},\tilde{\mathbf{y}}} - G_{\mathbf{x},\mathbf{y}} + (\epsilon - 1) \lfloor G_{\tilde{\mathbf{x}},\tilde{\mathbf{y}}} - G_{\mathbf{x},\mathbf{y}} \rfloor \Big\}.$$

We take the expectation from the absolute value of distance deviation and have

$$\begin{aligned}
\mathbb{E}\Big[\, |D_H(\tilde{\mathbf{x}}, \tilde{\mathbf{y}}) - D_H(\mathbf{x}, \mathbf{y})| \, \Big] &\leq \frac{1+\epsilon}{\epsilon(1-\epsilon)} \Big\{ \mathbb{E}\Big[\, |G_{\tilde{\mathbf{x}},\tilde{\mathbf{y}}} - G_{\mathbf{x},\mathbf{y}}| \, \Big] \\
&\quad + (1-\epsilon) \mathbb{E}\Big[\, | \lfloor G_{\tilde{\mathbf{x}},\tilde{\mathbf{y}}} - G_{\mathbf{x},\mathbf{y}} \rfloor | \, \Big] \Big\} \\
&\leq \frac{1+\epsilon}{\epsilon(1-\epsilon)} \Big\{ (2-\epsilon) \mathbb{E}\Big[\, |G_{\tilde{\mathbf{x}},\tilde{\mathbf{y}}} - G_{\mathbf{x},\mathbf{y}}| \, \Big] + (1-\epsilon) \Big\} \\
&\leq \frac{1}{\epsilon} \Big(2n\beta + n \frac{\beta^2}{2} (3\epsilon - \epsilon^2 - 2) + 1 + \epsilon \Big),
\end{aligned}$$

for $0 < \epsilon < 1/(2n-1)$. Equation (3.3) represents the equivalent conductance between a non-ideal row pair in terms of individual bits. Following lemma is

directly derived from this equation and justifies the inequalities employed in the latter derivations.

Lemma 3.3. *For a non-ideal memristor arrays with accuracy parameter ϵ, the difference between the noisy and actual equivalent conductances can be bounded as*

$$|G_{\tilde{\mathbf{x}},\tilde{\mathbf{y}}} - G_{\mathbf{x},\mathbf{y}}| \leq \frac{\epsilon(1-\epsilon)}{1+\epsilon} \sum_{i=1}^{n} |x_i - \tilde{x}_i| + \frac{\epsilon(1-\epsilon)}{1+\epsilon} \sum_{i=1}^{n} |y_i - \tilde{y}_i| + \frac{(1-\epsilon)^2}{1+\epsilon} \sum_{i=1}^{n} |x_i y_i - \tilde{x}_i \tilde{y}_i| \,. \tag{3.5}$$

3.3.2 Multiple measurements

We have been studied the case in which every stored bit in the memristor is read from the memory only once. For this scenario, we obtained the confidence bound on the measured hamming distance of two sequences $\mathbf{x}$ and $\mathbf{y}$. In order to have more reliable measurements, we can employ multiple reads for every single bit stored in two sequences. Returning to the ideal case, suppose that every bit is measured from the memristor $2m+1$ times independently. More precisely, for a stored bit x_i, we read the memory $2m+1$ times and $\tilde{x}_i^{(1)}, \cdots, \tilde{x}_i^{(2m+1)}$ are the noisy measurements. Therefore, $P(\tilde{x}_i^{(j)} = x_i) = 1 - \beta$, for $1 \leq j \leq 2m+1$.

Now, it is up to us to how to use these measured values to estimate the Hamming distance in a reasonable fashion. We consider two different scenarios for this sake.

Scenario I: One naive way to estimate a good measurement $\tilde{x}_i$ based on these $2m+1$ reads is by majority rule, i.e.

$$\tilde{x}_i = \begin{cases} x_i & \text{if at least } m+1 \text{ of } \tilde{x}_i^{(j)}\text{s agree to } x_i, \\ 1 - x_i & \text{otherwise.} \end{cases}$$

The same argument holds for vector $\mathbf{y}$. Now, we can analyze the confidence bounds for hamming distance measurement in this scenario. First let us review some simple lemmas regarding this scenario.

Lemma 3.4. *In Scenario I, for $2m+1$ reads for sequence $\mathbf{x}$ we have*

$$1-\alpha = \mathsf{P}(\tilde{x}_i = x_i) = \sum_{j=m+1}^{2m+1} \binom{2m+1}{j} (1-\beta)^j \beta^{2m+1-j} \tag{3.6}$$

Proof. Since all the reads are independent, by majority rule,

$$\begin{aligned}\mathsf{P}(\tilde{x}_i = x_i) &= \mathsf{Prob}[\text{at least } m+1 \text{ of } \tilde{x}_i^{(j)}\text{s agree to } x_i] \\ &= \sum_{j=m+1}^{2m+1} \binom{2m+1}{j} (1-\beta)^j \beta^{2m+1-j}\end{aligned}$$

□

Lemma 3.5. *In Scenario I, for $2m+1$ reads for sequences $\mathbf{x}$ and $\mathbf{y}$, we have*

$$1-\alpha' = \mathsf{P}(\tilde{x}_i\tilde{y}_i = x_i y_i) = \sum_{j=m+1}^{2m+1} \binom{2m+1}{j} (1-\gamma)^j \gamma^{2m+1-j}, \tag{3.7}$$

where $\gamma = \beta(1-\beta) + \frac{\beta^2}{2}$.

Proof. Since all the reads are independent, by majority rule we have

$$\begin{aligned}\mathsf{P}(\tilde{x}_i\tilde{y}_i = x_i y_i) &= \mathsf{Prob}[\text{at least } m+1 \text{ of } \tilde{x}_i^{(j)}\tilde{y}_i^{(j)}\text{s agree to } x_i y_i] \\ &= \sum_{j=m+1}^{2m+1} \binom{2m+1}{j} (1-\gamma)^j \gamma^{2m+1-j}\end{aligned}$$

□

Theorem 3.6. *In Scenario I, for $2m+1$ reads for sequences $\mathbf{x}$ and $\mathbf{y}$, we have*

$$\mathbb{E}\Big[\, |D_H(\tilde{\mathbf{x}}, \tilde{\mathbf{y}}) - D_H(\mathbf{x}, \mathbf{y})| \,\Big] \leq 2n(\alpha + \alpha').$$

Proof. The proof is similar to the argument we made for single-read scenario.

$$\begin{aligned}\mathbb{E}\Big[\,|D_H(\tilde{\mathbf{x}},\tilde{\mathbf{y}}) - D_H(\mathbf{x},\mathbf{y})|\,\Big] &\leq \mathbb{E}\Big[\sum_{i=1}^{n}|x_i-\tilde{x}_i| + \sum_{i=1}^{n}|y_i-\tilde{y}_i| + 2\sum_{i=1}^{n}|x_iy_i-\tilde{x}_i\tilde{y}_i|\Big] \\ &\leq n\alpha + n\alpha + 2n\alpha' \\ &= 2n(\alpha+\alpha'),\end{aligned}$$

where α and α' are defined in equations (3.6) and (3.7). □

Figure 3.3 depicts the improvement in the normalized confidence bounds for different number of reads.

In certain situations, it is possible that Hamming weights of $\mathbf{x}$ and $\mathbf{y}$ are known a priori. Therefore, we can expect to have better confidence intervals for Hamming distance deviation. Following lemma represents the improvement obtained in confidence interval precisely.

Lemma 3.7. *For multiple reads scenario, if $W_H(\mathbf{x})$ and $W_H(\mathbf{y})$ are known, then*

$$\mathbb{E}\Big[\,|D_H(\tilde{\mathbf{x}},\tilde{\mathbf{y}}) - D_H(\mathbf{x},\mathbf{y})|\,\Big] \leq 2n\alpha'.$$

Proof. Recall from equation (3.2) that Hamming distance of two vectors can be described in terms of their Hamming weights and the equivalent conductance measured in the memristor array. Hence, when $W_H(\mathbf{x})$ and $W_H(\mathbf{y})$ are known, then

$$\mathbb{E}\Big[\,|D_H(\tilde{\mathbf{x}},\tilde{\mathbf{y}}) - D_H(\mathbf{x},\mathbf{y})|\,\Big] \leq 2\sum_{i=1}^{n}\mathbb{E}\Big[\,|x_iy_i-\tilde{x}_i\tilde{y}_i|\,\Big] = 2n\alpha'$$

□

Scenario II: We have been following the multiple reads approach in which every bit of all vectors (two vectors in Hamming distance computation) stored in the memrsitor array are read more than once. In scenario I, every stored bit, e.g.

x_i was read from the array $2m+1$ times and finally $\tilde{x}_i$ was estimated from these reads based on majority rule. Now, consider the situation where we first estimate the Hamming distance of $\mathbf{x}$ and $\mathbf{y}$ for every read from the memristor and then we decide how to make a final estimations for Hamming distance, based on the estimations in hand. More precisely, assume vectors $\mathbf{x}$ and $\mathbf{y}$ are read $2m+1$ times and vectors $\tilde{\mathbf{x}}^{(j)}$ and $\tilde{\mathbf{y}}^{(j)}$ are obtained, for $1 \leq j \leq 2m+1$. For every j, we define jth Hamming distance estimation

$$D_H^{(j)} = D_H(\tilde{\mathbf{x}}^{(j)}, \tilde{\mathbf{y}}^{(j)}).$$

Now, we take the average of these measured distances as a final estimation for $D_H(\mathbf{x}, \mathbf{y})$, as follows

$$\tilde{D}_H(\mathbf{x}, \mathbf{y}) = \frac{1}{2m+1} \sum_{j=1}^{2m+1} D_H^{(j)}.$$

Lemma 3.8. *For multiple reads scenario II, the confidence interval is bounded as follows*

$$\mathbb{E}\Big[\Big|\tilde{D}_H(\mathbf{x}, \mathbf{y}) - D_H(\mathbf{x}, \mathbf{y})\Big|\Big] \leq 4n\beta - n\beta^2$$

Proof. The estimated Hamming distance is defined by equation (3.4). We can write

$$\begin{aligned}
\mathbb{E}\Big[\Big|\tilde{D}_H(\mathbf{x}, \mathbf{y}) - D_H(\mathbf{x}, \mathbf{y})\Big|\Big] &= \mathbb{E}\Big[|\frac{1}{2m+1} \sum_{j=1}^{2m+1} D_H^{(j)} - D_H(\mathbf{x}, \mathbf{y})|\Big] \\
&\leq \frac{1}{2m+1} \sum_{j=1}^{2m+1} \mathbb{E}\Big[|D_H(\tilde{\mathbf{x}}^{(j)}, \tilde{\mathbf{y}}^{(j)}) - D_H(\mathbf{x}, \mathbf{y})|\Big] \\
&\leq 4n\beta - n\beta^2
\end{aligned}$$

□

Notice that multiple-read Scenario II introduces no benefits compared to the single-read case.

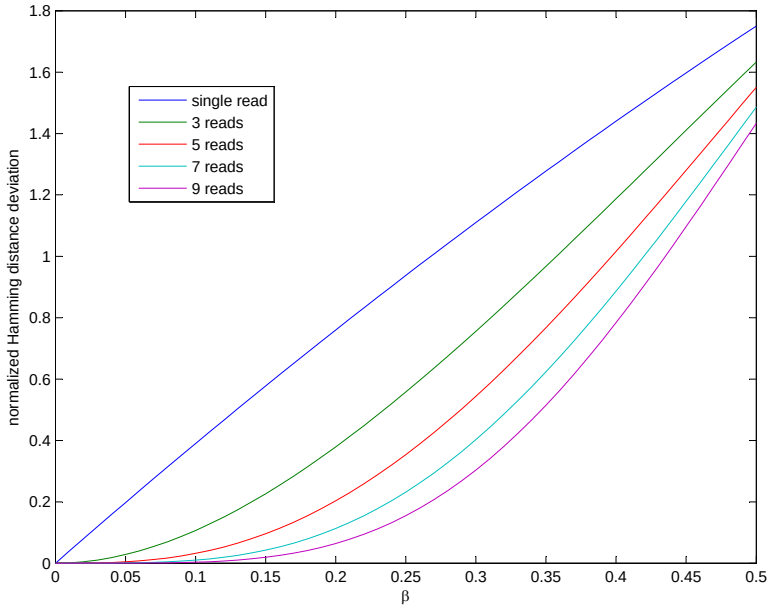

FIGURE 3.3: Confidence bounds for multiple-reads: BSC model

3.4 Noise modeling: AWGN

In previous analysis we considered a simple BSC model for noisy reads from the memristor, in which every bit was read correctly with probability $1 - \beta$ and was read as the flipped one with probability β. Therefore, all the post-operations on the vectors were employed on the hard values of measurements. Now, we turn our

consideration into a soft-valued measurement. More explicitly, we assume that every bit is additively distorted with a Gaussian noise $N \sim \mathcal{N}(0, \sigma_N^2)$. For instance, suppose bit x is stored in the memristor. As the noisy measurement, $\tilde{x} = x + N$ is the soft-valued read for x (Figure 3.4). We will analyze the confidence bounds for this model in the following.

3.4.1 Single measurement

Let vectors $\mathbf{x} = (x_1, \cdots, x_n)$ and $\mathbf{y} = (y_1, \cdots, y_n)$ be two binary vectors stored in a memristor array. Every bit is measured from the memristor with respect to the Gaussian model described above, i.e. $\tilde{\mathbf{x}} = (\tilde{x}_1, \cdots, \tilde{x}_n)$ and $\tilde{\mathbf{y}} = (\tilde{y}_1, \cdots, \tilde{y}_n)$ are measured vectors in which

$$\tilde{x}_i = x_i + N_i^{(x)} \text{ for } 1 \leq i \leq n, \tag{3.8}$$

and

$$\tilde{y}_i = y_i + N_i^{(y)} \text{ for } 1 \leq i \leq n. \tag{3.9}$$

Notice that all of the noise components $N_i^{(x)}$ and $N_i^{(y)}$ are i.i.d. from a Gaussian distribution $\mathcal{N}(0, \sigma_N^2)$.

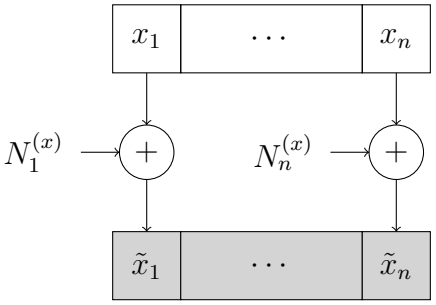

FIGURE 3.4: Noisy measurement: AWGN model

Before turning into the distance analyses, let us first review some useful properties of Gaussian random variables.

Lemma 3.9. *For two random variables $N_1, N_2 \sim \mathcal{N}(0, \sigma_N^2)$, we have*

1. $\mathbb{E}[|N_1|] = \mathbb{E}[|N_2|] = \sigma_N \sqrt{\frac{2}{\pi}}$
2. $\mathbb{E}[|N_1 N_2|] \leq \sigma_N^2$.

Proof. Define the folded Gaussian random variable $N_1^+ = |N_1|$ and $N_2^+ = |N_2|$.

1. The probability density function of N_1^+ is

$$f_{N_1^+}(\eta) = \frac{2}{\sqrt{2\pi\sigma_N^2}} e^{-\frac{-\eta^2}{2\sigma_N^2}} \text{ for } \eta \geq 0.$$

   Taking expectation from this distribution yields $\mathbb{E}[|N_1|] = \mathbb{E}[N_1^+] = \sigma_N \sqrt{\frac{2}{\pi}}$. Similarly, $\mathbb{E}[|N_2|] = \sigma_N \sqrt{\frac{2}{\pi}}$.

2. From Cauchy-Schwarz inequality we have

$$\mathbb{E}[|N_1 N_2|] = |\mathbb{E}[|N_1||N_2|]| \leq \sqrt{\mathbb{E}[N_1^2]\mathbb{E}[N_2^2]} = \sigma_N^2$$

□

Recall from equation (3.4) that

$$|D_H(\tilde{\mathbf{x}}, \tilde{\mathbf{y}}) - D_H(\mathbf{x}, \mathbf{y})| \leq \sum_{i=1}^{n} |x_i - \tilde{x}_i| + \sum_{i=1}^{n} |y_i - \tilde{y}_i| + 2 \sum_{i=1}^{n} |x_i y_i - \tilde{x}_i \tilde{y}_i| .$$

However, vectors $\tilde{\mathbf{x}}$ and $\tilde{\mathbf{y}}$ are real-valued (and not binary) vectors and we can not employ Hamming distance operation on these vectors. Instead, we define a Hamming-like distance metric for real-valued vectors. For two real-valued vectors

$\tilde{\mathbf{x}}$ and $\tilde{\mathbf{y}}$, define Hamming-like distance

$$D_{HL}(\tilde{\mathbf{x}}, \tilde{\mathbf{y}}) = \sum_{i=1}^{n} |\tilde{x}_i - \tilde{y}_i|^2.$$

Notice that for binary vectors, the Hamming-like distance defined above simplifies to the typical Hamming distance metric.

Therefore, we can evaluate the deviation in the Hamming distance as follows

$$|D_{HL}(\tilde{\mathbf{x}}, \tilde{\mathbf{y}}) - D_H(\mathbf{x}, \mathbf{y})| \leq \sum_{i=1}^{n} \left|x_i^2 - \tilde{x}_i^2\right| + \sum_{i=1}^{n} \left|y_i^2 - \tilde{y}_i^2\right| + 2\sum_{i=1}^{n} \left|x_i y_i - \tilde{x}_i \tilde{y}_i\right|.$$

Replacing equations (3.8) and (3.9) in the latter expression yields

$$\begin{aligned} |D_{HL}(\tilde{\mathbf{x}}, \tilde{\mathbf{y}}) - D_H(\mathbf{x}, \mathbf{y})| \leq & \sum_{i=1}^{n} |2x_i N_i^{(x)} + {N_i^{(x)}}^2| + \sum_{i=1}^{n} |2y_i N_i^{(y)} + {N_i^{(y)}}^2| \\ & + 2\sum_{i=1}^{n} |x_i N_i^{(y)} + y_i N_i^{(x)} + N_i^{(x)} N_i^{(y)}|. \end{aligned}$$

Taking the expectation from both sides yields

$$\begin{aligned} \mathbb{E}\Big[\, |D_{HL}(\tilde{\mathbf{x}}, \tilde{\mathbf{y}}) - D_H(\mathbf{x}, \mathbf{y})| \,\Big] &\leq 2n\big(\sigma_N \sqrt{\frac{2}{\pi}} + \sigma_N^2\big) + 2n\big(\sigma_N^2 + \sigma_N \sqrt{\frac{2}{\pi}}\big) \\ &= 4n\sigma_N \big(\sqrt{\frac{2}{\pi}} + \sigma_N\big). \end{aligned}$$

For the Gaussian noise model, we have been assuming that the memristor array is ideal, i.e. $\epsilon = 0$, and we derived the confidence bounds on the Hamming distance deviation. Now, we can employ the same procedure to derive the effect of non-ideality of the memristor array on the confidence bounds. Recall from Lemma 3.2 that it has been shown that for enough hardware accuracy, Hamming distance of two binary vectors can be calculated by a single equivalent conductance measurement.

Let $\tilde{\mathbf{x}} = (\tilde{x}_1, \cdots, \tilde{x}_n)$ and $\tilde{\mathbf{y}} = (\tilde{y}_1, \cdots, \tilde{y}_n)$ be the noisy measurements of binary vectors $\mathbf{x} = (x_1, \cdots, x_n)$ and $\mathbf{y} = (y_1, \cdots, y_n)$ stored in the memristor, where equations (3.8) and (3.9) describe the Gaussian noisy model. In the previous subsection, we showed that for $0 < \epsilon < 1/(2n-1)$, the Hamming distance deviation can be bounded as follows

$$\mathbb{E}\Big[\,|D_H(\tilde{\mathbf{x}},\tilde{\mathbf{y}}) - D_H(\mathbf{x},\mathbf{y})|\,\Big] \leq \frac{1+\epsilon}{\epsilon(1-\epsilon)}\Big\{(2-\epsilon)\mathbb{E}\Big[\,|G_{\tilde{\mathbf{x}},\tilde{\mathbf{y}}} - G_{\mathbf{x},\mathbf{y}}|\,\Big] + (1-\epsilon)\Big\}.$$

From equation (3.5), we have

$$\begin{aligned}
\mathbb{E}\Big[\,|G_{\tilde{\mathbf{x}},\tilde{\mathbf{y}}} - G_{\mathbf{x},\mathbf{y}}|\,\Big] &\leq \frac{\epsilon(1-\epsilon)}{1+\epsilon}\sum_{i=1}^{n}\mathbb{E}\Big[\,|x_i - \tilde{x}_i|\,\Big] + \frac{\epsilon(1-\epsilon)}{1+\epsilon}\sum_{i=1}^{n}\mathbb{E}\Big[\,|y_i - \tilde{y}_i|\,\Big] \\
&\quad + \frac{(1-\epsilon)^2}{1+\epsilon}\sum_{i=1}^{n}\mathbb{E}\Big[\,|x_i y_i - \tilde{x}_i\tilde{y}_i|\,\Big] \\
&\leq n\frac{\epsilon(1-\epsilon)}{1+\epsilon}\sigma_N\sqrt{\frac{2}{\pi}} + n\frac{\epsilon(1-\epsilon)}{1+\epsilon}\sigma_N\sqrt{\frac{2}{\pi}} + n\frac{(1-\epsilon)^2}{1+\epsilon}\Big(\sigma_N\sqrt{\frac{2}{\pi}} + \sigma_N^2\Big) \\
&= n\frac{(1-\epsilon)}{1+\epsilon}\sigma_N\Big((1+\epsilon)\sqrt{\frac{2}{\pi}} + (1-\epsilon)\sigma_N\Big).
\end{aligned}$$

Using the obtained bound, we have proved the following theorem in the recent discussions.

Theorem 3.10. *For Gaussian noisy measurements and* $0 < \epsilon < 1/(2n-1)$*, the confidence bound on the Hamming distance deviation is*

$$\mathbb{E}\Big[\,|D_H(\tilde{\mathbf{x}},\tilde{\mathbf{y}}) - D_H(\mathbf{x},\mathbf{y})|\,\Big] \leq \frac{1}{\epsilon}\Big\{n(2-\epsilon)\sigma_N\Big((1+\epsilon)\sqrt{\frac{2}{\pi}} + (1-\epsilon)\sigma_N\Big) + (1-\epsilon)\Big\}.$$

3.4.2 Multiple measurements

Thus far, we analyzed the confidence intervals for single-read scenario with respect to a Gaussian model. Now, like we did for BSC model, we can investigate the effect of multiple reads on the confidence interval improvements. More precisely, assume

two binary vectors $\mathbf{x} = (x_1, \cdots, x_n)$ and $\mathbf{y} = (y_1, \cdots, y_n)$ stored in a memristor array. Each of two vectors $\mathbf{x}$ and $\mathbf{y}$ are read from the memristor m times and we obtain measurement vectors $\tilde{\mathbf{x}}^{(j)} = (\tilde{x}_1^{(j)}, \cdots, \tilde{x}_n^{(j)})$ and $\tilde{\mathbf{y}}^{(j)} = (\tilde{y}_1^{(j)}, \cdots, \tilde{y}_n^{(j)})$ for $1 \leq j \leq m$. By the Gaussian model setup,

$$\tilde{x}_i^{(j)} = x_i + N_{i,j}^{(x)} \text{ for } 1 \leq i \leq n \text{ and } 1 \leq j \leq m,$$

and

$$\tilde{y}_i^{(j)} = y_i + N_{i,j}^{(y)} \text{ for } 1 \leq i \leq n \text{ and } 1 \leq j \leq m,$$

where all the noise components are i.i.d. from $\mathcal{N}(0, \sigma_N^2)$.

Now, we have to estimate vectors $\tilde{\mathbf{x}}$ and $\tilde{\mathbf{y}}$ based on noisy observations $\tilde{\mathbf{x}}^{(j)}$ and $\tilde{\mathbf{y}}^{(j)}$. For this model where we are dealing with zero-mean Gaussian noise, a natural approach is taking the average from noisy measured vectors. Therefore, we define two new vectors $\tilde{\mathbf{x}} = (\tilde{x}_1, \cdots, \tilde{x}_n)$ and $\tilde{\mathbf{y}} = (\tilde{y}_1, \cdots, \tilde{y}_n)$ as follows

$$\tilde{\mathbf{x}} = \frac{1}{m} \sum_{j=1}^{m} \tilde{\mathbf{x}}^{(j)} = \frac{\tilde{\mathbf{x}}^{(1)} + \cdots + \tilde{\mathbf{x}}^{(m)}}{m},$$
$$\tilde{\mathbf{y}} = \frac{1}{m} \sum_{j=1}^{m} \tilde{\mathbf{y}}^{(j)} = \frac{\tilde{\mathbf{y}}^{(1)} + \cdots + \tilde{\mathbf{y}}^{(m)}}{m}.$$

As described in the beginning of the setup, an additive Gaussian noise distorts the measurements. Hence, entries of vectors $\tilde{\mathbf{x}}$ and $\tilde{\mathbf{y}}$ can be expressed as

$$\tilde{x}_i = x_i + \frac{1}{m} \sum_{j=1}^{m} N_{i,j}^{(x)} \text{ for } 1 \leq i \leq n,$$
$$\tilde{y}_i = y_i + \frac{1}{m} \sum_{j=1}^{m} N_{i,j}^{(y)} \text{ for } 1 \leq i \leq n.$$

Having these estimated vectors $\tilde{\mathbf{x}}$ and $\tilde{\mathbf{y}}$, now we can analyze the expected deviation in Hamming distance of vectors $\mathbf{x}$ and $\mathbf{y}$. Recall that Hamming-like distance of two real-valued vectors $\tilde{\mathbf{x}}$ and $\tilde{\mathbf{y}}$ is defined as

$$D_{HL}(\tilde{\mathbf{x}}, \tilde{\mathbf{y}}) = \sum_{i=1}^{n} |\tilde{x}_i - \tilde{y}_i|^2.$$

Therefore, we can express the deviation between the actual Hamming distance of vectors $\mathbf{x}$ and $\mathbf{y}$ and the Hamming-like distance between two noisy measurements $\tilde{\mathbf{x}}$ and $\tilde{\mathbf{y}}$. We have

$$\begin{aligned}
|D_{HL}(\tilde{\mathbf{x}}, \tilde{\mathbf{y}}) - D_H(\mathbf{x}, \mathbf{y})| &\leq \sum_{i=1}^{n} \left|x_i^2 - \tilde{x}_i^2\right| + \sum_{i=1}^{n} \left|y_i^2 - \tilde{y}_i^2\right| + 2\sum_{i=1}^{n} |x_i y_i - \tilde{x}_i \tilde{y}_i| \\
&= \sum_{i=1}^{n} \left| \frac{1}{m^2} \Big(\sum_{j=1}^{m} N_{i,j}^{(x)} \Big)^2 + \frac{2}{m} x_i \sum_{j=1}^{m} N_{i,j}^{(x)} \right| \\
&+ \sum_{i=1}^{n} \left| \frac{1}{m^2} \Big(\sum_{j=1}^{m} N_{i,j}^{(y)} \Big)^2 + \frac{2}{m} y_i \sum_{j=1}^{m} N_{i,j}^{(y)} \right| \\
&+ 2\sum_{i=1}^{n} \left| \frac{1}{m^2} \sum_{j=1}^{m} N_{i,j}^{(x)} \sum_{j=1}^{m} N_{i,j}^{(y)} + \frac{1}{m} x_i \sum_{j=1}^{m} N_{i,j}^{(y)} + \frac{1}{m} y_i \sum_{j=1}^{m} N_{i,j}^{(x)} \right|
\end{aligned}$$

Taking the expectation from this inequality and using Lemma 3.9 yield

$$\begin{aligned}
\mathbb{E}\Big[\left| D_{HL}(\tilde{\mathbf{x}}, \tilde{\mathbf{y}}) - D_H(\mathbf{x}, \mathbf{y}) \right| \Big] &\leq 2n\Big(\frac{\sigma_N^2}{m} + \sigma_N \sqrt{\frac{2}{\pi}} \Big) + 2n\Big(\frac{\sigma_N^2}{m} + \sigma_N \sqrt{\frac{2}{m\pi}} \Big) \\
&= 2n\sigma_N \Big(2\frac{\sigma_N}{m} + \sqrt{\frac{2}{\pi}} + \sqrt{\frac{2}{m\pi}} \Big).
\end{aligned}$$

Figure 3.5 depicts the improvement in the normalized confidence bounds for different number of reads.

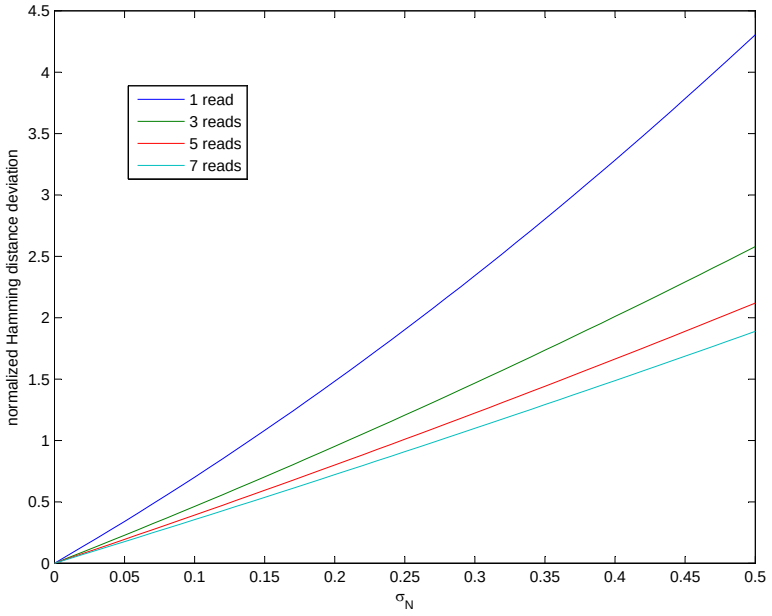

FIGURE 3.5: Confidence bounds for multiple-reads: AWGN model

3.5 Conclusion

To put the discussions of this chapter in nutshell, we studied different models for approximate computing for a specific function which was calculating the Hamming distance between two binary vectors. Two canonical models, BSC and AWGN were investigated in different situations. These situations vary with or without taking

the effect of noise in measurements and non-ideality of resistive arrays and dealing with single or multiple read scenarios.

Chapter 4

Conclusion

In this thesis, we explored the notion of approximate synchronization and computation. Having stated the challenge of energy/time/hardware issues, seeking approximate solutions makes sense. Firstly, we studied the approximate synchronization problem and as the contribution, upper bounds on the optimal required rate were provided, for uniform and non-uniform i.i.d. sources. We formulated the problem in the context of distributed source coding and studied that from an information-theoretic perspective. They showed that how much rate one can save allowing a limited distortion in the synched files.

The results regarding approximate synchronization have been submitted to the 2016 *IEEE Information Theory Workshop* (ITW) [RSTD16], as a part of the contributions.

In the second part of the thesis, we explored the notion of approximate solutions this time for computation. We centered our focus on Hamming distance calculation of two binary vectors. We provided the motivation for in-memory computing and extended the results from noise-free computation to noisy computation. As the first step, we mathematically modeled the noise in measurements and considered two canonical model for that, bit-flipping and Gaussian noise. Confidence

bounds on deviation in the distance due to noise were studied for two single and multiple measurements scenarios.

References

[Bra14] M. Braverman, "Interactive communication and coding theory," in *Int. Conf. Math.*, 2014.

[BSYD13] N. Bitouze, F. Sala, S. M. S. T. Yazdi, and L. Dolecek, "A practical framework for efficient file synchronization," in *Proc. of the Allerton Conf. on Comm., Control, and Comp.*, Sep.-Oct. 2013.

[CC15] Y. Cassuto and K. Crammer, "In-Memory Hamming Similarity Computation in Resistive Arrays," in *Proc. IEEE Int. Symp. Inf. Theory*, Hong Kong, 2015.

[CKY14a] Y. Cassuto, Kvatinsky, and E. Yaakobi, "Information-Theoretic Sneak-Path Mitigation in Memristor Crossbar Arrays," in *IEEE Int. Symp. Inf. Theory (submitted)*, 2014.

[CKY14b] ——, "On the Channel Induced by Sneak-Path Errors in Memristor Arrays," in *IEEE SPCOM*, Indian Institute of Science Bangalore (invited paper), 2014.

[DG01] S. Diggavi and M. Grossglauser, "On transmission over deletion channels," in *Proc. of the Allerton Conf. on Comm., Control, and Comp.*, Illinois, 2001.

[DG06] ——, "Information transmission over a finite buffer channel," in *IEEE Trans. Info. Theory*, vol. 52, no. 1, 2006, pp. 1226–1237.

[DMP07] S. Diggavi, M. Mitzenmacher, and H. D. Pfister, "Capacity Upper Bounds for the Deletion Channel," in *Proc. IEEE Int. Symp. Inf. Theory*, St. Petersburg, Russia, June 2007, pp. 1716–1720.

[Dob67] R. L. Dobrushin, "Shannon's theorems for channels with synchronization errors," *Problems of Inform. Transm*, vol. 3, no. 4, pp. 11–26, 1967.

[Gal61] R. G. Gallager, "Sequential decoding for binary channels with noise and synchronization errors," *Lincoln Lab. Group Report*, 1961.

[KM10] Y. Kanoria and A. Montanari, "On the deletion channel with small deletion probability," in *Proc. IEEE Int. Symp. Inf. Theory*, Austin, Texas, Jul. 13-18 2010, pp. 1002–1006.

[Lev66] V. I. Levenshtein, "Binary codes capable of correcting deletions, insertions and reversals," *Soviet Physics Doklady*, vol. 10, no. 8, pp. 845–848, Feb. 1966.

[Mit09] M. Mitzenmacher, "A survey of results for deletion channels and related synchronization channels," *Probability Surveys*, vol. 6, pp. 1–33, 2009.

[MRT11] N. Ma, K. Ramchandran, and D. Tse, "Efficient file synchronization: A distributed source coding approach," in *Proc. IEEE Int. Symp. Inf. Theory*, St. Petersburg, Russia, Jul.–Aug. 2011, pp. 583–587.

[MTZ03] Y. Minsky, A. Trachtenberg, and R. Zippel, "Set reconciliation with nearly optimal communication complexity," in *IEEE Trans. on Info. Theory*, vol. 49, no. 9, 2003, pp. 2213–2218.

[MV12] M. Mitzenmacher and G. Varghese, "The complexity of object reconciliation, and open problems related to set difference and coding," in

Proc. of the Allerton Conf. on Comm., Control, and Comp., Sep.-Oct. 2012, pp. 1126–1132.

[RSTD16] A. Reisizadehmobarakeh, C. Schoeny, C.-Y. Tsai, and L. Dolecek, "Approximate File Synchronization: Upper Bounds and Interactive Algorithms," in *Submitted to Information Theory Workshop (ITW)*, Cambridge, UK, 2016.

[SBSD14] C. Schoeny, N. Bitouze, F. Sala, and L. Dolecek, "Efficient file synchronization: Extensions and simulations," in *Proc. of IEEE Asilomar Conf. on Sig., Syst., and Comps.*, Nov. 2014.

[TKMS10] A. Tauman Kalai, M. Mitzenmacher, and M. Sudan, "Tight Asymptotic Bounds for the Deletion Channel with Small Deletion Probabilities," in *Proc. IEEE Int. Symp. Inf. Theory*, 2010.

[Tri09] A. Tridgell, "Efficient algorithms for sorting and synchronization," Ph.D. dissertation, Australian National University, 2009, 2009.

[VSR15] R. Venkataramanan, V. N. Swamy, and K. Ramchandran, "Low-complexity interactive algorithms for synchronization from deletions, insertions, and substitutions," in *IEEE Trans. on Info. Theory*, vol. 61, no. 10, 2015, pp. 5670–5689.

[VTR11] R. Venkataramanan, S. Tatikonda, and K. Ramchandran, "Bounds on the optimal rate for synchronization from insertions and deletions," in *Proc. of the Info. Theory and Applications Workshop*, San Diego, CA, Feb. 2011.

[VTR13] ——, "Achievable rates for channels with deletions and insertions," in *IEEE Trans. on Info. Theory*, vol. 59, no. 11, Nov. 2013, pp. 6990–7013.

[VZR10] R. Venkataramanan, H. Zhang, and K. Ramchandran, "Interactive low-complexity codes for synchronization from deletions and insertions," in *Proc. of the Allerton Conf. on Comm., Control, and Comp.*, 2010, pp. 1412–1419.

[YD14] S. M. S. T. Yazdi and L. Dolecek, "A deterministic, polynomial-time protocol for synchronizing from deletions," in *IEEE Trans. Info. Theory*, vol. 60', no. 1, Jan. 2014, pp. 397–407.