"Information-Friction" for Noiseless
Compressive Sensing Decoding

Researchers
Carnegie Mellon University, The Chinese University of Hong Kong

Abstract
The fundamental problem considered in this paper is, basically, “what is the energy consumed for the implementation of compressive sensing algorithm on a circuit?” We use the proposed “bit-meters” measure as a proportional measurement for energy, i.e., the product of number bits transmitted and the distance of information transport. Be Analogous to the friction imposed on relative motion between two surfaces, this model is so-called “information-friction” model. By using this “information-friction” model, a fundamental lower bound for the implementation of compressive sensing algorithms on a circuit is provided. Further, we explore and compare a series of decoding algorithms based on different implementation-circuits. As our second main result, an order-tight asymptotic result on a fixed precision bits for the regime $m = \mathcal{O}(k)$ ($m$ is the number of measurements and $k$ is the number of non-zero entries in the compressible vector) is provided. We thus attest in this paper that the asymptotic lower bound is order-optimal with a sub-linear precision $k$ such that $k = n^{1-\beta}$ ($\beta \in (0, 1)$, $n$ is the total number of input entries) since a proposed algorithm with corresponding construction of implementation-circuit can achieve an upper bound with the same order.

Keywords: energy-efficient, compressive sensing, circuit implementation

I. Introduction

The “information-friction” model is firstly considered in [12] for finding a trade-off between energy consumed in encoding/decoding processes and transmission power in a communication system. Compressive sensing, spurred by [4, 9] recently has been fully explored using information theory, and order-optimal algorithms for encoding and decoding based on a construction of a spare graph have been given, for instance, in [1, 10]. In certain practice, like some distributed cellphones networks, a naturally generated ensemble of measurement matrix and compressed signal is achievable. And for the base-station, after receiving the compressed signal, it needs to decode the compressed signal and get the needed information as efficient as possible. Therefore, it’s natural for us to ask that if these algorithms for compressive sensing above are also energy-efficient for decoding, if we plan to implement them on physical circuits.

To answer this question, the first problem is to find the fundamental limit on energy for the implementations of compressive sensing decoding algorithms on a circuit. The setting for compressive sensing is traditional such that the compressible signal is assumed to be a $k$-sparse real vector $X$ of length-$n$. And the corresponding compressed signal is a length-$m$ real vector $Y$ which is computed using a $m \times n$ real measurement matrix $A$ denoted by $A$. The goal is to efficiently recover the input vector $X$ as a recovery vector $\hat{X}$ with a small average block error probability $P_e^{blk} = \Pr(E^{blk} = 1)$ such that $E^{blk} = 1$ if, roughly speaking, the normalized $l_1$ distance between the input vector $X$ and the recovery vector $\hat{X}$ is larger than $2^{-Q}$ for a fixed constant $Q$; otherwise, $E^{blk} = 0$. We thus use notation Q as the precision of the number of bits in order to implement compressive sensing decoding algorithms on a circuit. In the real case, we only get finite bits to store the value for the entries. Thus $Q$ is indeed reasonable to be made as a constant in our analysis.

As an initial work, we focus on the “information-friction” model, and find a lower bound of “bit-meters”, namely, the product of the number of bits moved and the distance these bits are moved. Using this particular metric called “bit-meters”, we assume the energy consumed is proportional to the “bit-meters” used in order to support a computation. In the first part of the main results, we show that for a fixed precision $Q$, the required “bit-meters” (energy) for decoding a compressed signal can be no smaller than $\Omega \left( \frac{nk}{\log n} \sqrt{\frac{\log \frac{1}{P_e}}{\log n}} \right)$ asymptotically at the regime

1The “bit-meters” metric is firstly suggested in [12], as a possible supplantation of VLSI model introduced by Thompson and others in [3, 5, 15, 16, 17, 19] (and explored further in [2, 7, 13]) for measuring the energy consumed in a circuit.

2This example is also given in [1].

3As the usual setting, we always assume $n > m > 1$ throughout the following pages.
\( m = O(k) \) and \( k = n^{1-\beta} \) where constant \( \beta \in (0, 1) \). This asymptotic lower bound is proved to be tight since a multi-stage algorithm modified from SHO-FA in [11] achieves an upper bound \( O\left(\sqrt{n}k \sqrt{\log \frac{1}{\epsilon}}\right) \) asymptotically as our second main result.

II. Background

In this section, we formalize the employed models, including both the compressive sensing model and the decoding circuit model.

A. Compressive Sensing Model

For the compressive sensing model, we consider two basic models, probabilistic and combinatorial. We assume the length of input vector are considerably large. Therefore the two models are asymptotically equivalent. For the ease of analysis, we adopt the probabilistic one with independent property to avoid tedious calculations. And for the purpose of presentation and elucidation, the combinatorial model are used to give us an intriguing insights of both fundamental limit and performances of batches of delicate decoding algorithms.

**Definition 1** (Compressive Sensing Model \((n,m,p)\)). A bounded length-\( n \) “compressible” vector \( \mathbf{X} \in \mathbb{R}^n \) is the input vector with each entry \( X_i \in \mathbb{R} \) satisfies \(|X_i| \leq U \) (with a constant upper bound \( U \geq 0 \)) has probability \( p \) to be non-zero and probability \( 1-p \) to be zero. We assume \( p = o(1) \) as the sparse assumption. The measurement matrix is a real matrix denoted by \( \mathbf{A} \in \mathbb{R}^{m \times n} \). And the length-\( m \) “compressed” vector \( \mathbf{Y} \in \mathbb{R}^m \) is the output vector such that \( \mathbf{Y} = \mathbf{A} \mathbf{X} \). Base on the output vector \( \mathbf{Y} \), a recovery vector \( \hat{\mathbf{X}} \) is decoded.

**Definition 2** (Compressive Sensing Model \((n,m,k)\)). A bounded length-\( n \) “compressible” vector \( \mathbf{X} \in \mathbb{R}^n \) is the \( k \)-sparse input vector which contains exactly \( k \) non-zero entries \( X_i \in \mathbb{R} \) satisfies \(|X_i| \leq U \) (with a constant upper bound \( U \geq 0 \)). We assume \( k = o(n) \) as the sparse assumption. The measurement matrix is a real matrix denoted by \( \mathbf{A} \in \mathbb{R}^{m \times n} \). And the length-\( m \) “compressed” vector \( \mathbf{Y} \in \mathbb{R}^m \) is the output vector such that \( \mathbf{Y} = \mathbf{A} \mathbf{X} \). Base on the output vector \( \mathbf{Y} \), a recovery vector \( \hat{\mathbf{X}} \) is decoded.

Note that by the strong law of large number (see, for instance the excellent textbook [11]), the number of non-zero entries of the input vector \( \mathbf{X} \) in the probabilistic compressive sensing model will be bounded in a constant range containing \( k \) (e.g. \( k/2, 3k/2 \)) with an exponential probability of error \( e^{-\Theta k} \), which is actually, is negligible compared with the error probability we could achieve as a result in [14].

Moreover, since we will analyze the performance of compressive sensing algorithms in a bit stream representation and communication, then in addition we need to define the average block error probability based on quantization and a given norm \( \| \cdot \|_{\ell_q} \) of interest for \( 0 \leq p = \infty \).

**Definition 3** (Reconstruction Error, Precision, Average Block Error Probability). Given the input vector \( \mathbf{X} \) and recovery vector \( \hat{\mathbf{X}} \), the Reconstruction Error is defined as \( \|\mathbf{X} - \hat{\mathbf{X}}\|_{\ell_q} \) and the Relative Error is defined further as \( \|\mathbf{X} - \hat{\mathbf{X}}\|_{\ell_q} / \|\mathbf{X}\|_{\ell_q} \). Let \( Q \) denote the Precision, i.e., the required number of bits for reconstructing the input vector \( \mathbf{X} \), then the Average Block Error Probability is defined by \( P_{blk}^k = \Pr(E_{blk} = 1) \) where \( E_{blk} = 1 \) if the relative error satisfies \( \|\mathbf{X} - \hat{\mathbf{X}}\|_{\ell_q} / \|\mathbf{X}\|_{\ell_q} > 2^{-Q} \); Otherwise, \( E_{blk} = 0 \).

B. Decoding Circuit Model

For the decoding circuit model, we arrange the nodes (input and output) on a lattice \( \Lambda \subset \mathbb{R}^2 \), a discrete additive subgroup of the 2-D Euclidean space \( \mathbb{R}^2 \) which spans \( \mathbb{R}^2 \). Furthermore, each point (i.e.the nodes) on the lattice \( \Lambda \) (i.e.the circuit) endowed with a constant packing radius \( \rho(\Lambda) > 0 \) to ensure the distance between nodes for real implementation. Considering the coherence with [12], we expound the generalized circuit model using the same order of definitions as [12] did.

**Definition 4** (Lattice \((\rho(\Lambda))\)). A Lattice \((\rho(\Lambda))\) is defined as a lattice \( \Lambda \) spans \( \mathbb{R}^2 \) with a constant packing radius \( \rho(\Lambda) > 0 \).

**Definition 5** (Substrate). A Substrate is defined as a compact subspace \( \mathcal{V} \subset \mathbb{R}^2 \) which is also a \( \Lambda \)-subspace.

**Definition 6** (Grid \((\rho(\Lambda), \mathcal{V})\)). A Grid \((\rho(\Lambda), \mathcal{V})\) is a lattice \( \Lambda_\mathcal{V} \) in Substrate \( \mathcal{V} \) defined as the intersection of Lattice \((\rho(\Lambda))\) and \( \mathcal{V} \) such that \( \Lambda_\mathcal{V} = \Lambda \cap \mathcal{V} \).
**Definition 7** (Decoding Circuit, Sub-Circuit, Computational Nodes, Input-nodes, Output-nodes). The substrate $V$ together with a collection $S \subset \Lambda_V$ of points (called Computational Nodes, or simply Nodes) inside the Grid $\Lambda_V$, is called a Decoding Circuit, and is denoted by $\text{DecCkt} = (V, S)$. A Sub-Circuit denoted by $\text{SubCkt} = (V_{\text{sub}}, S_{\text{sub}})$ is constituted by an open and bounded subset $V_{\text{sub}}$ of $V$ and by the subset of Computational Nodes $S_{\text{sub}} \subset \Lambda \cap V_{\text{sub}}$. Furthermore, Computational Nodes can receive and store messages from the recovery vector $X$ or other nodes, and generate and transmit messages for the output vector $Y$ or other nodes. The Output-nodes are defined as the nodes for storing the recovery vector $\hat{X}$, and the Input-nodes are defined as the nodes for storing the received output vector $Y$.

WLOG, we assume that each node, no matter input or output, can behave as a bridge for communication between other nodes, thus exactly $n + m$ nodes ($n$ output nodes and $m$ input nodes) are enough for sustaining the functionality of the decoding circuit $\text{DecCkt} = (V, S)$ such that $|S| = n + m$, where $n$ is the length of input vector $X$ and $m$ is the length of the output vector $Y$.

**Definition 8** (Link). The computational nodes can accomplish noiseless communication with one another through unidirectional Links, within which binary-strings of messages are transmitted.

**Definition 9** (Communication Distance). Given a decoding circuit $\text{DecCkt} = (V, S)$ which has mixed input-nodes and output-nodes, if a decoding algorithm $\text{DEC}$ is implemented with two nodes $x$ and $y$ communicating, the corresponding Communication Distance $D_{\text{DEC}}$ is defined as the Euclidean distance between nodes $x, y \in V \cap \Lambda$ for the implementation of algorithm $\text{DEC}$ such that $D_{\text{DEC}} = \sup_{x, y \in S} \|x - y\|_2$.

**Definition 10** (Bit-meters Measure). The Bit-meters Measure (denoted by $\mu(\cdot)$) of a link is a function defined as the product of total number of bits carried by the messages on the link and the communication distance between the nodes at the ends of link. The bit-meters ($\mu(\cdot)$) for the decoding circuit $\text{DecCkt}$ and sub-circuits $\text{SubCkt}$s are the sums of bit-meters for all the links with nodes on its end contained entirely in the corresponding circuits.

Note that the Bit-meters Measure $\mu$ defined on links (or equivalently nodes which sending bits) satisfies the non-negativity, null empty set and countable additivity. Thus indeed the function $\mu$ is a measure from the power set $\mathcal{P}(S)$ of set of nodes $S$ to the extended real number line.

Our main goal focus on the asymptotic lower bound and upper bounds of bit-meters for the decoding circuit of compressive sensing, i.e., find the bounds of $\mu(S)$ as $n \to \infty$. In practice, it’s reasonable for us to relate bit-meters with energy consumed for the decoding process, as [12] elaborated using batch of different scenarios. Note that all the discussions are based on the implementation model summarized below.

**Definition 11** (Implementation Model $(\rho(\Lambda), \mu)$). Implementation Model $(\rho(\Lambda), \mu)$ denotes the model embed with Decoding Circuit $\text{DecCkt}$ and Bit-meters Measure $\mu$.

### III. Decoding Algorithms

In this section, we use a heuristic way to explore different constructions for the implementation-circuits of compressive sensing algorithm. The basic issues here are the locations of the two types of nodes (input and output) and how they communicate with one another. We discuss two cases, one is the centralization of input nodes, the other one is distributively arrangement of the nodes. By our discussion, for our interested regime $m = O(k)$, the later design always dominates the former one, we will show that local-decoding helps a lot to reduce the energy consumed and approach the order-optimal trade-off.

### A. Centralized-Decoding Algorithm

Centralization of input-nodes, which serves as the simplest construction, are going to be introduced and analyzed. A direct intuition is, this design is better for those algorithms have a relatively higher decoding complexity and lager measurements, namely, have a lager number of input-nodes need to talk to one another frequently. The benefit is a better error probability.

However, for the those algorithms with a relatively sparse measurement matrix such that the average block error probability $P_{e_{blk}}$ is of the order satisfies $P_{e_{blk}} = \Omega(e^{-m})$, the centralized design involves a “gap” between the bit-meters consumed and the fundamental limit introduced in the next section [IV-A] for our interested regime $m = O(k)$ as the direct theorem below indicates.

*which is automatically a $\sigma$-algebra. Thus the measure $\mu$ is well-defined.
Figure 1. This conceptual graph illustrates one possible arrangement of input-nodes and output-nodes on a implementation circuit, whereby the lower bound of bit-meters in Theorem 1 could be derived. Within the decoding circuit, all \( k \) input-nodes are located in the central part and all \( n \) output-nodes are put in the surrounding area of the central part.

**Theorem 1.** For the Compressive Sensing Model \((n, m, k)\) in a Centralized Decoding Circuit \(\text{DecCkt}\) implemented on the Implementation Model \((\rho(\Lambda), \mu)\) that achieves an average block error probability \(P_{\text{blk}}^{e.m} = \Omega(e^{-m})\) and a fixed precision \(Q\), if the decoder satisfies the regime \(m = O(k)\) and the sub-linear regime \(k = n^{1-\beta}\) where \(\beta \in (0, 1)\), then the decoder bit-meters are asymptotically upper bounded as:

\[
\text{bit-meters}(\text{DecCkt}) = \Omega\left(\sqrt{nk \log \frac{1}{P_{\text{blk}}}}\right).
\]

Proof: We only consider the bit-meters from the input-nodes to the output-nodes. Since after the input-nodes collaborates to decode the recovery vector \(\hat{X}\), the \(\Omega(k)\) bits information are required to transmit to the output-nodes with average distance \(\Omega(\sqrt{n})\). Thus we have

\[
\text{bit-meters}(\text{DecCkt}) = \Omega\left(k\sqrt{n}\right).
\]

Then we combine \(P_{\text{blk}}^{e.m} = \Omega(e^{-m})\) and conclude that \(\text{bit-meters}(\text{DecCkt}) = \Omega\left(\sqrt{nk \log \frac{1}{P_{\text{blk}}}}\right)\).

**B. Distributive-Decoding Algorithms**

We now propose two energy-efficient compressive sensing algorithms, both use the idea of local decoding to reduce the order of bit-meters. Instead of arranging all the input nodes related to received output vector on the central part of the circuit, we put them distributively among the circuit with carefully designed algorithms. As the intuition, since a large fraction of the communication are only carried within a small region, then roughly, the consumed energy are reduced prominently. And indeed, the implication above can be verified in the latter part mathematically. Leaving the formal definitions for next section(Section IV-B), we orchestrate the descriptions of algorithms here using a stage-by-stage manner accompanied by schematic graphs. The analysis of the performance will be given in the later section(see Section IV).

1) Chain Algorithm:

a) First Stage: The input vector containing \(n\) entries of real number is firstly divided into \(k\) separated sub-groups which are going be compressed individually. Each sub-group contains \(n/k\) entries. Note that we need to make sure that if a sub-group only contains a single non-zero entry, then it would be filtered out and resolved within the fixed precision. The decoding process can be conducted independently among all these \(k\) sub-groups. Thus, the corresponding decoder only need to deal with these sub-groups locally, i.e., it only needs to communicate within the local regions for the corresponding sub-groups respectively, which is an intuitive method for saving energy since the distances for communication are reduced greatly for most of the communication between nodes. A more rigorous depiction of the decoding circuit with nodes-arrangement is provided in the following section (see also Section IV). In this stage, if we use a measurement constant \(c\) number of measurements for each sub-group, then a constant proportion \(\rho\) of the total \(k\) non-zero entries (\(\rho k\) entries) can be located and solved in \(Q\)-bits precision with a high probability which is exponential in \(k\). Further define \(\phi = 1/(1 - \rho)\) to be a parameter for the remaining stages.
Figure 2. The schematic graph for the 1st stage, which illustrates the division of input vector and the construction of measurement matrix $\mathcal{A}$. In the first box with color of calamine blue, we exemplify part of the Identification Phase by a sub-group of five left nodes and three right nodes. Note that for each node on the right, the weights of edges connected to it should be made unique, which is the requirement for Identification Phase. In Verification Phase, the connection is kept as the same whereas the weights $e^{i\Theta}$ (Here we keep using the same notations in [1], where $i$ and $j$ are indexes for entries in the measurement matrix $\mathcal{A}$, $\iota$ denotes the positive square root of $-1$ in order to avoid confusions) for non-zero entries in the measurement matrix $\mathcal{A}$ are chosen uniformly in $[0, \pi/2]$.

Figure 3. The schematic graph for the 2nd stage, which illustrates the conflation processes in the coming stages. The sub-groups in the left block with gray color, are merged two-by-two (assume $\rho = 1/2$ and set $\phi = 2$), form the new sub-groups in the right block with light blue color. Note that in a same way as the first stage described in III-B2a and exemplifies in Figure III-B1a, the two phases (Identification Phase and Verification Phase) are also implemented for each of the new sub-group with the same number of measurements $c$ as the previous stages.

b) Second Stage up to $\log_\phi (k/\log_2 k)$-th Stage: We further combine $\phi = 1/(1 - \rho)$ of the resulting sub-groups come from first stages together in the second stage, which can be regarded as the “reserve processes” of the dichotomy. Note that in the $i$-th stage the conflation only processed in a local region with area of order approximately $\log_2 k(\phi^{i-1}n/k)$. In the $i$-th stage, $k/\phi^{i-2}$ many sub-groups are merged into $k/\phi^{i-1}$ new sub-groups. Each new sub-group contains $\phi^{i-1}n/k$ entries of the input vector. Also, the filtering step in the first stage III-B2a is implemented on each new sub-groups and again, like in the first stage, the corresponding decoder need to merely handle the information locally. The algorithm continue the conflation up to the $\log_\phi (k/\log_2 k)$-th stage, and still, use measurement constant $c$ as the measurements we used for each sub-group. As a result, approximately 1/2 of the total $k/\phi^{i-1}$ remaining non-zero entries can be located and solved in $Q$-bits precision with a high probability $^6$ which is of order $1/k$.

c) $(\log_\phi (k/\log_2 k) + 1)$-th Stage (Wind-up Stage): After $\log_\phi (k/\log_2 k)$ stages, the algorithm stops conflation, and globally decode the entire length-$n$ input vector given the information from the previous stages III-B2a and III-B1b in the last stage with $O(\log_2 k)$ measurements. In this sense, for the decoder, the information are communicated within the entire decoding circuit, which helps improve the performance with respect to the error probability. Actually, as an overall performance, the algorithm could achieve an average block error probability $P_{blk}$ of order $1/k$ and the bit-meters of order $O\left(\sqrt{\frac{nk}{\log n}} \sqrt{\log \frac{1}{P_{blk}}}\right)$.

2) Shotgun Algorithm:

a) First Stage: In a similar way to the first chain algorithm III-B1, the input vector containing $n$ entries of real number is firstly divided into $k$ separated sub-groups which are going be compressed individually. Each sub-group contains $n/k$ entries. Note that we need to make sure that if a sub-group only contains a single non-zero entry, then it would be filtered out and resolved within the fixed precision. The decoding process can be conducted independently

---

$^5$The details are given in the Section IV

$^6$Again, the details are given in the Section IV
within the entire decoding circuit, which helps improve the performance with respect to the error probability.

And the shortage for the chain algorithm is that for large size of input vector $X$, it necessarily requires several powerful central node within each local sub-circuit to handle the gathered information. For the shotgun algorithm, actually, as an overall performance, the algorithm could achieve an average block error probability $P_{blk}$ which is exponential in $k$. Further define $\varphi = 1/(1 - \sigma)$ to be a parameter for the remaining stages.

b) Second Stage up to $\log_{\varphi}(k/\log_2 k)$-th Stage: We further combine $\varphi = 1/(1 - \sigma)$ of the resulting sub-groups come from first stages uniformly at random in the second stage. Note that the combination is independent for each stage, and in the $i$-th stage the combination area is of order approximately $\log_\varphi k(\varphi^{i-1} n/k)$. In the $i$-th stage, $k/\varphi^{i-2}$ many sub-groups are merged into $k/\varphi^{i-1}$ new sub-groups. Each new sub-group contains $\varphi^{i-1} n/k$ entries of the input vector. Similarly, the filtering step in the first stage III-B2a is implemented on each new sub-groups and again, like in the first stage, the corresponding decoder need to merely handle the information locally. The algorithm continue the independent combination processes up to the $\log_{\varphi}(k/\log_2 k)$-th stage, and still, use measurement constant $c^\prime$ as the measurements we used for each sub-group. As a result, approximately $1/2$ of the total $k/\varphi^{i-1}$ remaining non-zero entries can be located and solved in $Q$ bits precision with a high probability $\Theta(1)$ which is exponential in $k$. Further define $\varphi = 1/(1 - \sigma)$ to be a parameter for the remaining stages.

c) $(\log_{\varphi}(k/\log_2 k) + 1)$-th Stage (Wind-up Stage): After $\log_{\varphi}(k/\log_2 k)$ stages, the algorithm stops combination, and globally decode the entire length-$n$ input vector given the information from the previous stages III-B2a and III-B1b in the last stage with $O(\log_2 k)$ measurements. In this sense, for the decoder, the information are communicated within the entire decoding circuit, which helps improve the performance with respect to the error probability. Actually, as an overall performance, the algorithm could achieve an average block error probability $P_{blk}$ of order $1/k$ and the bit-meters of order $O\left(\frac{nk}{\log n} \sqrt{\log \frac{1}{P_{blk}}} \right)$.

As a summary, the first chain algorithm combine local sub-circuits sequentially and transmit the information together gradually to resolve the input vector $X$, which is in performance better than the second shotgun algorithm. And the shortage for the chain algorithm is that for large size of input vector $X$, it necessarily requires several powerful central node within each local sub-circuit to handle the gathered information. For the shotgun algorithm, every node has the same functionality and the decoder merely needs to filter out the possible single non-zero entry in each local sub-circuit at each stage. Below we restate the Theorem 4 as the main result proved in Section IV-B using the algorithms given above.

Theorem 2. For the Compressive Sensing Model $(n, m, k)$ in a decoding circuit DecCkt implemented on the Implementation Model $(\rho(\Lambda), \mu)$ that achieves an average block error probability $P_{blk} = \Omega(e^{-m})$ and a fixed precision $Q$, if the decoder satisfy the regime $m = O(k)$ and the sub-linear regime $k = n^{1 - \beta}$ where $\beta \in (0, 1)$, then the decoder bit-meters are asymptotically

$^7$The details are given in the Section IV.
$^8$A more concise description is in the Section IV.
$^9$Again, the details are given in the Section IV.
Figure 5. This graph illustrates the flow of distributive-decoding algorithms. We decode local information stage by stage and end up with a clearing stage to improve performance. As a result stated in \( \mathcal{O}(k) \) under our assumptions and fix our interested regime \( m = \Theta(k) \), the bit-meters are upper bounded by \( \Theta \left( \sqrt{n} \log n \left( \log \left( \frac{1}{\mathcal{P}_{\text{blke}}} \right) \right) \right) \).

Figure 6. This graph illustrates the stencil partition on circuit. The sub-lattice which has a larger parallelepiped defines the sub-circuits. For instance, for the sub-circuit on the left-up corner contains the fundamental parallelepipeds, the order of quotient \( \lambda = 9 \), and it has 6 input-nodes and 10 output-nodes. Moreover, it has 1 output-node in the inner parallelepiped.

upper bounded as:

\[
\text{bit-meters(DecCkt)} = \Theta \left( \sqrt{\frac{n}{\log n}} \left( \log \frac{1}{\mathcal{P}_{\text{blke}}} \right) \right).
\]

IV. Main Results

A. Fundamental Limit

In this section, a lower bound of bit-meters for all possible compressive sensing decoding algorithms is considered. The idea of showing this lower bound comes from [12], which is using a "stencil" to divide the entire circuit into several sub-circuits, and find the minimal number of bits communicated between each sub-circuit, times a distance induced by both the side-length of sub-circuits and a fraction parameter introduced in the following definition. (Note that we define the circuit using lattice, so we called each sub-circuit "parallelepiped" sometimes in the coming context \(^{10}\)).

**Definition 12 (Stencil-partition).** The Stencil \((\lambda, \eta, u)\) on the Implementation Model \((\rho(\Lambda), \mu)\) is defined as a sub-lattice \( \Lambda_0 \subseteq \Lambda \) with order of quotient \( |\Lambda/\Lambda_0| = \lambda \) or the cosets \( u + \Lambda_0 = \{u + v : v \in \Lambda_0\} \) where \( u \in \Lambda \) induce a partition of a circuit into sub-circuits, each occupying the same substrate area \( \text{det}(\Lambda_0) \). And fraction parameter \( \eta \) defines the inner parallelepiped and the outer parallelepiped for each sub-circuit. If any computational node lies on the boundary of an outer parallelepiped, then it is arbitrarily included in one of the sub-circuit. Let the \( i \)-th sub-circuit have \( m_i \) input-nodes and \( n_i \) output-nodes within the outer parallelepiped and \( n_i^{\text{inside}} \) output-nodes inside the inner parallelepiped.

\(^{10}\)Although we are using a 2-D substrate represented by a 2-D sub-space in Euclidean space, for the convenience of possible generalization in the future, we just use the term "parallelepiped" instead of "parallelogram".
Lemma 1. Let $L$ denote the number of sub-circuits by Stencil-partition. Then the packing radius $\rho(\Lambda_0)$ of the outer parallelepipeds is given by

$$\rho(\Lambda_0) = \sqrt{\frac{\sigma(\Lambda_0)(n + m)}{\sigma(\Lambda)L}} \rho(\Lambda).$$

Proof: By definition 11 of the Implementation Model ($\rho(\Lambda), \mu$), the lattice $\Lambda$ has a packing radius $\rho(\Lambda) > 0$, and since for 2-D lattice, there exists a positive packing density $\sigma(\Lambda) > 0$ such that

$$\sigma(\Lambda) = \frac{\pi \rho^2(\Lambda)}{\det(\Lambda)}.$$

Similarly for the sub-lattice $\Lambda_0$, we also have a positive packing density $\sigma(\Lambda_0) > 0$ such that $\sigma(\Lambda_0) = \frac{\pi \rho^2(\Lambda_0)}{\det(\Lambda_0)}$. Moreover, the cardinality of the quotient $\Lambda/\Lambda_0$ equals to $\det(\Lambda_0)/\det(\Lambda)$. Since $L = \frac{n + m}{\det(\Lambda_0)}$, we conclude that

$$\rho(\Lambda_0) = \sqrt{\frac{\sigma(\Lambda_0)(n + m)}{\sigma(\Lambda)L}} \rho(\Lambda).$$

Lemma 2. On the Implementation Model ($\rho(\Lambda), \mu$), for any fraction parameter $\eta > 0$, there exists a point $u \in \Lambda$ for Stencil $(\lambda, \eta, u)$ such that the number of output-nodes covered by the Stencil is lower bounded by

$$\sum_{i} n_{i}^{\text{inside}} \geq n (1 - 2\eta)^2.$$  \hspace{1cm} (1)

Proof: Note that $n (1 - 2\eta)^2$ is the expectation of number of output-nodes covered by the Stencil, if the point $u \in \Lambda$ is uniformly distributed. Thus at least there exists one point $u$ satisfies the requirement.

Lemma 3. On the Implementation Model ($\rho(\Lambda), \mu$), let $L$ be the number of sub-circuits, for any fraction parameter $\eta > 0$ and any choice of $u \in \Lambda$ for Stencil $(\lambda, \eta, u)$, the number of sub-circuits covered by the inner parallelepiped which satisfy $m_i \leq \min (2m/L, n_i)$ is larger or equal to $\min ((1 - R)/ (1 + R), 1/2) L$ where $R$ is the rate of compressive sensing defined by $R = m/n$.

Proof: Assume $n > m$, firstly we choose the origin $O$ of the Stencil such that the location of output-nodes satisfies the equation given above. Then for this fixed choice of $u \in \Lambda$, we’re going to consider the worst location of the input-nodes which minimizes the fraction of input-nodes satisfy $m_i \leq \min (2m/L, n_i)$. We define an inner sub-circuit as non-locally decodable if $m_i \leq \min (2m/L, n_i)$ and otherwise we call it locally decodable. Figure IV-A explicitly gives an example. Thus for the first case $m_i \leq 2m/L$, the minimal fraction of “non-locally decodable” nodes is 1/2. Similarly, the minimal fraction of “good” nodes is $(1 - R)/ (1 + R)$ for the second case $m_i \leq n_i$. Since the total number of input-nodes is $m$, then with the consideration of overlapping cases, $\min ((1 - R)/ (1 + R), 1/2)$ is a lower bound for the fraction of input-nodes with $m_i \leq \min (2m/L, n_i)$. Below we prove the claim explicitly.

\footnote{Actually $m_i \leq \min (2m/L, n_i)$ is merely a sufficient condition for a sub-circuit to be non-locally decodable, however, we will use the term “non-locally decodable” to imply that the sub-circuit satisfies $m_i \leq \min (2m/L, n_i)$.}
Let $\chi = \min(2m/L, (n + m)/2L)$, then the fraction $f_1$ of “locally decodable” nodes among all sub-circuits defined by the stencil satisfies $f_1 \leq m/\chi L$, which implies the fraction $f_2$ of “non-locally decodable” nodes satisfies $f_2 = 1 - f_1 \geq 1 - m/\min(m/2, (n + m)/2) = \min(1/2, (1 - R)/(1 + R))$.

Lemma 4. If at most $H(\mathbf{X})/3$ bits of information is available to obtain an estimate $\hat{\mathbf{X}}$ of a variable $\mathbf{X}$ with entropy $H(\mathbf{X})$, then $\Pr[\hat{\mathbf{X}} \neq \mathbf{X}] \geq 1/9$.

Proof: Modifying Fano’s inequality \cite{6} applied to reconstructing the message $\mathbf{X}$ by defining an error random variable $E(\mathbf{X}, \hat{\mathbf{X}})$ such that

$$E(\mathbf{X}, \hat{\mathbf{X}}) = \begin{cases} 1 & \text{if } \hat{\mathbf{X}} = \mathbf{X} \\ 0 & \text{if } \hat{\mathbf{X}} \neq \mathbf{X} \end{cases}$$

Then since the input vector $\mathbf{X}$, the output vector $\mathbf{Y}$ and the recovery vector $\hat{\mathbf{X}}$ form a Markov chain $\mathbf{X} \rightarrow \mathbf{Y} \rightarrow \hat{\mathbf{X}}$, we get $H(\mathbf{X}) = H(\mathbf{X} | \hat{\mathbf{X}}) + I(\mathbf{X}; \hat{\mathbf{X}})$.

Thus, elaborately modify Fano’s inequality \cite{6},

$$H(\mathbf{X} | \hat{\mathbf{X}}) = H(E(\mathbf{X}, \hat{\mathbf{X}}), \mathbf{X} | \hat{\mathbf{X}})$$

$$= H(E(\mathbf{X}, \hat{\mathbf{X}}) | \hat{\mathbf{X}}) + \Pr[E(\mathbf{X}, \hat{\mathbf{X}}) = 0] H(\mathbf{X} | \hat{\mathbf{X}}, E(\mathbf{X}, \hat{\mathbf{X}}) = 0)$$

$$+ \Pr[E(\mathbf{X}, \hat{\mathbf{X}}) = 1] H(\mathbf{X} | \hat{\mathbf{X}}, E(\mathbf{X}, \hat{\mathbf{X}}) = 1)$$

$$\leq h_b(P_e) + P_e H(\mathbf{X}).$$

Given the available information $I$ of at most $H(\mathbf{X})/3$ bits, the error probability $P_e := \Pr[\hat{\mathbf{X}} \neq \mathbf{X}]$ is lower bounded by

$$P_e H(\mathbf{X}) + h_b(P_e) \geq H(\mathbf{X}) - I(\mathbf{X}; \hat{\mathbf{X}}) = H(\mathbf{X}) - H(\mathbf{X})/3 = 2H(\mathbf{X})/3,$$

where $h_b(\cdot)$ on the LHS is the binary entropy function (will also appear in the later parts).

Then since $n > m > 1$ we have

$$P_e \geq \frac{2H(\mathbf{X})/3 - 1}{H(\mathbf{X})} \geq \frac{2}{3} - \frac{1}{2} > \frac{1}{9}.$$

Lemma 5. For the Compressive Sensing Model $(n, m, p)$ in a decoder circuit $\text{DecCkt}$ implemented on the Implementation Model $(\rho, \lambda, \mu)$, if the relative error satisfies $||\mathbf{X} - \mathbf{\hat{X}}||_{\ell_q} / ||\mathbf{X}||_{\ell_q} \leq 2^{-Q}$, then there exists a constant $C_0(\mathbf{X}, \ell_q) < 1$ only depend on the input vector $\mathbf{X}$ and norm $\ell_q$ such that at least $C_0npQ$ bits need to be received by the out-put nodes asymptotically.

Proof: Let $Q_i$ denote the number of bits for each entry $\hat{X}_i$ of the recovery vector $\hat{\mathbf{X}}$ that it shares the same bits with the input vector $\mathbf{X}$. Hence $2^{-Q_i - 1} \leq |\hat{X}_i^j - X_i^j| / |X_i| \leq 2^{-Q_i + 1}$ for all $i$. Suppose $||\mathbf{X} - \mathbf{\hat{X}}||_{\ell_q} / ||\mathbf{X}||_{\ell_q} \leq 2^{-Q}$, then we get

$$\left( \sum_{i=1}^{k} |X_i - \hat{X}_i|^q \right)^{1/q} \leq 2^{-Q} \left( \sum_{i=1}^{k} |X_i|^q \right)^{1/q},$$

which implies

$$\left( \frac{2^{-qQ_i + 1} \sum_{i=1}^{k} |X_i|^q}{\sum_{i=1}^{k} |X_i|^q} \right)^{1/q} \leq 2^{-Q}.$$

Since $|X_i| \geq 0$ for all $i$, recall that as our assumption there is constant $U \geq 0$ such that $|X_i| \leq U$ for any $i$ and by Jensen’s Inequality (see, for instance in book \cite{14}) we get

$$|U|^q \sum_{i=1}^{k} Q_i \geq \sum_{i=1}^{k} |X_i|^q Q_i \geq \sum_{i=1}^{k} |X_i|^q U,$$

thus $\sum_{i=1}^{k} Q_i \geq C_0kQ$ with $C_0 = \sum_{i=1}^{k} |X_i|^q / kU < 1$. And the asymptotic result follows as $n \rightarrow \infty$. 

\[\square\]
Lemma 6. For the Compressive Sensing Model \((n, m, p)\) in a decoder circuit \(\text{DecCkt}\) implemented on the Implementation Model \((\rho(\Lambda), \mu)\), for any decoder sub-circuit \(\text{SubCkt}\) obtained via Stencil-partition with \(m_i \leq \min(2m/L, n_i)\), if bit-meters(\(\text{SubCkt}\)) \(\leq n\rho(\Lambda_0)C_0 (n_i^{\text{inside}} - m_i) pQ/3\), then \(P_e^{\text{blk}} \geq p^{2m/L}/9\), given \(L\) is the number of sub-circuits.

Proof: In the \(i\)-th sub-circuit, since the number of bit-meters for the sub-circuit with the assumption \(m_i \leq \min(2m/L, n_i)\) is smaller than \(n\rho(\Lambda_0)C_0 (n_i^{\text{inside}} - m_i) pQ/3\), and the distance between the outer parallelepiped and the inner parallelepiped is bounded from below by \(n\rho(\Lambda_0)\), at most \(C_0 (n_i^{\text{inside}} - m_i) pQ/3\) bits of information \(I\) can be communicated from outside the outer parallelepiped to the inside of inner parallelepiped.

Now since \(m_i \leq n_i\), corresponding to the \(n_i\) output nodes, if among these \(n_i\) entries we have \(m_i\) entries in the input vector \(X\) take non-zero values, then the decoder cannot solve all values of the \(Q\) bits in output-nodes. We denote this event by \(L\). Then \(L\) occurs with probability at least \(p^{2m/L}\) since \(m_i \leq 2m/L\).

Conditioning on the event \(L\), from Lemma 4 using Fano’s inequality [8], if the received entropy is smaller than \(C_0 (n_i^{\text{inside}} - m_i) pQ/3\), the average block error probability defined by \(P_e^{\text{blk}} = \Pr(E^{\text{blk}} = 1)\) where \(E^{\text{blk}} = 1\) if \(\|X - \hat{X}\|_{\ell_2}/\|X\|_{\ell_2} > 2^{-Q}\) and otherwise \(E^{\text{blk}} = 0\), will be larger than \(1/9\). Thus, given the assumption in Lemma 6, the (unconditional) error probability for recovering the \(n_i\) entries of input vector \(X\) with precision \(Q\) in the \(i\)-th sub-circuit is lower bounded by \(p^{2m/L}/9\). Also, for the entire circuit, since the average block error probability \(P_e^{\text{blk}}\) is larger than that for any sub-circuit, we obtain the Lemma 6.

Theorem 3. For the Compressive Sensing Model \((n, m, p)\) in a decoding circuit \(\text{DecCkt}\) implemented on the Implementation Model \((\rho(\Lambda), \mu)\) that achieves an average block error probability \(P_e^{\text{blk}} \geq 0\), the decoder bit-meters are lower bounded as:

\[
\text{bit-meters}(\text{DecCkt}) \geq \frac{\rho(\Lambda)}{24\sqrt{2}} C_0 \left(\frac{n}{4} - m\right) pQ \sqrt{\left(\frac{1}{2R} - 1 - R^2 + 2R + 2/\mu\right)} \log_2 10 P_e^{\text{blk}}.
\]

Proof: The outer parallelepiped of the Stencil divide the circuit into \(L\) sub-circuits. Let the \(i\)-th sub-circuit have \(m_i\) input-nodes and \(n_i\) output-nodes within the outer parallelepiped and \(n_i^{\text{inside}}\) output-nodes inside the inner parallelepiped. Using Lemma 2 and Lemma 3 we can choose a fixed origin \(O\) of the Stencil such that at least \((1 - 2\eta)^2\) fraction of the \(n\) output-nodes are covered by the inner parallelepiped. Moreover, note that the number of sub-circuits covered by the inner Stencil with \(m_i \leq \min(2m/L, n_i)\) is \(\min((1 - R)/(1 + R), 1/2)\) \(L\), which will be used in the later part.

Next, setting \(L = 2m/\log_2 10P_e^{\text{blk}}\), from Lemma 6 if we assume the bit-meters for a “non-locally decodable” sub-circuit are smaller than \(n\rho(\Lambda_0)C_0 (n_i^{\text{inside}} - m_i) pQ/3\), then the average block error probability \(P_e^{\text{blk}}\) is lower bounded as

\[P_e^{\text{blk}} \geq p^{2m/L}/9 = p^{\log_2 10P_e^{\text{blk}}}/9 = 10P_e^{\text{blk}}/9,
\]

which is a contradiction. Thus for each “non-locally decodable” sub-circuit, i.e., the sub-circuit satisfies \(m_i \leq \min(2m/L, n_i)\) obtained via the Stencil-partition, let \(\mu(i)\) denote the bit-meters for the \(i\)-th sub-circuit generated by stencil partition, we must have for all \(i\) with \(m_i \leq \min(2m/L, n_i)\), \(\mu(i) \geq C_0 (n_i^{\text{inside}} - m_i) pQn\rho(\Lambda_0)/3\).

From Lemma 2 \(\sum_{i=1}^L n_i^{\text{inside}} \geq n (1 - 2\eta)^2\), therefore, we can bound the total bit-meters in the decoding circuit by

\[
\text{bit-meters}(\text{DecCkt}) \geq \sum_{i=1}^L \mu(i) \\
\geq \sum_{m_i \leq \min(2m/L, n_i)} \mu(i) + \sum_{m_i \geq 2m/L} \mu(i) \\
\geq \sum_{m_i \leq \min(2m/L, n_i)} C_0 (n_i^{\text{inside}} - m_i) pQn\rho(\Lambda_0)/3 + \sum_{m_i \geq 2m/L} \mu(i)
\]

Now we define three types of sub-circuits under the condition \(m_i \geq 2m/L\), first we use \(LD_1\) to denote the set of \(i\) such that \(m_i \geq 2m/L\) and \(\mu(i)\) satisfies \(\mu(i) \geq C_0 (n_i^{\text{inside}} - m_i) pQn\rho(\Lambda_0)/3\); then secondly let \(LD_2\) denote the set of \(i\) such that \(m_i \geq 2m/L\) and \(\mu(i)\) satisfies \(\mu(i) < C_0 (n_i^{\text{inside}} - m_i) pQn\rho(\Lambda_0)/3\); finally let \(LD_3\) denote the set of \(i\)
such that \( m_i \leq \min(2m/L, n_i) \), then it follows that

\[
\text{bit-meters}(\text{DecCkt}) \geq \sum_{i \in LD_1 \cup LD_2} C_0 (n_i^{\text{inside}} - m_i) p Q \eta \rho(\Lambda_0)/3 + \sum_{i \in LD_2} \mu(i)
\]

\[
\geq \frac{1 - R}{2(1 + R)} \sum_{i=1}^L C_0 (n_i^{\text{inside}} - m_i) p Q \eta \rho(\Lambda_0)/3.
\]

\[
\geq \frac{1 - R}{2(1 + R)} C_0 ((1 - 2\eta)^2 n - m) p Q \eta \rho(\Lambda_0)/3.
\]

(3)

Since by Lemma 1 \( \rho(\Lambda_0) \leq \frac{1}{2} \rho(\Lambda) \sqrt{n + m}/L = \frac{1}{2} \rho(\Lambda) \sqrt{\log p 10 P^{\text{blk}}_c (1 + 1/2\eta)} \), substituting \( \rho(\Lambda_0) \) into the equation (3) we get

\[
\text{bit-meters}(\text{DecCkt}) \geq \frac{\eta \rho(\Lambda)}{6 \sqrt{2}} C_0 ((1 - 2\eta)^2 n - m) p Q \sqrt{\left(\frac{1}{2R} - \frac{1 - R^2 + 2R}{(1 + R)^2}\right)} \sqrt{\log p 10 P^{\text{blk}}_c}.
\]

Choosing \( \eta = 1/4 \) yields the Theorem 3.

Hence for the regime \( m = O(k) \) of our interests, one could derive the following result in order expression, which serves as the benchmark for our design of algorithm.

**Corollary 1.** For the Compressive Sensing Model \((n, m, k)\) in a decoding circuit \(\text{DecCkt} \) implemented on the Implementation Model \((\rho(\Lambda), \mu)\) that achieves an average block error probability \(P^{\text{blk}}_c = \Omega(e^{-m})\) and a fixed precision \(Q\), if the decoder satisfy the regime \( m = O(k) \), then the decoder bit-meters are asymptotically lower bounded as:

\[
\text{bit-meters}(\text{DecCkt}) = \Omega \left( \frac{nk}{\log n} \sqrt{\frac{1}{P^{\text{blk}}_c}} \right).
\]

**Proof:** Base on the result of Theorem 3 since asymptotically, in the Compressive Sensing Model \((n,m,p)\), the expectation of number of non-zero entries in the input vector \(X\) is concentrated at \(k = np\). By Hoeffding’s inequality, we can bound number of non-zero entries in the input vector \(X\) in the Compressive Sensing Model \((n,m,p)\) in the range \([k/2, 3k/2]\) with probability \(1 - e^{-\Theta(k)}\). Hence asymptotically, if we have \(P^{\text{blk}}_c = \Omega(e^{-m})\), we can substitute \(k = np\) in the inequality 2 and get

\[
\text{bit-meters}(\text{DecCkt}) \geq \frac{\rho(\Lambda)}{24 \sqrt{2}} C_0 \left( \frac{n}{4} - m \right) \frac{kQ}{n} \sqrt{\left(\frac{1}{2R} - \frac{1 - R^2 + 2R}{(1 + R)^2}\right)} \sqrt{\log p 10 P^{\text{blk}}_c}
\]

(4)

which differ from the original lower bound in the inequality 2 by simply a constant.

Since \(k = o(n)\) by the sparse assumption, in the regime \(m = O(k)\), we get \(R = m/n = o(1)\). And since \(m = O(k)\), letting \(R = k/n\) we can asymptotically bound the bit-meters as

\[
\text{bit-meters}(\text{DecCkt}) = \Omega \left( \frac{nk}{\log n} \sqrt{\frac{1}{P^{\text{blk}}_c}} \right).
\]

B. Achievability

Firstly we define the encoding matrix and the corresponding decoding steps. After that, several properties follow and whereby we are able to prove the achievability results.

1) Choice of Encoding Matrix:

2) Decoding Steps:

**Lemma 7.** For a decoding circuit with decoding steps defined above, it satisfies the following properties (where \(i\) denotes the index of stages):

- The communication distance \(D_{\text{DecCkt}}\) is upper bounded by

\[
D^i_{\text{DecCkt}} = \begin{cases} O(\sqrt{2^{1-i} n/k}) & \text{if } i = 1, 2, \ldots, \log_2(k/\log_2 k) \\ O(\sqrt{n}) & \text{if } i = \log_2(k/\log_2 k) + 1; \end{cases}
\]
• The number of decoding steps $N_{\text{DecCkt}}^{i\text{-th Stage}}$ is upper bounded by

$$N_{\text{DecCkt}}^{i\text{-th Stage}} = \begin{cases} \mathcal{O}(k/2^{i-1}) & \text{if } i = 1, 2, \ldots, \log_2(k/\log_2 k) \\ \mathcal{O}(\sqrt{k}) & \text{if } i = \log_2(k/\log_2 k) + 1 \end{cases}$$

• The bit-precision arithmetic required in each communication between nodes is upper bounded by

$$B_{\text{DecCkt}} = \mathcal{O}(1).$$

**Theorem 4.** For the Compressive Sensing Model $(n, m, k)$, there exists a compressive sensing algorithm for measurement matrix $A$ with the following properties:

1. For every input vector $X \in \mathbb{R}^n$, with probability $1-\mathcal{O}(1/\sqrt{k})$ over the choice of $A$, the algorithm produces a recovery vector $\hat{X}$ such that $||X - \hat{X}||_1/||X||_1 \leq 2^{-Q}$;
2. The number of measurements $m \leq 2ck + \sqrt{k}$ where $c$ is the measurements constant;
3. The number of decoding steps by the algorithm is $N_{\text{DecCkt}} = \mathcal{O}(k)$.

**Theorem 5.** For the Compressive Sensing Model $(n, m, k)$ in a decoding circuit $\text{DecCkt}$ implemented on the Implementation Model $(\rho(\Lambda), \mu)$ that achieves an average block error probability $P_{\text{blk}} = \Omega(e^{-m})$ and a fixed precision $Q$, if the decoder satisfy the regime $m = \mathcal{O}(k)$ and the sub-linear regime $k = n^{1-\beta}$ where $\beta \in (0, 1)$, then the decoder bit-meters are asymptotically upper bounded as:

$$\text{bit-meters}(\text{DecCkt}) = \mathcal{O}
\left(\sqrt{nk \log n \log \frac{1}{P_{\text{blk}}}}\right).$$

**Proof:** From Lemma 7, the number of decoding steps $N_{\text{DecCkt}}$ is upper bounded by

$$N_{\text{DecCkt}}^{i\text{-th Stage}} = \begin{cases} \mathcal{O}(k/2^{i-1}) & \text{if } i = 1, 2, \ldots, \log_2(k/\log_2 k) \\ \mathcal{O}(\sqrt{k}) & \text{if } i = \log_2(k/\log_2 k) + 1 \end{cases},$$

where $i$ denotes the index of stages. It implies the times of communications between nodes in the circuit is also of the same order as the number of decoding steps for each $i$-th stage. And from Lemma 7 the total bit-meters is bounded by:

$$\text{bit-meters}(\text{DecCkt}) = \sum_{i=1}^{\log_2(k/\log_2 k)+1} N_{\text{DecCkt}}^{i\text{-th Stage}} B_{\text{DecCkt}}$$

$$= \sum_{i=1}^{\log_2(k/\log_2 k)} \mathcal{O}\left((\log n + Q) \sqrt{nk/2^{i-1}}\right) + \mathcal{O}\left(\log n \sqrt{nk}\right)$$

$$= \sum_{i=1}^{\log_2(k/\log_2 k)} \mathcal{O}\left((\log n) \sqrt{nk/2^{i-1}}\right) + \mathcal{O}\left(\log n \sqrt{nk}\right)$$

$$= \mathcal{O}\left(\log n \sqrt{nk}\right)$$

$$= \mathcal{O}\left(\sqrt{nk \log n \log \frac{1}{P_{\text{blk}}}}\right).$$

We get equation 5 because as the assumption the precision parameter $Q$ is fixed which is reasonable in many applications. Summing up all terms in equation 5 yields equation 6. And by Theorem 4 the average block error probability $P_{\text{blk}}$ satisfies $P_{\text{blk}} = \mathcal{O}(1/\sqrt{k})$, thus $\mathcal{O}(\sqrt{\log n}) = \mathcal{O}(\sqrt{\log k})$ in the sub-linear regime $k = n^{1-\beta}$ where $\beta \in (0, 1)$. Hence $\mathcal{O}(\sqrt{\log n}) = \mathcal{O}(\sqrt{1/\log P_{\text{blk}}})$ implies equation 7. Therefore as the conclusion we get $\text{bit-meters}(\text{DecCkt}) = \mathcal{O}\left(\sqrt{nk \log n \log \frac{1}{P_{\text{blk}}}}\right)$.

**References**

[1] Mayank Bakshi, Sidharth Jaggi, Sheng Cai, and Minghua Chen. Sho-fa: Robust compressive sensing with order-optimal complexity, measurements, and bits. In Communication, Control, and Computing (Allerton), 2012 50th Annual Allerton Conference on, pages 786–793. IEEE, 2012.

[2] Sandeep N Bhatt, Gianfranco Bilardi, and Geppino Pucci. Area-time tradeoffs for universal vlsi circuits. *Theoretical Computer Science*, 408(2):143–150, 2008.
[3] Richard P Brent and HT Kung. The area-time complexity of binary multiplication. Journal of the ACM (JACM), 28(3):521–534, 1981.

[4] Emmanuel J Candès, Justin Romberg, and Terence Tao. Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information. Information Theory, IEEE Transactions on, 52(2):489–509, 2006.

[5] BM Chazelle and LM Monier. Towards more realistic models of computation for vlsi. 1981.

[6] Herman Chernoff. A measure of asymptotic efficiency for tests of a hypothesis based on the sum of observations. The Annals of Mathematical Statistics, pages 493–507, 1952.

[7] Richard Cole and Alan Siegel. Optimal vlsi circuits for sorting. Journal of the ACM (JACM), 35(4):777–809, 1988.

[8] Thomas M Cover and Joy A Thomas. Elements of information theory. John Wiley & Sons, 2012.

[9] David L Donoho. Compressed sensing. Information Theory, IEEE Transactions on, 52(4):1289–1306, 2006.

[10] Michael T Goodrich and Michael Mitzenmacher. Invertible bloom lookup tables. In Communication, Control, and Computing (Allerton), 2011 49th Annual Allerton Conference on, pages 792–799. IEEE, 2011.

[11] Geoffrey Grimmett and David Stirzaker. Probability and random processes, volume 2. Oxford Univ Press, 1992.

[12] Pulkit Grover. Information-friction and its impact on minimum energy per communicated bit. In Information Theory Proceedings (ISIT), 2013 IEEE International Symposium on, pages 2513–2517. IEEE, 2013.

[13] Mark R Kramer and Jan van Leeuwen. The vlsi complexity of boolean functions. In Logic and Machines: Decision Problems and Complexity, pages 397–407. Springer, 1984.

[14] Marek Kuczma. An introduction to the theory of functional equations and inequalities: Cauchy’s equation and Jensen’s inequality. Springer, 2008.

[15] Charles Eric Leiserson. Area-efficient vlsi computation. Technical report, DTIC Document, 1981.

[16] Carver Mead and Lynn Conway. Introduction to VLSI systems, volume 1080. Addison-Wesley Reading, MA, 1980.

[17] Bhabani P Sinha and Pradip K Srimani. A new parallel multiplication algorithm and its vlsi implementation. In Proceedings of the 1988 ACM sixteenth annual conference on Computer science, pages 366–372. ACM, 1988.

[18] Clark D Thompson. Area-time complexity for vlsi. In Proceedings of the eleventh annual ACM symposium on Theory of computing, pages 81–88. ACM, 1979.

[19] Clark D. Thompson. The VLSI complexity of sorting. Springer, 1981.

[20] Clark David Thompson. A complexity theory for VLSI. PhD thesis, Carnegie-Mellon University, 1980.