Practical MU-MIMO Detection and LDPC Decoding through Digital Annealing

Po-Shao Chen, Wei Tang, Zhengya Zhang
Department of Electrical and Computer Engineering, University of Michigan, Ann Arbor

***Abstract*—Digital annealing has been successfully applied to solving combinatorial optimization (CO) problems. It is more flexible, robust, and easier to deploy on edge platforms compared to its counterparts including quantum annealing and analog and in-memory Ising machines. In this work, we apply digital annealing to compute-intensive communication digital signal processing problems, including multi-user detection in multiple-input and multiple-output (MU-MIMO) wireless communication systems and decoding low-density parity-check (LDPC) codes. We show that digital annealing can achieve near maximum likelihood (ML) accuracy for MIMO detection with even lower complexity than the conventional minimum mean square error (MMSE) detection. In LDPC decoding, we enhance digital annealing by introducing a new cost function that improves decoding accuracy and reduces computational complexity compared to the standard formulations.**

***Index Terms*—Combinatorial optimization, digital annealing, MIMO detection, LDPC decoding**

I. Introduction

Combinatorial optimization (CO) solves problems involving discrete variables in combinatorial structures [1]. CO has proven to be highly relevant in solving modern-day problems such as logistics and route planning, scheduling, network design, and resource allocation.

CO methods have also been studied in the field of communication digital signal processing (DSP). Recent research has proposed using CO methods for multi-user multiple-input multiple-output (MU-MIMO) detection [2], [3] and for decoding low-density parity-check (LDPC) codes [4], [5]. The common approach among these studies is to formulate the problems in either an Ising model [6] or a quadratic unconstrained binary optimization (QUBO) model [7]. The Ising model represents discrete variables as magnetic spins on a lattice structure, while the QUBO model expresses these variables as binary quadratic equations. The problems are then solved using either quantum annealing [8] or classical annealing [9], as depicted in Fig. 1.

Quantum annealing leverages the unique properties of quantum bits (qubits) [10], [11]. The goal of quantum annealing is to find the ground state, which corresponds to the configuration with the lowest energy. However, quantum computers operate at extremely low temperatures near 0 K and require radio frequency and magnetic shielding [12]. The significant overhead involved in operating quantum computers, which includes cooling and shielding requirements, combined with their high sensitivity, renders them unsuitable for communication DSPs that need to be deployed on user terminals or base stations.

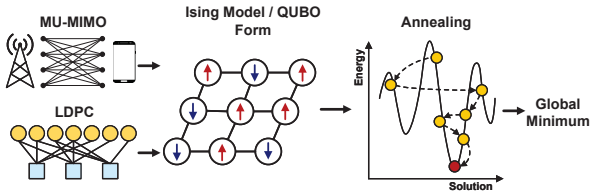

Fig. 1: Illustration of steps to solve a CO problem.

On the other hand, classical annealing begins by initiating a solution and incrementally making randomized modifications in an iterative manner. This process enables the exploration of a broad solution space guided by a temperature parameter that gradually decreases over time to avoid local minima. Classical annealing has been implemented using CMOS circuits, including digital annealers [9], [13]–[15], as well as analog forms using compute-in-memory [16] and coupled oscillators [17]–[19]. Analog annealers tend to be more energy-efficient and can provide higher performance, but they often suffer from increased sensitivity and scalability limitations, making them suitable only for smaller problem sizes. Additionally, analog annealers are limited in connectivity, restricting the types of problems that can be mapped onto them. Digital annealers, on the other hand, can overcome these challenges and present themselves as a general-purpose solution that can be practically deployed on communication DSPs.

Although recent research has started to apply annealing to communication DSPs, the results are not yet close to being a practical solution. Quantum annealers have been used in MIMO detection [2] and LDPC decoding [5], but it is currently far from feasible to deploy quantum computers in remote base stations, let alone user terminals. More fundamentally, the accuracy, measured in bit error rate (BER) under a range of signal-to-noise ratio (SNR), of MIMO detection and LDPC decoding is not on par with the state-of-the-art alternatives to annealing. A digital annealer was used in an LDPC decoder [4], but the BER performance is unclear. A coupled-oscillator annealer was used in a MIMO detector [3], but its scalability is limited.

To overcome these challenges, this paper focuses on digital annealers, which are more robust, scalable, general-purpose, and readily deployable compared to quantum and analog annealers. We emphasize both the accuracy and complexity of our digital annealing solution. Specifically, we address MIMO detection and LDPC decoding problems, contributing in two

978-3-9826741-0-0/DATE25/© 2025 EDAA

key areas:

- We demonstrate the mapping of various MIMO detection problems onto several types of digital annealers. Our results indicate that with appropriate choices, digital annealing not only surpasses state-of-the-art approaches in accuracy but also achieves this with reduced complexity.
- We analyze mappings of LDPC decoding problems onto annealers and show that the use of ancillary variables leads to significantly decreased accuracy. We propose a new mapping for digital annealing that eliminates ancillary variables. This mapping not only improves the accuracy of LDPC decoding but also reduces the hardware cost.

With demonstrated accuracy and complexity, our contributions aim to pave the way for the practical adoption of digital annealer DSPs.

II. Background

To use quantum annealing, a CO problem is first represented as an objective function to be minimized. Subsequently, the problem is formulated in an Ising model or a QUBO model. In both models, the system's energy is typically represented by the following general form:

$$E = \sum_i h_i q_i + \sum_{i<j} J_{ij} q_i q_j, \tag{1}$$

where E represents the energy of the system; i and j are indices of individual variables, also known as spins or qubits; h_i is the coefficient of the linear term that represents the bias associated with variable i; q_i and q_j represent the variables i and j, having bipolar values of $\{$-1, +1$\}$ in an Ising model and binary values of $\{0, 1\}$ in a QUBO model; and J_{ij} is the coefficient of the quadratic term, representing the interaction strength or coupling between variables i and j.

Prior studies on MIMO detection and LDPC decoding have employed both Ising and QUBO models. The coefficients of these models are subsequently mapped onto the qubits and their interactions on a quantum annealer [2], [5] or onto classical (non-quantum) annealers [3], [4]. Note that classical annealing is not restricted solely to Ising and QUBO models, allowing flexible problem formulation to achieve optimal solutions.

A. Digital Annealer

A digital annealer operates by randomly selecting a solution by flipping variables (between 0 and 1, or -1 and +1). To decide whether a solution is accepted, the following criterion is used:

$$e^{-\beta \Delta E} > random(0, 1), \tag{2}$$

where $\beta = \frac{1}{T}$ and T denotes the temperature. ΔE represents the change in energy from the current solution to the new solution obtained after the variable flip. At the beginning of the annealing process, β is set to β_{min}, representing a high temperature. During this stage, the annealer explores a larger search space and accepts more random flips, even when $\Delta E > 0$. As the annealing progresses, the temperature decreases to cool down the system. One popular choice is using a geometric cooling schedule. In each iteration, β is scaled up until it reaches β_{max}, at which point only flips that reduce the energy ($\Delta E < 0$) are accepted.

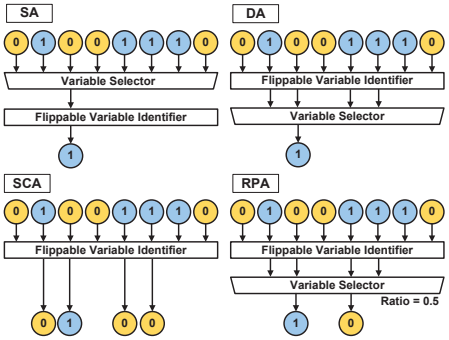

Fig. 2: Different types of annealing policies.

B. Annealing Policies

A digital annealer implements an iterative annealing policy to find the optimal solution. Several types of policies can be employed, including simulated annealing (SA) [20], digital annealing (DA) [20], stochastic cellular automata annealing (SCA) [13], and ratio-controlled parallel annealing (RPA) [14], as illustrated in Fig. 2. In SA, a random variable is selected and flipped if it satisfies Eq. (2). SA relies on a trial-and-error process. In contrast, DA, SCA, and RPA follow a different strategy. They first identify all variables that can be flipped based on Eq. (2). Among the identified flippable variables, DA chooses one variable to flip in each iteration, whereas SCA flips all the flippable variables at once. RPA utilizes a ratio-based approach, selecting a portion of the flippable variables to flip based on a predefined ratio.

In terms of computation, SA is sequential, while DA, SCA, and RPA parallelize the identification of flippable variables, making them faster but incurring a higher computational cost. Among DA, SCA, and RPA, DA is the slowest as it flips one variable at a time, SCA is the fastest as it flips all flippable variables simultaneously, and RPA falls in between depending on the ratio. However, SCA may introduce large disturbances to the system, leading to convergence challenges.

III. Multi-User MIMO Detection Problem

In a $N_t \times N_r$ uplink multi-user MIMO (MU-MIMO) wireless communication system, N_t represents the number of transmit antennas (e.g., one per mobile user) and N_r denotes the number of receive antennas at the base station. The uplink received data symbols at the base station $\mathbf{y} \in \mathbb{C}^{N_r \times 1}$ can be expressed as:

$$\mathbf{y} = \mathbf{H_c}\mathbf{x} + \mathbf{n}, \tag{3}$$

where $\mathbf{H_c} \in \mathbb{C}^{N_r \times N_t}$ represents the channel matrix, $\mathbf{x} \in \mathbb{C}^{N_t \times 1}$ represents the transmitted data symbols from users, and $\mathbf{n} \in \mathbb{C}^{N_r \times 1}$ is the noise. Maximum likelihood (ML) detection aims to find the most likely transmitted data symbols $\mathbf{x}$ given the received data $\mathbf{y}$. However, the complexity of the ML detection method is often too high to be feasible, especially for systems with large antenna arrays and high-order modulations. Therefore, a common practice is to use approximate methods, such as minimum mean square error (MMSE) detection [21]. MMSE detection is computationally efficient, but its BER is inferior compared to ML.

A. Ising Model Formulation

Annealing can be employed to tackle the ML detection problem. The ML solution aims to find $\mathbf{x}^*$ that minimizes the following cost function:

$$\mathbf{x}^* = \arg\min_{\mathbf{x}} ||\mathbf{y} - \mathbf{H_c}\mathbf{x}||^2, \tag{4}$$

where $\mathbf{x}$ can be expressed as:

$$\mathbf{x} = [x_1, x_2, ..., x_{N_t}] \in \{\pm 1\}^{N_t}, \tag{5}$$

in the case of binary phase-shift keying (BPSK) modulation.

To formulate the problem using an Ising model, a bipolar-valued vector $\mathbf{s}$ is introduced. The vector is obtained by augmenting $\mathbf{x}$ with a '1' at the end:

$$\mathbf{s} = [x_1, x_2, ..., x_{N_t}, 1]^{\mathrm{T}}. \tag{6}$$

Next, the coefficient matrix $\mathbf{J}$ is obtained by first augmenting the channel matrix $\mathbf{H_c}$ with a column of received data symbol vector $\mathbf{y}$, and then computing the Gram matrix $\mathbf{J}$:

$$\begin{aligned} \mathbf{H_c^{'}} &= [\mathbf{H_c}, \mathbf{y}] \in \mathbb{R}^{N_r \times (N_t+1)}, \\ \mathbf{J} &= \mathbf{H_c^{'}}^{\mathrm{T}} \mathbf{H_c^{'}} \in \mathbb{R}^{(N_t+1)\times(N_t+1)}. \end{aligned} \tag{7}$$

The Ising model of the ML detection can then be written as:

$$\mathbf{s}^* = \arg\min_{\mathbf{s}} \sum_{i=1}^{N_t+1} \sum_{j=1}^{N_t+1} J_{ij} s_i s_j, \text{ subject to } s_{N_t+1} = 1, \tag{8}$$

where J_{ij} is the element at the i^{th} row and j^{th} column in $\mathbf{J}$. For higher order modulations, such as quadrature phase-shift keying (QPSK) and quadrature amplitude modulation (QAM), the complex symbols can be expanded to real values before mapping them to an Ising model [22].

B. Digital Annealing Results

In [3], analog coupled oscillators were employed to implement annealing for MIMO detection. Our research takes a different approach by experimenting with a digital annealer using various annealing policies to investigate the optimal way to leverage them.

In digital annealing, the convergence and computational latency are measured based on the number of iterations. As depicted in Fig. 2, each iteration in SA involves selecting one variable and flipping it if it satisfies the condition in Eq. (2). On the other hand, in DA, SCA, and RPA, an iteration is comprised of identifying all the variables that can be flipped, which can be conveniently parallelized, and subsequently flipping one variable in DA, all variables in SCA, or a subset of variables in RPA. The computational complexity per iteration for SA is $\mathcal{O}(1)$, while for DA, SCA, and RPA, it is $\mathcal{O}(n)$, where n denotes the length of the input vector. However, DA, SCA, and RPA can leverage parallel computing hardware, resulting in fewer iterations and faster convergence.

We first compare the quality of different annealing policies for a digital annealer in Fig. 3 for MIMO detection in an example 16×64 QPSK MIMO system. This system consists of 16 transmit antennas (or 16 single-antenna user terminals) and 64 receive antennas, which is typical in a massive MIMO uplink system. Remarkably, even with only 50 iterations, DA quickly surpasses the performance of the MMSE detector and approaches ML. However, SA, SCA, and RPA (with a 0.5 ratio) all lag behind DA and underperform compared to MMSE. After increasing the number of iterations to 500, SA, SCA, and RPA show gradual improvement but still significantly underperform when compared to DA. These results are attributed to SA's slow convergence and SCA's inability to introduce sufficient randomness due to flipping all flippable variables.

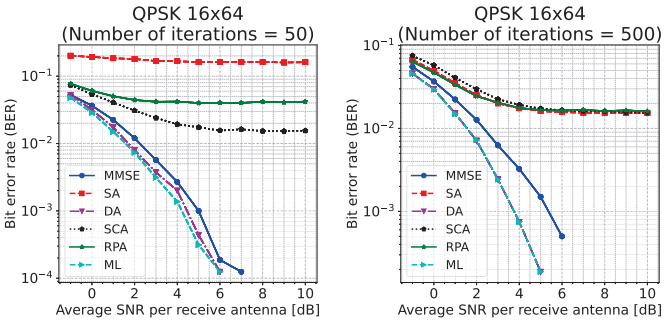

Fig. 3: BER comparison between MMSE, ML, and digital annealing using different annealing policies in a QPSK 16x64 MIMO system.

Across various antenna configurations including 16×16, 16×64, and 32×128, a digital annealer using DA consistently outperforms MMSE and matches ML, as shown in Fig. 4. In small to medium-scale MIMO setups like 16×16, the convergence of the digital annealer tends to be slower. However, in massive MIMO configurations that involve a larger number of receive antennas, the convergence can be achieved in under 100 iterations. This improvement is attributed to the better conditioned channels in massive MIMO systems [23]. It is important to highlight that a digital annealer offers a complexity advantage as it eliminates the need for matrix inversion which is required by an MMSE detector.

In Fig. 5, we present a comparison of the complexity among digital annealer (using the DA policy), MMSE, and ML detectors for 16×16 and 32×128 MIMO system. Here, the complexity is measured in terms of the number of operations. In the case of a 16×16 QPSK MIMO system, the digital annealer exhibits complexity that is approximately 7 orders of magnitude lower than ML. For a larger, 32×128 QPSK MIMO system, the digital annealer's complexity is even lower due to faster convergence, thanks to the better conditioned channel in massive MIMO.

The complexity of a digital annealer can be further reduced by implementing an early termination method. It has been observed that in a majority of cases, the cost function converges to a stable state after a small number of iterations. By applying early termination, the complexity of a digital annealer can be reduced by decreasing the average number of iterations, without compromising accuracy. Fig. 6 demonstrates the BER with early termination, showing minimal change, while the average number of iterations is reduced by 38% by implementing early

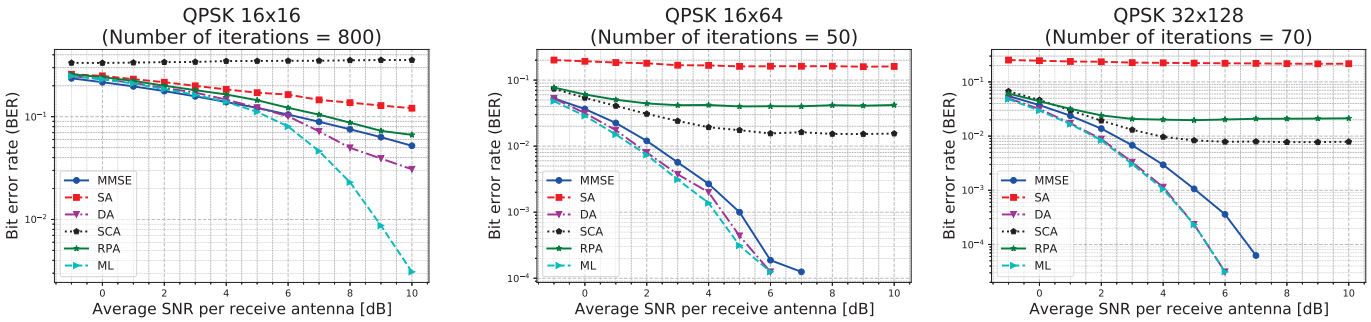

Fig. 4: Digital annealing results in different MIMO QPSK systems.

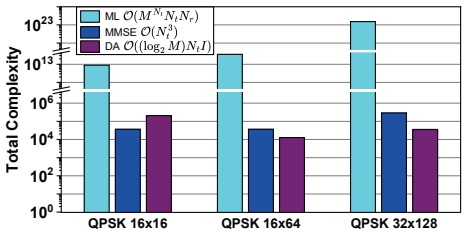

Fig. 5: Comparison of complexity for MU-MIMO detectors. N_t: number of transmit antennas; N_r: number of receive antennas; M: modulation size; I: number of annealing iterations.

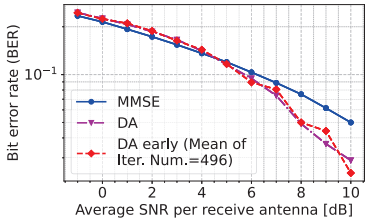

Fig. 6: BER comparison between MMSE, ML, digital annealing with and without early termination in a QPSK 16×16 MIMO system.

termination. The superior accuracy and complexity of digital annealer make it a competitive solution for MIMO detection, particularly massive MIMO detection.

Table I compares our digital annealing-based MU-MIMO detector to the state-of-the-art solutions [2], [3]. Our digital annealer was synthesized in a 28nm CMOS technology. The operating frequency is 200 MHz. The digital annealer is the most flexible, capable of supporting any MIMO configuration beyond the three evaluated in this work. The digital annealer offers competitive BER, silicon area, power, and energy per bit, with throughput that can be further increased by parallelizing using more copies. A full comparison is limited because [3] was only simulated and some results were projected, without reporting the area or power of an implementation of the oscillator Ising machine.

IV. LDPC Decoding Problem

An LDPC code is defined by an $m \times n$ binary parity check matrix $\mathbf{H}$ with a low density of 1's. A $1 \times n$ vector $\mathbf{x}$ is a codeword if and only if $\mathbf{H}\mathbf{x}^{\mathrm{T}} = \mathbf{0} \pmod 2$ [24]. In other

TABLE I: Annealer-based MU-MIMO Detector Comparison

	Digital Annealer (This work)	Quantum Annealing [2]	Oscillator Ising Machine [3]
Implementation	28 nm CMOS synthesized	D-Wave 2000Q	simulated
MIMO Size	16×16, 16×64, 32×128	Max: 40×40	16×64
Error Rate	$< 10^{-3}$ (BER, SNR=5dB)†	$\approx 10^{-3}$(BER, SNR=13dB)‡	$\approx 10^{-3}$ (SER*, SNR=5dB)†
Coefficient J_{ij} Precision	FP32	6 to 7 bits	N/A
Throughput	1.93 Mbps†	0.32 Mbps‡	3.2 Gbps†§
Area	1.5 mm^2	N/A	N/A
Power	475.8 mW	16 kW	N/A
Energy/Bit	0.25 uJ/Bit†	0.05 J/Bit‡	N/A

*SER: Symbol Error Rate. †Decoding QPSK 16x64 MIMO. ‡Decoding QPSK 16x16 MIMO. §Estimated from 10 cycles of oscillation and 1GHz oscillator frequency.

words, each of the m rows of $\mathbf{H}$ represents a parity check, and a codeword must satisfy all m parity checks.

In a communication system, data is encoded in a codeword $\mathbf{x}$ and modulated to $f(\mathbf{x})$ before being transmitted over a channel. For example, using BPSK modulation, $x = 0$ and 1 are modulated to $f(x) = +1$ and -1, respectively for transmission. The received, noise-corrupted word, is represented as $\mathbf{y}$. To decode the transmitted codeword, the ML decoder selects the codeword $\mathbf{x}^*$ that is most likely to have been sent given the received word $\mathbf{y}$. The ML decoding becomes cost prohibitive as the number of possible codewords increases exponentially with the length of the codeword. Instead of ML, approximate belief propagation (BP) decoding is commonly used [25].

A. QUBO Model Formulation

ML decoding involves enumerating *every possible codeword* $\mathbf{x}$ and calculating the Euclidean distance between the received word $\mathbf{y}$ and the transmitted modulated word $f(\mathbf{x})$. The codeword with the smallest distance is chosen as the most likely transmitted codeword, which can be expressed as:

$$\mathbf{x}^* = \arg\min_{\mathbf{x}\in\mathbf{X}} ||\mathbf{y} - f(\mathbf{x})||^2, \tag{9}$$

where $\mathbf{X}$ represents the set of possible codewords.

The problem can be formulated using a QUBO model [5], where the cost function is the Euclidean distance defined in Eq. (9). A binary-valued vector $\mathbf{s}$ is introduced to represent the codeword $\mathbf{x}$. However, unlike in MIMO decoding, where

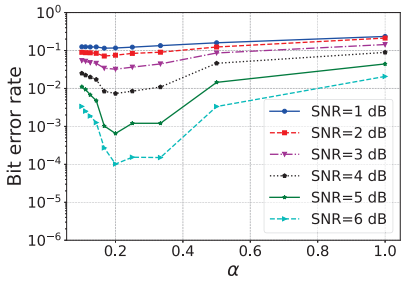

Fig. 7: Tuning α in a range of SNRs for a (128, 64) code.

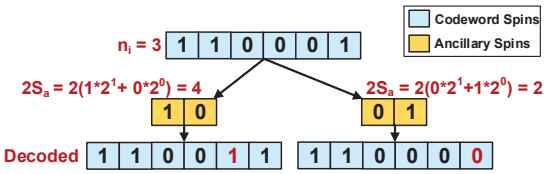

Fig. 8: Error correction being affected by ancillary variables.

each variable in $\mathbf{s}$ can freely take the values of either 0 or 1, LDPC decoding requires the vector $\mathbf{s}$ to be a valid codeword that satisfies $\mathbf{Hs}^{\mathrm{T}} = \mathbf{0} \pmod 2$.

In prior work, this constraint is represented by a satisfier cost function F_{sat}, and is combined with the probability-based distance cost function F_{dist} to form the overall cost function [5]. With these, the QUBO model for the ML decoding can be expressed as:

$$\mathbf{s}^* = \arg\min_{\mathbf{s}} \left(W_1 \sum_{i=1}^{m} F_{sat}(c_i) + W_2 F_{dist}(\mathbf{s}) \right), \qquad (10)$$

where W_1 and W_2 represent the weights for the satisfier function and the distance function, respectively. Here, c_i refers to the i-th parity check. The satisfier function, F_{sat}, is computed for each parity check c_i. If the parity check c_i is satisfied, $F_{sat}(c_i) = 0$.

We define the ratio $\alpha = W_1/W_2$. Tuning α adjusts the relative importance of the valid codeword and shortest distance metrics. To optimize the performance of a given code under a certain SNR, α can be fine-tuned as illustrated in Fig. 7.

B. Ancillary Variable Usage

In previous works on the QUBO formulation of the LDPC decoding problem [4], [5], ancillary variables are required to represent the satisfier cost function. For each parity check c_i, the satisfier function can be defined as follows:

$$F_{sat}(c_i) = (n_i - 2S_a(n_i))^2, \quad n_i = \sum_{\forall j \in c_i} s_j. \qquad (11)$$

Here, $j \in c_i$ denotes the bit index of variable s_j ($s_j \in \{0, 1\}$) involved in the parity check c_i. n_i denotes the count of 1 variables ($s_j = 1$) involved in the parity check c_i. S_a represents the value of the ancillary variables. For instance, if $n_i = 2$, two ancillary variables are set to [0, 1], representing the value $S_a = 1$. Consequently, $F_{sat}(c_i) = 0$, indicating that the even parity is satisfied. On the other hand, if $n_i = 3$, two ancillary variables can be set to either [0, 1] or [1, 0] to represent $S_a = 1$ or 2, respectively, as shown in Fig. 8. Consequently, $F_{sat}(c_i) = 1$, indicating that the even parity is not satisfied.

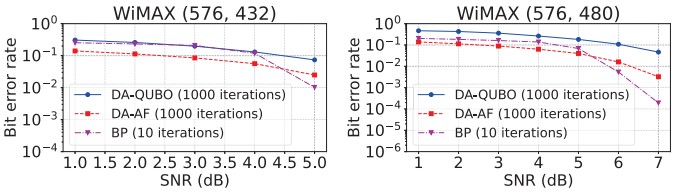

Fig. 9: BER Comparison between BP and digital annealing using both QUBO and AF formulations.

The ancillary variables play a crucial role in maintaining the decoding dynamics. They are selected based on n_i and are assigned to represent the *closest even number*. Nevertheless, this approach has notable drawbacks. In the example shown in Fig. 8, if $n_i = 3$, the generated ancillary variables can represent 2 or 4, excluding 0, 6, or 8, etc. Consequently, the range of decoding possibilities is limited, allowing for only 2 or 4 as the number of 1 variables in c_i. Even the choice between 2 or 4 is arbitrary as they are equidistant from 3. This approach inherently restricts the error correction capability and the feasible movements that can be made.

C. Ancillary-Free (AF) Formulation

The QUBO formulations of the LDPC decoding problem face a challenge due to the need for ancillary variables. These ancillary variables do not hold any physical significance and their inclusion introduces complexity and reduces accuracy.

Inspired by the gradient descent bit-flipping decoding of LDPC codes [26], we consider an alternative, ancillary-free (AF) cost function and convert the ML decoding problem to the following (for BPSK modulation $f(\mathbf{x})$):

$$\mathbf{s}^* = \arg\min_{\mathbf{s}} - \left(W_1 \sum_{i=1}^{m} \prod_{\forall j \in c_i} s_j + W_2 \sum_{j=1}^{n} s_j y_j \right). \qquad (12)$$

Here, $s_j \in \{-1, +1\}$ now represents the variable corresponding to bit j in the transmitted and modulated code, and y_j denotes the received value for bit j. For parity check c_i, $\prod_{\forall j \in c_i} s_j = 1$ if the even parity check is satisfied, or -1 if not. Therefore, the first term in the cost function, $\sum_{i=1}^{m} \prod_{\forall j \in c_i} s_j$, provides a measure of satisfaction of the parity checks, with a larger value indicating higher satisfaction. The second term in the cost function, $\sum_{j=1}^{n} s_j y_j$, computes the inner product between the codeword $\mathbf{s}$ and the received word $\mathbf{y}$. This term serves as a distance metric, where a larger value signifies a closer match between the codeword and the received word. By combining both terms, the decoding aims to find the codeword that satisfies all the even parities and is closest to the received word. Similarly, the ratio $\alpha = W_1/W_2$ can be adjusted to optimize the decoding performance.

It is worth noting that this AF cost function does not work for a quantum annealer because it is not in Ising or QUBO form. However, it works for a digital annealer, which is a key advantage over a quantum annealer.

D. Digital Annealing Results

We present a comparison of the performance between a digital annealer using the DA policy and the AF formulation

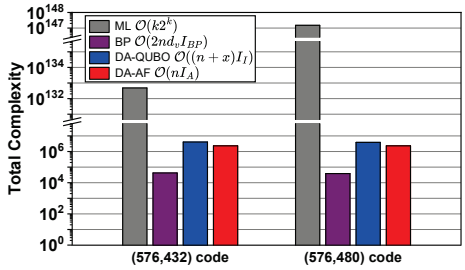

Fig. 10: Comparison of complexity for LDPC codes. n, k: number of bits and number of message bits in the codeword; d_v: average bit degree, $d_v = 3.67$ for the (576, 432) code and $d_v = 3.33$ for the (576, 480) code, and nd_v represents the total number of 1's in the parity check matrix; x: number of ancillary spins for DA-QUBO, $I_{BP} = 10$, $I_Q = I_A = 1000$: number of iterations for BP decoding, DA-QUBO, and DA-AF.

(denoted DA-AF) and a digital annealer using the DA policy and the QUBO formulation (denoted DA-QUBO) in Fig. 9 for two practical LDPC codes used in the WiMax standard [27]. The codes considered are a (576, 432) code and a (576, 480) code, both having 576 bits in a codeword, but with 432 and 480 message bits respectively. It is worth noting that these codes are longer and more practically relevant than the short codes that have been attempted previously for both quantum [5] and digital annealers [4]. Additionally, conventional BP decoding results are included for reference.

For both codes, DA-AF outperforms DA-QUBO, indicating the limitations of the previous QUBO formulation that relies on the use of ancillary spins. In fact, in certain low SNR regions, DA-AF can outperform BP. By tuning α based on the SNR, further improvements in BER can be achieved.

Fig. 10 provides a comparison of the computational complexity, measured in terms of the number of operations required, for different decoding methods applied to the two WiMax LDPC codes. The complexity of BP decoding is directly influenced by the number of 1's in the parity check matrix [25]. Since LDPC codes typically have a low density of 1's in their parity check matrix, the complexity of BP decoding tends to be competitive. However, it is important to note that BP decoding can face challenges in terms of wiring complexity and low memory utilization [25]. These challenges arise primarily due to the irregular and low-density structure of 1's in the parity check matrix of LDPC codes. These complexities and inefficiencies are not considered in this complexity comparison. On the other hand, ML decoding involves conducting trials on 2^k codewords, where k is the number of message bits. For practical LDPC codes with k in the range of hundreds or thousands, ML decoding becomes infeasible due to its impractical computational complexity.

The complexity of DA-QUBO is determined by both the number of bits in the codeword n, and the number of ancillary bits x. For the (576, 432) code, $x = 432$, and for the (576, 480) code, $x = 384$. In contrast, the complexity of DA-AF depends solely on n, so it is more efficient than DA-QUBO. Additionally, while DA-AF exhibits higher complexity than BP, it has the advantage of easy parallelization and full utilization, leading to enhanced efficiency and performance compared to a parallel BP architecture. To further optimize the computational complexity of DA-AF, an early termination method, as previously mentioned, can be employed. This makes DA-AF a promising alternative to BP for LDPC decoding.

TABLE II: Annealer-based LDPC Decoder Comparison

	This Work	[4]	[5]
Cost Function	Ancillary-Free	QUBO w/ ancillary	QUBO w/ ancillary
Implementation	28 nm CMOS synthesized	Fujitsu Digital Annealer	D-Wave 2000Q
Code Length	576	32, 128, 256	420
Throughput	0.58 Mbps	1.2 Kbps	10 Mbps
Area	1.11 mm^2	N/A	N/A
Power	266.4 mW	N/A	16 kW
Energy/Bit	0.46 uJ/Bit	N/A	1.6 mJ/Bit

Table II compares annealing-based LDPC decoder implementations. Our digital annealing-based LDPC decoder is synthesized in a 28nm CMOS technology with an operating frequency of 200 MHz. The digital annealer was evaluated on a 576-bit LDPC code, which is longer and more complex than those studied for other annealers [4], [5]. The digital annealer can support any block length beyond the one shown. It offers competitive silicon area, power, and energy per bit, with throughput that can be further increased by parallelizing using more copies. Compared to another digital annealer [4], our compact design achieves more than two orders of magnitude higher throughput. Compared to the quantum annealer [5], our digital annealer consumes three orders of magnitude less energy per bit, making it suitable for edge platforms.

V. Conclusions

We analyze the effectiveness of digital annealing in solving complex CO problems including MIMO detection and LDPC decoding. Our findings indicate that for MIMO detection, digital annealing with the DA policy surpasses the performance of conventional MMSE detection and closely approaches ML detection. Additionally, the digital annealer exhibits comparable or even lower complexity in comparison to MMSE detection. For LDPC decoding, we identify a drawback in the previous QUBO formulation that relies on ancillary variables, resulting in decreased accuracy. We propose a new AF formulation for digital annealing that eliminates the need for ancillary variables. This AF formulation not only enhances the decoding accuracy but also reduces the complexity, making it a practical alternative to conventional BP decoding.

Acknowledgment

The research is supported in part by the Defense Advanced Research Projects Agency (DARPA) under the Quantum-Inspired Classical Computing (QuICC) Program, and the Center for Ubiquitous Connectivity (CUbiC) sponsored by the Semiconductor Research Corporation (SRC) and DARPA under the JUMP 2.0 program.

REFERENCES

[1] B. H. Korte, J. Vygen, B. Korte, and J. Vygen, *Combinatorial optimization*. Springer, 2011, vol. 1.

[2] M. Kim, D. Venturelli, and K. Jamieson, "Leveraging quantum annealing for large mimo processing in centralized radio access networks," in *Proceedings of the ACM special interest group on data communication*, 2019, pp. 241–255.

[3] J. Roychowdhury, J. Wabnig, and K. P. Srinath, "Performance of oscillator ising machines on realistic mu-mimo decoding problems," 2021.

[4] M. Tawada, S. Tanaka, and N. Togawa, "A new ldpc code decoding method: Expanding the scope of ising machines," in *2020 IEEE International Conference on Consumer Electronics (ICCE)*, 2020, pp. 1–6.

[5] S. Kasi and K. Jamieson, "Towards quantum belief propagation for ldpc decoding in wireless networks," in *Proceedings of the 26th Annual International Conference on Mobile Computing and Networking*, 2020, pp. 1–14.

[6] B. A. Cipra, "An introduction to the ising model," *The American Mathematical Monthly*, vol. 94, no. 10, pp. 937–959, 1987.

[7] F. Glover, G. Kochenberger, and Y. Du, "A tutorial on formulating and using qubo models," *arXiv preprint arXiv:1811.11538*, 2018.

[8] P. Hauke, H. G. Katzgraber, W. Lechner, H. Nishimori, and W. D. Oliver, "Perspectives of quantum annealing: Methods and implementations," *Reports on Progress in Physics*, vol. 83, no. 5, p. 054401, 2020.

[9] M. Sao, H. Watanabe, Y. Musha, and A. Utsunomiya, "Application of digital annealer for faster combinatorial optimization," *Fujitsu Scientific and Technical Journal*, vol. 55, no. 2, pp. 45–51, 2019.

[10] M. W. Johnson, M. H. Amin, S. Gildert, T. Lanting, F. Hamze, N. Dickson, R. Harris, A. J. Berkley, J. Johansson, P. Bunyk *et al.*, "Quantum annealing with manufactured spins," *Nature*, vol. 473, no. 7346, pp. 194–198, 2011.

[11] F. F. Silva, P. M. Carvalho, and L. A. Ferreira, "A quantum computing approach for minimum loss problems in electrical distribution networks," *Scientific Reports*, vol. 13, no. 1, p. 10777, 2023.

[12] D-Wave. (2018) D-wave solver document. [Online]. Available: https://docs.dwavesys.com/docs/latest/ (accessed: Sep. 12, 2024)

[13] K. Yamamoto, K. Kawamura, K. Ando, N. Mertig, T. Takemoto, M. Yamaoka, H. Teramoto, A. Sakai, S. Takamaeda-Yamazaki, and M. Motomura, "Statica: A 512-spin 0.25m-weight annealing processor with an all-spin-updates-at-once architecture for combinatorial optimization with complete spin–spin interactions," *IEEE Journal of Solid-State Circuits*, vol. 56, no. 1, pp. 165–178, 2021.

[14] K. Kawamura, J. Yu, D. Okonogi, S. Jimbo, G. Inoue, A. Hyodo, López García-Arias, K. Ando, B. H. Fukushima-Kimura, R. Yasudo, T. Van Chu, and M. Motomura, "Amorphica: 4-replica 512 fully connected spin 336mhz metamorphic annealer with programmable optimization strategy and compressed-spin-transfer multi-chip extension," in *2023 IEEE International Solid-State Circuits Conference (ISSCC)*, 2023, pp. 42–44.

[15] T. Takemoto, M. Hayashi, C. Yoshimura, and M. Yamaoka, "A 2 × 30k-spin multi-chip scalable cmos annealing processor based on a processing-in-memory approach for solving large-scale combinatorial optimization problems," *IEEE Journal of Solid-State Circuits*, vol. 55, no. 1, pp. 145–156, 2020.

[16] Y. Su, J. Mu, H. Kim, and B. Kim, "A scalable cmos ising computer featuring sparse and reconfigurable spin interconnects for solving combinatorial optimization problems," *IEEE Journal of Solid-State Circuits*, vol. 57, no. 3, pp. 858–868, 2022.

[17] W. Moy, I. Ahmed, P.-w. Chiu, J. Moy, S. S. Sapatnekar, and C. H. Kim, "A 1,968-node coupled ring oscillator circuit for combinatorial optimization problem solving," *Nature Electronics*, vol. 5, no. 5, pp. 310–317, 2022.

[18] H. Lo, W. Moy, H. Yu, S. Sapatnekar, and C. H. Kim, "An ising solver chip based on coupled ring oscillators with a 48-node all-to-all connected array architecture," *Nature Electronics*, vol. 6, no. 10, pp. 771–778, 2023.

[19] T. Wang and J. Roychowdhury, "Oim: Oscillator-based ising machines for solving combinatorial optimisation problems," in *Unconventional Computation and Natural Computation: 18th International Conference, UCNC 2019, Tokyo, Japan, June 3–7, 2019, Proceedings 18*. Springer, 2019, pp. 232–256.

[20] M. Aramon, G. Rosenberg, E. Valiante, T. Miyazawa, H. Tamura, and H. G. Katzgraber, "Physics-inspired optimization for quadratic unconstrained problems using a digital annealer," *Frontiers in Physics*, vol. 7, 2019. [Online]. Available: https://www.frontiersin.org/journals/physics/articles/10.3389/fphy.2019.00048

[21] C. Studer, S. Fateh, and D. Seethaler, "Asic implementation of soft-input soft-output mimo detection using mmse parallel interference cancellation," *IEEE Journal of Solid-State Circuits*, vol. 46, no. 7, pp. 1754–1765, 2011.

[22] Z. I. Tabi, Marosits, Z. Kallus, P. Vaderna, I. Gódor, and Z. Zimborás, "Evaluation of quantum annealer performance via the massive mimo problem," *IEEE Access*, vol. 9, pp. 131 658–131 671, 2021.

[23] J. Hoydis, C. Hoek, T. Wild, and S. ten Brink, "Channel measurements for large antenna arrays," in *2012 International Symposium on Wireless Communication Systems (ISWCS)*, 2012, pp. 811–815.

[24] R. Gallager, "Low-density parity-check codes," *IRE Transactions on Information Theory*, vol. 8, no. 1, pp. 21–28, 1962.

[25] A. J. Blanksby and C. J. Howland, "A 690-mw 1-gb/s 1024-b, rate-1/2 low-density parity-check code decoder," *IEEE Journal of solid-state circuits*, vol. 37, no. 3, pp. 404–412, 2002.

[26] T. Wadayama, K. Nakamura, M. Yagita, Y. Funahashi, S. Usami, and I. Takumi, "Gradient descent bit flipping algorithms for decoding ldpc codes," in *2008 International Symposium on Information Theory and Its Applications*, 2008, pp. 1–6.

[27] "Ieee p802.16e-2005," *Part 16: Air Interface for Fixed and Mobile Broadband Wireless Access Systems Amendment for Physical and Medium Access Control Layers for Combined Fixed and Mobile Operation in Licensed Bands*, 2005.