Towards Quantum Annealing for Multi-user NOMA-based Networks

Eldar Gabdulsattarov, #Khaled Rabie, °Xingwang Li, and Galymzhan Nauryzbayev$^{\boxtimes}$

School of Engineering and Digital Sciences, Nazarbayev University, Nur-Sultan, Z05H0K4, Kazakhstan
#Department of Engineering, Manchester Metropolitan University, Manchester, M15 6BH, UK
°School of Physics and Electronic Information Engineering, Henan Polytechnic University, Jiaozuo 454000, China
Email: {eldar.gabdulsattarov, galymzhan.nauryzbayev}@nu.edu.kz, #k.rabie@mmu.ac.uk, °lixingwang@hpu.edu.cn

Abstract—**Quantum Annealing (QA) uses quantum fluctuations to search for a global minimum of an optimization-type problem faster than classical computers. To meet the demand for future internet traffic and mitigate the spectrum scarcity, this work presents the QA-aided maximum likelihood (ML) decoder for multi-user non-orthogonal multiple access (NOMA) networks as an alternative to the successive interference cancellation (SIC) method. The practical system parameters such as channel randomness and possible transmit power levels are taken into account for all individual signals of all involved users. The brute force (BF) and SIC signal detection methods are taken as benchmarks in the analysis. The QA-assisted ML decoder results in the same BER performance as the BF method outperforming the SIC technique, but the execution of QA takes more time than BF and SIC. The parallelization technique can be a potential aid to fasten the execution process. This will pave the way to fully realize the potential of QA decoders in NOMA systems.**

Index Terms—**Quantum annealing, non-orthogonal multiple access (NOMA), maximum likelihood (ML).**

I. INTRODUCTION

There has been a drastic rise in the traffic of advanced multimedia applications such as real-time conferencing, online video gaming and virtual reality in recent years. Moreover, according to the Cisco Annual Report, the total number of internet users will reach 5.3 billion by 2023 [1]. To meet this ever-growing demand on traffic, it is crucial to consider some optimization improvements in the quality of data transmission in terms of the bandwidth, error rate, and delay. One of the actions that can be taken into account is to implement Quantum Annealing (QA) technologies to solve NP-hard optimization problems in wireless communication.

The authors in [2]–[5] suggested the centralized radio access network (C-RAN) as a promising and cost-effective design architecture for future wireless networks, where multiple base stations (BSs) are interconnected with a centralized data center. The C-RAN's data centers support most of the BSs' signal processing operations. It is envisioned that the QA computers placed at the C-RAN centers can facilitate optimization problems, paving the way for full integration of the presented methodologies in real-world communication systems.

One of the computational complex problems in wireless networks is to obtain an optimal scheduling in multi-user (MU) systems, which can help one effectively employ network resources. The authors in [6] introduced the QA approach to find an optimal solution for the time-division multiple access

(TDMA) transmission scheduling problem in a wireless sensor network using a tree topology. The results revealed that QA outperforms the other scheduling solutions in terms of the runtime and proximity to the optimal solution.

An essential part of cellular base-band processing that can enhance the quality of communication is the error correction code (ECC), which mitigates the errors in received bit-streams caused by the noise in a wireless environment. However, the error correction code protocols with exceptional performance require significant computational resources for the decoding process at the receiver. For this reason, the implementation of QA computers was proposed in [2] for efficient decoding of low-density parity-check codes in uplink systems. The method achieved a bit error rate (BER) of 10^{-8} at the signal-to-noise ratio (SNR) of 2.5-3.5 dB lower than the standard algorithm for decoding ECCs on a field-programmable gate array (FPGA).

Another key aspect in the performance of wireless communication systems is a signal detection. To achieve the full potential of multi-user systems, BSs have to efficiently decode superimposed data streams received from different users. Maximum likelihood (ML) usually ends up with an optimum solution for decoding the received signals, but it is an NP-hard problem. The computational cost of ML rises exponentially with the number of users and data rate [3]. For large-scale networks, the ML decoding process might last expiration of acknowledgement timeout. The authors in [3]–[5] employed a QA approach to speed up this process. Particularly, [3] presented the QuAMax approach for ML decoding in MU MIMO systems using QA. The performance of the solution was tested on the quantum annealer with 2048 qubits considering different modulation schemes and multiple users. The method achieves the same BER as ZF does, but with significantly lower computation time in the case of binary phase-shift keying (BPSK) and quadrature phase-shift keying (QPSK) modulation types. Moreover, the QA approach outperforms the conventional decoders in terms of the computable size of ML problems. For instance, the work in [3] was extended by [4], where an experiment on the recently released QA hardware with 5640 qubits was performed. The results revealed that the new computer is able to provide better scaling for embedding the ML MIMO decoding problem. In addition, the study showed that today's QA machines can achieve identical BER as the ML sphere decoder for 4×4 MIMO using 16-QAM (quadrature amplitude modulation).

The current communication technologies support multiple users using orthogonal multiple access (OMA) techniques such as TDMA, frequency- and code-division multiple access (i.e., FDMA and CDMA) schemes. However, all these techniques have certain shortcomings such as precise time synchronization, limited frequency carriers and code length. These drawbacks get more perceptible for a large number of users [7]. Non-orthogonal multiple access (NOMA) is a technology that allows all users in a cell operate in the common time and frequency, that, in turn, improve the spectral efficiency (SE). The successive interference cancellation (SIC) is considered as a promising decoding technique for NOMA that decodes symbol-by-symbol until the desired one is detected. However, the high computational cost, latency and error propagation of SIC make it impractical to implement NOMA in the current wireless communication systems [7], [8]. The traditional ML decoder supports high-quality decoding but suffers from high computation time. To address this issue, a NOMA-ML decoder based on QA was proposed in [5] for a two-user system model, with a BPSK modulation and constant channel gains.

In contrast to the previous studies, in this work, we consider the NOMA-based uplink network comprising multiple users under practical scenarios and different modulation types. The NOMA-ML decoding problem was firstly reformulated into a QUBO model for multiple modulation schemes including BPSK, QPSK, $16-$QAM and $64-$QAM. Moreover, the generalized QUBO expressions of the proposed system model are derived for the multi-user network under the previously defined modulation types. We have investigated how the number of users in a cell affects the signal decoding performance. Furthermore, the QA-assisted ML decoder was compared with SIC and Brute Force (BF) techniques in terms of BER performance and simulation time. Finally, the influence of the parallelization method on the execution time of the QA machine is examined.

II. SYSTEM MODEL

The system model considered here represents an uplink NOMA-based network scenario (depicted in Fig. 1) comprising BS and N end-users, denoted by U_k, with $k \in \mathcal{A} = \{1, 2, \ldots, N\}$. It is assumed that all nodes are deployed with a single antenna. The channel between the user of interest and BS is given by $\tilde{h}_k$ and assumed to be independent and identically distributed random variables (RVs) following Rayleigh fading.

A. Non-orthogonal Multiple Access

Due to the considered uplink mode of communication, all end-users simultaneously transmit their individual messages to BS while ensuring the NOMA principle to be true through the path loss associated with their corresponding distances[1]. Hence, the received signal at BS can be written as

$$y = \sum_{k=1}^{N} s_k h_k + n, \qquad (1)$$

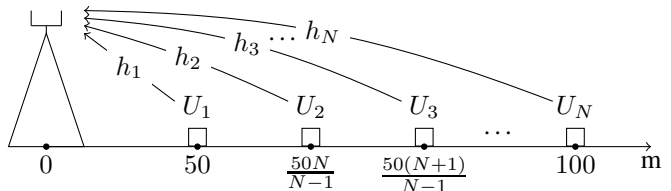[1]Note that the similar assumption can be made using the principle of messages' priorities.

Fig. 1. An illustration of uplink NOMA with N users.

where s_k is the transmitted message of user k, with the transmit power defined as $\mathbb{E}\{s_k s_k^*\} = P_k$. For the sake of simplicity, we assume that all end-users transmit with the same power levels, *i.e.*, $P = P_1 = P_2 = \ldots = P_N$. h_k is modelled as $h_k = \tilde{h}_k / \sqrt{d_k^\tau}$, where d_k and τ indicate the corresponding distance and path-loss exponent. n is an additive white Gaussian noise (AWGN) term, with zero mean and variance σ_n^2.

B. Maximum Likelihood Detection

The receiver implements the ML detection as a decoding method, which output, given by an optimal symbol vector $\hat{\mathbf{s}}_{ML}$, is formulated as

$$\hat{\mathbf{s}}_{ML} = \underset{\hat{s}_1, \hat{s}_2, \ldots \hat{s}_N}{\arg \min} |y - \mathbf{s}\mathbf{h}^{\mathrm{T}}|^2, \qquad (2)$$

where y is the received signal, $\mathbf{s}$ and $\mathbf{h}$ are the $1 \times N$ vectors representing the transmit symbols and channel gains of end-users, respectively.

C. Quantum Annealing

Quantum computing implements the concepts of quantum mechanics to solve high-computational problems faster than classical computers. There are several types of quantum computers. If the quantum circuit model controls the qubits using quantum logic gates to solve a wide range of problems, QA is an effective tool to solve optimization problems with given objective functions. Specifically, the quantum processing unit (QPU) uses quantum fluctuations to search for the global minimum solution. However, modern quantum computers are not that perfect yet and QA might end up with a local minimum solution due to the analogue noises in the device [9]. To mitigate the internal errors in QPU, the annealing parameters could be modified based on the specifics of the problem. To minimize these errors, QA can be simulated many times, and the optimal solution will be the answer with the lowest energy level and most frequent occurrences. The annealing parameter that controls the number of samples per simulation is the *number of reads* (R), it was decided to set it to 1000. Another annealing parameter that could impact the performance of the QA machine is the *single-sample annealing time* (T_a), which was set to $20 \ \mu s$ without any pausing time [10].

D. Quadratic Unconstrained Binary Optimization

To solve an optimization problem by a means of QA, the objective function should be reformulated in a form of the quadratic unconstrained binary optimization (QUBO) model or the Ising model. These formulations can be used interchangeably, and it is easy to switch from one to another. At the end
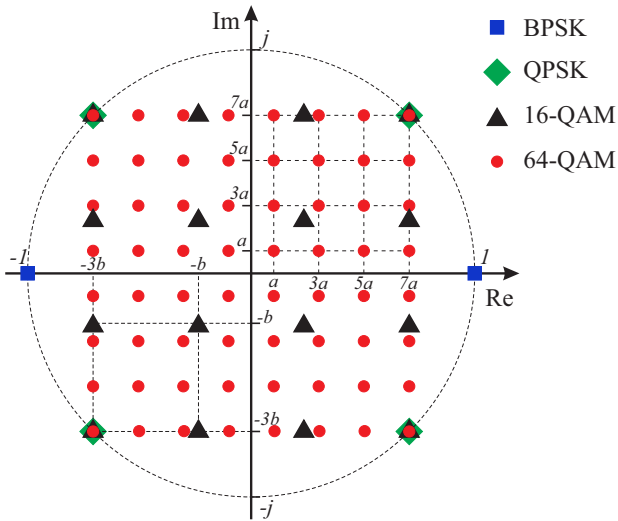
Fig. 2. Constellations of different modulation types.

Fig. 3. BER vs. the number of users per cell under BPSK.

of annealing process, the best solution is given by the qubit sequence with a minimum energy state. The model is given by an upper-diagonal matrix Q with a size of $M \times M$

$$\hat{q}_1, \hat{q}_2, \ldots, \hat{q}_M = \underset{q_1, q_2, \ldots q_M}{\arg\min} \sum_{i \leq j}^{M} Q_{ij} q_i q_j, \qquad (3)$$

where M denotes the number of qubits, and each variable q_i takes binary numbers. By applying the property $q_i^2 = q_i q_i = q_i$, and substituting the symbols in (2) by the corresponding binary equivalent form, it becomes feasible to formulate the NOMA ML problem in terms of the QUBO form. The corresponding diagonal Q_{ii} and non-diagonal Q_{ij} terms of the QUBO model are represented by linear and quadratic coefficients, accordingly.

To implement the obtained QUBO model on QPU, the matrix coefficients have to be mapped into physical qubits of a QA chip. The linear coefficients define the qubit biases, while quadratic coefficients represent the coupling strengths [11]. However, QPU's physical qubits are not fully connected, in the Chimera graph each qubit has six neighbours, and, in Pegasus topology, each qubit is connected with 15 neighbour qubits [12]. Therefore, sometimes it is not possible to directly apply the QUBO model onto real QA machine.

III. FORMULATION OF NOMA ML PROBLEM IN A QUBO MODEL

The ML expression in (2) should be described by a QUBO model to solve it with QA. Taking into account (1) and the complex numbers due to the broadcast transmission, we split the received signal y and channel coefficients into real and imaginary parts as follows: $y = y_R + jy_I$ and $h_k = h_{k,R} + jh_{k,I}$. Each symbol, denoted by s_k, can be transformed into the qubit form. The methodology for representing symbols in terms of the binary variables depends on the modulation type. The considered modulations are presented in Fig. 2.

1) BPSK: The BPSK symbols of a form $s_k \in \{\pm 1\}$, can be reformulated into the QUBO form by $s_k = 2q_k - 1$. In this case, M in (3) takes a value of the number of users, N.
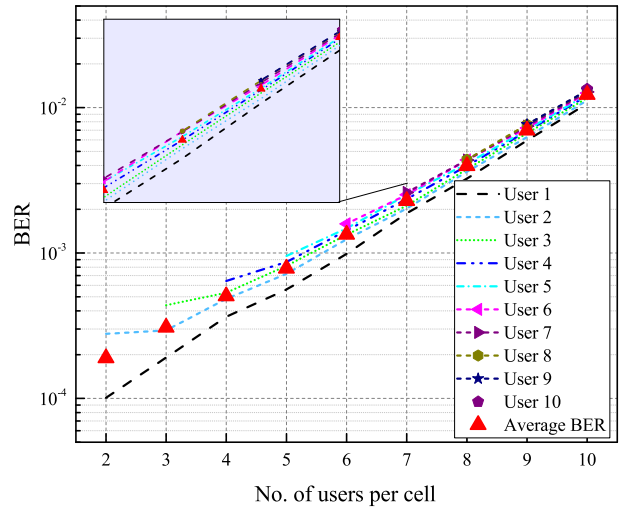
2) QPSK: As shown in Fig. 2, the QPSK symbols, $s_k \in \{\frac{\pm 1 \pm j}{\sqrt{2}}\}$, reside on a unit circle with $90°$ phase difference between two consecutive symbols. They can be represented by two qubits; therefore, the term M in (3) becomes the double of a number of users, $2N$. Finally, the QPSK symbols can be expressed as $s_k = [(2q_{2k-1} - 1) + j(2q_{2k} - 1)]/\sqrt{2}$.

3) 16-QAM: The 16-QAM modulation encodes four information bits into a complex symbol and correspondingly includes 16 constellation points. The symbols $s_k \in \{\pm b \pm jb, \pm b \pm j3b, \pm 3b \pm jb, \pm 3b \pm j3b\}$, where $b = 1/(3\sqrt{2})$, require four qubits to be adapted for the QUBO model, i.e., $s_k = [(4q_{4k-3} + 2q_{4k-2} - 3) + j(4q_{4k-1} + 2q_{4k} - 3)]/3\sqrt{2}$. M in (3) becomes the quadruple of a number of users, $4N$.

4) 64-QAM: On the other hand, the 64-QAM modulation returns a complex symbol by encoding 16 information bits. As shown in Fig. 2, the constellation points are defined as per Fig. 2 and require six qubits to be adopted for the QUBO model; therefore, the term M in (3) becomes the sixtuple of a number of users, $6N$. These symbols can be re-expressed as $s_k = a(8q_{6k-5} + 4q_{6k-4} + 2q_{6k-3} - 7) + ja(8q_{6k-2} + 4q_{6k-1} + 2q_{6k} - 7)$.

IV. IMPLEMENTATION AND RESULTS DISCUSSION

A. Implementation

The system model considered in the analysis represents a N-end user uplink NOMA system that employs BPSK modulation. As illustrated in Fig. 1, the base station is located at a coordinate 0. We assume that the first and the farthest users are located 50 and 100 m away from the base station, respectively, while the other intermediate nodes are evenly distributed in-between. For practical purposes, we apply the standardized scenarios described in [13] and [14]. The performance of the exploited solvers (i.e., BF, SIC and QA) were examined by varying the transmit power from -40 dBm to 24 dBm. Moreover, the environment of data transmission is considered to be a free space with line-of-sight propagation which relates to the path-loss exponent $\tau = 2$ [15].
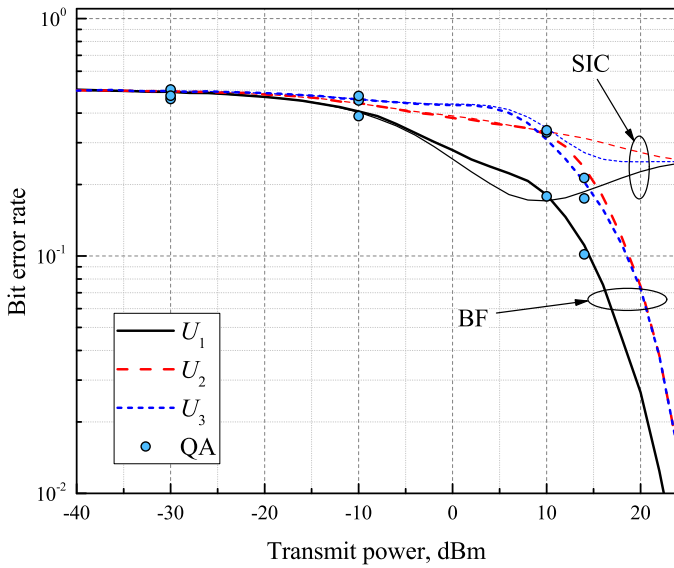
Fig. 4. The BER performance vs. the transmit power for a three-user NOMA scenario, with different decoding techniques.
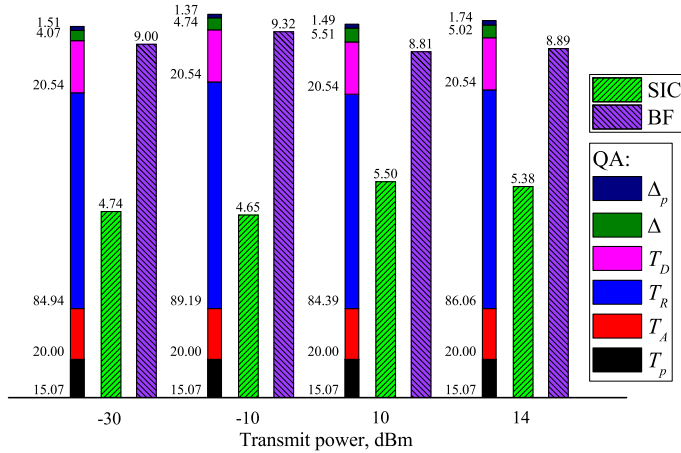

Fig. 5. Comparison between the decoding techniques in terms of simulation time for five (5) samples (in ms).

The QA method was implemented on the proposed system model using D-Wave's Advantage QPU [12]. It is important to mention that the parallelization procedure was used to save the simulation time on the QA computer [3], i.e., 5 instances of the problem were run at the same time on QPU. Since the 3-user BPSK problem requires only 3 logical qubits, 5 instances would occupy 15 logical qubits. In the analysis of simulation time, the following terms are used as follows. *QPU Programming Time* (T_p) is the time taken for programming the couplers and biases of the chip in accordance with the QUBO model. *QPU Sampling Time* (T_s) is the total time for simulation of R samples and consists of the annealing time (T_a), the readout time (T_r), and the delay time (T_d) for every single sample [16]. For the sake of simplicity, we declare the following terms: the *total QPU Annealing Time* ($T_A = R \cdot T_a$), the *total QPU Readout Time* ($T_R = R \cdot T_r$), and the *QPU Delay Time* ($T_D = R \cdot T_d$). T_p, T_s, *QPU Access Overhead*

Table I. Simulation time (in ms) of the decoding techniques.

Technique	for 5 samples	for 1 sample
SIC	5.066	3.958
BF	9.006	4.535
QA		
QPU Service Time	148.116	107.984
T_p	15.069	15.068
T_A	20	20
T_R	86.145	46.68
T_D	20.54	20.54
Δ	4.835	3.86
Δ_p	1.527	1.836

Time (Δ) and *Post-processing Overhead Time* (Δ_p) constitute *QPU Service Time*, which is considered as the total time taken for the QA simulation excluding internet delay.

B. Discussion

In Fig. 3, we examine the influence of the number of users per cell on the BER performance in the NOMA system. Particularly, the transmit power and the AWGN power were set to 10 dBm and −60 dBm, respectively. As expected, the average BER performance degrades with the increase in the number of users per cell. It is seen from the plot that, for the number of users greater than 6, the difference between the BER curves becomes quite similar to each other. Therefore, we apply the 3-user NOMA scenario in the further analysis.

In Figs. 4 and 5, we compare the SIC, BF and QA decoding techniques in terms of the BER performance and simulation time, with the noise power set to $\sigma_n^2 = -30$ dBm. QA was simulated for particular transmit power levels, i.e., $P = \{-30, -10, 10, 14\}$ dBm. The number of QA simulations per power level was taken to ensure a 1% accuracy with respect to the BER performance of BF. Overall, about 380 simulations (with 5 problems at a time) were performed on QPU.

Fig. 4 illustrates the BER performances of each decoding method. The BER curves can be characterized by different behaviour. It can be seen from the U_1's curve, the SIC and BF performances coincide up to −5 dBm, and then SIC starts outperforming BF over a short transmit power range (i.e., from −5 dBm to 10 dBm). However, afterwards, SIC starts experiencing noticeable performance degradation. At the same time, U_2 has identical BER for SIC and BF up to 10 dBm. After this point, there is a slight improvement of the SIC curve, but, in general, its performance is much worse than that of BF. What stands out for U_3 is the equal performance of SIC and BF for transmit power less than 5 dBm, thereafter, the difference between them begins to grow considerably. The performance of BF improves remarkably after 5 dBm, whereas the SIC curve enhances minimally until 15 dBm with the following saturation after this point. In general, the QA, BF and SIC curves follow the same trend till 10 dBm, but after this level, the SIC performance degrades substantially, while the BER result of QA is approximately the same as BF. The reason for the SIC's low performance could be insufficient differences between the end-users' power levels.

Fig. 5 presents the timing data for each technique. As can be seen from the histogram plot, SIC shows the fastest

execution among the presented techniques. As expected, the second fastest technique is BF, since it needs to iterate over all possible combinations. The QA execution time interval consists of several timing sub-intervals. While T_p, T_A and T_D are the same for all cases, T_R shows a slight variation, since the reading time depends on the position of exploited physical qubits on the chip and on the number of used qubits (more time is needed for a larger number of qubits) [16]. The dependence on the number of qubits could be noticed in Table I, i.e., T_R for 1 sample is twice as fast as T_R for 5 samples. Overall, the total time taken for QA execution is about 30 times more than the SIC simulation. However, it is important to mention that QA needs to simulate the same problem R number of times (in our case it is set to 1000). Furthermore, the problem could be addressed by the parallelization method. From Table I, it is clearly noticeable that the execution of 5 instances in parallel is more than 3 times faster than executing the same number of samples in a sequential order. Therefore, one could potentially conclude that the overall QA simulation time can be decreased by simulating multiple problem instances at the same time.

V. Conclusion

To mitigate the problem of spectrum scarcity and present an alternative for SIC, this work aims at evaluating the performance of QA-aided ML detection for NOMA systems. The ML problem was firstly described in terms of the QUBO model to enable the integration of the problem with QPU. For 3-user NOMA under BPSK modulation, the BER performance of QA is approximately the same as the BF method, but QA takes longer time to execute due to the current hardware specifics. The parallelization method seems to be a potential solution that could decrease the execution time of QA. Additionally, the analogous noises in QA could be suppressed with coming of the next-generation QA hardware, and the time taken for error alleviation would be reduced. This will pave the way for computing time-dependent operations on QA including the QA-assisted ML decoding technique for future NOMA systems.

VI. Acknowledgement

Appendix A
QUBO Model Coefficients

In this section, we present the coefficients for the QUBO model for different modulation types[2].

1) BPSK: The corresponding Q_{ij}^{B} values can be found as

$$Q_{ii}^{\mathrm{B}} = P\left[-4h_{i,\mathrm{R}}\left(\sum_{l=1}^{N}h_{l,\mathrm{R}}-h_{i,\mathrm{R}}\right)-4h_{i,\mathrm{I}}\left(\sum_{l=1}^{N}h_{l,\mathrm{I}}-h_{i,\mathrm{I}}\right)\right]$$
$$-\sqrt{P}\left(4y_{\mathrm{R}}h_{i,\mathrm{R}}+4y_{\mathrm{I}}h_{i,\mathrm{I}}\right), \tag{A.1}$$
$$Q_{ij}^{\mathrm{B}} = P\left(8h_{i,\mathrm{R}}h_{j,\mathrm{R}}+8h_{i,\mathrm{I}}h_{j,\mathrm{I}}\right). \tag{A.2}$$

[2]Note that the equations involving the positive integer variables i and j must satisfy the condition $i < j$.

2) QPSK: The Q_{ij}^{Q} values can be found using (A.3), (A.4) and the equations given below

$$Q_{(2i-1),(2i)}^{\mathrm{Q}} = 0, \quad \forall i \geq 1, \tag{A.5}$$
$$Q_{(2i-1),(2j-1)}^{\mathrm{Q}} = Q_{(2i),(2j)}^{\mathrm{Q}} = \frac{P}{2}\left(8h_{i,\mathrm{R}}h_{j,\mathrm{R}}+8h_{i,\mathrm{I}}h_{j,\mathrm{I}}\right), \tag{A.6}$$
$$Q_{(2i-1),(2j)}^{\mathrm{Q}} = \frac{P}{2}\left(8h_{i,\mathrm{I}}h_{j,\mathrm{R}}-8h_{i,\mathrm{R}}h_{j,\mathrm{I}}\right), \tag{A.7}$$
$$Q_{(2i),(2j-1)}^{\mathrm{Q}} = \frac{P}{2}\left(-8h_{i,\mathrm{I}}h_{j,\mathrm{R}}+8h_{i,\mathrm{R}}h_{j,\mathrm{I}}\right). \tag{A.8}$$

3) 16-QAM: The $Q_{ij}^{16\mathrm{Q}}$ values can be found using

$$Q_{(4i-3),(4i-1)}^{16\mathrm{Q}} = Q_{(4i-3),(4i)}^{16\mathrm{Q}}$$
$$= Q_{(4i-2),(4i-1)}^{16\mathrm{Q}} = Q_{(4i-2),(4i)}^{16\mathrm{Q}} = 0, \tag{A.13}$$
$$Q_{(4i-3),(4i-2)}^{16\mathrm{Q}} = Q_{(4i-1),(4i)}^{16\mathrm{Q}} = \frac{8P}{9}\left(|h_{i,\mathrm{R}}|^2+|h_{i,\mathrm{I}}|^2\right), \tag{A.14}$$
$$Q_{(4i-3),(4j-2)}^{16\mathrm{Q}} = Q_{(4i-2),(4j-3)}^{16\mathrm{Q}} = Q_{(4i),(4j-1)}^{16\mathrm{Q}}$$
$$= Q_{(4i-1),(4j)}^{16\mathrm{Q}} = \frac{P}{9}\left(8h_{i,\mathrm{R}}h_{j,\mathrm{R}}+8h_{i,\mathrm{I}}h_{j,\mathrm{I}}\right), \tag{A.15}$$
$$Q_{(4i-2),(4j-2)}^{16\mathrm{Q}} = Q_{(4i),(4j)}^{16\mathrm{Q}} = \frac{P}{9}\left(4h_{i,\mathrm{R}}h_{j,\mathrm{R}}+4h_{i,\mathrm{I}}h_{j,\mathrm{I}}\right), \tag{A.16}$$
$$Q_{(4i-1),(4j-1)}^{16\mathrm{Q}} = Q_{(4i-3),(4j-3)}^{16\mathrm{Q}}$$
$$= \frac{P}{9}\left(16h_{i,\mathrm{R}}h_{j,\mathrm{R}}+16h_{i,\mathrm{I}}h_{j,\mathrm{I}}\right), \tag{A.17}$$
$$Q_{(4i-2),(4j-1)}^{16\mathrm{Q}} = Q_{(4i-3),(4j)}^{16\mathrm{Q}} = \frac{8P}{9}\left(-h_{i,\mathrm{R}}h_{j,\mathrm{I}}+h_{i,\mathrm{I}}h_{j,\mathrm{R}}\right), \tag{A.18}$$
$$Q_{(4i-1),(4j-2)}^{16\mathrm{Q}} = Q_{(4i),(4j-3)}^{16\mathrm{Q}}$$
$$= \frac{P}{9}\left(-8h_{i,\mathrm{I}}h_{j,\mathrm{R}}+8h_{i,\mathrm{R}}h_{j,\mathrm{I}}\right), \tag{A.19}$$
$$Q_{(4i-2),(4j)}^{16\mathrm{Q}} = \frac{P}{9}\left(-4h_{i,\mathrm{R}}h_{j,\mathrm{I}}+4h_{i,\mathrm{I}}h_{j,\mathrm{R}}\right), \tag{A.20}$$
$$Q_{(4i),(4j-2)}^{16\mathrm{Q}} = \frac{P}{9}\left(-4h_{i,\mathrm{I}}h_{j,\mathrm{R}}+4h_{i,\mathrm{R}}h_{j,\mathrm{I}}\right), \tag{A.21}$$
$$Q_{(4i-1),(4j-3)}^{16\mathrm{Q}} = \frac{P}{9}\left(-16h_{i,\mathrm{I}}h_{j,\mathrm{R}}+16h_{i,\mathrm{R}}h_{j,\mathrm{I}}\right), \tag{A.22}$$
$$Q_{(4i-3),(4j-1)}^{16\mathrm{Q}} = \frac{P}{9}\left(-16h_{i,\mathrm{R}}h_{j,\mathrm{I}}+16h_{i,\mathrm{I}}h_{j,\mathrm{R}}\right). \tag{A.23}$$

4) 64-QAM: The derivations of $Q_{ij}^{64\mathrm{Q}}$ values are omitted due to the submission page limit and will be included in the extended version, if accepted.

References

[1] "Cisco Annual Internet Report - Cisco Annual Internet Report (2018–2023) White Paper," Cisco, March 2020, [Online]. Available: https://www.cisco.com/c/en/us/solutions/collateral/executive-perspectives/annual-internet-report/white-paper-c11-741490.html.

[2] S. Kasi and K. Jamieson, "Towards quantum belief propagation for LDPC decoding in wireless networks," *26th Annu. Int. Conf. Mobile Comput. Netw.,* London, UK, pp. 1-14, Sep. 2020.

$$Q^{\mathrm{Q}}_{(2i-1),(2i-1)} = \frac{P}{2}\left[-4h_{i,\mathrm{R}}\left(\left(\sum_{l=1}^{N}h_{l,\mathrm{R}}-h_{i,\mathrm{R}}\right)-\left(\sum_{l=1}^{N}h_{l,\mathrm{I}}-h_{i,\mathrm{I}}\right)\right)-4h_{i,\mathrm{I}}\left(\left(\sum_{l=1}^{N}h_{l,\mathrm{R}}-h_{i,\mathrm{R}}\right)+\left(\sum_{l=1}^{N}h_{l,\mathrm{I}}-h_{i,\mathrm{I}}\right)\right)\right]$$
$$-\sqrt{2P}\left(2y_{\mathrm{R}}h_{i,\mathrm{R}}+2y_{\mathrm{I}}h_{i,\mathrm{I}}\right) \tag{A.3}$$

$$Q^{\mathrm{Q}}_{(2i),(2i)} = \frac{P}{2}\left[-4h_{i,\mathrm{R}}\left(\left(\sum_{l=1}^{N}h_{l,\mathrm{R}}-h_{i,\mathrm{R}}\right)+\left(\sum_{l=1}^{N}h_{l,\mathrm{I}}-h_{i,\mathrm{I}}\right)\right)-4h_{i,\mathrm{I}}\left(-\left(\sum_{l=1}^{N}h_{l,\mathrm{R}}-h_{i,\mathrm{R}}\right)+\left(\sum_{l=1}^{N}h_{l,\mathrm{I}}-h_{i,\mathrm{I}}\right)\right)\right]$$
$$-\sqrt{2P}\left(-2y_{\mathrm{R}}h_{i,\mathrm{I}}+2y_{\mathrm{I}}h_{i,\mathrm{R}}\right) \tag{A.4}$$

$$Q^{16\mathrm{Q}}_{(4i-3),(4i-3)} = \frac{P}{9}\left[-4h_{i,\mathrm{R}}\left(h_{i,\mathrm{R}}+3\left(\left(\sum_{k=1}^{N}h_{k,\mathrm{R}}-h_{i,\mathrm{R}}\right)-\left(\sum_{k=1}^{N}h_{k,\mathrm{I}}-h_{i,\mathrm{I}}\right)\right)\right)\right.$$
$$\left.-4h_{i,\mathrm{I}}\left(h_{i,\mathrm{I}}+3\left(\left(\sum_{k=1}^{N}h_{k,\mathrm{R}}-h_{i,\mathrm{R}}\right)+\left(\sum_{k=1}^{N}h_{k,\mathrm{I}}-h_{i,\mathrm{I}}\right)\right)\right)\right]-\frac{\sqrt{2P}}{3}\left(4y_{\mathrm{R}}h_{i,\mathrm{R}}+4y_{\mathrm{I}}h_{i,\mathrm{I}}\right) \tag{A.9}$$

$$Q^{16\mathrm{Q}}_{(4i-2),(4i-2)} = \frac{P}{9}\left[-4h_{i,\mathrm{R}}\left(h_{i,\mathrm{R}}+\frac{3}{2}\left(\left(\sum_{k=1}^{N}h_{k,\mathrm{R}}-h_{i,\mathrm{R}}\right)-\left(\sum_{k=1}^{N}h_{k,\mathrm{I}}-h_{i,\mathrm{I}}\right)\right)\right)\right.$$
$$\left.-4h_{i,\mathrm{I}}\left(h_{i,\mathrm{I}}+\frac{3}{2}\left(\left(\sum_{k=1}^{N}h_{k,\mathrm{R}}-h_{i,\mathrm{R}}\right)+\left(\sum_{k=1}^{N}h_{k,\mathrm{I}}-h_{i,\mathrm{I}}\right)\right)\right)\right]-\frac{\sqrt{2P}}{3}\left(2y_{\mathrm{R}}h_{i,\mathrm{R}}+2y_{\mathrm{I}}h_{i,\mathrm{I}}\right) \tag{A.10}$$

$$Q^{16\mathrm{Q}}_{(4i-1),(4i-1)} = \frac{P}{9}\left[-4h_{i,\mathrm{R}}\left(h_{i,\mathrm{R}}+3\left(\left(\sum_{k=1}^{N}h_{k,\mathrm{R}}-h_{i,\mathrm{R}}\right)+\left(\sum_{k=1}^{N}h_{k,\mathrm{I}}-h_{i,\mathrm{I}}\right)\right)\right)\right.$$
$$\left.-4h_{i,\mathrm{I}}\left(h_{i,\mathrm{I}}+3\left(-\left(\sum_{k=1}^{N}h_{k,\mathrm{R}}-h_{i,\mathrm{R}}\right)+\left(\sum_{k=1}^{N}h_{k,\mathrm{I}}-h_{i,\mathrm{I}}\right)\right)\right)\right]-\frac{\sqrt{2P}}{3}\left(-4y_{\mathrm{R}}h_{i,\mathrm{I}}+4y_{\mathrm{I}}h_{i,\mathrm{R}}\right) \tag{A.11}$$

$$Q^{16\mathrm{Q}}_{(4i),(4i)} = \frac{P}{9}\left[-4h_{i,\mathrm{R}}\left(h_{i,\mathrm{R}}+\frac{3}{2}\left(\left(\sum_{k=1}^{N}h_{k,\mathrm{R}}-h_{i,\mathrm{R}}\right)+\left(\sum_{k=1}^{N}h_{k,\mathrm{I}}-h_{i,\mathrm{I}}\right)\right)\right)\right.$$
$$\left.-4h_{i,\mathrm{I}}\left(h_{i,\mathrm{I}}+\frac{3}{2}\left(-\left(\sum_{k=1}^{N}h_{k,\mathrm{R}}-h_{i,\mathrm{R}}\right)+\left(\sum_{k=1}^{N}h_{k,\mathrm{I}}-h_{i,\mathrm{I}}\right)\right)\right)\right]-\frac{\sqrt{2P}}{3}\left(-2y_{\mathrm{R}}h_{i,\mathrm{I}}+2y_{\mathrm{I}}h_{i,\mathrm{R}}\right) \tag{A.12}$$

[3] M. Kim, D. Venturelli, and K. Jamieson, "Leveraging Quantum Annealing for Large MIMO Processing in Centralized Radio Access Networks," in *Proc. of the ACM Special Interest Group on Data Communication,* pp. 241-255, Aug. 2019.

[4] Z. I. Tabi *et al.*, "Evaluation of Quantum Annealer Performance via the Massive MIMO Problem," *IEEE Access,* vol. 9, pp. 131658-131671, 2021.

[5] R. C. Kizilirmak, "Quantum Annealing Approach to NOMA Signal Detection," *12th Int. Symp. Commun. Syst. Netw. Digital Signal Process. (CSNDSP),* Porto, Portugal, pp. 1-5, Jul. 2020.

[6] F. Ishizaki, "Computational Method Using Quantum Annealing for TDMA Scheduling Problem in Wireless Sensor Networks," *13th Int. Conf. Signal Process. Commun. Syst. (ICSPCS),* Australia, pp. 1-9, Dec. 2019.

[7] R. C. Kizilirmak, "Non-Orthogonal Multiple Access (NOMA) for 5G Networks", *Towards 5G Wireless Networks - A Physical Layer Perspective*. London, United Kingdom: IntechOpen, 2016 [Online]. Available: https://www.intechopen.com/chapters/52822

[8] B. Makki *et al.*, "A Survey of NOMA: Current Status and Open Research Challenges," *IEEE Open J. Commun. Soc.,* vol. 1, pp. 179-189, 2020.

[9] "Error Sources for Problem Representation" *D-Wave System Documentation,* 2021 [Online]. Available: https://docs.dwavesys.com/docs/latest/c_qpu_ice.html [Accessed Apr. 2022].

[10] "Solver Parameters," *D-Wave System Documentation,* 2021 [Online]. Available: https://docs.dwavesys.com/docs/latest/c_solver_parameters.html [Accessed Apr. 2022].

[11] "Constraints Example: Minor-Embedding" *D-Wave System Documentation,* 2021 [Online]. Available: https://docs.dwavesys.com/docs/latest/c_gs_7.html[Accessed Apr. 2022].

[12] "D-Wave QPU Architecture: Topologies" *D-Wave System Documentation,* 2021 [Online]. Available: https://docs.dwavesys.com/docs/latest/c_gs_7.html[Accessed Apr. 2022].

[13] "LTE; Evolved Universal Terrestrial Radio Access (E-UTRA); Base Station (BS) radio transmission and reception (3GPP TS 36.104 version 9.4.0 Release 9)," ETSI, France, Tech. Specification. RTS/TSGR-0436104v940, Jul. 2010.

[14] "LTE; Evolved Universal Terrestrial Radio Access (E-UTRA); User Equipment (UE) radio transmission and reception (3GPP TS 36.101 version 14.3.0 Release 14)," ETSI, France, Tech. Specification. RTS/TSGR-0436101ve30, Apr. 2017.

[15] T. Rappaport, "Mobile Radio Propagation: Large-Scale Path Loss," *Wireless Communications Principles and Practice*, 2nd ed., Prentice Hall, 2001, pp. 69-138.

[16] "Operation and Timing" in *D-Wave System Documentation*, 2021 [Online]. Available: https://docs.dwavesys.com/docs/latest/c_qpu_timing.html [Accessed 5 Apr. 2022].