# Quantum Annealing for Communication Problems

Gengzhi Yang

November 27, 2024

It seems that applying Quantum Annealing techniques on communication problems could result in a much faster speed. Here I note some of the problems and corresponding solutions.

# 1 MIMO

The original work [KVJ19] uses the technique described in Section 5, and the resources need are shown in the figure 1 below. Part of their data is claimed that can be found at [She+16], which leads

Table 2: Logical (physical) number of qubits required for various configurations of the elementary adiabatic quantum ML decoder. For each configuration, **bold font** indicates non-feasibility on the current (2,031 physical qubit) D-Wave machine with Chimera connectivity.

Config.	BPSK	QPSK	16-QAM	64-QAM
10 × 10	10 (40)	20 (120)	40 (440)	60 (1K)
20  imes 20	20 (120)	40 (440)	80 (2K)	120 (4K)
40  imes 40	40 (440)	80 (2K)	160 (7K)	240 (15K)
60 × 60	60 (1K)	120 (4K)	240 (15K)	360 (33K)

Figure 1. MIMO resource needed
--------------------------------

to the Website here. Other data should has been generated in random. There are many numerical experiments are done, the biggest size requires around 1,000 qubits. For the real-world data mentioned above, QuAMax requires 2  $\mu$ s for BPSK and 2-10  $\mu$ s for QPSK.

For the follow-up work [KVJ20], they developed a hybrid classical-quantum algorithm for the MIMO problem and a new annealing schedule(Reverse Annealing). Part of the data is from [KVJ19]. They claimed a faster converging time and bigger success probability.

### **1.1** Vector Perturbation Precoding

The downlink VPP problem is to find an optimal perturbation vector  $v^*$  that minimizes the transmit power at the base station. The optimization problem is briefly introduced in Section 5.3.

For the size of the problem they are solving, "We evaluate our proposed QA based VPP (QAVP) technique on a real Quantum Annealing device over a variety of design and machine parameter settings. With existing hardware, QAVP can achieve a BER of  $10^{-4}$  with 100µs compute time, for a 6×6 MIMO system using 64 QAM modulation at 32 dB SNR." [Quoted from [Kas+21]]

## 1.2 Design for Coherent Ising Machines

In work [SVJ22], they designed a special QUBO form to adapt to the Coherent Ising Machine. This adpatation could be seen at Section 5.4. They performed their numercial experiments on a CIM simulator. The numerical experiments are done in the configuration of 16 users, 16 antennas/32 antennas.

# 1.3 Reconfigurable Antenna MIMO Systems

This optimization problem (Section 5.5) is firstly introduced in [KSJ24]. In this paper, they are aiming at quite small settings, e.g. 4 transmitters and 4 receivers.

# 2 LDPC

The technique is described in Section 6.

In work [KJ20], it is said that their embedding design can support LDPC codes of block length up to 420 bits on real state-of-the-art QA hardware with 2,048 qubits. For the error rate, "*Results* on the real-world quantum annealer show that QBP achieves a bit error rate of  $10^{-8}$  in 20 µs and a 1,500 byte frame error rate of  $10^{-6}$  in 50 µs at signal-to-noise ratio of 9 dB over a Gaussian noise channel".

# 3 Polar Codes

In work [KKJ22], they detailed explained how to transforme the Polar codes decoding process into a QUBO form, which can be seen in Section 7. They describe their problem size as "We experimentally evaluate HyPD on a state-of-the-art QA device with 5,627 qubits, for Polar codes of block length 1,024 bits, in Rayleigh fading channels. ... Our decoder targets 1,024-bit 5G-NR Polar codes with BPSK modulation and 200 message data bits, which is typically the maximum UCI payload in LTE and 5G-NR eMBB scenarios." The implementation is said to follow the 3GPP Multiplexing and channel coding standard.

# 4 Quantum Annealing

## 4.1 Two Equivalent Formulations

#### 4.1.1 Ising Spin Glass Form

For spins  $s_i \in \{+1, -1\},\$ 

$$\hat{s}_1, \cdots, \hat{s}_N = \arg\min_{\{s_1, \cdots, s_N\}} \left( \sum_{i < j} g_{ij} s_i s_j + \sum_i f_i s_i \right).$$

$$(1)$$

### 4.1.2 QUBO Form

Quadratic unconstrained binary optimization is formulated as follows: For  $q_i \in \{0, 1\}$ ,

$$\hat{q}_1, \cdots, \hat{q}_N = \arg\min_{\{q_1, \cdots, q_N\}} \left( \sum_{i \le j} Q_{ij} q_i q_j \right).$$
<sup>(2)</sup>

Here  $Q \in \mathbb{R}^{n \times n}$  is a upper triangular matrix.

Given a QUBO formulation, by doing  $q_i \leftrightarrow \frac{1}{2}(s_i + 1)$ , we could have

$$\begin{aligned} \hat{s}_{1}, \cdots, \hat{s}_{N} &= \arg \min_{\{s_{1}, \cdots, s_{N}\}} \left( \sum_{i \leq j} Q_{ij} (\frac{s_{i}+1}{2}) (\frac{s_{j}+1}{2}) \right) \\ &= \arg \min_{s_{1}, s_{2}, \cdots, s_{N}} \left( \sum_{i < j} \frac{Q_{ij}}{4} s_{i} s_{j} + \sum_{i} \frac{Q_{ii}}{4} s_{i}^{2} + \sum_{i} \frac{Q_{ii}}{2} s_{i} + \sum_{\{j:i < j\}} \frac{Q_{ij}}{4} s_{i} + \sum_{\{j:j < i\}} \frac{Q_{ij}}{4} s_{i} \right) \\ &= \arg \min_{s_{1}, s_{2}, \cdots, s_{N}} \left( \sum_{i < j} \frac{Q_{ij}}{4} s_{i} s_{j} + \sum_{i} \frac{Q_{ii}}{2} s_{i} + \sum_{\{j:i < j\}} \frac{Q_{ij}}{4} s_{i} + \sum_{\{j:j < i\}} \frac{Q_{ij}}{4} s_{i} \right) \\ &= \arg \min_{s_{1}, s_{2}, \cdots, s_{N}} \left( \sum_{i < j} \frac{Q_{ij}}{4} s_{i} s_{j} + \left( \sum_{i} \frac{Q_{ii}}{2} + \sum_{\{j:i < j\}} \frac{Q_{ij}}{4} + \sum_{\{j:j < i\}} \frac{Q_{ij}}{4} \right) s_{i} \right), \end{aligned}$$
(3)

which leads to  $g_{ij} = \frac{Q_{ij}}{4}, f_i = \sum_i \frac{Q_{ii}}{2} + \sum_{\{j:i < j\}} \frac{Q_{ij}}{4} + \sum_{\{j:j < i\}} \frac{Q_{ij}}{4}.$ 

### 4.2 Coherent Ising Machines

Given the Ising optimization problem

$$\arg\min_{s\in\{\pm1\}} -\sum_{j\neq i} J_{ij} s_i s_j,\tag{4}$$

the CIM can be modelled by real valued variable  $x_i$  where  $s_i = sign(x_i)$ . And the whole process is

$$\frac{\mathrm{d}x_i}{\mathrm{d}t} = (1-p)x_i - x_i^3 + \epsilon(t)\sum_{j\neq i} J_{ij}x_j.$$
(5)

The dynamics above can be further enhanced to

$$\frac{\mathrm{d}x_i}{\mathrm{d}t} = (1-p)x_i - x_i^3 + \epsilon(t)e_i \sum_{j \neq i} J_{ij}x_j,$$

$$\frac{\mathrm{d}e_i}{\mathrm{d}t} = -\beta(x_i^2 - a)e_i.$$
(6)

# 5 From MIMO to QUBO Form

The Multiple Input Multiple Output (MIMO) decoding problem is about given  $N_t$  mobile users and  $N_r$  antennas (Additionally assume  $N_r \ge N_t$ ), solving the maximum likelihood detection problem, which can be described as a least-square optimization:

$$\hat{v} = \arg \min_{v \in \mathbb{C}^{N_t}, v \text{ generated in } O^{N_t}} \|y - Hv\|^2.$$
(7)

Here  $|O| = 2^Q$ , i.e. any element in v is generated by Q bits.  $H = H^I + iH^Q$  is the wireless channel and  $y = H\overline{v} + n$ , where  $\overline{v} \in \mathbb{C}^{N_t}$  is the input symbol and n is the noise. One conventional approach is to use the QR factorization on H.

# 5.1 ML-to-QA Problem Reduction

Let consider the symbol  $v = [v_1, v_2, \dots, v_{N_t}] \in \mathbb{C}^{N_t}$ . If  $v_i$  is generated by O where  $|O| = 2^Q$ , we are requiring  $N = QN_t$  solution variables. Now we are in searching of some transform function  $T(\cdot)$  such that  $T^{-1}(v_i)$  is a Q-bit representation.

#### 5.1.1 Binary Modulation

In this case,  $v_i = \{\pm 1\}$ . Choose Q = 0 and  $T(q_i) = 2q_i - 1$  converts the problem into the QUBO form.

#### 5.1.2 QPSK Modulation

In this case,  $v_i = \{\pm 1 \pm 1i\}$ . Choose Q = 1 and  $T(q_i) = 2q_i^0 - 1 + i(2q_i^1 - 1)$  converts the problem into the QUBO form. Here  $q_i = [q_i^0, q_i^1]$ .

### 5.2 Higher-order Modulation

Here we focus on 16 quadrature amplitude modulation (16-QAM). For an 1-D constellation  $\begin{bmatrix} 00 & 01 & 10 & 11 \end{bmatrix}$ , since each  $v_i$  is sampled from such constellation, we need Q = 2 and choose  $T(q_i) = 4q_i^0 + 2q_i^1 - 3$ , which takes values of  $\{-3, -1, 1, 3\}$ .

Thus for a 2-D constellation

$$\begin{bmatrix} 0011 & 0111 & 1011 & 1111 \\ 0010 & 0110 & 1010 & 1110 \\ 0001 & 0101 & 1001 & 1101 \\ 0000 & 0100 & 1000 & 1100 \end{bmatrix}$$
(8)

we just need to take  $v_i = (4q_i^0 + 2q_i^1 - 3) + i(4q_i^2 + 2q_i^3 - 3)$ , representing  $v_i = \{-3, -1, 1, 3\} + i\{-3, -1, 1, 3\}$ .

However, practical wireless communication systems use a different bit-to-symbol mapping as

$$\begin{bmatrix} 0010 & 0110 & 1110 & 1010\\ 0011 & 0111 & 1111 & 1011\\ 0001 & 0101 & 1101 & 1001\\ 0000 & 0100 & 1100 & 1000 \end{bmatrix},$$

$$(9)$$

which is called Gray code. A naive approach should be simply considering the 4-PAM constellation  $\begin{bmatrix} 00 & 01 & 11 & 10 \end{bmatrix}$ , and take the Gray-coded bit-to-symbol mapping as the transformation, namely  $T(q_i) = 2(2q_i^0 - 1) + 2(q_i^0 - q_i^1)^2 - 1$ . Here we successfully turned the symbol into QUBO variables. But the resulting expansion of the ML norm would yield cubic and quartic terms.

Alternatively, we can just use the QuAMax transform at the receiver, flip the 3rd and 4th bit conditioned on the 2nd bit of  $[q_i^0, q_i^1, q_i^2, q_i^3]$  yields

$$\begin{bmatrix} 0011 & 0100 & 1011 & 1100\\ 0010 & 0101 & 1010 & 1101\\ 0001 & 0110 & 1001 & 1110\\ 0000 & 0111 & 1000 & 1111 \end{bmatrix} .$$
(10)

Denote the intermediate code as  $[b_i^0, b_i^1, b_i^2, b_i^3]$ . After that, do

$$g_i^0 = b_i^0, \quad g_i^1 = b_i^0 \oplus b_i^1, \quad g_i^2 = b_i^1 \oplus b_i^2, \quad g_i^3 = b_i^2 \oplus b_i^3.$$
(11)

This gives us the Gray code.

#### 5.3 Vector Perturbation Precoding

In VPP, the user data symbol  $u \in \mathbb{C}^{N_r}$  is perturbed by an integer  $v \in \mathbb{C}^{N_r}$ . The optimization becomes

$$v^* = \arg\min \left\| H^{\dagger}(HH^{\dagger})^{-1}(u+\tau v) \right\|^2.$$
 (12)

Here  $\tau$  is a constant.

Consider the k-th entry in v as  $a_k + ib_k$ , we then can represent  $a_k$  or  $b_k$  as

$$\sum_{m=1}^{t} 2^{m-1} q_m - 2^t q_{t+1},\tag{13}$$

which takes all the integers from  $[-2^t, 2^t - 1]$ . Plug everything back yields a QUBO form.

### 5.4 Complex to Real Dilation

Take

$$H' = \begin{bmatrix} \Re(H) & -\Im(H) \\ \Im(H) & \Re(H) \end{bmatrix}, y' = \begin{bmatrix} \Re(y) \\ \Im(y) \end{bmatrix}, v' = \begin{bmatrix} \Re(v) \\ \Im(v) \end{bmatrix},$$
(14)

We can get an equivalent real valued system as

$$v = \arg\min \|y' - H'v'\|^2.$$
 (15)

Inspired by the Coherent ising machines, take an initial guess symbol  $x_m$ , denote the solution as u, then define  $d = u - x_m$ . Since u and  $x_m$  are symbols(dilated version), we know each entry in d must be an even integer. Then we can always write

$$d = Ts, \tag{16}$$

where  $T = [2^n I_{2N_t}, 2^{n-1} I_{2N_t}, \cdots, 2I_{2N_t}, I_{2N_t}], s = [s_0, s_1, \cdots, s_{2N_t-1}]^{\top}, s_i = \pm 1$ . The optimization problem can be re-formulated as

$$\arg\min_{d} \|y' - H'(d + x_m)\| = \arg\min_{d} \|(y' - H'x_m) - H'd\|.$$
(17)

The remaining problem is that the problem above has both linear and Quadratic terms. However, this can be solved by adding one ancilla qubit.

## 5.5 Reconfigurable Antenna MIMO

For two diagonal matices X of size  $NN_T$  and Y of size  $NN_R$  (their elements should be 0's and 1's), the optimization goal is

$$\max_{X,Y} \operatorname{Tr}(XG^{\top}YG),\tag{18}$$

subject to

$$\sum_{i=kN+1}^{(k+1)N} x_{ii} = 1, \sum_{i=kN+1}^{(k+1)N} y_{ii} = 1,$$
(19)

for k range from 0 to  $N_T - 1$  or  $N_R - 1$ .

Prolong the diagonal lines of X and Y and store them in one vector denoted as b, take  $Q = \begin{bmatrix} 0 & \frac{1}{2}T\\ \frac{1}{2}T^{\top} & 0 \end{bmatrix}$ , where  $T = G^{\odot 2}$  is of size  $NN_T \times NN_R$ , we have the following quadratic form:

$$\arg\max b^{\top}Qb, \quad P_k(b) = \sum_{i=kN+1}^{(k+1)N} b_i - 1 = 0.$$
(20)

It is possible to convert the optimization problem above into a quadratic form and use the constraint as part of the optimization objective function. After that just use the CIM solver.

# 6 LDPC Codes

The Low Density Parity Check Code (LDPC) is a linear error correcting code. A binary (N, K)LDPC code is a linear block code described functionally by a sparse parity check matrix H of size  $(N - K) \times N = M \times N$ . It is said to be a  $(d_b, d_c)$  regular code if each column of H contains exactly  $d_b$  1's and each row of H contains exactly  $d_c$  1's.

## 6.1 LDPC Encoder

Consider a message u of length K, M = N - K.

- 1. Convert H into  $[P|I_{N-K}]$  by Gaussian elimination. Here P is a matrix of size  $M \times K$ .
- 2. Construct  $G = [I_K | P^\top]$ . G is of size  $K \times (M + K) = K \times N$ .
- 3. Encode u into c = uG. Note here the "matrix-vector" multiplication is in the modulo-2 sense.

This way of encoding ensures that the modulo two bit-sum at every check node is zero. Notice that the first 3-bits are exactly the original message itself!

## 6.2 LDPC Decoder

We call a binary bit string  $x = [x_0, x_1, \dots, x_{N-1}]$  as the decoded message if  $xH^{\top} = 0$  (in the modulo 2 sense). That is the reason why H is called the parity-check matrix.

# 6.3 QUBO Form

We are designing a loss function for x such that the minimum satisfies  $xH^{\top} = 0$  (In the modulo 2 sense). A straight forward way is to introduce some ancilla qubits, say  $\{q_{e_{ik}}\}_i$ , and define

$$L_i = xH(:,i) - 2(q_{e_{i1}} + 2q_{e_{i2}} + \dots + 2^m q_{e_{ik}}).$$
(21)

The latter term defined any even number that can be possibly expressed by xH(:,i).

Also we can add some distance function  $\Delta_i := (q_i - \mathbb{P}(q_i = 1|y_i))^2$ , where y is the vector received. The final QUBO form could be a linear combination of these two.

# 7 Polar Codes

Polar codes are a linear block error-correcting codes. Pre-decide the K most reliable bit-channels, encode the message  $m = [m_0, \dots, m_{K-1}]$  into  $u = [u_0, \dots, u_{N-1}]$ , set the rest of the bits as 0. The encoding process can be easily described as

$$x = uG_N,\tag{22}$$

where  $G_N = G_2^{\otimes n}$ ,  $G_2 = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}$ . *u* is the input vector and *x* is the encoded codeword.

Alternatively, for the input  $u_0$  and  $u_1$ , encode it as  $[u_0 \oplus u_1, u_1]$ . Same process also applys for two input vectors: for input  $u_L$  and  $u_R$ , encode them as  $[u_L \oplus u_R, u_R]$ . An example is provided as follows:

$$u_0, u_1, u_2, u_3 \longrightarrow [u_0 \oplus u_1, u_1], [u_2 \oplus u_3, u_3] \longrightarrow [u_0 \oplus u_1 \oplus u_2 \oplus u_3, u_1 \oplus u_3, u_2 \oplus u_3, u_3].$$
(23)

Set  $q = [q_0, \dots, q_{N-1}]$  as our solution vector, and any  $a_i$  is ancilla qubit for computation. The QUBO form consists 3 parts: For any node T for performing the XOR operation,

$$C_N(T) = \sum_{i,j} (b_i + b_j - a_k - 2a_{k+1})^2.$$
(24)

The  $b_i$  and  $b_j$  is determined from earlier nodes. Still take our example in Equation 23,

$$C_N(T_{1,0}) = (q_0 + q_1 - a_1 - 2a_2)^2,$$
  

$$C_N(T_{1,1}) = (q_2 + q_3 - a_3 - 2a_4)^2,$$
  

$$C_N(T_{2,0}) = (a_1 + a_3 - a_5 - 2a_6)^2 + (q_1 + q_3 - a_7 - 2a_8)^2.$$
(25)

And the final codeword is  $b = [a_5, a_7, a_3, q_3]$ .

The second constraint is about the frozen bits. As stated as before, some bits are intentionally set as 0.

$$C_F(q_i) = q_i. (26)$$

The third constraint is about how our codeword matches with the received vector y.

$$C_R(b_j) = \sum_j (b - \mathbb{P}(b_j = 1|y))^2.$$
 (27)

This can only obtained from other estimations.

The final QUBO form should be a weighted linear combination of the three constraints mentioned before.

# References

- [Kas+21] Srikar Kasi et al. "Quantum annealing for large MIMO downlink vector perturbation precoding". In: ICC 2021-IEEE International Conference on Communications. IEEE. 2021, pp. 1–6.
- [KJ20] Srikar Kasi and Kyle 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.
- [KKJ22] Srikar Kasi, John Kaewell, and Kyle Jamieson. "The design and implementation of a hybrid classical-quantum annealing polar decoder". In: GLOBECOM 2022-2022 IEEE Global Communications Conference. IEEE. 2022, pp. 5819–5825.
- [KSJ24] Ioannis Krikidis, Abhishek Kumar Singh, and Kyle Jamieson. "Optimizing Reconfigurable Antenna MIMO Systems with Coherent Ising Machines". In: *arXiv preprint arXiv:2403.12571* (2024).
- [KVJ19] Minsung Kim, Davide Venturelli, and Kyle 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.
- [KVJ20] Minsung Kim, Davide Venturelli, and Kyle Jamieson. "Towards hybrid classical-quantum computation structures in wirelessly-networked systems". In: Proceedings of the 19th ACM Workshop on Hot Topics in Networks. 2020, pp. 110–116.
- [She+16] Clayton Shepard et al. "Understanding real many-antenna MU-MIMO channels". In: 2016 50th Asilomar Conference on Signals, Systems and Computers. IEEE. 2016, pp. 461–467.
- [SVJ22] Abhishek Kumar Singh, Davide Venturelli, and Kyle Jamieson. "Perturbation-based formulation of maximum likelihood MIMO detection for coherent ising machines". In: GLOBE-COM 2022-2022 IEEE Global Communications Conference. IEEE. 2022, pp. 2523–2528.