RESEARCH ARTICLE

Detecting multiple communities using quantum annealing on the D-Wave system

Christian F. A. Negre[1], Hayato Ushijima-Mwesigwa[2], Susan M. Mniszewski[2]*

1 Theoretical Division, Los Alamos National Laboratory, Los Alamos, NM, United States of America,
2 Computer, Computational, & Statistical Sciences Division, Los Alamos National Laboratory, Los Alamos, NM, United States of America

* smm@lanl.gov

Abstract

A very important problem in combinatorial optimization is the partitioning of a network into communities of densely connected nodes; where the connectivity between nodes inside a particular community is large compared to the connectivity between nodes belonging to different ones. This problem is known as community detection, and has become very important in various fields of science including chemistry, biology and social sciences. The problem of community detection is a twofold problem that consists of determining the number of communities and, at the same time, finding those communities. This drastically increases the solution space for heuristics to work on, compared to traditional graph partitioning problems. In many of the scientific domains in which graphs are used, there is the need to have the ability to partition a graph into communities with the "highest quality" possible since the presence of even small isolated communities can become crucial to explain a particular phenomenon. We have explored community detection using the power of quantum annealers, and in particular the D-Wave 2X and 2000Q machines. It turns out that the problem of detecting at most two communities naturally fits into the architecture of a quantum annealer with almost no need of reformulation. This paper addresses a systematic study of detecting two or more communities in a network using a quantum annealer.

1 Introduction

The use of networks spans across many scientific domains. To showcase how broad this spectrum is, we highlight the following examples: molecules are chemical networks with atoms connected to each other [1–3]; living cells follow a communication pattern described by a network [4]; and human interactions create different social networks [5]. The study of networks has resurged in recent times due to the availability and capability of producing and storing data from a large number of applications. Advances in crystallography, for example, allows chemists to get atomistic representations of complex proteins; social media speeds up human communication; etc. [6].

In all of the previous mentioned scientific domains in which graphs are used, there is the imminent need to have the ability to partition a graph into communities with the "highest

Competing interests: The authors have declared that no competing interests exist.

quality" possible. In other words we want the best community split to be able to recognize particular features of the network such as the presence of small communities that could become crucial to explain a particular phenomenon [2]. Another feature that needs to be properly revealed is the boundary between communities which could be crucial for classification.

Fig 1a shows the molecular structure of a small protein composed of seven amino acids [7]. In Fig 1b, we show the result of partitioning a graph built out of the the connectivity between orbitals of different atoms. This connectivity is computed out of the density matrix of the system, which is an object that accounts for the electronic structure. The graph is composed of 300 nodes/orbitals and 1794 edges/connections. This community splitting was performed using the technique presented in this paper, rendering an optimum with 7 communities. We can clearly see how, roughly, each amino acid that composes the protein is classified as belonging to a single community. This is far from being trivial. In order to recognize each amino acid component, a computational chemist will need to perform a careful inspection of the chemical structure of the protein, which could become a time-consuming task if the system is larger.

Due to the large size of network datasets, many meaningful properties on networks are expensive to compute, especially when dealing with NP-hard graph problems. Post Moore's era supercomputing has provided us an opportunity to explore new approaches for traditional graph algorithms on quantum computing architectures [8]. These machines are attractive alternatives to traditional computers in helping to solve the aforementioned graph problems [9]. In a recent paper we have developed a formulation for graph partitioning to be performed on a quantum annealer [8]. In this paper we present a formulation for performing community detection of multiple communities (two or more).

The general concept of community structure was introduced for the first time by Girvan and Newman in 2002 [10] to describe the general appearance (the skeleton) of a network. A network can be divided into sets of nodes belonging to different communities (also called clusters). Nodes within any of these communities are highly connected (high intraconnectivity); whereas nodes in different communities are less connected (low interconnectivity). This natural division of a graph into communities differs from the usual graph partitioning (GP) problem in that there are no restrictions on the size of the communities (see Fig 2). Methods involving both community detection and graph partitioning are considered unsupervised machine learning techniques [11]. The total number of feasible solutions for k-community
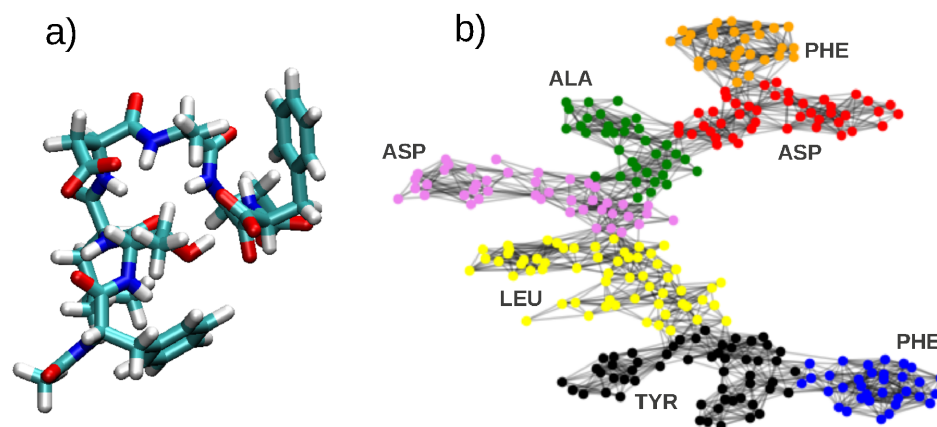
Fig 1. a) Molecular representation of a small protein composed of seven amino acids (PHE, ASP, ALA, ASP, LEU, TYR and PHE). C, O, N, and H atoms are represented in cyan, read, blue, and white colors respectively. b) Representation of the orbital connectivity associated graph with nodes colored by the communities they belong to. Note PHE and ASP appear twice in the amino acid sequence.

https://doi.org/10.1371/journal.pone.0227538.g001
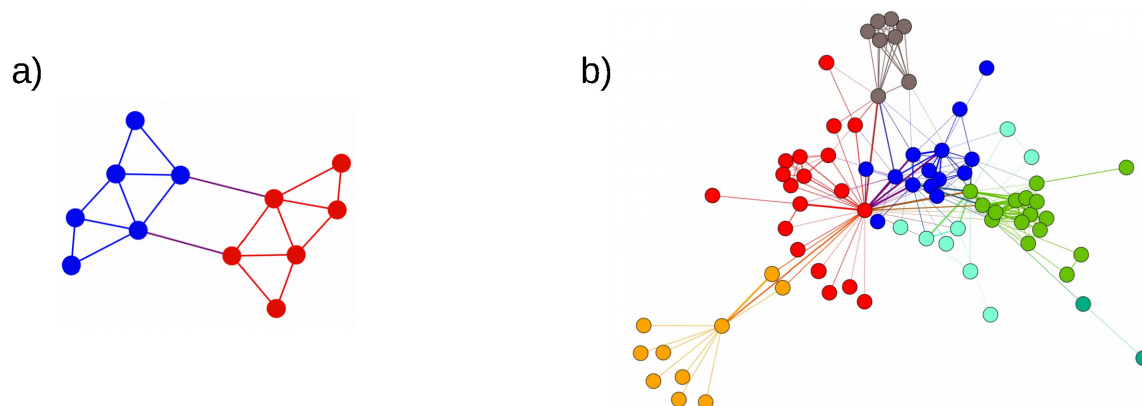
Fig 2. Graph partitioning (a) showing two parts of equal size (six nodes) depicted with red and blue colors; and (b), example of community detection showing 6 different communities of the Les Miserables coappearance social network.

detection over a network of n nodes is given by the Bell number $B_n = \frac{1}{e}\sum_{k=0}^{\infty}\frac{k^n}{k!}$ which has an upper bound of $\left(\frac{0.792n}{\ln(n+1)}\right)^n$ [12]. This implies an extensive search space for feasible solutions as compared to the regular balanced GP problem. It also implies the need to combine heuristics and recursive optimization methods to search for the optimal community structure. Several algorithms have been proposed based on greedy techniques [13, 14], simulated annealing [15, 16], genetic algorithm [17], spectral optimization [18] and extreme optimization methods [19, 20]. Recent work in [21] proposes multi-community detection using quantum hardware for signed graphs.

The notion of "more connected" within a community is arbitrary and the existence of a unique definition is still under debate, however, the metric proposed by Girvan and Newman based on the modularity has been well accepted in the field [10, 22, 23]. Other metrics and methods include k-means [24], spectral partitioning [25], hierarchical grouping [26], modularity density [22], min-max cut [27], and normalized cut [25] among others. Moreover, Neukart, et al., implemented a quantum-assisted clustering method on the D-Wave system [28]. The modularity metric is used to quantify the quality of a community structure by comparing the connectivity of edges within communities with the connectivity of an equivalent network where edges would be placed randomly.

Maximizing the modularity with respect to the community partitions is a problem that has been largely studied and different methods have been proposed. Despite all these efforts, modularity maximization still remains a field that needs further exploration. Two main problems have been identified. The first problem is the existence of a resolution limit, in which a very small community could be missed if the resulting community structure also contains large communities [29]. The second problem is the existence of several minima with the same modularity which prevents or makes it difficult to obtain the global minima [30]. These two problems are related to what we have previously loosely defined as highly resolved community structure.

A method for further refining community structures has been recently introduced [31]. The authors proposed an exact algorithm for bi-partitioning a network using split and merge movements over communities of an existing partition. Using this approach the authors were able to further improve the modularity values and, as a consequence, the quality of the community structure. We will use some of these highly refined results as a point of comparison with the ones obtained from the method we are introducing.

Quantum computers of the annealer type, such as the D-Wave 2X and 2000Q, minimize the Ising objective function as follows making use of the quantum entanglement effect [32]. The objective function can be written as follows:

$$O(\mathbf{h}, \mathbf{J}, \mathbf{s}) = \sum_i h_i s_i + \sum_{i<j} J_{ij} s_i s_j \tag{1}$$

where $s_i \in \{-1, +1\}$ are magnetic spin variables where the result is encoded; h_i and J_{ij} are local magnetic fields and coupling strengths that encode the problem Hamiltonian. At the hardware level, the D-Wave quantum computer is composed of qubits with sparse connectivity as a fixed sparse graph, known as a *Chimera* graph. During the annealing process each qubit can be in a "superposition" state (both a "-1" and a "+1" simultaneously). The superposition lasts until an outside event causes it to collapse into either a "-1" or a "+1" state. The result of the annealing process is a low-energy ground state **s**, consisting of an Ising spin for each qubit value $\in \{-1, +1\}$. This makes quantum computers useful to tackle NP-hard complex problems including optimization, machine learning and sampling problems. Maximization problems can also be solved by the D-Wave by using the negative of Eq 1 as the objective function. The formulation where variables take values of either 0 or 1 is called the quadratic unconstrained binary optimization or QUBO formulation and it is an alternative representation that can easily be translated to or from the Ising model. An Ising model can become a QUBO through the transformation, $s = 2x - 1$.

Current D-Wave platforms have physical constraints such as limited precision, sparse connectivity, and number of available qubits. Embedding is required to map a problem onto the hardware *Chimera* graph prior to annealing. Purely quantum approaches are limited by the number of graph nodes/variables that can be represented on the hardware, 46 for the D-Wave 2X and 64 for the D-Wave 2000Q. Due to the fact of limited variables, larger problems require hybrid quantum-classical approaches.

In this paper we describe the method for performing community detection based on the modularity metric using the D-Wave quantum annealer. We carefully derive the formulation of the problem as a QUBO. Results are compared with existing benchmarks using "state of the art" tools.

2 Formulation

Let $G = (V, E)$ be a weighted graph with nodes i in V and edges ij in E such that the corresponding adjacency matrix A is defined as follows:

$$A_{ij} = \begin{cases} 0, & \text{if} \quad i = j \\ w_{ij}, & \text{if} \quad i \neq j \end{cases} \tag{2}$$

with w_{ij} being the weight of edge ij. We can then construct a modularity matrix B as the difference between A and a matrix constructed as an outer product of the vector degree **g**. Here, the node degree g_i is defined as $g_i = \sum_j A_{ij}$. Newman's expression for the modularity matrix B can be written as follows:

$$B = A - \frac{\mathbf{g}\mathbf{g}^T}{2m} \tag{3}$$

where, $2m = \sum_i g_i$. Or equivalently:

$$B_{ij} = A_{ij} - \frac{g_i g_j}{2m} = A_{ij} - \frac{g_i g_j}{\sum_l g_l} \tag{4}$$

For partitioning the graph into at most two communities, if $s_i \in \{-1, 1\}$, is a binary variable indicating which community node i belongs to, then the modularity Q for any given partition is given by

$$Q = \frac{1}{2m} \sum_{i,j} B_{ij} \cdot \frac{s_i s_j + 1}{2} \tag{5}$$

Community detection requires maximizing the modularity Q. The rows and columns of the matrix B sum to zero, thus, we have $\sum_{i,j} B_{ij} \frac{s_i s_j + 1}{2} = \sum_{i,j} B_{ij} \frac{s_i s_j}{2}$. In addition, the term $\frac{s_i s_j + 1}{2}$ can be viewed as a product of binary variables $x_i \in \{0, 1\}$. We can therefore write Eq (5) in matrix form as

$$Q = \frac{1}{4m} \mathbf{s}^T B \mathbf{s}$$

where $\mathbf{s}$ is a column vector with entries s_i. A transformation from a QUBO to Ising formulation can be given by

$$\mathbf{s}^T B \mathbf{s} = 4\mathbf{x}^T B \mathbf{x} - 4\mathbf{x}^T B \mathbb{1} + \mathbb{1}^T B \mathbb{1}$$

where $\mathbf{x}$ is a vector such that $x_i \in \{0, 1\}$, $\mathbb{1}$ is a vector of all ones, and $\mathbf{s} = 2\mathbf{x} - \mathbb{1}$. However, since $B\mathbb{1} = \mathbf{0}$, we have

$$\mathbf{s}^T B \mathbf{s} = 4\mathbf{x}^T B \mathbf{x} \tag{6}$$

Eq (6) essentially shows that both QUBO and Ising formulations are equivalent for modularity maximization. Thus, the maximum modularity for at most two communities is given by

$$\max_{\mathbf{s}} \left(\frac{1}{4m} \mathbf{s}^T B \mathbf{s} \right) \quad \text{or} \quad \max_{\mathbf{x}} \left(\frac{1}{m} \mathbf{x}^T B \mathbf{x} \right) \tag{7}$$

which are clearly unconstrained quadratic optimization problems, suitable to be solved by quantum annealers.

Now, if we are interested in partitioning the graph into at most k communities, the modularity for any given partition is given by

$$Q = \frac{1}{2m} \sum_{i,j} B_{ij} \cdot \delta(c_i, c_j) \tag{8}$$

where $1 \leq c_i \leq k$ is the community node i belongs to and the function $\delta(c_i, c_j) = 1$, if $c_i = c_j$ or 0 otherwise; with $i \neq j$ for $1 \leq i$ and $j \leq |V|$. Eq 8 poses the problem that the function δ is not necessarily a quadratic binary variable function.

In our previous work, we demonstrated dividing a graph into two communities [8]. Additionally, we introduced the concept of a logical super-node which allows the partition of graphs into more than two parts in an "all at once" k-concurrent fashion (see Fig 3). Each logical super-node represents a graph node. A super-edge represents a connection between graph nodes. The logical super-node utilizes a one-hot encoding to represent the selection of 1 out of k communities [33]. In this work, we will use the concept of the logical super-node and a
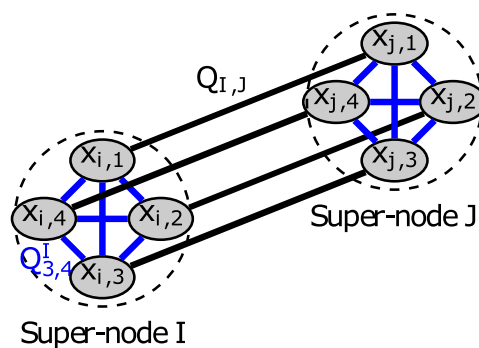
Fig 3. An example of the super-node concept used in *k*-concurrent community detection is shown for partitioning into 4 parts. Two super-nodes *I* and *J* consisting of four subnodes ($x_{i/j,k}$) where each are connected by a super-edge $Q_{I,J}$. Internal edges $Q_{l,m}^{I/J}$ where $l, m \in \{1 - 4\}$ are set to enforce the selection of only one subnode to be equal to "1" after the annealing. The super-edge $Q_{I,J}$ is shown with connections between corresponding super-nodes.

proper formulation leading to a QUBO problem to partition the graph into more than two communities in a *k*-concurrent fashion.

In this section, we will generalize the modularity matrix formulation for an arbitrary number of communities. For this, let $G = (N, E)$ be a graph for which the final result of the community split is a set composed of communities C_i with the following property: $N = \{C_1 \cup C_2 \cup \ldots\}$. For a given community, C, the value

$$Q_C = \frac{1}{2m} \sum_{(i,j) \in C} \left(A_{ij} - \frac{g_i g_j}{2m} \right)$$

computes the modularity of community C within the QUBO formulation.

Suppose that now we want to partition G into at most k communities. Let $x_{i,j}$ be the decision variables such that

$$x_{i,j} = \begin{cases} 1 & \text{if node } i \text{ is in community } j \\ 0 & \text{otherwise.} \end{cases}$$

Strictly speaking, the set $\{x_{i,1}, \ldots x_{i,k}\}$ is a logical super-node as depicted in Fig 3. Since each node must be in exactly one community, the following constraint needs to be fulfilled.

$$\sum_{j=1}^{k} x_{i,j} = 1 \tag{9}$$

Let

$$\mathbf{x}_j = \begin{bmatrix} x_{1,j} \\ x_{2,j} \\ \vdots \\ x_{n,j} \end{bmatrix},$$

be the vector state, then:

$$Q(\mathbf{x}) = \max_{\mathbf{x}} \left(\sum_{j=1}^{k} \mathbf{x}_j^T B \mathbf{x}_j \right) \tag{10}$$

subject to $\sum_{j=1}^{k} x_{i,j} = 1$, for $i = 1, \ldots, n$ with $x_{i,j} \in \{0, 1\}$ for $i = 1, \ldots, n$ and $j = 1, \ldots, k$. A corresponding relaxation is given by

$$\max_{\mathbf{x}} \left(\sum_{j=1}^{k} \mathbf{x}_j^T B \mathbf{x}_j + \sum_{i=1}^{n} \gamma_i (\sum_{j=1}^{k} x_{i,j} - 1)^2 \right) \tag{11}$$

with $x_{i,j} \in \{0, 1\}$, for $i = 1, \ldots, n$ and $j = 1, \ldots, k$. γ_i are the corresponding relaxation coefficients.

The derivation that follows has already been shown in reference [8]. We will give a summary for completeness here. We start by considering the following equality:

$$(\sum_{j=1}^{k} x_{i,j} - 1)^2 = (\sum_{j=1}^{k} x_{i,j})^2 - 2 \sum_{j=1}^{k} x_{i,j} + 1$$

Let now $\mathbf{Z}_i$ be the $\mathcal{N} \times \mathcal{N}$ zero matrix (with $\mathcal{N} = k \times n$) whose jth diagonal element is 1 if and only if $j \equiv i \pmod{n}$. For example, in $\mathbf{Z}_1$ every 1st, $(n+1)^{\text{th}}$, $(2n+1)^{\text{th}}$, $\ldots$, $((k-1)n+1)^{\text{th}}$ diagonal element is 1 and has zero everywhere else. Then

$$(\sum_{j=1}^{k} x_{i,j})^2 = \mathbf{X}^T \mathbf{Z}_i \mathbb{1}_{\mathcal{N} \times \mathcal{N}} \mathbf{Z}_i \mathbf{X}$$

where $\mathbf{X}$ is the matrix $(\mathbf{x}_1, \ldots, \mathbf{x}_j, \ldots, \mathbf{x}_k)$ formed by the $\mathbf{x}_j$ column vectors. In consequence,

$$\sum_{j=1}^{k} x_{i,j} = \mathbb{1}_{\mathcal{N}}^T \mathbf{Z}_i \mathbf{X}.$$

Hence,

$$(\sum_{j=1}^{k} x_{i,j} - 1)^2 = \mathbf{X}^T \mathbf{Z}_i \mathbb{1}_{\mathcal{N} \times \mathcal{N}} \mathbf{Z}_i \mathbf{X} - 2 \mathbb{1}_{\mathcal{N}}^T \mathbf{Z}_i \mathbf{X} + 1$$

and

$$\begin{aligned}
\sum_{i=1}^{n} \gamma_i (\sum_{j=1}^{k} x_{i,j} - 1)^2 &= \sum_{i=1}^{n} \gamma_i (\mathbf{X}^T \mathbf{Z}_i \mathbb{1}_{\mathcal{N} \times \mathcal{N}} \mathbf{Z}_i \mathbf{X} \\
&\quad - 2 \mathbb{1}_{\mathcal{N}}^T \mathbf{Z}_i \mathbf{X} + 1) \\
&= \mathbf{X}^T \sum_{i=1}^{n} \gamma_i (\mathbf{Z}_i \mathbb{1}_{\mathcal{N} \times \mathcal{N}} \mathbf{Z}_i) \mathbf{X} \\
&\quad - 2 \sum_{i=1}^{n} \gamma_i \mathbb{1}_{\mathcal{N}}^T \mathbf{Z}_i \mathbf{X} + \sum_{i=1}^{n} \gamma_i.
\end{aligned}$$

Let D_γ be a diagonal matrix such that

$$D_\gamma = \text{diag}(\gamma_1, \ldots, \gamma_n)$$

and $\mathbf{B}_\Gamma$ be a block matrix with $k \times k$ blocks, where each block is equal to D_γ, then

$$\sum_{i=1}^{n} \gamma_i \mathbf{Z}_i \mathbb{1}_{\mathcal{N}\times\mathcal{N}} \mathbf{Z}_i = \mathbf{B}_\Gamma$$

and

$$\sum_{i=1}^{n} \gamma_i \mathbb{1}_{\mathcal{N}}^T \mathbf{Z}_i \mathbf{X} = \Gamma^T \mathbf{X}.$$

where we have defined $\Gamma^T = \sum_{i=1}^{n} \gamma_i \mathbb{1}_{\mathcal{N}}^T \mathbf{Z}_i$. So,

$$\sum_{i=1}^{n} \gamma_i \left(\sum_{j=1}^{k} x_{i,j} - 1\right)^2 = \mathbf{X}^T \mathbf{B}_\Gamma \mathbf{X} - 2\Gamma^T \mathbf{X} + \sum_{i=1}^{n} \gamma_i.$$

Therefore, we have

$$\max_{\mathbf{x}} (\mathbf{X}^T (\beta \mathcal{B} + \mathbf{B}_\Gamma) \mathbf{X} - 2\Gamma^T \mathbf{X}) \tag{12}$$

with $x_{i,j} \in \{0, 1\}$, being the elements of matrix $\mathbf{X}$ with $i = 1, \ldots, n, j = 1, \ldots, k$. $\mathcal{B}$ is a block diagonal matrix with the modularity matrix B on the diagonal, and β just a tunable parameter to control the weight of the term.

3 Results and discussion

We used the python NetworkX tools [34] for pre- and post-processing of the example graphs. The size of these graphs in most cases is larger than the number of nodes or variables that can be embedded on the D-Wave 2X and 2000Q. Therefore, we have used the hybrid quantum-classical tool, *qbsolv*, developed by D-Wave [35]. The *qbsolv* software takes the full problem in QUBO format as input and makes multiple calls to the D-Wave to solve subQUBOs for global minimization, followed by tabu search for local minimization. It can be called directly through the D-Wave Ocean application programming interface (API) or from the command line. Resulting bitstrings of zeros and ones are translated based on the optimization problem's representation.

The quality of a community structure is determined by evaluating the modularity metric. This varies from 0 to 1, with a larger value being preferable. We have used the Zachary (karate club) graph to compare the results of the modularity metric obtained with other methods. The aforementioned is a social network of friendships between 34 members of a karate club at a US university in the 1970s. With only 34 nodes and 78 edges, this graph is considered an archetypal social network that has been extensively used to benchmark graph algorithms [36]. From Table 1 we observe that modularity is similar across all methods, however the quantum

Table 1. Values of modularity are shown for the Zachary network computed with different algorithms: Girvan and Newman (GN) [10]; Clauset et al. (CNM) [37]; Duch and Arenas (DA) [20]; and the D-Wave quantum annealer (QA).

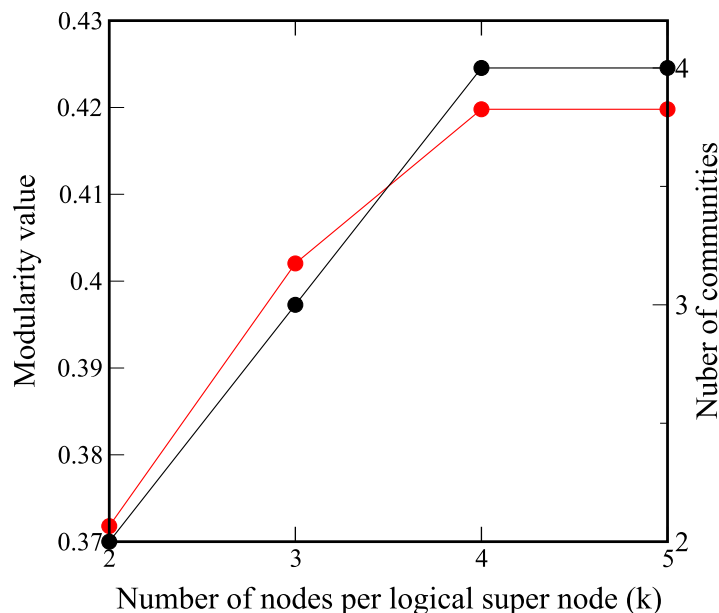| Method	Modularity
GN	0.401
CNM	0.381
DA	0.419
Newman	0.419
QA	0.420

Fig 4. Zachary karate club network showing the different communities found with the D-Wave quantum annealer.

annealer approach results in the best value based on the community structure. This value is identical to the record value obtained by Blondel et al. showing four communities with a modularity of 0.41979 [14]. A representation of the community structure obtained by this method can be seen in Fig 4.

In Fig 5 we can see how the numbers of communities and the modularity reach a maximum when the number of nodes (or variables) per logical super-node increases. Modularity and number of communities reach values of 0.42 and 4 respectively. Instead of increasing the number of nodes per super-node iteratively, one could directly use a large number such as $k = n$ and get the same result. Currently, the D-Wave machines are limited in the problem size as number of variables that can be embedded and run on the *Chimera* graph architecture. The 2X is limited to 46 variables, while the 2000Q can run problems up to 64. The sparse connectivity of the *chimera* graph requires that some variables will be represented by chains of qubits [38]. This can quickly use up the available qubits on the 2X (up to 1152) or the 2000Q (up to 2048). The penalty constant γ that is used to constrain each graph node to be in only one community can vary depending on the graph. Tuning is required for each new network problem.

We have computed the community structure for the following set of benchmark social networks: a coappearance network of characters in the novel Les Miserables (LesMiserables) [39]; an undirected social network of frequent associations between 62 dolphins in a community living off Doubtful Sound, New Zealand (Dolphins) [40]; a book co-purchasing network where nodes represent books about US politics and edges represent frequent co-purchasing of books (PoliticalBooks) [23]; a collaboration network of Jazz musicians (Jazz) [41]; and a metabolic network of the nematode C. elegans (Elegans) [20]. Results are shown in Table 2.

Results for the Zachary and Dolphins benchmarks match the highly refined values reported in [31] which, to our knowledge are the "best" in the literature ever obtained. The quantum annealer performs slightly better for the Dolphins network, when compared with non-refined results in the literature [42]. Our results on LesMiserables and PoliticalBooks benchmarks are similar in comparison to the refined values of 0.56001 and 0.52724 respectively [42]. On the
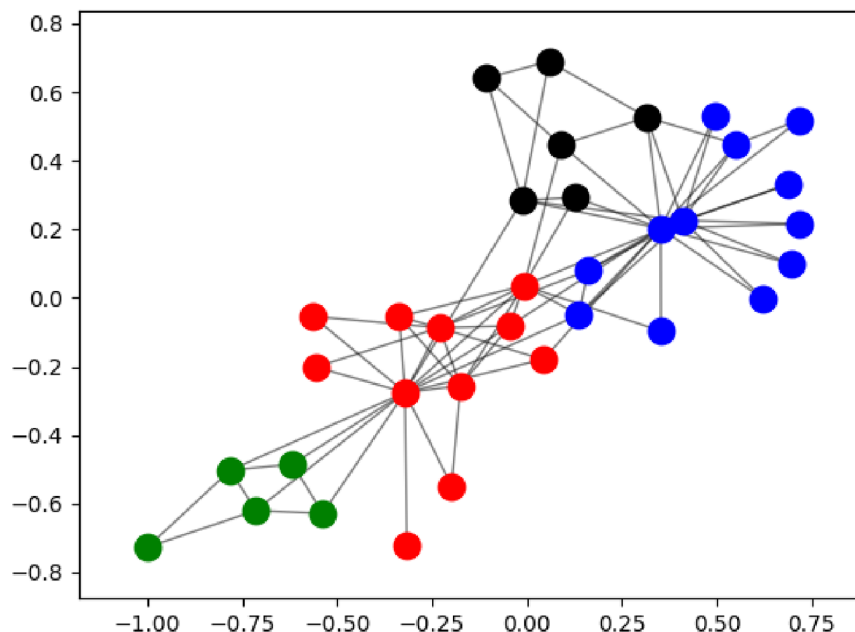
Fig 5. Convergence of community detection for the Zachary network with increasing number of nodes (or variables) per logical super-node. The black dots show the number of communities that are obtained and the red dots show the modularity values.

other hand, results for the Jazz and Elegans benchmarks are similar in comparison, but lower than the values reported in the literature [20].

As a mode of comparison, we have also run the same calculation using a Classical annealer (CA) to solve the QUBO problem as implemented in the D-Wave's ocean API [43] (See Table 2). To our surprise, the CA performs as well as the QA for the case of the Zachary graph (smallest case in the set). This means that there must be a benefit of formulating the modularity method as a QUBO problem that goes beyond the Quantum nature of the D-Wave annealer. When the size of the graph increases, the performance of the CA gets compromised. For the largest graph this QUBO approach solved with the CA shows a notably poorer performance. The parameters for the classical annealing were chosen as the default with the exception of `num_reads` which was set to 1000 to increase the quality of the results. We have also explored different inverse temperature programs by changing the variable `beta_range` and saw no improvement on the results. One of the benefits of using quantum annealing is that there is no need to proceed recursively. The total default time for annealing is 20 μs for the case of the D-Wave. brown However, one can make a reasonable analogy between the

Table 2. The results of *k*-concurrent community detection on benchmark graphs. N, E, N_{com}, and *Mod.* are the number of nodes, number of edges, number of communities and modularity respectively. We show results from both the D-Wave quantum annealer (QA) and a Classical annealer (CA).

Graph	N	E	N_{com}	*Mod.* (QA)	*Mod.* (CA)
Zachary	34	78	4	0.41979	0.41979
Dolphins	62	159	5	0.52852	0.52761
LesMiserables	77	254	6	0.55861	0.53071
PoliticalBooks	105	441	4	0.52555	0.52008
Jazz	198	2742	3	0.44447	0.44205
Elegans	453	2040	5	0.41728	0.20031

annealing time in QA and the number of iterations in CA. Problems having many local minima (that could come from large sized system) would require longer annealing times for QA [44].

For the case of the quantum annealer we used the default number of reads of 50 and no spin reversal transform was applied. The annealing time was 20 μs which is the default value for D-Wave 2000Q. For every case, the value of γ was set such that the penalty constant of Eq 2 is not violated and the relative importance of the values of B are not undermined. The values of γ used for all the graphs are as follows: -5 Zachary and dolphins, -6 for LesMiserables, -10 for PoliticalBooks, -525 for Jazz, and -33 for Elegans. Note that these values will depend upon the case analyzed and we do not have a direct way to determine them based on matrix B. For the case of CA the number was manually changed to get the best possible results without violating the one hot encoding constraint (nodes cannot belong to multiple communities). For the case of CA the values of γ that we used are: -5 for Zachary and Dolphins, -7 for LesMiserables, -10 for PoliticalBooks, -20 for Jazz, and -20 for Elegans. The score performed by the quantum annealer reflects the size of the machine architecture. A small graph results in a clean embedding on the *Chimera* graph, with small chains to compensate for missing connections. As we said before, when graphs are larger we start having longer chains which can hinder the annealing performance [38]. For even larger graphs, hybrid quantum-classical approaches such as *qbsolv* are used for orchestrating the use of the D-Wave solver on subproblems. Lower performance using *qbsolv* may come from the mixing of classical and quantum processing. The quantum state initially created is no longer coming from the full Hamiltonian, but several pieces of it and a full pseudo solution is reconstructed. We strongly believe that having more qubits available and increased connectivity will improve the performance of this technique.

A particular property of the modularity matrix is that it can be thresholded with almost no effect on the quality of the community structure. This statement has a significant impact since it is saying that we can save qubits and reduce the size of the problem to be embedded. Here we have tested this hypothesis by using the Zachary graph and thresholding the modularity matrix weights before solving the problem for community detection. From Table 3 we can see that we can remove up to 40% of the edges (by thresholding) and still get the same community structure with the same modularity value.

Note that, with this technique, the modularity remains unchanged up to a large threshold value because the community structure does not change and we are still evaluating the modularity with the original (non-thresholded) matrix. In other words, the annealing is performed

Table 3. Results for *k*-concurrent community detection with thresholding on the Zachary benchmark are shown. *Thres.*, E, N_{com}, and *Mod.* are the threshold values, number of edges, number of communities and modularity respectively.

Thres.	E	N_{com}	Mod.
0.00	561	4	0.41978
0.02	544	4	0.41978
0.05	411	4	0.41978
0.06	334	4	0.41978
0.07	300	4	0.41510
0.08	244	3	0.39907
0.10	227	3	0.39907
0.15	169	2	0.37179
0.25	110	2	0.37179

with the QUBO generated from the thresholded modularity matrix whereas the evaluation of the modularity, instead, is performed with the original modularity matrix.

It is difficult to disentangle the combined effect of *qbsolv* and the quantum annealing. However, if we look at the thresholding of the modularity matrix, we can see that the number of edges can be fairly reduced down to 300 without changing the result of the modularity. This means that, at least, 300×4×4 = 4800 chimera edges are required. The D-Wave 2000Q can embed a fully connected 64 node graph requiring at least 4096 chimera edges. This means that the 300 edges thresholded version of the Zachary graph is comparable to the maximum fully connected graph that can be embedded in the D-Wave 2000Q machine.

4 Conclusion

In this paper we have demonstrated the ability of a quantum computer and in particular a quantum annealer to solve community detection as an optimization problem. From the analysis of the results we observe that the quantum computer can render a highly optimized community structure. In the case of the Zachary graph, we reach the actual record value. For other other benchmark graphs, including the larger graphs such as Jazz and Elegans, the quality of the community structure is comparable to "state of the art" results.

One of the most notable observations is that by using this quantum annealing technique with the *k*-concurrent method, we obtain the community structure "all at once" within the annealing time. There is no need to implement an iterative process as is the case for heuristic methods running on classical computers. In principle, the quantum annealer is designed to explore the full search space during the annealing time. Limitations such as a reduced number of available qubits and sparse connectivity can be detrimental to the quality of the results. brown Improvements of the results presented in this work remain tied to the evolution of quantum annealers towards better scaling, and better exploration of the solution space.

Acknowledgments

Author Contributions

Conceptualization: Christian F. A. Negre, Hayato Ushijima-Mwesigwa, Susan M. Mniszewski.

Data curation: Christian F. A. Negre, Hayato Ushijima-Mwesigwa.

Formal analysis: Hayato Ushijima-Mwesigwa.

Funding acquisition: Susan M. Mniszewski.

Investigation: Christian F. A. Negre, Hayato Ushijima-Mwesigwa, Susan M. Mniszewski.

Methodology: Christian F. A. Negre.

Software: Christian F. A. Negre.

Supervision: Susan M. Mniszewski.

Validation: Christian F. A. Negre.

Writing – original draft: Christian F. A. Negre, Hayato Ushijima-Mwesigwa, Susan M. Mniszewski.

Writing – review & editing: Christian F. A. Negre, Hayato Ushijima-Mwesigwa, Susan M. Mniszewski.

References

1. Niklasson AMN, Mniszewski SM, Negre CFA, Cawkwell MJ, Swart PJ, Mohd-Yusof J, et al. Graph-based linear scaling electronic structure theory. The Journal of Chemical Physics. 2016; 144(23):234101. https://doi.org/10.1063/1.4952650 PMID: 27334148

2. Rivalta I, Sultan MM, Lee NS, Manley GA, Loria JP, Batista VS. Allosteric pathways in imidazole glycerol phosphate synthase. Proceedings of the National Academy of Sciences. 2012; 109(22):E1428–E1436. https://doi.org/10.1073/pnas.1120536109

3. Negre CFA, Morzan UN, Hendrickson HP, Pal R, Lisi GP, Loria JP, et al. Eigenvector centrality for characterization of protein allosteric pathways. PNAS. 2018. https://doi.org/10.1073/pnas.1810452115 PMID: 30530700

4. Jeong H, Tombor B, Albert R, Oltvai ZN, Barabási AL. The large-scale organization of metabolic networks. Nature. 2000; 407(651). https://doi.org/10.1038/35036627 PMID: 11034217

5. Ugander J, Karrer B, Backstrom L, Marlow C. The Anatomy of the Facebook Social Graph. CoRR. 2011;abs/1111.4503.

6. Fortunato S, Hric D. Community detection in networks: A user guide. Physics Reports. 2016; 659:1–44. https://doi.org/10.1016/j.physrep.2016.09.002.

7. Ghale P, Kroonblawd M, Mniszewski S, Negre C, Pavel R, Pino S, et al. Task-based Parallel Computation of the Density Matrix in Quantum-based Molecular Dynamics using Graph Partitioning. SIAM Journal on Scientific Computing. 2017; 39(6):C466–C480. https://doi.org/10.1137/16M109404X

8. Ushijima-Mwesigwa H, Negre CFA, Mniszewski SM. Graph Partitioning Using Quantum Annealing on the D-Wave System. In: Proceedings of the Second International Workshop on Post Moore's Era Supercomputing (PMES). PMES'17. New York, NY, USA: ACM; 2017. p. 22–29. Available from: http://doi.acm.org/10.1145/3149526.3149531.

9. Shaydulin R, Ushijima-Mwesigwa H, Safro I, Mniszewski S, Alexeev Y. Community Detection Across Emerging Quantum Architectures. In: Proceedings of the Third International Workshop on Post Moore's Era Supercomputing (PMES). PMES'18; 2018. p. 12–14.

10. Girvan M, Newman MEJ. Community structure in social and biological networks. Proceedings of the National Academy of Sciences. 2002; 99(12):7821–7826. https://doi.org/10.1073/pnas.122653799

11. Géron A. Hands-On Machine Learning with Scikit-Learn and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems. O'Reilly Media; 2017. Available from: https://www.amazon.com/Hands-Machine-Learning-Scikit-Learn-TensorFlow/dp/1491962291?SubscriptionId=AKIAIOBINVZYXZQZ2U3A&tag=chimbori05-20&linkCode=xm2&camp=2025&creative=165953&creativeASIN=1491962291.

12. Berend D, Tassa T. Improved bounds on Bell numbers and on moments of sums of random variables. Probability and Math Statistics. 2010.

13. Newman MEJ. Fast algorithm for detecting community structure in networks. Phys Rev E. 2004; 69:066133. https://doi.org/10.1103/PhysRevE.69.066133

14. Blondel VD, Guillaume JL, Lambiotte R, Lefebvre E. Fast unfolding of communities in large networks. Journal of Statistical Mechanics: Theory and Experiment. 2008; 2008(10):P10008. https://doi.org/10.1088/1742-5468/2008/10/P10008

15. Kirkpatrick S, Gelatt CD, Vecchi MP. Optimization by Simulated Annealing. Science. 1983; 220(4598):671–680. https://doi.org/10.1126/science.220.4598.671 PMID: 17813860

16. Guimera R, Nunes Amaral LA. Functional cartography of complex metabolic networks. Nature. 2005; 433(895). https://doi.org/10.1038/nature03288 PMID: 15729348

17. Goldberg DE. Genetic Algorithms in Search, Optimization and Machine Learning. 1st ed. Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc.; 1989.

18. Fortunato S. Community detection in graphs. Physics Reports. 2010; 486(3):75–174. https://doi.org/10.1016/j.physrep.2009.11.002.

19. Boettcher S, Percus AG. Optimization with Extremal Dynamics. Phys Rev Lett. 2001; 86:5211–5214. https://doi.org/10.1103/PhysRevLett.86.5211 PMID: 11384460

20. Duch J, Arenas A. Community detection in complex networks using extremal optimization. Phys Rev E. 2005; 72:027104. https://doi.org/10.1103/PhysRevE.72.027104

21. Zahedinejad E, Crawford D, Adolphs C, Oberoi JS. Multi-Community Detection in Signed Graphs Using Quantum Hardware. arXiv preprint arXiv:190104873. 2019.

22. Newman MEJ, Girvan M. Finding and evaluating community structure in networks. Physical review E. 2004; 69(2):026113. https://doi.org/10.1103/PhysRevE.69.026113

23. Newman MEJ. Modularity and community structure in networks. Proceedings of the National Academy of Sciences. 2006; 103(23):8577–8582. https://doi.org/10.1073/pnas.0601602103

24. Hartigan JA, Wong MA. Algorithm AS 136: A K-Means Clustering Algorithm. Journal of the Royal Statistical Society Series C (Applied Statistics). 1979; 28(1):100–108.

25. Shi J, Malik J. Normalized cuts and image segmentation. IEEE Transactions on Pattern Analysis and Machine Intelligence. 2000; 22(8):888–905. https://doi.org/10.1109/34.868688

26. Ward JH Jr. Hierarchical Grouping to Optimize an Objective Function. Journal of the American Statistical Association. 1963; 58(301):236–244. https://doi.org/10.1080/01621459.1963.10500845

27. Ding CHQ, He X, Zha H, Gu M, Simon HD. A min-max cut algorithm for graph partitioning and data clustering. In: Proceedings 2001 IEEE International Conference on Data Mining; 2001. p. 107–114.

28. Neukart F, Dollen DV, Seidel C. Quantum-Assisted Cluster Analysis on a Quantum Annealing Device. Frontiers in Physics. 2018; 6:55. https://doi.org/10.3389/fphy.2018.00055

29. Fortunato S, Barthélemy M. Resolution limit in community detection. Proceedings of the National Academy of Sciences. 2007; 104(1):36–41. https://doi.org/10.1073/pnas.0605965104

30. Good BH, de Montjoye YA, Clauset A. Performance of modularity maximization in practical contexts. Phys Rev E. 2010; 81:046106. https://doi.org/10.1103/PhysRevE.81.046106

31. Cafieri S, Hansen P, Liberti L. Improving heuristics for network modularity maximization using an exact algorithm. Discrete Applied Mathematics. 2014; 163:65–72. https://doi.org/10.1016/j.dam.2012.03.030.

32. D-Wave Systems.

33. Harris HSL David Money. 3.4.2 State Encodings. Elsevier; 2007. Available from: https://app.knovel.com/hotlink/khtml/id:kt009519R3/digital-design-computer/state-encodings.

34. Hagberg A, Swart P, S Chult D. Exploring network structure, dynamics, and function using NetworkX. In: Proc. SciPy 2008; 2008. p. 11–16. Available from: http://networkx.github.io/.

35. Booth M, Reinhardt SP, Roy A. Partitioning Optimization Problems for Hybrid Classical/Quantum Execution. D-Wave Technical Report Series. 2017;14(1006A-A):1–9.

36. Zachary WW. An Information Flow Model for Conflict and Fission in Small Groups. Journal of Anthropological Research. 1977; 33(4):452–473. https://doi.org/10.1086/jar.33.4.3629752

37. Clauset A, Newman MEJ, Moore C. Finding community structure in very large networks. Phys Rev E. 2004; 70:066111. https://doi.org/10.1103/PhysRevE.70.066111

38. Lanting T, Przybysz AJ, Smirnov AY, Spedalieri FM, Amin MH, Berkley AJ, et al. Entanglement in a Quantum Annealing Processor. Phys Rev X. 2014; 4:021041.

39. Knuth DE. The Stanford GraphBase: A Platform for Combinatorial Computing. Addison-Wesley, Reading, MA; 1993.

40. Lusseau D, Schneider K, Boisseau OJ, Haase P, Slooten E, Dawson SM. The bottlenose dolphin community of Doubtful Sound features a large proportion of long-lasting associations. Behavioral Ecology and Sociobiology. 2003; 54(4):396–405. https://doi.org/10.1007/s00265-003-0651-y

41. Kaur S, Singh S, Kaushal S, Sangaiah AK. Comparative analysis of quality metrics for community detection in social networks using genetic algorithms. Neural Network World. 2016; 26(6):625–641.

42. Fu YH, Huang CY, Sun CT. A community detection algorithm using network topologies and rule-based hierarchical arc-merging strategies. PLOS ONE. 2017; 12(11):1–30. https://doi.org/10.1371/journal.pone.0187603

43. D-Wave's Classical SA; 2018.

44. Albash T, Lidar DA. Demonstration of a Scaling Advantage for a Quantum Annealer over Simulated Annealing. Physical Review X. 2018; 8(3). https://doi.org/10.1103/PhysRevX.8.031016