ALMA MATER STUDIORUM · UNIVERSITY OF BOLOGNA

School of Science
Department of Computer Science and Engineering
Master's Degree in Computer Science

Clustering aggregation on a neutral atom quantum computer

Supervisor:
Prof. Stefano Lodi

Submitted by:
Riccardo Scotti

Cosupervisors:
Dr. Gabriella Bettonte
Dr. Antonio Costantini

Session II
Academic Year 2023/2024

UNIVERSITY OF BOLOGNA

Abstract

School of Science
Department of Computer Science and Engineering

Master's Degree in Computer Science

Clustering aggregation on a neutral atom quantum computer

by Riccardo Scotti

TODO

Acknowledgements

TODO

Contents

Abstract **iii**

Acknowledgements **v**

1 Introduction **1**

2 Background **3**
2.1 Clustering 3
2.2 Quantum computing 4
2.3 Quantum annealing 5
2.4 Neutral atoms technology 5

3 State of the art **7**

4 Proposed method **9**

5 Experimental setup **11**
5.1 Quantum platforms 11
5.2 Dataset 12
5.3 Evaluation metrics 12
5.4 Description of experiments 14

6 Results **15**
6.1 Fresnel experiment 15

7 Conclusions **17**

Bibliography **23**

Chapter 1

Introduction

TODO

Chapter 2

Background

2.1 Clustering

Clustering is an unsupervised data analysis technique that can be informally defined as the partitioning of a set of objects into groups called clusters; such partitioning must ensure that objects within a same cluster are more similar[1] to each other than to objects in other clusters. [7]

Clustering is widely employed in numerous scenarios that require to group together similar data points, or to extract knowledge from a set of objects in the absence of any prior information. For instance, it is used in marketing and finance as a profiling tool; in image processing and computer vision, as a segmentation technique [6], where it plays a pivotal role in various fields, such as remote image sensing [17] and digital forensics [12]; by energy distribution companies, to optimize the allocation of resources to end users.

2.1.1 Formal definition

Let $D = \{x_1, ..., x_n\}$ be a set of objects, also called *dataset*; a clustering for D is a collection of k elements $\mathcal{C} = \{c_1, ..., c_k\}$, with $\mathcal{C} \subseteq \mathcal{P}(D)$, such that the following properties hold:

$$D = \bigcup_{i=1}^{k} c_i \tag{2.1}$$

$$c_i \cap c_j = \emptyset \quad \forall c_i, c_j \in \mathcal{C}. \tag{2.2}$$

2.1.2 Clustering aggregation

Clustering aggregation, also known as clustering ensemble, is a technique that aims to improve the robustness and overall quality of clustering results, by combining multiple clustering solutions into a single one [19].

The motivation for utilizing clustering aggregation arises from the observation that no single clustering algorithm is universally optimal for all types of data or applications. Variability in initial conditions, distance metrics, and the inherent characteristics of the dataset can lead to different results, even when using the same algorithm. Aggregating these clustering solutions helps overcoming discrepancies and reduces the risk of selecting a sub optimal solution [18].

[1] Provided that a binary ordering relation is defined on the set of objects.

2.2 Quantum computing

Quantum computing is a set of computational models and paradigms that combines concepts of computer science, physics and engineering. It offers an alternative perspective on computing to that of classical computing, both in terms of theoretical framework, as well as regarding the physical realization of machines.

The basic ideas of quantum computing were established by Richard Feynman in a 1982 paper, discussing the usage of computers to perform physics simulations [4]. Feynman observed that, due to their inherent complexity, quantum systems could not be feasibly simulated by classical computers, and introduced the idea of computing machines based on quantum principles [3]. A subsequent work by Deutsch demonstrated, via the construction of a quantum Turing machine, that quantum computers are computationally equivalent to classical computers, and could therefore be useful for scopes beyond mere quantum simulation [1].

2.2.1 Qubits and quantum gates

A classical computer can be analyzed, from an abstract point of view, as a system of bits, which represent information, and logical gates, which manipulate information; analogously, a quantum computer is composed of quantum bits, or qubits for short, and quantum gates.

A quantum computer differs from a classical computer in the fact that it is, at its core, a quantum system, and as such it abides to the laws of quantum mechanics. Some of these laws, referred to as the postulates of quantum mechanics, give a synthetic description of qubits and quantum gates [10].

Postulate 1 (State space) *The state of a quantum system is described by a unit vector in a Hilbert space $\mathcal{H}$.*

Postulate 2 (Evolution) *The evolution of a system from one state to another is described by a unitary operator.*

Postulate 3 (Composition of systems) *When two systems, whose state spaces are $\mathcal{H}_1$ and $\mathcal{H}_2$, are treated as a unique, combined system, then the state space for the overall system is the tensor product of the single state spaces $\mathcal{H}_1 \otimes \mathcal{H}_2$.*

Postulate 4 (Measurement) *Given a system with state*

$$|\psi\rangle = \sum_i \alpha_i |\varphi_i\rangle ,$$

performing a measurement on the system will yield label i as result with probability $|\alpha_i|^2$ and leave the system in the state $|\varphi_i\rangle$.

From postulate 1 follows one of the principal characteristics of qubits, which goes under the name of *superposition*. While the state of a classical bit can only have one of its possible values (i.e. either `0` or `1`), the state of a qubit is a linear

combination, or superposition, of multiple values. Supposing that a qubit can only have two values $|0\rangle$ and $|1\rangle$, it assumes the general form

$$|\psi\rangle = \alpha\,|0\rangle + \beta\,|1\rangle\,,$$

where (α, β) is a unit vector of a Hilbert space $\mathcal{H}$, or alternatively $|\alpha|^2 + |\beta|^2 = 1$.

Postulates 2 gives a mathematical description of quantum gates, and from it it is possible to derive the property that all quantum gates are invertible.

Postulate 3 states that multiple qubits can be analyzed as a single system and introduces the notion of *entanglement*, the condition in which multiple qubits mutually affect the state of each other.

Postulate 4 describes the operation of measurement; while the measurement of a classical bit does not, in principle, affect its value, measurement in quantum mechanics is a destructive operation that breaks superposition and puts a qubit in a single state. Furthermore, it explicitates the probabilistic nature of quantum systems.

2.2.2 Advantages of quantum computing

The peculiar characteristics of quantum computers, some of which mentioned in 2.2.1, make them better suited than classical computer at solving various tasks. An important result in quantum computing is Shor's algorithm for number factorization in polynomial time, which is not only a groundbraking result in computer science and mathematics, but has also important implications for cryptography and security [16]. Other examples of remarkable results are in quantum chemistry simulation [9] and in solving optimization problems for logistics and finance [2].

However, despite its promising advantages, quantum computing remains a mainly theoretical field, due to significant technological challenges in building reliable machines. Current quantum processors are limited by issues such as noise, error rates and decoherence, which hinder practical applications for most large scale problems [13].

2.3 Quantum annealing

2.4 Neutral atoms technology

Chapter 3

State of the art

Chapter 4

Proposed method

Chapter 5

Experimental setup

This chapter provides a description of the experimental setup used to evaluate the clustering aggregation protocol in different quantum computing environments, both on a simulator and on real hardware. The protocol was tested on three distinct platforms: a simulator for a neutral atom quantum computer, specifically the Pulser simulator by PASQAL; the Fresnel neutral atom quantum computer, also developed by PASQAL; the Advantage quantum annealer with Pegasus topology, developed by D-Wave.

5.1 Quantum platforms

5.1.1 Programmable neutral-atom arrays

5.1.2 QuTiP

The Quantum Toolbox in Python, also known as QuTiP, is an object oriented Python framework designed to represent and simulate the dynamics of open and closed quantum systems [8].

The core of information representation in QuTiP is the `Qobj` class, which is used to represent both quantum states (such as pure states or density matrices) and quantum operators (such as Hamiltonians or measurement operators). The `Qobj` class offers methods to perform matrix operations like addition, multiplication and tensor product, which are essential to manipulate quantum systems; sparse matrix encoding of objects allows these operations to be executed relatively efficiently.

To perform simulations using QuTiP, it is necessary to specify the initial conditions of the system, for example ket vectors or density matrices, and a Hamiltonian or other operators, which describe the time evolution and possible dissipations. It is then possible to select different kinds of solvers, such as the Monte Carlo wave function method for a Schrödinger equation describing evolution of pure states, or a numerical approach for a Lindblad master equation in the case of open systems.

Results are presented in the form of evolved states at fixed intervals throughout the simulation time windows. Additionally, QuTiP allows the calculation of expectation values for observables (such as spin, position, or energy) over time.

Since QuTiP does not directly support neutral atom systems, Pulser contains a library that acts as an interface to QuTiP, in order to simplify the definition of neutral-atom arrays and the simulation of experiments with laser pulses.

5.2 Dataset

The dataset used to test the protocol is shown in figure 5.1. It was first introduced by Gionis, Mannila and Tsaparas to test classical clustering aggregation algorithms [5]; it was then used by Li and Latecki to test a clustering aggregation protocol that uses simulated annealing [11].

It is specifically designed to contain different shapes and configuration of points, in such a way that most clustering algorithms fail to produce a correct clustering. It is made up of 7 clusters and 788 points in total.

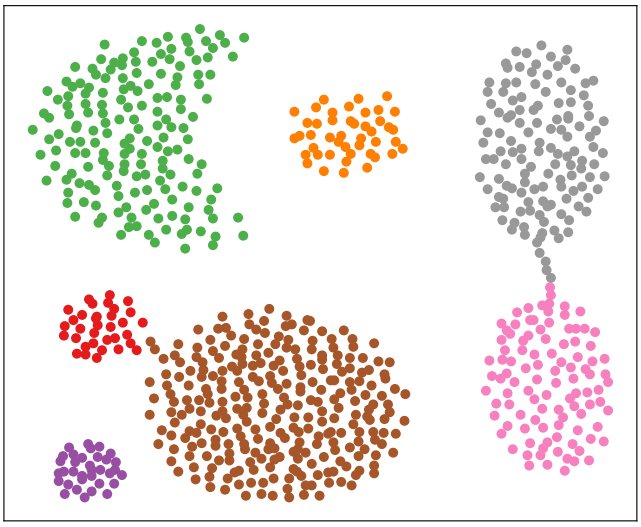

Figure 5.1: Dataset plot

5.3 Evaluation metrics

Different metrics were used to evaluate the quality of single clusters and of clustering algorithms as a whole. Silhouette score was used as a factor to weigh clusters in the clustering aggregation protocol; Rand index was used to compare the overall quality of the clustering algorithms and aggregation protocols.

5.3.1 Silhouette score

The silhouette score is a metric used to evaluate clustering quality, introduced by Rousseeuw [15]. The score is computed for each point in the dataset and reflects

both its cohesion with points in the same cluster, as well as the separation with respect to points in different clusters. Specifically, the silhouette score of a point p in the dataset is defined as

$$S_p = \frac{b_p - a_p}{\max(a_p, b_p)}, \tag{5.1}$$

where a_i is the average distance from the point to other points in the same cluster (intra-cluster distance), and b_i is the average distance to points in the nearest different cluster (inter-cluster distance). The score ranges between -1 and 1, where values close to 1 indicate that points are well-clustered, with high cohesion and good separation, while negative values suggest points may have been assigned to the wrong cluster.

A quantification of the quality for a single cluster can be obtained by computing the average of the silhouette score of the point it contains; more formally, for a cluster c_i, the formula for its average silhouette (AS_i) is

$$AS_i = \frac{\sum_{p \in c_i} S_p}{|c_i|}. \tag{5.2}$$

5.3.2 Rand index

The Rand index is a metric that quantifies the similarity between two data clusterings, first proposed by Rand [14]. It is computed by considering all possible pairs of points in the dataset and assessing whether they are assigned to the same or different clusters in the two clusterings being compared. The index assumes values between 0 and 1, with 1 indicating perfect agreement between the two clusterings.

Given two clusterings $\mathcal{C}_1$ and $\mathcal{C}_2$, their Rand index is computed as

$$RI = \frac{a + b}{a + b + c + d}, \tag{5.3}$$

where

- a is the number of point pairs that are put in the same cluster in both clusterings;
- b is the number of point pairs that are put in different clusters in both clusterings;
- c is the number of point pairs that are put in the same cluster in $\mathcal{C}_1$, but in different clusters in $\mathcal{C}_2$;
- d is the number of point pairs that are put in different clusters in $\mathcal{C}_1$, but in the same cluster in $\mathcal{C}_2$.

5.4 Description of experiments

5.4.1 Pulser experiment

The Pulser neutral atom computer simulator was used to test the aggregation protocol on the dataset discussed in 5.2;

Two different clustering algorithms were run on the dataset, DBSCAN and Spectral Clustering. Hyperparameters were tuned empirically, in order to ensure an overall amount of clusters inferior or equal to 14, so as not to exceed the amount of qubits the simulator can handle [8].

Chapter 6

Results

6.1 Fresnel experiment

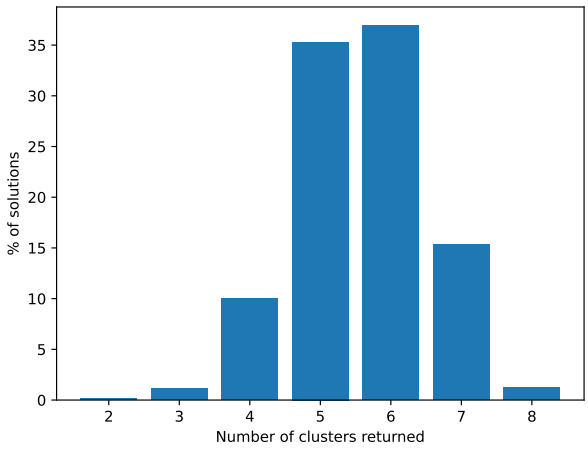

Figure 6.1: Histogram for probability of number of clusters returned in Fresnel experiment

Number of clusters	Probability (%)
2	0.12
3	1.11
4	10.0
5	35.31
6	36.91
7	15.31
8	1.23

Table 6.1: Probability of number of clusters returned in Fresnel experiment

Chapter 7

Conclusions

List of Figures

5.1 Dataset plot 12

6.1 Histogram for probability of number of clusters returned in Fresnel experiment 15

List of Tables

6.1 Probability of number of clusters returned in Fresnel experiment . 15

Bibliography

[1] David Deutsch. "Quantum theory, the Church–Turing principle and the universal quantum computer". In: *Proceedings of the Royal Society of London. A. Mathematical and Physical Sciences* 400.1818 (1985), pp. 97–117.

[2] Edward Farhi, Jeffrey Goldstone, and Sam Gutmann. *A Quantum Approximate Optimization Algorithm*. 2014. arXiv: 1411.4028 [quant-ph]. URL: https://arxiv.org/abs/1411.4028.

[3] Richard P. Feynman. "Quantum Mechanical Computers". In: *Between Quantum and Cosmos*. Princeton University Press, Dec. 2017, 523–548. DOI: 10.1515/9781400886975-036. URL: http://dx.doi.org/10.1515/9781400886975-036.

[4] Richard P. Feynman. "Simulating physics with computers". In: *International Journal of Theoretical Physics* 21.6–7 (June 1982), 467–488. ISSN: 1572-9575. DOI: 10.1007/bf02650179. URL: http://dx.doi.org/10.1007/BF02650179.

[5] Aristides Gionis, Heikki Mannila, and Panayiotis Tsaparas. "Clustering aggregation". In: *ACM Trans. Knowl. Discov. Data* 1.1 (2007), 4–es. ISSN: 1556-4681. DOI: 10.1145/1217299.1217303. URL: https://doi.org/10.1145/1217299.1217303.

[6] Rafael C. Gonzalez and Richard E. Woods. *Digital Image Processing, Global Edition*. en. 4th ed. London, England: Pearson Education, Sept. 2017, pp. 770–772.

[7] A. K. Jain, M. N. Murty, and P. J. Flynn. "Data clustering: a review". In: *ACM Comput. Surv.* 31.3 (1999), 264–323. ISSN: 0360-0300. DOI: 10.1145/331499.331504. URL: https://doi.org/10.1145/331499.331504.

[8] J.R. Johansson, P.D. Nation, and Franco Nori. "QuTiP: An open-source Python framework for the dynamics of open quantum systems". In: *Computer Physics Communications* 183.8 (Aug. 2012), 1760–1772. ISSN: 0010-4655. DOI: 10.1016/j.cpc.2012.02.021. URL: http://dx.doi.org/10.1016/j.cpc.2012.02.021.

[9] Ivan Kassal et al. "Simulating Chemistry Using Quantum Computers". In: *Annual Review of Physical Chemistry* 62.1 (May 2011), 185–207. ISSN: 1545-1593. DOI: 10.1146/annurev-physchem-032210-103512. URL: http://dx.doi.org/10.1146/annurev-physchem-032210-103512.

[10] Phillip Kaye, Raymond Laflamme, and Michele Mosca. *An Introduction to Quantum Computing*. Oxford University Press, Nov. 2006, pp. 38–51. ISBN: 9780191916724. DOI: 10.1093/oso/9780198570004.001.0001. URL: http://dx.doi.org/10.1093/oso/9780198570004.001.0001.

[11] Nan Li and Longin Jan Latecki. "Clustering aggregation as Maximum-Weight Independent Set". In: *Proceedings of the 25th International Conference on Neural Information Processing Systems - Volume 1*. NIPS'12. Lake Tahoe, Nevada: Curran Associates Inc., 2012, 782–790.

[12] Francesco Marra et al. "Blind PRNU-Based Image Clustering for Source Identification". In: *IEEE Transactions on Information Forensics and Security* 12.9 (2017), pp. 2197–2211. DOI: 10.1109/TIFS.2017.2701335.

[13] John Preskill. "Quantum Computing in the NISQ era and beyond". In: *Quantum* 2 (Aug. 2018), p. 79. ISSN: 2521-327X. DOI: 10.22331/q-2018-08-06-79. URL: http://dx.doi.org/10.22331/q-2018-08-06-79.

[14] William M. Rand. "Objective criteria for the evaluation of clustering methods". In: *Journal of the American Statistical Association* 66.336 (1971), pp. 846–850.

[15] Peter J. Rousseeuw. "Silhouettes: A graphical aid to the interpretation and validation of cluster analysis". In: *Journal of computational and applied mathematics* 20 (1987), pp. 53–65.

[16] Peter W. Shor. "Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer". In: *SIAM Journal on Computing* 26.5 (Oct. 1997), 1484–1509. ISSN: 1095-7111. DOI: 10.1137/s0097539795293172. URL: http://dx.doi.org/10.1137/S0097539795293172.

[17] Anne Solberg, Torfinn Taxt, and Anil Jain. "A Markov Random Field Model for Classification of Multisource Satellite Imagery". In: *Geoscience and Remote Sensing, IEEE Transactions on* 34 (Feb. 1996), pp. 100 –113. DOI: 10.1109/36.481897.

[18] Alexander Strehl, Joydeep Ghosh, and Claire Cardie. "Cluster ensembles—A knowledge reuse framework for combining multiple partitions". In: *Journal of Machine Learning Research* 3.Dec (2002), pp. 583–617.

[19] Salvador Vega-Pons and Jose Ruiz-Shulcloper. "A survey of clustering ensemble algorithms". In: *International Journal of Pattern Recognition and Artificial Intelligence* 25.03 (2011), pp. 337–372.