

quantum Case-Based Reasoning (qCBR)

Parfait Atchade Adelomou,^{1,2} Daniel Casado Fauli,¹ Elisabet Golobardes Ribé,¹ and Xavier Vilasis-Cardona¹

¹*Research Group on Data Science for the Digital Society La Salle - Universitat Ramon Llull Carrer de Sant Joan de La Salle, 42 08022 Barcelona (Spain) **

²*Lighthouse Disruptive Innovation Group, LLC 7 Broadway Terrace, Apt 1 Cambridge MA 02139 Middlesex County, Massachusetts (USA)*

(Dated: March 2021)

Case-Based Reasoning (CBR) is an artificial intelligence approach to problem-solving with a good record of success. This article proposes using Quantum Computing to improve some of the key processes of CBR, such that a Quantum Case-Based Reasoning (qCBR) paradigm can be defined. The focus is set on designing and implementing a qCBR based on the variational principle that improves its classical counterpart in terms of average accuracy, scalability and tolerance to overlapping. A comparative study of the proposed qCBR with a classic CBR is performed for the case of the Social Workers' Problem as a sample of a combinatorial optimization problem with overlapping. The algorithm's quantum feasibility is modelled with docplex and tested on IBMQ computers, and experimented on the Qibo framework.

KeyWords: Quantum Computing, Machine Learning, Case-Based Reasoning, Quantum Case-Based Reasoning, Artificial Intelligent, VQC, Variational Quantum Classifier,

I. INTRODUCTION

The social workers' problem (SWP) stands for solving the schedules of social workers visiting patients' homes while fitting both distance and time restrictions [1] and represents a class of combinatorial optimization problems, which lie in the NP complexity class. The standard way to solve this class of problems begins by establishing the cost function. Then, depending on its form, existing linear or quadratic programming methods such as Simplex [2] or Cplex [3] can be applied. More complex cost functions require more sophisticated numerical methods. Depending on the problem's complexity class, the algorithm can be improved by introducing some heuristics or restrictions in the objective function to reduce its computational cost for an approximate solution. When the size of the problem grows, the computational cost may soon become intractable for the current computational paradigms. In addition to the above, solving these problems is more challenging when the input data presents some overlapping issues or when outstanding accuracy is required.

In this paper, an approach combining adapting Case-Based Reasoning[4] to a quantum computing is proposed to solve this class of problems. This paradigm, denoted Quantum Case-Based Reasoning (qCBR), will address both the overlap in the input data and the accuracy problem. Furthermore, by directly constructing the framework, questions like the actual efficiency of a qCBR implementation at the present level of quantum technology, the tolerance concerning input overlap, the scalability and the applicability to other combinatorial optimization problems will be discussed. The paper is organized as follows. Section II, shows previous work on both

ensemble techniques and quantum machine learning approaches. In Section III, the Case-Based Reasoning will be explored by focusing on its features. Section IV introduces the necessary quantum fundamentals for this era to solve this problem. Implementing the proposed strategy and creating the quantum CBR (qCBR) is done in Section V. In Section VI, we present our experimental analysis results. In Section VII, we summarise, benchmark and present some open problems. Finally, Section VIII concludes previous results and outlines future work.

II. RELATED WORK

CBR is a problem solving approach widely considered in the literature with a large record of success. Application examples are a medical reasoning program that improves with experience [5], an individual prognosis of diabetes long-term risks[6], Case-Based Sequential ordering of songs for playlist recommendation [7], ranking order in financial distress prediction [8], monitoring the elderly at home [5], software control [4], in the medical field[4], sequencing problems [9] etc. In one of our previous works [1, 10], we observed part of the benefits of using the CBR instead of the Top-Down method. And the needs of empowering this problem-solving method based on human learning were seen.

Quantum computing stands as a new computing paradigm based on exploiting the principles of quantum mechanics and establishing the quantum bit (qubit) as the elementary unit of information. It emerged in the early 1990's from algorithms that were able to take advantage of quantum characteristics to show advantages over their classical counterparts, being Shor's algorithm [11] for integer factorization and Grover's algorithm [12] for searching in an unordered data sequence, the most famous. However, current quantum computing devices

* parfait.atchade@salle.url.edu

suffer from technological limitations, such as the number of qubits available and the noise and decoherence problems, such that they are still no match for their classical counterparts. This situation is known as the Noisy Intermediate-Scale Quantum (NISQ) era [13]. These limitations have forced the scientific community to develop handy tools for hybrid computing, mixing classical and quantum. Taking advantage of the variational principle, it is possible to solve combinatorial optimization problems and enhance one of this era's most promising fields; quantum machine learning (QML) [14, 15]. In this new approach, several techniques and methods already explored in Machine Learning (ML) are being worked on.

In the last two years, the number of algorithms based on QML have increased considerably since the first definition in 2014 [14]. This progress relies on the advances in decoherence control [16, 17] and error correction systems [18] combined with the availability of several quantum server providers in the cloud. Most of these new algorithms take after the variational principle, being the Variational Quantum EigenSolver (VQE) [19] and the Quantum Approximate Optimization Algorithm (QAOA) [20–22] the most famous. Other promising developments are the Quantum Neural Network (QNN) [23–25], the Quantum Support Vector Machine (QSVM) [26–28] and the data loading system [29, 30]. On the one hand, the following references [31, 32] highlight works done in the Top-Down philosophy. On the other hand, references [33–38] highlight the many contributions in quantum machine learning, from using the properties of quantum computing to finding new drugs as new ways to calculate the expected value, among others.

The literature shows examples of exploiting the possibilities of hybrid (classical-quantum) computing connecting it to CBR. For instance, in reference [39] a cognitive engine that uses CBR-QGA to adjust and optimize the radio parameters is presented. An initial quantum bit made up of the matching case parameters is used to avoid blindness of the initial population search and speed up optimization of the quantum genetic algorithm. References [40, 41] propose a new framework that can be adopted in many applications that require Computational Intelligence (CI) solutions. The framework is built under the concepts of Soft Computing (SC), where Fuzzy Logic (FL), Artificial Neural Network (ANN) and Genetic Algorithm (GA) are exploited to perform reasoning tasks based on soft cases. Also studies [42] focused on some vital blocks of the CBR were reviewed. It has focused on the quantum version of the k-NN algorithm that allows us to understand the fundamentals when transcribing classic machine learning algorithms into their quantum versions.

Reviewing state of the art, we have seen an interesting field known as Quantum Information Retrieval (QIR) [43–45] that uses the Gleason theorem [46] on the Measures on the Closed Subspaces of a Hilbert Space for information retrieval geometry [47]. It calculates the probability algebraically through the density matrix trace and

acts on a quantum projector. The projector can be any concept to recover. However, for the quantum CBR, we are not only interested in a great recovery system, but we also need to provide the qCBR with a synthesiser whose function will be to fine-tune the recovered data in the case of not being the optimal result since the qCBR has the process of "generate" a new outcome based on the retrieved information.

However, no quantum Case-Based Reasoning was found that can satisfy the requirements presented above. Such is the purpose of this paper.

III. CASE-BASED REASONING

CBR [4] is a machine learning technique based on solving new problems using experience, as humans do. The *experience* is represented as a case memory containing previously solved cases. The CBR cycle can be summarised in four steps: (1) Retrieval of the most similar cases, (2) Adaptation to those cases to propose a new solution to the new environment, (3) Validity check of the proposed solution and finally, (4) Storage following a learning policy. In the present work, the proposed qCBR modifies these phases as follows (see Fig.(1) and (3)).

The CBR technique could be summarized in two large blocks according to their functionality: a classifier and a synthesizer. One of the classical CBR advantages is its classifier's simplicity, being a k-nearest neighbors algorithm (K-NN) [50, 51] classifier a common option. This apparent advantage can lead to collateral problems [52] at the memory level, at the level of slowness when the volume of data grows considerably and at data synthesis. The synthesis block is in charge of adapting the experience and saving the new problem. Such adaptation

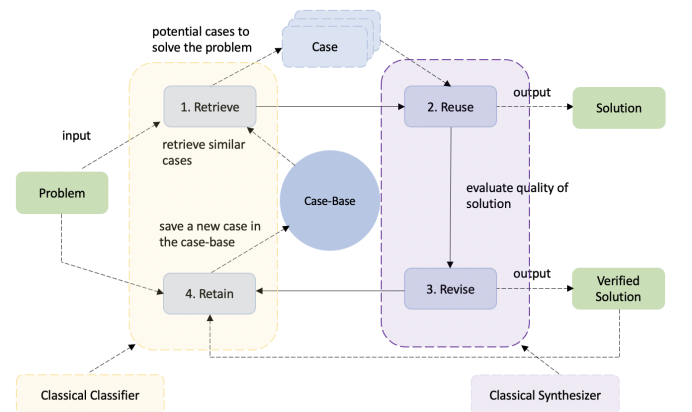


FIG. 1. Case-Based Reasoning block diagram. In this work, for the standard CBR, two essential blocks are distinguished: The classifier and the synthesizer. The classifier is made up of the retrieve and retain blocks and the re-use and revise blocks to make up the synthesis system.

Methods	Brute force	k-d tree method[48]	Ball tree method[49]
Training time complexity	$O(1)$	$O(dN\log(N))$	$O(dN\log(N))$
Training space complexity	$O(1)$	$O(dN)$	$O(dN)$
Prediction time complexity	$O(KNd)$	$O(K\log(N))$	$O(K\log(N))$
Prediction space complexity	$O(1)$	$O(1)$	$O(1)$

TABLE I. Table of the NN brute force's, k-d tree's, and Ball tree's complexity method. Where d , is the data dimensionality, N is the number of points in the training dataset and K is the algorithm's neighbours' number.

and classification can be costly (Table I) for considerably high data volumes [53]. From this follows that a different approach would be required to further empowering this technique.

The proposal of this note is to achieve such empowering in two steps. First by making a CBR with a quantum classifier [54] instead of a classical neural network, KNN[50, 51] or a Support Vector Machine (SVM) [55] since quantum classifiers offer outstanding accuracy and tolerate overlapping problems [56]. The second would be changing the classical synthesis technique for the Variational Quantum Eigensolver (VQE) [57–59] with *Initial_point*[60].

IV. QUANTUM CIRCUITS IN THE NISQ ERA

Quantum circuits are mathematically defined as operations on an initial quantum state. Quantum computing generally makes use of quantum states built from qubits, that is, binary states represented as $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$. Their number of qubits n commonly defines the states of a quantum circuit and, in general, the circuit's initial state $|\psi\rangle_0$ is the zero state $|0\rangle$. In general, a quantum circuit implements an internal unit operation U to the initial state $|\psi\rangle_0$ to transform it into the final output state $|\psi\rangle_f$. This gate U is wholly fixed and known for some algorithms or problems. In contrast, others define its internal functioning through a fixed structure, called Ansatz[61] (Parametrized Quantum Circuit (PQC)), and adjustable parameters θ [54]. Parameterized circuits are beneficial and have interesting properties in this quantum age since they broadly define the definition of ML and provide flexibility and feasibility of unit operations with arbitrary precision [14, 15, 62].

Figure (2) depicts the concept of hybrid computing (quantum + classical), which defines the NISQ. This takes advantage of quantum computing's capacity to solve complex problems, and the experience of classical optimization algorithms (COBYLA [63], SPSA [64], BFGS [65], etc.) to train variational circuits. Classical algorithms are generally an iterative scheme that searches for better candidates for the parameters θ at each step.

The value of the hybrid computing idea in the NISQ

era is necessary because it allows the scientific community to exploit the powers of both and reap the benefits of the constant acceleration of the oncoming quantum-computer development. With a good optimization system and a closed-loop system, the non-systematic noises could be automatically corrected during the optimization process.

Furthermore, with the insertion of information (data) into the variational circuit through the quantum gate U , learning techniques can be improved.

The Variational Quantum Circuit (VQC) [66, 67], consists of a quantum circuit that defines the base structure similar to neural network architecture (Ansatz) while, the variational procedure can optimize the types of gates (one or two-qubit parametric gates) and their free parameters. All this is summarized in a few very identifiable steps. First, the Ansatz must be designed, using a set of one- and two-qubit parametric gates. The Ansatz of this circuit can follow a particular path by exploiting the problem's characteristics. A critical block is measuring the quantum state resulting from the given Ansatz. Since the VQC is a feedback system, these measurements evaluate a cost function $CF(\theta)$ (see in the Appendix A 1) that encodes the problem. The classical optimizer has the role of optimizing the cost function to find the value of the parameters that minimize it.

V. IMPLEMENTATION

The work proposed in this article is the implementation of a quantum Case-Based Reasoning (qCBR) based on figure (1). The strategy to follow is to replace the classical classification: an Artificial Neural Network (ANN) or a Support Vector Machine (SVM) or the KNN with a quantum variational classifier that guarantees the required accuracy. And for the quantum synthesis system, use the VQE with and without *Initial_point* together with a probabilistic decision tree. Figure (3) shows the changes that will be introduced to obtain the qCBR, and figure (10) shows the detail of the functional blocks implemented with the specific problem of social workers. The two VQE blocks and the Variational Quantum Classifier are presented before detailing them.

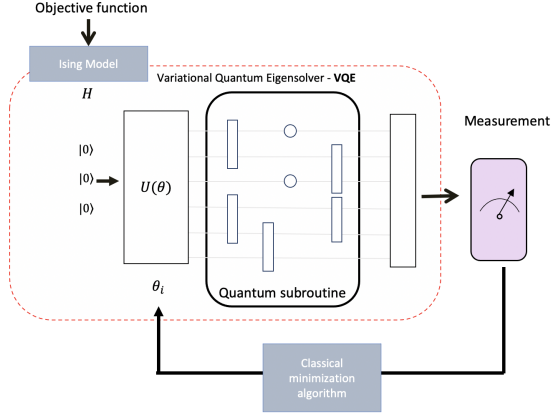


FIG. 2. VQE working principle based on the quantum variational circuit. Given an objective function that characterises the problem, with the help of the Ising Model block, we pass the objective function from the classical to the quantum domain. The ansatz is initialised with random values. Then, starting from these initial values (initial position) and depending on the measured value, a classical and external optimiser is used to feedback the new values of the ansatz parameters. So, until reaching the minimum energy value, equivalent to the ground state of our Hamiltonian, defined by the variational principle.

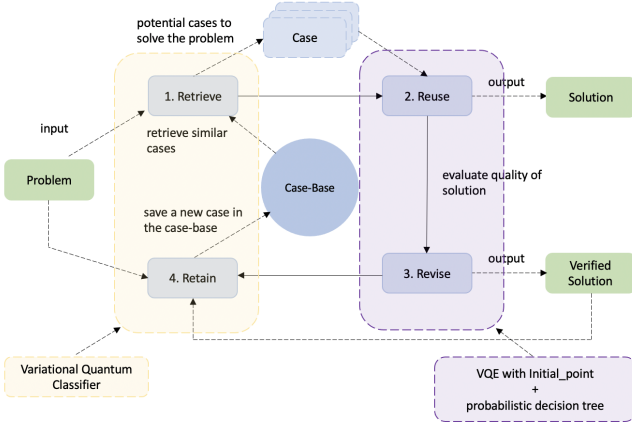


FIG. 3. Quantum Case-Based system block diagram. In this scheme, to convert the classic CBR into the quantum one, it is proposed to change the retrieve and retain blocks for a quantum variational classifier and the re-use and revise blocks for a synthesis system based on VQE with initial_point.

1. Variational Quantum Eigensolver

VQE (figure (4)) is a classical hybrid quantum algorithm that combines aspects of quantum mechanics with the classical algorithm, and its objective is to find approximate solutions to combinatorial problems. One of the fundamental approaches is to map combinatorial problems into a physics problem. That is, about a problem

that can be formulated in terms of a Hamiltonian Ising model. Therefore, the identification of the solution to the combinatorial problem is linked to finding the ground state of this physics problem. As a result, the goal is to find the ground state of this Hamiltonian. The unknown eigenvectors are prepared by varying the experimental parameters and calculating the Rayleigh-Ritz ratio [68] in a classical minimization, figure (4). At the end of the algorithm, the reconstruction of the eigenvector that is stored in the final set of experimental parameters that define the state would be performed.

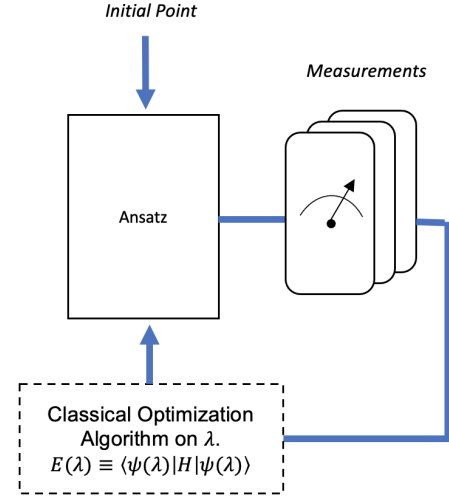


FIG. 4. We start from an initial position of the initial values of the Ansatz parameters. Normally we start from initial equiprobable values between all qubits. Then, depending on the measured value, a classical and external optimizer is used to feedback the new values of the ansatz parameters. So on until reaching the value of the minimum energy equivalent to the ground state of our Hamiltonian as defined by the variational principle. If we want to start the optimization from a point other than the initial state, we can load some positions (Initial point) equivalent to some energy values on the ansatz. In the case of qCBR, the key is that having saved the initial point values of each schedule already calculated will save optimization time since it will be possible to start from a fairly optimal point.

From the variational principle, the following equation $\langle H \rangle_{\psi(\vec{\theta})} \geq \lambda_i$ can be reached, with λ_i as eigenvector and $\langle H \rangle_{\psi(\vec{\theta})}$ as the expected value. This way, the VQE finds (1) as an optimal choice of parameters $\vec{\theta}$, that the expected value is minimized and that a lower eigenvalue is located.

$$\langle H \rangle = \langle \psi(\theta) | H | \psi(\theta) \rangle \quad (1)$$

The VQE is used here with and without *Initial_point*[60] for the synthesis task. In the detailed explanation of the work, more detail will be given. The *Initial_point* is

the starting point (initial parameter values) for the optimizer. Without this starting point, the VQE will search the ansatz for a preferred point, and if not, it will just calculate a random one. This possibility is essentially useful, such as when there are reasons to believe that the outcome position is close to a particular spot. Furthermore, in the qCBR, this *Initial_point* will be tremendously useful for reshaping the retrieved solution if the latter is not the most optimal.

2. Quantum classifiers

The variational quantum classifier belongs to the variational algorithms like VQE, where classically tunable parameters of a unit circuit are used to minimize the expected value of an observable. The great novelty resides in loading the data in the variational system.

We have designed a classifier that emulates neural networks solving the function $Wx + b$, with W and b the parameters and x , the sample data to be classified. The non-linearity of the quantum gates is used to implement the activation function $f(Wx + b)$ given $Wx + b$. The figure (5) provides us with the block diagram of the classifier. The optimization and parameters' (W, b) actualization are done in the first step, MSE between (\bar{y} and y), where y is the label associated with x and k of the labels.

The detailed operations of the classifier are given by the figure (6) where the quantum gates, R_y , R_x and C_{RZ} are used to define the block. In our, the optimization and parameters' (W, b) actualization are done through the fidelity cost between $\alpha_{c,q} F_{c,q}(\vec{\theta}, \vec{\omega}, \vec{x}_\mu)$ and $Y_c(\vec{x}_\mu)$, where \vec{x}_μ are the training points and $\vec{\alpha} = (\alpha_1, \dots, \alpha_C)$ are introduced as class weights to be optimized together with $\vec{\theta}$, $\vec{\omega}$, are the parameters and Q the numbers of the qubits. Counting on $Y_c(\vec{x}_\mu)$ as the fidelity vector for a perfect classification and $F_{c,q}(\vec{\theta}, \vec{\omega}, \vec{x}_\mu) = \langle \psi_c | \rho_q(\vec{\theta}, \vec{\omega}, \vec{x}) | \psi_c \rangle$.

Since a universal quantum classifier of n qubits is needed for the purpose of this paper, see figure (5), a sub-base in the Hilbert vector space of equitably dividing the hyperplane Z is described as follows. Let $B = (\{i, j, k, l, m, n, o, p\})$ be a sub-base within the Hilbert vector space, for the space of the classes C^{2^q} , the coordinates of the target classes are defined by expression (2) with q as the number of the qubits.

$$\begin{aligned} &\{i(1,0,0,0,0,0,0,0); j(0,1,0,0,0,0,0,0); k(0,0,1,0,0,0,0,0); \\ &l(0,0,0,1,0,0,0,0); m(0,0,0,0,1,0,0,0); n(0,0,0,0,0,1,0,0); \\ &o(0,0,0,0,0,0,1,0); p(0,0,0,0,0,0,0,1)\} \end{aligned} \quad (2)$$

The Ansatz design and data loading (variables x_i similar to neural networks)[29] are given by equation (3), and its analysis is detailed in A 1.

$$U = (\theta, x) = R_x(\theta_1 x + \theta_2) R_z(\theta_3) \quad (3)$$

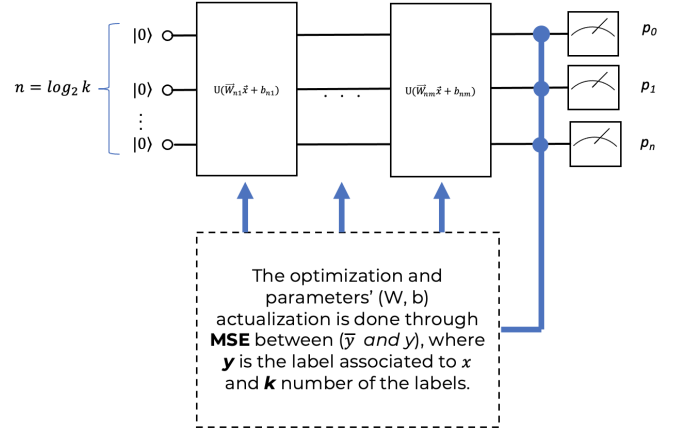


FIG. 5. This is the variational classifier's diagram block used in the qCBR, we use the data re-uploading technique to create an n -dimensional classifier as if it were a neural network where the non-linearity of the quantum gates will act as an activation function, and we will use the model $y = Wx + b$.

3. Memory Structure

Next, some test benches based on the memory structure described in figures (7) and (8) are defined to train the parameterized quantum circuit, and its performance is analyzed in terms of the circuit architecture. The results, discussions and annexe sessions will emphasize the classifier with or without entanglement and a comparative study with different ansatzes.

The memory structure of the qCBR's retention system is given in figure (7). The solution class (target) corresponds to the paths each Social Worker will take between the different patients for a specific schedule, representing these paths as an adjacency matrix, such as:

$$SOL_{SWP} = \begin{pmatrix} 0 & x_{0,1} & x_{0,2} \\ x_{1,0} & 0 & x_{1,2} \\ x_{2,0} & x_{2,1} & 0 \end{pmatrix} \quad (4)$$

where $x_{i,j}$ is a binary variable, the rows of the matrix represent the origin node and the column the destination node of the path.

Each solution class is represented as a label (e.g., 'A') and is related to the different initial points associated with each of the samples that make up the training dataset can be seen. This solution class is also associated with the result of the VQE.

A. The qCBR solving the Social Workers' Problem

A real problem already developed in these references [1][70] is used to correctly test the proposed and im-

distance function because one wants to make g_{ij} behave like d_{ij} , and thus be able to take full advantage of the initial objective function's behaviour.

Let ε be positive and represent a weighted degree parameter of the time window function; τ_i is the starting worker time of a slot of time for patient i and τ_j for the patient j . With d_{\max} as the maximum distance between all patients and d_{\min} the minimum one. Hence, let us define the non-negative time windows $T_{ij} = (\tau_i - \tau_j) > 0$.

To fill out the data structure created in the classifier to train and test its predictions, the parameters of the *Initial_point* obtained by VQE are abstracted from the result. Furthermore, it is composed of each class's coordinates with the following parameters: start time sT and end time eT of patient 1 to n , where n is the maximum number of patients in the app. Finally, figure (9) summarizes the data's representation and description that make up the training dataset.

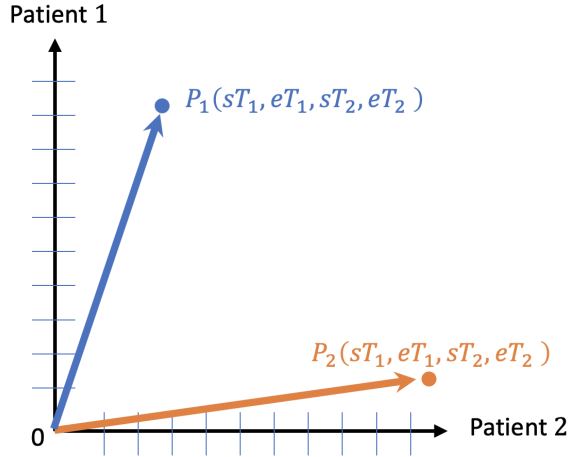


FIG. 9. The SWP represented in vector form to take advantage of the Hilbert vector space's characteristics within quantum computing. It is seen that each patient represents a dimension and the points that make up the dataset a dimension of $2n$ coordinates, n being the total number of patients. In this figure, to simplify the understanding, we use two patients, therefore, two dimensions.

Each coordinate's class corresponds to the VQE solution following the memory structure in figure (7). Where the class number of the classifier is given by equation (6) taking into account the conditions that every worker has a patient and that the workers are indistinguishable (that is, it doesn't matter whether the social worker m_1 takes care of the patient n_1 and m_2 takes care of n_2 or vice versa).

$$N_{SOL_{SWP}} = \frac{1}{m!} \sum_{k=0}^{m-1} (-1)^k \binom{m}{m-k} (m-k)^n \quad (6)$$

With n the number of patients, m the number of social workers and, knowing that the appearance of patients is

ordered in the schedule (from oldest to most recent), n_1 will be the patient with the first schedule and n_k the patient with the last one.

In this article all the tests done are for $n = 4$ with the data structure equal to $(sT_1, eT_1, sT_2, eT_2, sT_3, eT_3, sT_4, eT_4)$; an 8-dimensional vector for each social worker visit the patient. In this case, the number of qubits will be defined by $q = \log_2(N_{SOL_{SWP}}) = 3$. These qubits are used to instance the quantum classifier, and it is worth to mention that the classifier must have $N_{SOL_{SWP}}$ classes.

The detail of the qCBR's implementation and analysis is in the Appendix B.

VI. RESULTS

When testing the classifier, a section of the sample database, schedules previously solved by the VQE algorithm to obtain its corresponding true solution (*ground truth*), was used as test samples (applying "Leave-one-out" cross-validation). Then, the total accuracy of the classifier predictions was obtained based on the ratio between the number of labels predicted correctly and the total number of labels.

Figures (11) to (12) show the implementation outcomes performed in *qibo*[73] and *qiskit*[60, 74] to identify the best model architecture and represent functions similar to qCBR.

Tables (II) to (IV) show the global results of qCBR solving the SWP. In table (II), the outcome of the different tested scenarios can be observed. Varying the number of patients, social workers, and the quantum circuit's depth to see the global hit number of the qCBR. In table (III), we can observe the resolution of the SWP, considering five patients, four social workers and setting the depth of the quantum circuit to eight. Through this scenario, the behaviour of the qCBR can be observed considering the number of cases carried out. It can be seen how the system begins to give more than satisfactory results after exceeding the threshold of the 240 results stored in the case memory. Table (IV) repeats the steps of table (III) with the only change of the input data; the number of patients and social workers. Tables (V) to (VIII) show the result of the implementation of the classical CBR leveraged on ANN and KNN to solve the SWP.

Tables (III) and (IV) represent the outcomes of the qCBR and show better results than the ones obtained with the classical CBR (tables (V) and (VI)). Tables (VIII), (VII) and (II) show the degree of scalability of the qCBR as a function of the variation in the number of patients and social workers. It has also been seen that qCBR is much better shared with overlapping as we wanted to demonstrate.

Also, we experimented by skipping the Principal Component Analysis (PCA) module [72][75], Independent Component Analysis (ICA) [71] and creating a classifier of the same dimension as the data (8 dimensions).

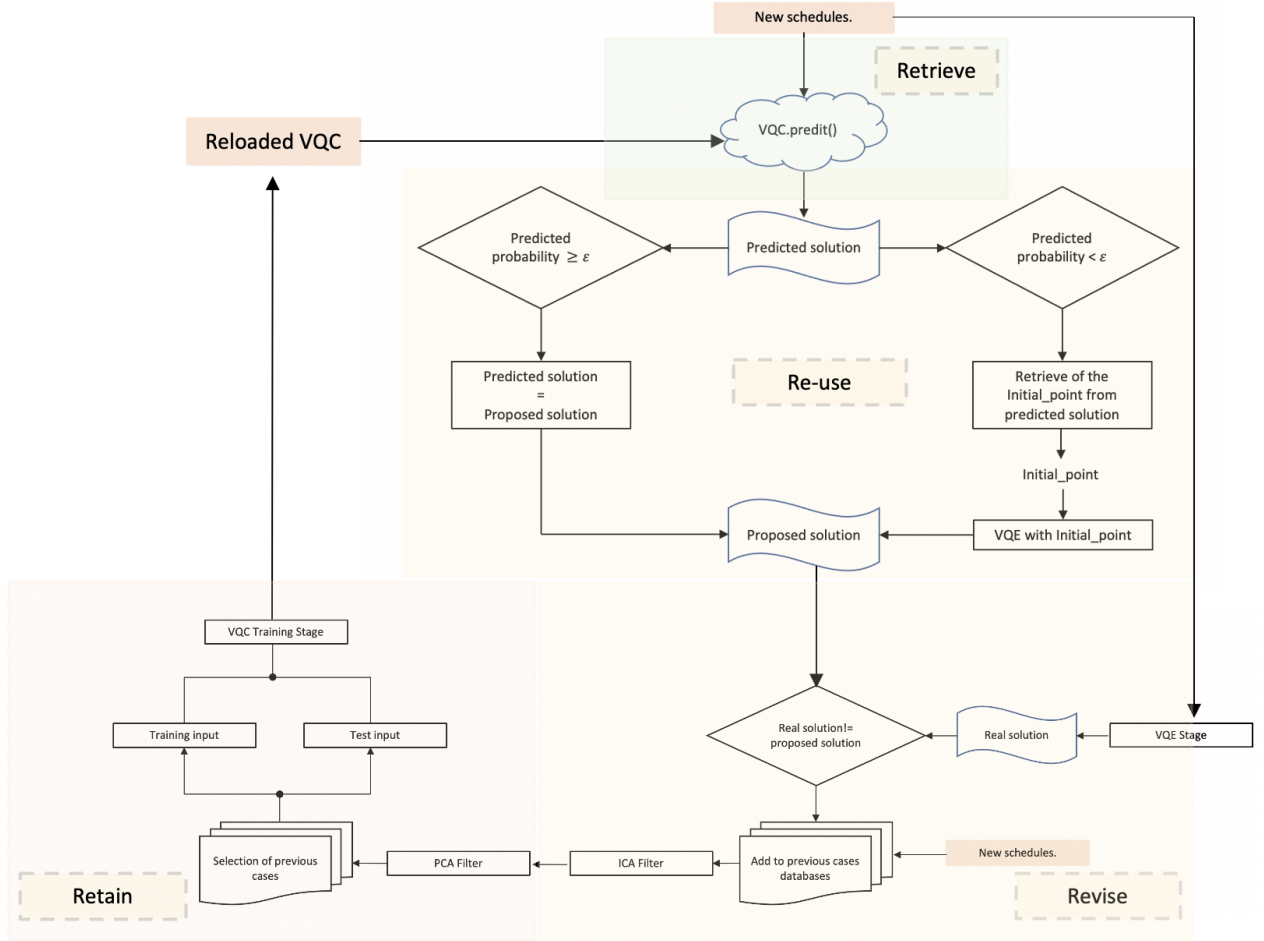


FIG. 10. Block diagram proposed for the resolution of qCBR, considering a real dataset with an overlap problem between the data components. This block diagram contemplates the treatment of the input data and the use of ICA[71] and PCA[72] before training and classifying the data. In this version, a classifier based on the re-uploader has been designed to be in charge of the classification tasks. And for the synthesis tasks, a decision tree passed in the classifier predictions have been used together with the VQE plus the *Initial_point*.

The results obtained have been very satisfactory at the Ansatz's accuracy and depth level. Still, the need to change the *BFGS* [65] optimizer to the *SPSA* [64] has become visible due to its slow convergence for the number of data and high parameters. Figure (11) describes the behaviour and compare the two scenarios.

Later the Re-use module was analyzed using VQE with *Initial_point* to synthesize the predicted results. In the graph shown in figure (12), it is observed how the algorithm, without initial parameters, tends to use a high energy constant of variation to quickly reach an approximation of the fundamental state. Which makes it have to progressively, after several iterations, n , reduce said constant to find the local minimum. On the other hand, when using an *Initial_point*, the algorithm does not need to start with a high variation to reach energy bands close to the ground state since it is much closer to said energy,

reducing the number of iterations necessary reach to the local minimum. We can then see how qCBR can afford to run VQE with *Initial_point* to refine the accuracy of its results since it requires fewer iterations to find the solution closest to the minimum, not assuming such a high computational cost as it would be running VQE without initial parameters.

The qCBR complexity (Table (IX)) is provided below where it can be seen that the *Retain* is the highest cost operation and has an exponential improvement compared to a *Retain* of a classic CBR that is usually of the order of $O(M^2(M + N))$ [26]

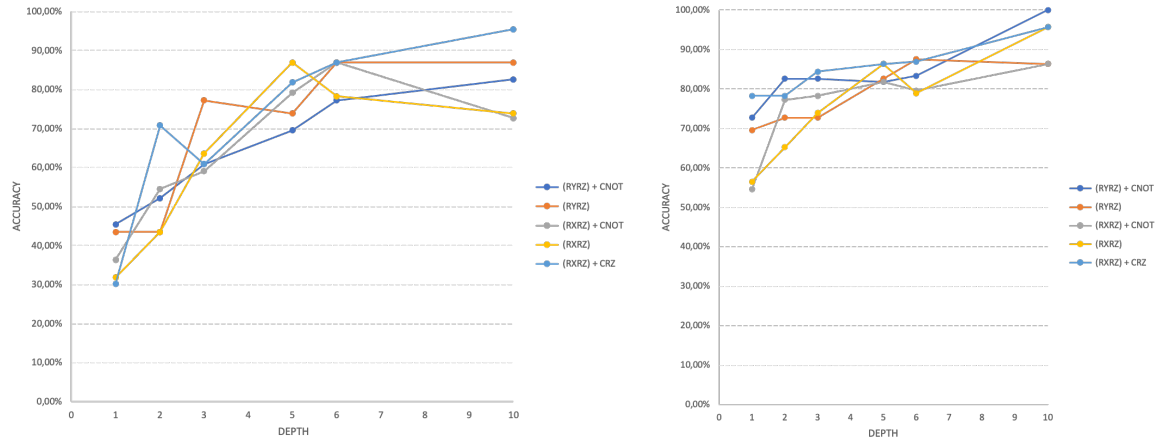


FIG. 11. Comparative graphs between different ansatzes, taking into account the classifier's accuracy as a function of depth. Represents the evolution of the ansatzes of dimension two and eight.

qCBR solving the Social Workers' Problem					
#Patients	#SW	#Qubits	#Layers	#Cases	Accuracy
3	2	6	2	580	82.5
4	3	12	3	580	82
5	2	20	4	580	82.5
5	3	20	5	580	87
5	4	20	8	580	92.8
5	4	20	10	580	100

TABLE II. The result of the qCBR with a variational classifier and using the VQE and the Initial_point with some decision trees as a synthesizer (10). This table shows the different studies made as a function of the quantum circuit's depth (layer). number of the patients and the social workers. The accuracy of the classifier is the maximum with the number of layers equal to 10. SW denotes Social Workers.

VII. DISCUSSIONS

Firstly, the proposed qCBR works very well and meets the objectives set using quantum computing to create efficient quantum Case-Based Reasoning. One of the issues to comment on is the improvement observed in figure (11) with respect to the 2 and 8-dimensional classifiers. Due to the small number of depths, but with many more parameters, the 8-dimensional classifiers have an average of about 25% of improvements over the 2-dimensional ones. With this result, in the case of not wanting an accuracy of around 95%, shallow depth could be used, and computation time saved, depending on the problems. Despite all these improvements, it is essential to highlight some

qCBR solving the Social Workers' Problem		
For 5 patients and 4 socials workers		
Layers	#Cases	Accuracy
8	20	-
	50	12.5
	100	72.5
	240	92.1
	340	95.5
	480	97.2
	500	98.7
	580	99.1

TABLE III. The result of the qCBR for a number of patients and social workers fixed at 5 and 4, respectively. The better behaviour of qCBR can be observed for some cases greater than 240. To have a good functioning of the qCBR, it must be iterated with the social workers' dataset one 239 times. And at the case number of 240, we will have an accuracy of 92%. The "accuracy" value is the percentage of the number of correct solutions found by the qCBR. A hyphen (-) denotes that no solution was found within the 20 cases. All these tests were done for the quantum circuit depth (layers) equal to 8.

aspects to refine. In the intelligent system that allows deciding the proposed solution, now, the average of the *Initial_point* of each solution class samples' *Initial_point* is used. It could still be seen based on the predicted solution, which *Initial_point* is the most suitable for the solution to propose. Thus, the cases to be re-used could

qCBR solving the Social Workers' Problem		
For 4 patients and 3 socials workers		
Layers	#Cases	Accuracy
8	20	-
	50	11.5
	100	73.1
	240	91.1
	340	91.9
	480	96.6
	500	98.1
	580	99.0

TABLE IV. The result of the qCBR for a number of patients and social workers fixed at 4 and 3, respectively. The better behaviour of qCBR can be observed for some cases greater than 240. To have a good functioning of the qCBR, it must be iterated with the social workers' dataset one 239 times. And at the case number of 240, we will have an accuracy of 91%. The "accuracy" value is the percentage of the number of correct solutions found by the qCBR. A hyphen (-) denotes that no solution was found within the 20 cases. All these tests were done for the quantum circuit depth (layers) equal to 8.

CBR with KNN solving the Social Workers' Problem		
For 5 patients and 4 socials workers		
Layers	#Cases	Accuracy
1	20	-
	50	42.9
	100	46.5
	240	52.6
	340	55.3
	480	56.8
	500	60.7
	580	63.1

TABLE V. The classical CBR result on KNN classifier for a number of patients and social workers fixed at 5 and 4, respectively. The better behaviour of this CBR can be observed for some cases greater than 240. To have a good functioning of the CBR, it must be iterated with the social workers' dataset one 239 times. And at the case number of 240, we will have an accuracy of 52.6%. The "accuracy" value is the percentage of the number of correct solutions found by the CBR leveraged on KNN, applying a 10-KFold cross-validation process. A hyphen (-) denotes that no solution was found within the 20 cases. All these tests were done for the layer equal to 1.

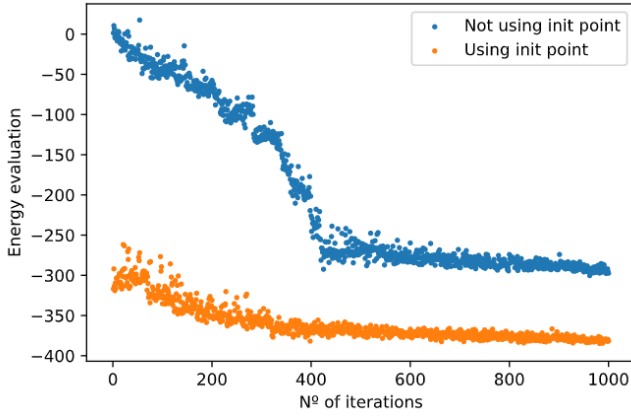


FIG. 12. Energy comparison between VQE algorithm without using or using the initial point. It is noticeable how the first one tends to stabilize after multiple iterations (approximately 400), starting the search for a minimum from a random starting point (depending on the seed provided). Meanwhile, the second one is capable of stabilizing and reach a solution close to the absolute minimum with much less iterations, starting from energy point evaluation close to the real energy solution

be better classified.

Also, one of the improvements is to train the classifier with noisy data further so that the qCBR can adapt to

real past situations that adjust to the new situation. Because, in practice, there is usually no past case strictly the same as a new one.

The last improvement is to generalize the qCBR to serve various types of problems (betting problem, financial, software maintenance, human reasoning, etc.). To get it, we must focus on designing the memory of the cases so that different data sizes can be indexed and train the classifier with several other data models.

Secondly, both QIR [44] and qCBR work with a data representation model based on a multidimensional vector in Hilbert space.

This offers the possibility for quantum algorithms to perform a clustering or discrimination of the data within this vector space.

The QIR analyses whether a certain entry is related to other types of documents previously studied and how the classic NLP techniques are performed [76, 77]. To do this, it projects the input vector introduced concerning the bases of the clusters built corresponding to each class with similar patterns.

At the same time, qCBR follows a similar process for predicting whether an input vector corresponds to a previously analysed class and calculates the probability that each type corresponds to the new vector from the prox-

CBR with KNN solving the Social Workers' Problem		
For 4 patients and 3 socials workers		
Layers	#Cases	Accuracy
1	20	-
	50	55.1
	100	58.5
	240	70.3
	340	71.1
	480	73.6
	500	74.8
	580	76.8

TABLE VI. The classical CBR result on KNN classifier for a number of patients and social workers fixed at 4 and 3, respectively. The better behaviour of this CBR can be observed for some cases greater than 100. To have a good functioning of the CBR, it must be iterated with the social workers' dataset one 239 times. And at the case number of 240, we will have an accuracy of 70.3%. The "accuracy" value is the percentage of the number of correct solutions found by the CBR leveraged on KNN, applying a 10-KFold cross-validation process. A hyphen (-) denotes that no solution was found within the 20 cases. All these tests were done for the layer equal to 1.

CBR leveraged by CNN solving the Social Workers' Problem				
#Patients	#SW	#Layers	#Cases	Accuracy
3	2	2	580	65.4
4	3	2	580	43.3
5	2	2	580	37.3
5	3	2	580	26.3
5	4	2	580	45.2

TABLE VII. CBR with a neural network classifier and a backtracking algorithm as a synthesizer. SW denotes Social Workers.

imity of each vector subspaces generated from each category.

The text representation is transformed to a numeric vector from a process called word2vec [78–80] and doc2vec [81, 82], and once the vector is obtained, the process to follow is identical to the one to follow by qCBR. In many cases, seeing references [83–86], QIR and NLP already predefine the classes to be analysed, either Pop,

CBR with KNN solving the Social Workers' Problem				
#Patients	#SW	#Layers	#Cases	Accuracy
3	2	1	580	95.6
4	3	1	580	77.8
5	2	1	580	47.8
5	3	1	580	44.7
5	4	1	580	63.1

TABLE VIII. CBR with a KNN classifier and a backtracking algorithm as a synthesizer. SW denotes Social Workers.

Methods	Complexity
Retrieve	$O(\log NM)$
Re-use	$O(K \log(N) + \log NM)$
Revise	$O(\log(N)) + ICA$
Retain	$O(\log NM) + PCA$

TABLE IX. Table of the complexity of qCBR counting the PCA cases and the ICA complexity. In this case, K, the number of shots of the VQE, is fixed to 50.

Rock, etc. By predefining that each axis of the Hilbert space corresponds to a type, this process is similar to the qCBR but without the synthesiser's ability.

The clustering process allows the algorithm to create classes and related documents without specifying the categories; therefore, in the case of QIR, it does not move away from an abstraction of the classical problem of "bag-of-words" parsers of spam.

The creation of the SWP vector subspace over the Hilbert vector space is similar in the references [87, 88] where the authors focus on filters, request and document retrieval.

It is worth noting that the qCBR does not present a barren plateau problem due to the low numbers of qubits, shallow quantum circuit and because we have used local cost functions as advocated by the barren plateau theorem [36].

VIII. CONCLUSIONS AND FURTHER WORK

We observed the outstanding performance of qCBR compared to its classical counterpart on the average accuracy, scalability and tolerance to an overlapping dataset. Some of the problems of standard and classical CBR have been mitigated in this work. With the design that has been proposed in this work, it has been possible to mea-

sure situations of difficult similarity between cases. Despite the non-linear and overlapping attributes, the classifier has been endowed with characteristics that serve to arrive at two similar topics that may seem quite different by having different values in features, but not very important. In the VQE with *Initial_point*, we can have different *Initial_point* associated with each training class sample with the same class. With the technique of the average of the "*Initial_point*", it is possible to solve this problem by providing the qCBR to distinguish the similarity between cases. Another issue that qCBR mostly solves is the time required to classify a new topic.

With the results of the two implementations (classical and quantum CBR), it is observed that the classical CBR designed with the KNN behaves better for some determined cases (table (VIII)). It is seen that the system has not finished learning thoroughly (table (V) and (VI)) contrary to the qCBR (table(III) and (IV)). This is due to its classifier's accuracy, without forgetting the significant contribution of its synthesis system.

Another improvement that qCBR introduces is when retaining cases, implementing a retention system that maintains model cases and that, together, synthesize the real and most important information. One of the improvements to consider is the implementation of quantum ICA. In this way, the classical ICA analysis's complexity cost will be significantly reduced. Also counting that the PCA is saved since we have an 8-dimensional classifier, the complexity of the qCBR would be that of the classifier plus some setup constants.

The other exciting line of the future is to design the memory of cases using the quantum technique of random-access memory (qRAM) [89] to improve the memory of stored cases.

ACKNOWLEDGMENTS

The authors greatly thank the IBMQ team, mainly Steve Wood. P.A. thanks Jennifer Ramírez Molino, the Qibo team, Adrian Perez-Salinas and Guillermo Alonso Alonso de Linaje for his support and comments on the manuscript.

Compliance with Ethics Guidelines

Conflict of interest: P. Atchade Adelmou, D. Casado Faulí, E. Golobardes Ribé and X. Vilasís-Cardona state that there are no conflicts of interest.

Appendix A: Variational Quantum Classifier

To date, two dominant categories allow to design quantum classifiers. Although almost all are inspired by the classical classifiers (kernel or neural networks) [53], there is a new category of classifiers that respond to the cur-

rent era of quantum computing (NISQ); hybrid and variational classifiers.

1. The Ansatz

The Ansatz design inherited from previous works [1, 10] [54]. The way to load the data into the Ansatz is inspired by [29] where the data (variable x) is entered using the weights and biases scheme. In this case, the single-qubit gate that serves as the building block for all Ansatz is given by (A1) similar to neural networks.

$$U = (\theta, x) = R_x(\theta_1 x + \theta_2) R_z(\theta_3) \quad (A1)$$

Being θ the vector of the parameters and R_x and R_y the unit gates of qubits used to create the Ansatz. To complement the experimentation scenario, it would be necessary to add the CNOT gate and the CRZ, which are the gates that help to achieve entanglement as seen in figure (13), (14) and (15).

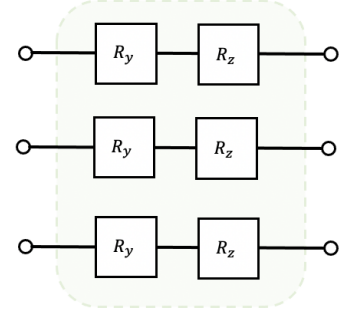


FIG. 13. R_y and R_z Ansatzes without entanglement used in qCBR experimentation.

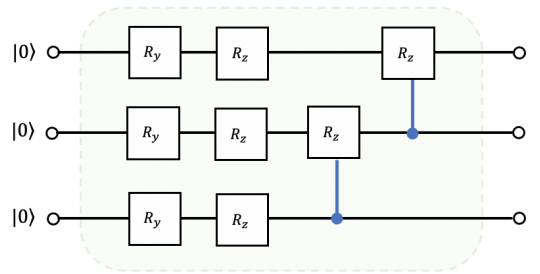


FIG. 14. R_y and R_z Ansatzes with CRZ entanglement used in qCBR experimentation.

The variational quantum classifier structure (figure (16) and (17)) is based on layers of trainable circuit blocks $L(i) = \prod_{i,j} U(i,j)$ and data coding, as shown in (3) for 8 dimensional or in (A1) for 2 dimensional data size. Additionally, the entanglement can be achieved using the CRZ or CNOT gates. The number of parameters

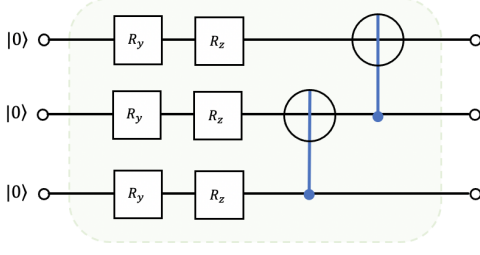


FIG. 15. R_y and R_z Ansatzes with $CNOT$ entanglement used in qCBR.

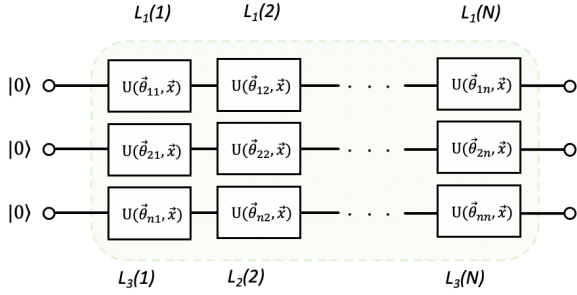


FIG. 16. Three-qubit quantum classifier circuit without entanglement.

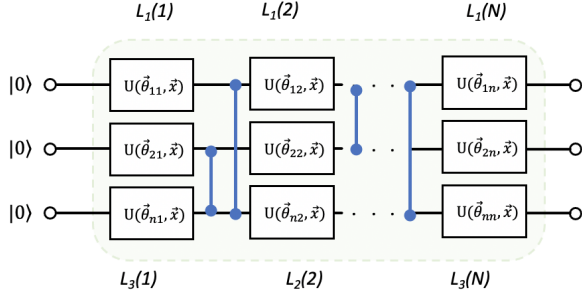


FIG. 17. Three-qubit quantum classifier circuit with entanglement by using CZ or $CNOT$ gates.

to optimize the classifier is given by (A2).

$$NumParam = (nL * 2d * L) \quad (A2)$$

In this case, with n the number of qubits, $n = 3$, L the number of layers (blocks), in the experiment, it is a variable data and d which is the dimension of the problem. In other words, d varies with the choice of Ansatz and whether or not entanglement is applied. In the case of the entanglements in figure (14), the d would be summed 1 (CRZ gate has one parameter), which equates to equation (A3).

$$NumParam = (nL * 2(d + 1) * L) \quad (A3)$$

2. Fidelity cost function

The similarity function follows the same strategy as the re-uploading and path; nevertheless, the Ansatz is different. It uses the definition of quantum fidelity associated with several qubits and maximizes said average fidelity between the test state and the final state corresponding to its class. Equation (A4) [15] defines the cost function used.

$$\begin{aligned} CF(\vec{\alpha}, \vec{\theta}, \vec{\omega}) &= \\ &= \frac{1}{2} \sum_{\mu=1}^M \sum_{c=1}^C \left(\sum_{q=1}^Q (\alpha_{c,q} F_{c,q}(\vec{\theta}, \vec{\omega}, \vec{x}_\mu) - Y_c(\vec{x}_\mu))^2 \right) \end{aligned} \quad (A4)$$

with

$$F_{c,q}(\vec{\theta}, \vec{\omega}, \vec{x}_\mu) = \langle \psi_c | \rho_q(\vec{\theta}, \vec{\omega}, \vec{x}) | \psi_c \rangle \quad (A5)$$

Where ρ_q is the reduced density matrix of the qubit to be measured, M is the total number of training point, C is the total number of the classes, \vec{x}_μ are the training points and $\vec{\alpha} = (\alpha_1, \dots, \alpha_C)$ are introduced as class weights to be optimized together with $\vec{\theta}$, $\vec{\omega}$, are the parameters and Q the numbers of the qubits. Counting on $Y_c(\vec{x}_\mu)$ as the fidelity vector for a perfect classification. This cost function (A4) is weighted and averaged over all the qubit that form this classifier. In order to complete the hybrid system, it is used for the classical part, the following minimization methods above cited: L-BFGS-B [90], COBYLA [63] and SPSA [64].

Appendix B: The qCBR's details

The operation of the **retrieve** (prediction) block is given by a new case (schedule). In this experimentation, the schedule that best adapts to the latest case to be solved is recovered with the predict method, which is executed at a time $O(\log(MN))$. It worth saying that, due to the SWP descriptions, a possible schedule change, a stage of understanding or interpretation is necessary, since an adequate resolution of the new schedule cannot be carried out if it is not understood with some completeness. This stage of understanding is a simple decision algorithm with minimal intelligence.

Once having the predicted solution, the synthesis block creates a new solution (proposed solution) by combining recovered solutions. To do this, the algorithm is divided into two main lines (figure (10)). A line that determines an acceptable degree of error (after a probabilistic study) that the predicted solution can be considered the proposed solution. The second branch is in charge of improving the expected solution towards a better-proposed solution. To do this, the *Initial_point* associated with the retrieved schedule is retrieved from the case memory, and the Variational Quantum Eigensolver is executed with very few shots, (k shots). The idea here is

to refine the new schedule's similarity with the recovered one. Operating the VQE with *Initial_point* provides the algorithm with parameter values through the initial point as a starting point for searching for the minimum eigenvalue (similarity between the two times) when the new time's solution point is believed to be close to a matter of the recovered schedule. This is how the **Re-use** block works. These operations have a complexity of $O(k \log(N) + \log(NM))$. Where N is the number of social workers, M is the number of patients and k is the number of shots.

The algorithm's processes to review the proposed solution are seen below the **Re-use** block in figure (10). It is essential to classify the best possible solution for the proposed prototype. The best possible solution is calculated with the VQE with the maximum resolution and depth (for the variational part). Once the solution is obtained, it is compared with the proposed solution and said solution with its characteristics is added to the new schedule before storing it (see figure (7) and (8)). The computational complexity of the Revise is determined by $O(\log(N) + ICA)$. In this work, access to data (states) is determined by $O(\log(MN))$ due to the characteristics of the inner products and superpositions.

One of the most critical blocks in this work is to **Retain**. This block is the heart of the CBR because it is the classifier and because it is the block that allows us to conclude that it has been learned from the previous cases. Not all instances (schedules) are saved in this job, leading to the excessively slow classifier. Therefore, in this part of the algorithm, the best cases (timetables) that summarize all the essential information are retained.

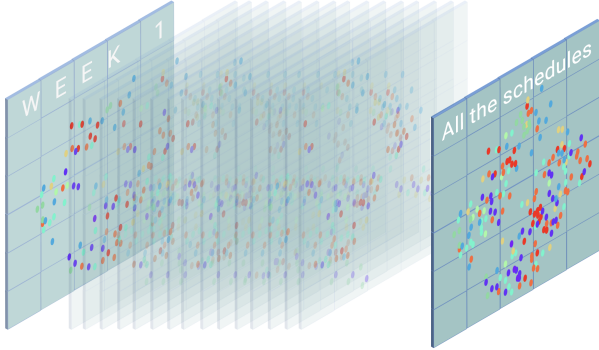


FIG. 18. Representation of the generation of the n weekly schedules of the SWP. The last plane, the one to the right of everything, represents all the SWP classes. The overlapping effect generated by the social workers' problem's characteristics and experimentation scenarios can be observed. This experimentation leads us to use the ICA technique to have a resulting dataset regardless of the schedules.

The **Retain** process begins with the treatment of schedules, searching for the algorithm's best efficiency, which is a challenge to solve in this block. In the case of SWP, the patient visits hours have a margin range of 30 min. Therefore, if one schedule starts at 9:00, the next

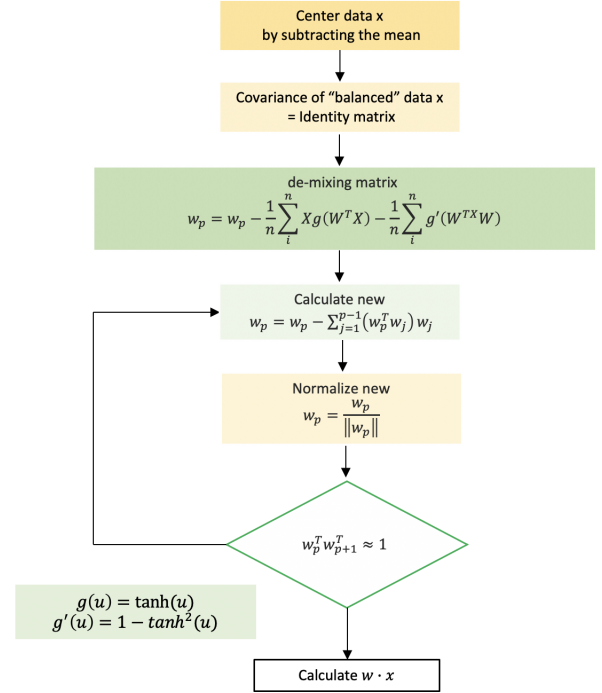


FIG. 19. Fundamental processes to apply ICA to SWP dataset. The first thing that is done is to centre the data x by subtracting the mean, balancing the data x removing its variance, and calculating the unmixing matrix of W . is calculated. Then the new value of w is calculated, and then w is normalized before checking if with the said value the algorithm converges or not. If it does not converge, a new w must be recalculated, and if it converges, calculate the scalar product of $\langle x, y \rangle$ to obtain the independent weekly schedules.

could begin at 9:30, leading to a dataset with overlap between schedules if many schedules have similar time ranges spread over different days of the week. In the case of non-linearity of the data, an almost perfect classifier with an average accuracy more significant than 80% would be needed to be combined with a data processing system and a decision tree.

In this work, we template both scenarios. First, get an excellent classifier and apply data processing techniques to help a poor classifier. Using the standard classifier, ICA [71] is applied to the original data to reduce the effects of the degree of overlap (figure (18)) without losing the fundamental characteristics of the data. Figure (19) summarizes the processes and operations applied to reduce the overlapping effect observed in the generation of SWP schedules. The complexity of this operation is noted as $O(ICA)$. The PCA is then used to reduce the data dimension from 8 to 2 and apply it to the designed variational classifier with the complexity equal to $O(PCA)$. Once the best time is determined, we retain the knowledge acquired at the time of the case's resolution.

- [1] Parfait Atchade-Adelomou, Elisabet Golobardes-Ribé, and Xavier Vilasis-cardona. Using the variational-quantum-eigensolver (vqe) to create an intelligent social workers schedule problem solver. In *International Conference on Hybrid Artificial Intelligence Systems*, pages 245–260. Springer, 2020.
- [2] Stephen L. Morgan and Stanley N. Deming. Simplex optimization of analytical chemical methods. *Analytical Chemistry*, 46(9):1170–1181, August 1974. doi:10.1021/ac60345a035.
- [3] Christian Bliklú, Pierre Bonami, and Andrea Lodi. Solving mixed-integer quadratic programming problems with ibm-cplex: a progress report. In *Proceedings of the twenty-sixth RAMP symposium*, pages 16–17, 2014.
- [4] Agnar Aamodt and Enric Plaza. Case-based reasoning: Foundational issues, methodological variations, and system approaches. *Artificial Intelligence Communications*, 7:39–59, 1994.
- [5] Eduardo Lupiani, Jose M. Juarez, Jose Palma, and Roque Marin. Monitoring elderly people at home with temporal case-based reasoning. *Knowledge-Based Systems*, 134:116–134, October 2017. doi:10.1016/j.knsys.2017.07.025.
- [6] Eva Armengol, Albert Palaudaries, and Enric Plaza. Individual prognosis of diabetes long-term risks: A cbr approach. *Methods of Information in Medicine-Methodik der Information in der Medizin*, 40(1):46–51, 2001.
- [7] Claudio Baccigalupo and Enric Plaza. Case-based sequential ordering of songs for playlist recommendation. In *Lecture Notes in Computer Science*, pages 286–300. Springer Berlin Heidelberg, 2006. doi:10.1007/11805816_22.
- [8] Hui Li and Jie Sun. Ranking-order case-based reasoning for financial distress prediction. *Knowledge-Based Systems*, 21(8):868–878, December 2008. doi:10.1016/j.knsys.2008.03.047.
- [9] Paolo Priore, David De La Fuente, Raúl Pino, and Javier Puente. Utilización del razonamiento basado en casos en la toma de decisiones. aplicación en un problema de secuenciación. *Dirección y Organización*, -(28), 2002.
- [10] Parfait Atchade-Adelomou, Elisabet Golobardes-Ribe, and Xavier Vilasis-Cardona. Using the parameterized quantum circuit combined with variational-quantum-eigensolver (vqe) to create an intelligent social workers’ schedule problem solver. *Arxiv*, 2020. arXiv:2010.05863.
- [11] P.W. Shor. Algorithms for quantum computation: discrete logarithms and factoring. In *Proceedings 35th Annual Symposium on Foundations of Computer Science*. IEEE Comput. Soc. Press, 1994. doi:10.1109/sfcs.1994.365700.
- [12] Lov K. Grover. A fast quantum mechanical algorithm for database search. In *Proceedings of the twenty-eighth annual ACM symposium on Theory of computing - STOC '96*. ACM Press, 1996. doi:10.1145/237814.237866.
- [13] John Preskill. Quantum computing in the nisq era and beyond. *Quantum*, 2:79, Aug 2018. URL: <http://dx.doi.org/10.22331/q-2018-08-06-79>, doi:10.22331/q-2018-08-06-79.
- [14] Maria Schuld, Ilya Sinayskiy, and Francesco Petruccione. An introduction to quantum machine learning. *Contemporary Physics*, 56(2):172–185, Oct 2014. URL: <http://dx.doi.org/10.1080/00107514.2014.964942>, doi:10.1080/00107514.2014.964942.
- [15] Adrián Pérez-Salinas, Alba Cervera-Lierta, Elies Gil-Fuster, and José I. Latorre. Data re-uploading for a universal quantum classifier. *Quantum*, 4:226, Feb 2020. URL: <http://dx.doi.org/10.22331/q-2020-02-06-226>, doi:10.22331/q-2020-02-06-226.
- [16] Katarzyna Roszak, Radim Filip, and Tomáš Novotný. Decoherence control by quantum decoherence itself. *Scientific Reports*, 5(1), June 2015. doi:10.1038/srep09796.
- [17] Robert Stárek, Michal Mičuda, Ivo Straka, Martina Nováková, Miloslav Dušek, Miroslav Ježek, Jaromír Fiurášek, and Radim Filip. Experimental quantum decoherence control by dark states of the environment. *New Journal of Physics*, 22(9):093058, September 2020. doi:10.1088/1367-2630/abb47d.
- [18] Joschka Roffe. Quantum error correction: an introductory guide. *Contemporary Physics*, 60(3):226–245, Jul 2019. URL: <http://dx.doi.org/10.1080/00107514.2019.1667078>, doi:10.1080/00107514.2019.1667078.
- [19] Alberto Peruzzo, Jarrod McClean, Peter Shadbolt, Man-Hong Yung, Xiao-Qi Zhou, Peter J. Love, Alán Aspuru-Guzik, and Jeremy L. O’Brien. A variational eigenvalue solver on a photonic quantum processor. *Nature Communications*, 5(1), July 2014. doi:10.1038/ncomms5213.
- [20] Edward Farhi, Jeffrey Goldstone, and Sam Gutmann. A quantum approximate optimization algorithm, 2014. arXiv:1411.4028.
- [21] G. G. Guerreschi and A. Y. Matsuura. QAOA for max-cut requires hundreds of qubits for quantum speed-up. *Scientific Reports*, 9(1), May 2019. doi:10.1038/s41598-019-43176-9.
- [22] Linghua Zhu, Ho Lun Tang, George S. Barron, F. A. Calderon-Vargas, Nicholas J. Mayhall, Edwin Barnes, and Sophia E. Economou. An adaptive quantum approximate optimization algorithm for solving combinatorial problems on a quantum computer, 2020. arXiv:2005.10258.
- [23] M. V. Altaisky, N. E. Kaputkina, and V. A. Krylov. Quantum neural networks: Current status and prospects for development. *Physics of Particles and Nuclei*, 45(6):1013–1032, November 2014. doi:10.1134/s1063779614060033.
- [24] Maria Schuld, Ilya Sinayskiy, and Francesco Petruccione. The quest for a quantum neural network. *Quantum Information Processing*, 13(11):2567–2586, August 2014. doi:10.1007/s11128-014-0809-8.
- [25] Kerstin Beer, Dmytro Bondarenko, Terry Farrelly, Tobias J. Osborne, Robert Salzmann, Daniel Scheiermann, and Ramona Wolf. Training deep quantum neural networks. *Nature Communications*, 11(1), February 2020. doi:10.1038/s41467-020-14454-2.
- [26] Patrick Rebentrost, Masoud Mohseni, and Seth Lloyd. Quantum support vector machine for big data classification. *Physical Review Letters*, 113(13), Sep 2014. URL: <http://dx.doi.org/10.1103/PhysRevLett.113.130503>, doi:10.1103/physrevlett.113.130503.
- [27] Vojtěch Havlíček, Antonio D. Córcoles, Kristan Temme, Aram W. Harrow, Abhinav Kandala, Jerry M. Chow, and Jay M. Gambetta. Supervised learning with quantum-

- enhanced feature spaces. *Nature*, 567(7747):209–212, March 2019. doi:10.1038/s41586-019-0980-2.
- [28] D. Willsch, M. Willsch, H. De Raedt, and K. Michielsen. Support vector machines on the d-wave quantum annealer. *Computer Physics Communications*, 248:107006, March 2020. doi:10.1016/j.cpc.2019.107006.
- [29] Adrián Pérez-Salinas, David López-Núñez, Artur García-Sáez, P. Forn-Díaz, and José I. Latorre. One qubit as a universal approximant, 2021. arXiv:2102.04032.
- [30] Maria Schuld and Nathan Killoran. Quantum machine learning in feature hilbert spaces. *Physical review letters*, 122(4):040504, 2019.
- [31] Parfait Atchade-Adelomou, Guillermo Alonso-Linaje, Jordi Albo-Canals, and Daniel Casado-Fauli. qrobot: A quantum computing approach in mobile robot order picking and batching problem solver optimization. *Algorithms*, 14(7), 2021. URL: <https://www.mdpi.com/1999-4893/14/7/194>, doi:10.3390/a14070194.
- [32] Saúl González-Bermejo, Guillermo Alonso-Linaje, and Parfait Atchade-Adelomou. Gps: Improvement in the formulation of the tsp for its generalizations type qubo, 2021. arXiv:2110.12158.
- [33] Zhikuan Zhao, Jack K Fitzsimons, Patrick Rebentrost, Vedran Dunjko, and Joseph F Fitzsimons. Smooth input preparation for quantum and quantum-inspired machine learning. *Quantum Machine Intelligence*, 3(1):1–6, 2021.
- [34] Lucas Lamata. Quantum machine learning and quantum biomimetics: A perspective. *Machine Learning: Science and Technology*, 1(3):033002, 2020.
- [35] Francesco Benfenati, Guglielmo Mazzola, Chiara Capecci, Panagiotis Kl Barkoutsos, Pauline J Ollitrault, Ivano Tavernelli, and Leonardo Guidoni. Improved accuracy on noisy devices by nonunitary variational quantum eigensolver for chemistry applications. *Journal of Chemical Theory and Computation*, 2021.
- [36] Marco Cerezo, Akira Sone, Tyler Volkoff, Lukasz Cincio, and Patrick J Coles. Cost function dependent barren plateaus in shallow parametrized quantum circuits. *Nature communications*, 12(1):1–12, 2021.
- [37] Guillermo Alonso-Linaje and Parfait Atchade-Adelomou. Eva: a quantum exponential value approximation algorithm, 2021. arXiv:2106.08731.
- [38] Parfait Atchade-Adelomou and Guillermo Alonso-Linaje. Quantum enhanced filter: Qfilter, 2021. arXiv:2104.03418.
- [39] Zhi jin Zhao and Hai chao Lai. A cognitive engine based on case-based reasoning quantum genetic algorithm. In *2012 IEEE 14th International Conference on Communication Technology*. IEEE, November 2012. doi:10.1109/icct.2012.6511219.
- [40] Salama A. Mostafa, Mohd Sharifuddin Ahmad, and M. A. Firdaus M. A soft computing modeling to case-based reasoning implementation. *International Journal of Computer Applications*, 47(7):14–21, June 2012. doi:10.5120/7199-9976.
- [41] Khaled Amaief and Jie Lu. Ontology-supported case-based reasoning approach for intelligent m-government emergency response services. *Decision Support Systems*, 55(1):79–97, April 2013. doi:10.1016/j.dss.2012.12.034.
- [42] Yeray Mezquita, Ricardo S. Alonso, Roberto Casado-Vara, Javier Prieto, and Juan Manuel Corchado. A review of k-NN algorithm based on classical and quantum machine learning. In *Distributed Computing and Artificial Intelligence, Special Sessions, 17th International Conference*, pages 189–198. Springer International Publishing, July 2020. doi:10.1007/978-3-030-53829-3_20.
- [43] Benjamin Piwowarski, Ingo Frommholz, Mounia Lalmas, and Keith Van Rijsbergen. What can quantum theory bring to information retrieval. In *Proceedings of the 19th ACM international conference on Information and knowledge management*, pages 59–68, 2010.
- [44] Alexander Lebedev and Andrei Khrennikov. Introductory review to quantum information retrieval, 2020. arXiv:2008.13541.
- [45] Kirsty Kitto, Peter Bruza, and Liane Gabora. A quantum information retrieval approach to memory. In *The 2012 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8. IEEE, 2012.
- [46] Andrew Gleason. Measures on the closed subspaces of a hilbert space. *Indiana Univ. Math. J.*, 6:885–893, 1957.
- [47] Cornelis Joost Van Rijsbergen. *The geometry of information retrieval*. Cambridge University Press, 2004.
- [48] Hemant M Kakde. Range searching using kd tree. from the citeseerx database on the World Wide Web: <http://citeseerx.ist.psu.edu/viewdoc/summary>, 2005.
- [49] Hazarath Munaga and Venkata Jarugumalli. Performance evaluation: Ball-tree and KD-tree in the context of MST. In *Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering*, pages 225–228. Springer Berlin Heidelberg, 2012. doi:10.1007/978-3-642-32573-1_38.
- [50] J.H. Friedman, F. Baskett, and L.J. Shustek. An algorithm for finding nearest neighbors. *IEEE Transactions on Computers*, C-24(10):1000–1006, October 1975. doi:10.1109/t-c.1975.224110.
- [51] K. Fukunaga and P.M. Narendra. A branch and bound algorithm for computing k-nearest neighbors. *IEEE Transactions on Computers*, C-24(7):750–753, July 1975. doi:10.1109/t-c.1975.224297.
- [52] Laura Lozano and Javier Fernández. Razonamiento basado en casos: Una visión general. *Recuperado el*, 2008.
- [53] Abdiansah Abdiansah and Retantyo Wardoyo. Time complexity analysis of support vector machines (SVM) in LibSVM. *International Journal of Computer Applications*, 128(3):28–34, October 2015. doi:10.5120/ijca2015906480.
- [54] Sukin Sim, Peter D. Johnson, and Alán Aspuru-Guzik. Expressibility and entangling capability of parameterized quantum circuits for hybrid quantum-classical algorithms. *Advanced Quantum Technologies*, 2(12):1900070, Oct 2019. URL: <http://dx.doi.org/10.1002/qute.201900070>, doi:10.1002/qute.201900070.
- [55] William S Noble. What is a support vector machine? *Nature Biotechnology*, 24(12):1565–1567, December 2006. doi:10.1038/nbt1206-1565.
- [56] Robert Wille, Rod Van Meter, and Yehuda Naveh. Ibm’s qiskit tool chain: Working with and developing for real quantum computers. In *2019 Design, Automation Test in Europe Conference Exhibition (DATE)*, pages 1234–1240, 2019. doi:10.23919/DATE.2019.8715261.
- [57] Alberto Peruzzo, Jarrod McClean, Peter Shadbolt, Man-Hong Yung, Xiao-Qi Zhou, Peter J. Love, Alán Aspuru-Guzik, and Jeremy L. O’Brien. A variational eigenvalue solver on a photonic quantum processor. *Nature Communications*, 5(1), Jul 2014. URL: <http://dx.doi.org/10.1038/ncomms5213>, doi:10.1038/ncomms5213.

- [58] Daochen Wang, Oscar Higgott, and Stephen Brierley. Accelerated variational quantum eigensolver. *Physical Review Letters*, 122(14), Apr 2019. URL: <http://dx.doi.org/10.1103/PhysRevLett.122.140504>, doi: 10.1103/physrevlett.122.140504.
- [59] Harper R Grimsley, Sophia E Economou, Edwin Barnes, and Nicholas J Mayhall. An adaptive variational algorithm for exact molecular simulations on a quantum computer. *Nature communications*, 10(1):1–9, 2019.
- [60] Robert Wille, Rod Van Meter, and Yehuda Naveh. Ibm’s qiskit tool chain: Working with and developing for real quantum computers. In *2019 Design, Automation & Test in Europe Conference & Exhibition (DATE)*, pages 1234–1240. IEEE, 2019.
- [61] G Arutyunov, S Frolov, and M Staudacher. Bethe ansatz for quantum strings. *Journal of High Energy Physics*, 2004(10):016–016, October 2004. doi:10.1088/1126-6708/2004/10/016.
- [62] Jacob Biamonte, Peter Wittek, Nicola Pancotti, Patrick Rebentrost, Nathan Wiebe, and Seth Lloyd. Quantum machine learning. *Nature*, 549(7671):195–202, Sep 2017. URL: <http://dx.doi.org/10.1038/nature23474>, doi: 10.1038/nature23474.
- [63] community The SciPy. Cobyla, January 2021. URL: <https://docs.scipy.org/doc/scipy/reference/optimize.minimize-cobyla.html>.
- [64] Spall James C. Spsa community, January 2001. URL: <https://www.jhuapl.edu/SPSA/>.
- [65] Dong C. Liu and Jorge Nocedal. On the limited memory BFGS method for large scale optimization. *Mathematical Programming*, 45(1-3):503–528, August 1989. doi:10.1007/bf01589116.
- [66] Michael A. Nielsen, Isaac Chuang, and Lov K. Grover. Quantum computation and quantum information. *American Journal of Physics*, 70(5):558–559, May 2002. doi: 10.1119/1.1463744.
- [67] Michael W Browne. Cross-validation methods. *Journal of Mathematical Psychology*, 44(1):108–132, March 2000. doi:10.1006/jmps.1999.1279.
- [68] Chai Wah Wu. On rayleigh–ritz ratios of a generalized laplacian matrix of directed graphs. *Linear Algebra and its Applications*, 402:207–227, June 2005. doi:10.1016/j.laa.2004.12.014.
- [69] Parfait Atchade-Adelomou. quantum cases based reasoning, 2021. URL: <https://github.com/pifparfait/qCBR>.
- [70] Parfait Atchade-Adelomou, Elisabet Golobardes-Ribé, and Xavier Vilasis-Cardona. Formulation of the social workers’ problem in quadratic unconstrained binary optimization form and solve it on a quantum computer. *Journal of Computer and Communications*, 8(11):44–68, 2020.
- [71] William K. Pratt and John Wiley. A wiley-interscience publication. In *Digital Image Processing*, pages 636–646. John Wiley & Sons, INC, 1978.
- [72] Zee Ma. A tutorial on principal component analysis, 2014. URL: <http://rgdoi.net/10.13140/2.1.1593.1684>, doi:10.13140/2.1.1593.1684.
- [73] Stavros Efthymiou, Sergi Ramos-Calderer, Carlos Bravo-Prieto, Adrián Pérez-Salinas, Diego García-Martín, Artur Garcia-Saez, José Ignacio Latorre, and Stefano Carrazza. Qibo: a framework for quantum simulation with hardware acceleration, 2020. [arXiv:2009.01845](https://arxiv.org/abs/2009.01845).
- [74] David C McKay, Thomas Alexander, Luciano Bello, Michael J Biercuk, Lev Bishop, Jiayin Chen, Jerry M Chow, Antonio D Córcoles, Daniel Egger, Stefan Filipp, et al. Qiskit backend specifications for openqasm and openpulse experiments. *arXiv preprint arXiv:1809.03452*, 2018.
- [75] Ewin Tang. Quantum-inspired classical algorithms for principal component analysis and supervised clustering, 2019. [arXiv:1811.00414](https://arxiv.org/abs/1811.00414).
- [76] Gobinda G Chowdhury. Natural language processing. *Annual review of information science and technology*, 37(1):51–89, 2003.
- [77] Elizabeth D Liddy. Natural language processing. 2001.
- [78] Yoav Goldberg and Omer Levy. word2vec explained: deriving mikolov et al.’s negative-sampling word-embedding method. *arXiv preprint arXiv:1402.3722*, 2014.
- [79] Xin Rong. word2vec parameter learning explained. *arXiv preprint arXiv:1411.2738*, 2014.
- [80] Kenneth Ward Church. Word2vec. *Natural Language Engineering*, 23(1):155–162, 2017.
- [81] Jey Han Lau and Timothy Baldwin. An empirical evaluation of doc2vec with practical insights into document embedding generation. *arXiv preprint arXiv:1607.05368*, 2016.
- [82] Donghwa Kim, Deokseong Seo, Suhyoung Cho, and Pil-sung Kang. Multi-co-training for document classification using various document representations: Tf-idf, lda, and doc2vec. *Information Sciences*, 477:15–29, 2019.
- [83] A Khrennikov, D Aert, B Wang, ED Buccio, and M Melucci. Quantum-like models for information retrieval and decision-making. steam-h: Science, technology, engineering, agriculture, mathematics & health, 2019.
- [84] Peter D Bruza and Richard J Cole. Quantum logic of semantic space: An exploratory investigation of context effects in practical reasoning. *arXiv preprint quant-ph/0612178*, 2006.
- [85] Kevin Lund and Curt Burgess. Producing high-dimensional semantic spaces from lexical co-occurrence. *Behavior research methods, instruments, & computers*, 28(2):203–208, 1996.
- [86] Scott Deerwester, Susan T Dumais, George W Furnas, Thomas K Landauer, and Richard Harshman. Indexing by latent semantic analysis. *Journal of the American society for information science*, 41(6):391–407, 1990.
- [87] Benjamin Piwowarski, Ingo Frommholz, Mounia Lalmas, and Keith Van Rijsbergen. Exploring a multidimensional representation of documents and queries (extended version). *arXiv preprint arXiv:1002.3238*, 2010.
- [88] Benjamin Piwowarski, Ingo Frommholz, Yashar Moshfeghi, Mounia Lalmas, and Keith van Rijsbergen. Filtering documents with subspaces. In *Lecture Notes in Computer Science*, pages 615–618. Springer Berlin Heidelberg, 2010. doi:10.1007/978-3-642-12275-0_60.
- [89] Miles Blencowe. Quantum RAM. *Nature*, 468(7320):44–45, November 2010. doi:10.1038/468044a.
- [90] Richard H. Byrd, Peihuang Lu, Jorge Nocedal, and Ciyu Zhu. A limited memory algorithm for bound constrained optimization. *SIAM Journal on Scientific Computing*, 16(5):1190–1208, September 1995. doi:10.1137/0916069.