QuantumEyes: Towards Better Interpretability of Quantum Circuits

Category: Research

Paper Type: application/design study

Fig. 1: The visualization system *QuantumEyes* enhances the interpretability of quantum circuits through the global analysis (A-C) to explain the operations of quantum gates and local analysis (A_2 and B_3) to reveal the implicit reasoning of basis state's measured probability. Probability Summary View (A) shows the overview of a quantum circuit regarding the measured probability. State Evolution View (B) supports the in-depth analysis of the basis states across each step. Gate Explanation View (C) explains the effects of quantum gates via the transformation of qubit states. State Comparison View (A_2 and B_3) with the geometrical visualization *dandelion chart* enables users to better understand the measured probability regarding amplitudes. The original quantum circuit (D) is given for a better comparison of *QuantumEyes* and quantum circuit diagrams.

Abstract— Quantum computing offers significant speedup compared to classical computing, which has led to a growing interest among users in learning and applying quantum computing across various applications. However, quantum circuits, which are fundamental for implementing quantum algorithms, can be challenging for users to understand due to their underlying logic, such as the temporal evolution of quantum states and the effect of quantum amplitudes on the probability of basis quantum states. To fill this research gap, we propose *QuantumEyes*, an interactive visual analytics system to enhance the interpretability of quantum circuits by visualizing the evolution of quantum states. Specifically, we develop a suite of visualizations for quantum circuit analysis at both global and local levels. For the global-level analysis, we present three coupled visualizations to delineate the changes of quantum states and the underlying reasons: a Probability Summary View to overview the probability evolution of quantum states; a State Evolution View to enable an in-depth analysis of the influence of quantum gates on the quantum states; a Gate Explanation View to show the individual qubit states and facilitate a better understanding of the effect of quantum gates. For the local-level analysis, we design a State Comparison View featuring a novel geometrical visualization *dandelion chart* to explicitly reveal how the quantum amplitudes affect the probability of the quantum state. We thoroughly evaluated *QuantumEyes* as well as the novel *dandelion chart* integrated into it through two case studies on different types of quantum algorithms and in-depth expert interviews with 12 domain experts. The results demonstrate the effectiveness and usability of our approach in enhancing the interpretability of quantum circuits.

Index Terms—Interpretability, Data Visualization, Quantum Circuits, Quantum Computing

1 Introduction

Quantum computing has experienced remarkable advancements in recent years. The rapid growth in the quality and quantity of quantum computers by leading IT companies such as IBM, Google and Amazon is making potential quantum advantages increasingly realistic for both theoretical quantum algorithms [23, 39, 57, 59] and emerging applications [6, 9, 12, 24, 36]. For example, quantum computing has shown its superior speedup on classical problems, such as *Grover's algorithm* for unstructured search [57], and *Shor's algorithm* for integer factoring [39], albeit only on small data sets currently. Meanwhile, researchers have successfully harnessed the power of quantum computing in various applications, such as machine learning [9], finance [36], and chemistry [24]. The quantum supremacy experiment by Google [6] has shown the potential advantage of quantum computers over their classical counterparts. The recent milestone claimed by Google in quantum error correction [1] has further improved its chance of success.

Building upon the proliferation of quantum computers, the number of people learning quantum computing has experienced rapid growth in recent years [38, 44]. However, prior research has identified that grasping abstract concepts in quantum computing remains challenging [31, 66]. For example, *quantum circuits*, the most fundamental routine to perform any quantum programs, lack the transparency and interpretability needed for easy comprehension [66]. Meanwhile, visualization has been proven to be an effective learning tool for educating people on obscure scientific concepts [37, 61]. Consequently, a graphical representation [19] known as quantum circuit diagrams was proposed decades ago and has been widely used in research papers and textbooks of quantum computing. Despite its prevalence, it primarily overviews a quantum circuit and has limitations in revealing deep insights into quantum circuits' behaviors. From a quantum circuit diagram, it is difficult for quantum computing developers and researchers to understand the functionality of each quantum gate and the final measured probability of each basis state. For example, the viewers cannot inspect the quantum states' initial generation and further evolution or the functionality of each quantum gate from a quantum circuit diagram. Thus, how to intuitively reveal the detailed inner workings of a quantum circuit still remains unclear, and a new visualization approach for better interpretability of quantum circuits is urgently needed.

However, it is non-trivial to fill this research gap. According to our extensive literature survey [7, 25, 31, 40, 49, 53, 63] and close collaborations with six quantum computing experts, the major challenges mainly come from the counter-intuitive nature and intrinsic complexity of ***quantum gate operations*** and ***measured probability*** of quantum circuits. First, the quantum gates are the fundamental operators to manipulate the status of qubits, which makes it crucial for domain users to understand quantum gate operations in order to interpret quantum circuits [31, 66]. But quantum gate operations are essential matrix multiplications that are difficult to visualize and explain. What makes matters worse is the matrix transformations of quantum gates involves complex numbers [35] that are counter-intuitive. Second, the measured probability, determined by the quantum amplitudes of each basis state, is critical to understand the output of quantum circuits. But users often do not possess a mathematical intuition regarding the underlying cause of each basis state's amplitude [49]. Also, quantum system states of multiple qubits can be entangled together rather than being a simple accumulation of multiple individual single-qubit states, and there will be 2^N possible basis states if the qubit number is N. It makes it extremely challenging to visualize multi-qubit states and the corresponding measured probabilities in a limited space [49].

To address the above challenges, we propose *QuantumEyes*, a visualization system to enhance the interpretability of quantum circuits. *QuantumEyes* can intuitively explain the functionality of each quantum gate and the measured probability of each basis state for a given quantum circuit. To ensure the effectiveness of our visual designs, We follow a user-centered design process [34] by working closely with six domain experts in quantum computing for over five months. By summarizing the expert feedback, we distilled design requirements in terms of two levels of analysis - global analysis and local analysis. For the global analysis, we further propose three coordinated views to enhance the interpretability of quantum gates' operations: a Probability Summary View summarizes the changes of all quantum states along a circuit (Fig. 1 A), a State Evolution View supports analyzing how quantum gates affect the evolution of multiple quantum states over time (Fig. 1 B), and a Qubit Explanation View further explains the quantum gates' effect from the view of the single qubit and its acting quantum gate (Fig. 1 C). For the local analysis, we propose a State Comparison View with a novel geometrical visualization *dandelion chart* (Fig. 3), which can visually explain the measured probability of each basis state based on the quantum amplitudes. To evaluate the usefulness and effectiveness of *QuantumEyes*, we present two case studies based on the quantum circuits with different quantum circuit architectures and qubit numbers, i.e., *Grover's Algorithm* and *Quantum Fourier Transform*. We further conduct in-depth interviews with 12 domain experts with carefully-designed tasks. The results show that *QuantumEyes* can effectively help developers and researchers better understand the behaviors of quantum circuits.

The major contributions of this paper can be summarized as follows:

- We formulate the design requirements for improving the interpretability of quantum circuits by working closely with quantum computing experts.
- We introduce *QuantumEyes*, an interactive visualization system to assist quantum computing users in intuitively understanding the behaviors of quantum circuits, including three coordinated views to support the global analysis and a single view with a novel geometrical design named *dandelion chart* to facilitate the local analysis.
- We conduct two case studies and in-depth user interviews with domain experts to demonstrate the effectiveness and usability of *QuantumEyes*.

To further benefit quantum computing developers and researchers, we have made our system *QuantumEyes* publicly accessible online[1]. Also, we have published the novel design for quantum state visualization *dandelion chart* as an independent NPM package[2].

2 Related Work

Our work is relevant to prior research on visualization of quantum circuit evolution and quantum state visualization.

2.1 Quantum State Visualization

Many existing approaches studied how to represent quantum states, the mathematical description of the state of a quantum system. We classify existing visual representations for quantum states based on whether the visualization is state vector-based or probability-aware.

State vector-based approaches. The goal of the state vector-based approach is to visualize the quantum amplitudes of quantum states. The most widely-used representation in the quantum computing community is *Bloch Sphere* [10], which is integrated into many popular quantum computing SDKs like IBM Qiskit [27] and Google Cirq [21] to visualize quantum states. Bloch Sphere leverages a point on the unit sphere to represent the quantum amplitude of a pure single-qubit state. Meanwhile, Bloch Sphere can also reflect two important visual effect, i.e., single-qubit rotation gates and statistical mixtures of pure states. Prior work has introduced various extensions of Bloch Sphere [4, 32, 64]. Also, many researchers have studied how to represent quantum states using 2D shapes. Wille et al. [65] visualized the components of state vectors using a tree-like design. Several studies explored how to better visualize quantum states by enabling multi-qubit visualization, such as the stellar representation [8] and the visualization based on multi-qubit Bloch vectors [62]. However, several issues exist in the above visualization approaches. First, these visualizations do not enable a direct comparison of the probabilities of basis states, making it hard for users to inspect the measured probability. Second, 3D representations

[1] https://quantumeyes.github.io/
[2] https://www.npmjs.com/package/dandelion_chart

have been proven less effective than 2D counterparts when conducting precise measurements [5, 56].

Probability-aware approaches. Some prior work focused on improving the state vector-based approach by explicitly visualizing the measured probability based on the state vector representation. For example, Ruan et al. [49] introduced a 2D geometrical visualization to highlight the impact of the state vector on the probability. Galambos et al. [20] utilized a fractal representation of a multiple-qubit system via a set of rectangles. Also, Chernega et al. studied several variants [13, 15] based on Triada of Malevich's squares [14], which mapped the state vectors of a qubit onto the vertices of a triangle. Similarly, Miller et al. [33] proposed an interface with an embedded node-like graph to explain the quantum circuits and stabilizer groups in quantum error correction, allow the visualization of the update of quantum states. Although the prior work can visualize the probability of quantum states, they still suffer from scalability issues. Most studies can only support the visualization of one qubit [13–15] or two qubits [20, 49], whereas most of the accessible quantum computers are already exceeding this number of qubits. Therefore, it is crucial to enable the inspection of the quantum states with many qubits. Our work aims to support quantum state visualization with multiple qubits, while preserving the property of being probability aware.

2.2 Visualization of Quantum Circuit Evolution

We categorize existing work into two groups: depending on whether the proposed visualization technique is for a specific algorithm or general quantum circuits.

Algorithm-specific visualization. Visualization approaches in this category often aims at a specific quantum algorithm without the generalizability for general quantum programs. For example, Tao et al. [54] utilized Bloch Sphere and a disk-like design to portray the evolution of each quantum states along each step of *Shor's algorithm*. Karafyllidis et al. [28] studied how to visually explain the QFT algorithm by visualizing the changes of the probability of each quantum state, but it cannot support the trace-back analysis of basis states. Meanwhile, two online platforms [29, 43] enabled users to visually understand quantum states and quantum circuits in Quantum Error Correction, respectively. But it is challenging to extend to other quantum circuits, which significantly limits their benefits and impact.

Generally-applicable visualization. Generally-applicable methods mainly focus on the explanation of general quantum circuits. Unlike algorithm-specific explainability, they can be applied to arbitrary quantum circuits and are more flexible. One common approach is leveraging measured probability to depict each step's behavior in a quantum circuit. For example, Williams [66] and Lin et al. [31] showed the probabilities of all possible states after each quantum gate to interpret the gate's functionality. However, the vertical coordinates of their visualizations are based on qubits and basis states, making it challenging for users to inspect how quantum gates directly affect the probabilities of basis states. Lamy [47] studied how to reveal the gate effect by visualizing the change of quantum state in each step with a rainbow box design, while preserving the visualization of phases. Moreover, Van de Wetering [60] proposed a graphical representation of a linear map between qubits. Another type of work focuses on explaining the noise in quantum circuits. For example, Ruan et al. [48] introduced a visualization approach for the awareness of noise hidden in quantum computers and compiled quantum circuits. Meanwhile, Quirk [46] and Q-Sphere [41] also enable users to interact with quantum circuits via a web-based platform. While all the above methods focus on visualizing the quantum circuit evolution via the sequence of basis states' probability, our work aims to depict the development of a quantum circuit by visualizing the relationship of all basis states across all steps, enabling users to inspect the creation and development of quantum states with greater clarity.

3 Background

This section introduces the background of quantum computing relevant to our study, including quantum states and quantum circuits.

3.1 Quantum State

In quantum computing, quantum states are the mathematical entities that provide the probability of multiple basis states. The true power of quantum computing derives from the exponentially increasing state space, as there will be 2^N basis states simultaneously for a specific quantum state with N qubits [25, 26]. Recalling that for one qubit, the single-qubit state can be expressed as $|\phi\rangle = \alpha|0\rangle + \beta|1\rangle$. The notation ~~in~~ $|\phi\rangle$, namely Bra-ket notation [18], is a standard mathematical framework used frequently to represent quantum states. Generally, any quantum state with n qubits can be represented as a linear combination of 2^n basis states:

$$|\phi\rangle = \alpha \cdot |0\cdots00\rangle + \beta \cdot |0\cdots01\rangle + \cdots + \gamma \cdot |1\cdots11\rangle, \quad (1)$$

where the complex number $\alpha, \beta, \cdots \gamma$ are called quantum amplitudes (*a.k.a.* amplitudes) which is used to describe the basis state (*e.g.*, $|0\cdots01\rangle$) of a quantum state. An arbitrary amplitude (*e.g.*, α) can be expressed as a complex number:

$$\alpha = a + b \cdot i, \quad (2)$$

where a is the real part, and $b \cdot i$ is the imaginary part (i is the imaginary unit). Note that the amplitude of any quantum state can be used to determine the probability of measuring the corresponding basis state, which can be written as follows:

$$Pr(|0\cdots00\rangle) = |\alpha|^2 = |a|^2 + |b|^2. \quad (3)$$

Since the amplitudes of all basis states satisfy a normalization constraint that the sum of the probabilities of all basis states equals 1, thus all amplitudes satisfy $|\alpha|^2 + |\beta|^2 + \cdots + |\gamma|^2 = 1$. Note: We use the phrase "measured probability" in this paper to refer to the probability of a certain basis state if the quits were measured. Also, we leverage the quantum simulator to access the measured probabilities before the final measurement at the end of the circuit execution.

3.2 Quantum Circuit

Similar to classical circuits, quantum circuits implement quantum algorithms by a sequence of physical gates acting on one or more qubits. The manipulation of a quantum circuit can be represented as a calculation of unitary matrices [67]. In this paper, we refer to each manipulation module highlighted by the grey rectangle as a *block*. Thus, the execution of an arbitrary quantum circuit consists of the matrix calculation of a set of blocks, as illustrated in Fig. 4C.

Upon completing the final quantum gate, the execution result would be measured for the probability distribution of all basis states. Note that the intermediate quantum state after each gate's unitary transformation can be measured if the device is a quantum simulator [11]. In contrast, only the final quantum state can be obtained for a real quantum computer due to the collapse of the quantum state upon measurement. Hence, for intermediate states, the visualization takes place in a "god mode" where the probabilities are known although the state is not actually measured.

4 Informing the Design

In this section, we first introduce the preliminary study, along with the design requirements distilled from the study. Then we introduce the dataset we used to fulfill the design requirements.

4.1 Preliminary Study

The primary goal of the preliminary study is to collect the design requirements faced in the routine tasks of quantum computing users. Following the guideline [50] of task abstractions for the design study, we designed the preliminary study as follows:

Participants: The study involved six domain experts (**P1-6**) (6 males, $age_{mean} = 36.5$, $age_{sd} = 4.9$) in quantum computing from educational institutions and a national research laboratory. Specifically, **P1-3** are professors from three different universities in Singapore and the U.S. **P4** is a research scientist from Pacific Northwest National Laboratory, and **P5-6** are two Ph.D. students whose research direction is quantum computing. Among them, **P1-2** and **P5** are working on Quantum Machine Learning, while **P3-4** and **P6** study Quantum Systems, Quantum Chemistry and Quantum Error Modeling, respectively. All the experts have an average of 6.8 years of research and development experience in quantum computing.

Procedures: For five months, we collaborated closely with the experts in quantum computing to conduct the preliminary study. To ensure our visualization system was tailored to seamlessly fit into domain users' routine tasks, we divided the whole procedure into two separate sessions. First, we began the first session by performing one-on-one, semi-structured, hour-long interviews with all the domain experts. During the interview, we posed carefully-crafted questions (see Appendix A) relevant to the interpretability improvement of quantum circuits. For the second session, we summarized the initial design requirements and developed a low-fidelity prototype to meet the basic needs according to their feedback. Next, we presented this prototype to the experts for iterative expert tests in the next three months. They were then asked to explore the prototype freely and share their concerns and suggestions in a think-aloud manner; we then use their feedback to refine and improve the prototype accordingly. Throughout the study, we meticulously recorded observations and notes for each interview and discussion.

4.2 Design Requirements

We distilled the collected feedback from the preliminary study to inform our design. Overall, we summarized users' general process as two levels of analysis, *i.e.*, global analysis and local analysis. Specifically, the global analysis aims to explain the effects of quantum gates from a high-level perspective, while the local analysis provides a more fine-grained explanation for the *quantum states* by illustrating the rationale of the measured probability of each basis state.

For the global analysis, users need to be aware of the functionality of quantum gates of quantum circuits from the following aspects:

R1 **Provide an overall summary of the quantum circuit.** Five participants (**P1-4, P6**) emphasized the importance of providing users with a coarse-grained overview of the whole quantum circuit. They all agreed with the idea of summarizing the evolution of quantum states regarding the temporal changes of *probabilities*, making it easier for users to interactively select the blocks of interest from a large number of operations of quantum gates. **P2** also mentioned the necessity to break the *blocks* into a linear sequence of the individual gate operation, namely *steps*, to illustrate the effect of each quantum gate better.

R2 **Explain the effect of quantum gates visually.** All participants (**P1-6**) strongly suggested that the visual designs should focus on the detailed explanation of quantum gates, which are the most basic ingredients of quantum circuits. Specifically, three participants (**P1, P5-6**) encouraged us to utilize the qubit states to depict the evolution of quantum states. Meanwhile, the other three experts (**P2-4**) also expressed the need to *"visualize the quantum gate's effect via comparing how the amplitudes change the measured probability before and after the quantum gate."*

R3 **Support the trace-back analysis of quantum states.** Three participants (**P1, P3, P5**) expected the approach to enable the trace-back analysis of quantum states. They all confirmed that it is significant to visually reveal how a specific quantum state was generated from the beginning of the quantum circuit. *"I have no idea about how a quantum state is formed and by what kind of gate operations before,"* **P1** commented, *"I hope it can inform me of its evolution intuitively."* Moreover, **P3** emphasized that the intuitive visualization of the original quantum circuit can significantly flatten the learning curves for domain users.

For local analysis, the following requirements are crucial for visualizing quantum states:

R4 **Explain the probability of basis states visually.** All participants (**P1-6**) confirmed that it would significantly help to inform users of each basis state's probability change, enhancing their confidence in understanding the effects of the quantum gates. In particular, four participants (**P1-3, P5**) emphasized the importance of visually correlating the amplitudes and probabilities other than by a set of individual visualizations (*e.g.*, several bar charts), because they believed that the explicit and correlated visual channels could intuitively highlight how amplitudes determined the measured probabilities.

R5 **Support the visualization of multi-qubit quantum states.** According to the suggestions from four participants (**P1, P3, P5-6**), the most widely-used visualization for quantum states, *i.e.*, Bloch Sphere, cannot support the multi-qubit state visualization. **P3** commented that this issue is unacceptable because the real power of quantum computing, *i.e.*, entanglement, requires multiple qubits. **P6** also said *"I really hope there exists a visual representation to make the multi-qubit state more intuitive."*

R6 **Address the issues of visual scalability.** **P1** and **P3** pointed out the issue of visual scalability. Specifically, P1 emphasized that scalability issues are typical quantum-specific problems that need to be addressed. Also, **P1** commented *"There will be a substantial quantity of basis states in the common cases."* **P3** also comments that visualizing many basis states is a complex task, given the requirement to display both the probability and amplitudes of each basis state concurrently.

4.3 Dataset

Building upon the above design requirements, we developed the system *QuantumEyes* based on *Qiskit* [42], which is an open-sourced framework for the implementation of quantum circuits. We utilized a quantum simulator, *i.e.*, *AerSimulator* [2], to extract quantum states. The raw dataset extracted contains the properties of the quantum circuit: the sequence and implementation of quantum gates on the individual qubits, the state vectors of the quantum states over each step, and the transformation matrices of the quantum gates.

Next, to obtain the probability of each basis state, we leveraged Equation 3 to calculate the amplitudes from the quantum state's state vector. Also, we decomposed the matrix of state vector by rows to extract all basis states of a quantum state, making it available for analysis of the trajectory of the quantum states (see Appendix E). Furthermore, based on the principle of unitary transformation [55], we deconstructed each block (Fig. 4 C) into multiple *steps* (Fig. 4 B) to better clarify the workflow of a quantum circuit.

5 *QuantumEyes*

We proposed *QuantumEyes*, an interactive visualization system to enhance the interpretability of quantum circuits. The architecture of *QuantumEyes* consists of three tightly-connected modules: (1) data storage module, (2) data processing module, and (3) visualization module. In particular, the data storage module stores all raw input data of the original quantum circuit. The data processing module supports the data preparation procedure before visualization, including the probability calculation of each quantum state, the decomposition of state vectors for state evolution analysis, and the generation of the transformation representation based on the qubit states. The visualization module reveals insights hidden in the quantum circuits, where three views (*i.e.*, Probability Summary View, State Evolution View, and Gate Explanation View) are applied for the global analysis and the State Comparison View with a novel design (*i.e.*, *dandelion chart*) is used for the local analysis. Furthermore, we also implement an original quantum circuit (Fig. 4 C), enabling domain users to efficiently conduct the comparative analysis with our visual designs. The system interface of *QuantumEyes* is shown in Appendix D.

5.1 Probability Summary View

We propose the Probability Summary View (Fig. 2 A) to provide an intuitive summary of the quantum circuit in terms of probability changes of basis states over each step (**R1**).

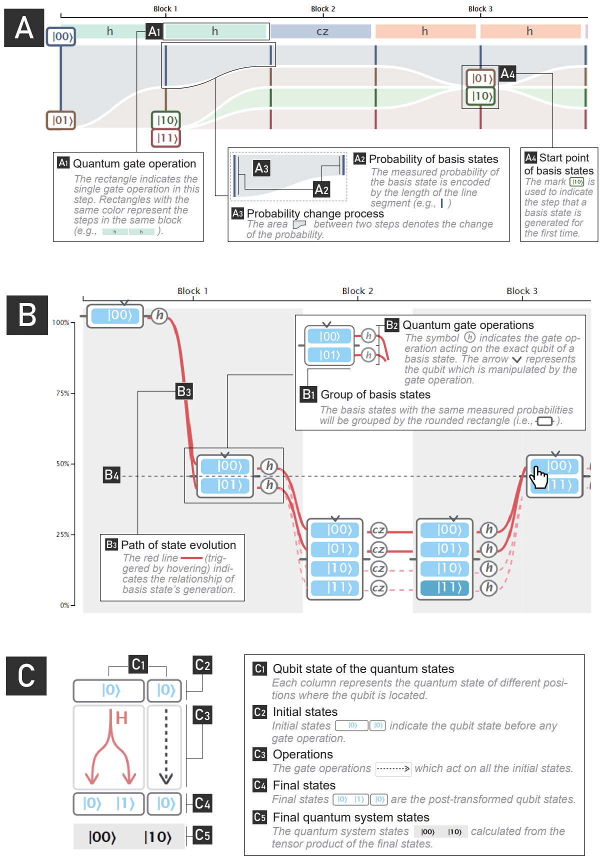


Fig. 2: The three coordinated views in *QuantumEyes* for global analysis. (A) Probability Summary View summarizes a quantum circuit via all basis states' temporal change of probabilities. (B) State Evolution View supports a fine-grained analysis of the basis states' evolution across each step. (C) Gate Explanation View visually explains the effect of quantum gates from the perspective of the qubit state.

We leverage the stacked area chart to portray the basis state's measured probability on each step, where the length of line segments encodes the probability (Fig. 2 A_2). Specifically, we use a set of line segments arranged vertically to reveal the probability proportion at each step. The total vertical length of all line segments at each step is a constant as the sum of all basis states' probabilities will always be 1. Also, we utilize the area (Fig. 2 A_3) to highlight the probability change of each basis state between steps. Moreover, we use a set of rectangles to denote the hierarchy of blocks and steps, where the rectangles in the same color are in a common block (Fig. 2 A_1). Note that the order of qubit labeling in the annotation is from left to right, while the qubit order in the view of Original Quantum Circuit is from bottom to top. Furthermore, we append the annotations (*e.g.*, $|01\rangle$) at the left-most area (Fig. 2 A_4) to depict the creation of a basis state. To enable the drill-down analysis from the summary of the quantum circuit (**R1**), users can interactively brush the steps of interest in Probability Summary View.

5.2 State Evolution View

The State Evolution View (Fig. 2 B) enables a drill-down analysis of the evolution of quantum states such as the separation and merging of quantum states for Hadamard gates [3] (**R2**). The design also supports the trace-back analysis (**R3**), making users aware of how a basis state was generated and further transformed by quantum gates.

We visualize the evolution of all the basis states using a graph-like design. Due to the consistency of the encoding of the horizontal axis, State Evolution View can also support a summary of quantum circuits (**R1**), enabling users to better compare with the Probability Summary View. The horizontal coordinate indicates the steps of the quantum circuit, while the vertical coordinate represents the basis state's measured probability. Meanwhile, prior work has also explored the visualization of quantum state evolution. For instance, Lin et al. [31] and Karafyllidis et al. [28] studied how to explain the behaviors of the overall quantum circuit using the encoding of color. Williams [66] utilized the length to indicate the measured probability of single qubits. Different from them, we use the vertical height of each basis state to encode their respective measured probability, which can better support the probability analysis by inspecting the pattern distribution of basis states. We use rounded rectangles to represent the entity of the basis state. Meanwhile, those basis states with the same probability are grouped by the outer rectangle (*i.e.*,), as shown in Fig. 2 B_1, where the outer rectangles' short line segments refer to each group's measured probability. Moreover, we encode the evolving relationship between the two steps using pink dotted lines. To indicate the gate operation, we use a symbol with the acronym inside after each basis state (Fig. 2 B_2); we then mark the qubit that the quantum gate acts on by the arrows. Note that the rectangles of basis states will be colored in light blue if the phase (*i.e.*, the sign of the state vector) is positive; otherwise, it will be colored in blue for the negative phase. We enable flexible interactions to enhance the usability of the system for users within the domain (**R3**). Precisely, users can hover over the specific state to analyze the evolution path highlighted in red lines (Fig. 2 B_3).

5.3 Gate Explanation View

The Gate Explanation View (Fig. 2 C) aims to allow users to understand a gate operation based on the qubit state (**R2**). We first deconstruct the quantum system states (*e.g.*, $|01\rangle$) into qubit states (*e.g.*, $|0\rangle$ and $|1\rangle$); we then visualize the explanation via a table-like design. We group it into the *global* category since it is actually proposed to explain the quantum gates, which belong to the components of the quantum circuits.

We define an arbitrary transformation as three parts, *i.e.*, the initial state, operation, and the final state; we then represent the three parts with the table's first, second, and third row, respectively. The column denotes each qubit in the original basis state. Meanwhile, we apply various colored lines (*e.g.*, for Hadamard gates) to represent the operation of quantum gates acting on the individual qubits (Fig. 2 C_3). Note that the operation will be represented as the dotted grey line if no quantum gate acts on a qubit. The list of all the implemented representations of gate operations can be found in Appendix F.

We use a transformation of the Hadamard gate as an example (Fig. 2 C) for illustration. Given a basis state of $|00\rangle$, the post-processed initial state is $|0\rangle$ and $|0\rangle$ for the first and second qubit respectively. After the Hadamard gate acting on the first qubit $|0\rangle$, the quantum state of the first gate converts to a state in superposition, *i.e.*, $|0\rangle$ and $|1\rangle$ each with a probability of 0.5, while the second qubit without any gate operation keeps the original state, *i.e.*, $|0\rangle$. Thus, the final state will be $|00\rangle$ ($|0\rangle \otimes |0\rangle$) and $|10\rangle$ ($|1\rangle \otimes |0\rangle$) through the tensor product operation.

5.4 State Comparison View

To enable the explanation of measured probability (*i.e.*, local analysis), we propose *dandelion chart*, a novel geometrical representation to visually explain the measured probabilities of basis states (Fig. 3). According to the quantum theory, we encode the amplitudes by 2D shapes to visualize arbitrary quantum states, including multi-qubit states (**R5**). We also visually correlate the probability with the corresponding amplitudes based on geometry principles to explicitly explain the measured probability of basis states (**R4**). Moreover, *dandelion chart* allows users to mitigating the visual clutter of numerous basis states via a geometry-based approach (**R6**).

The *dandelion chart* is incorporated into the State Comparison View to facilitate the comparison between two quantum states using a pair of *dandelion charts*, as shown in Fig. 4 $B_{4\text{-}7}$.

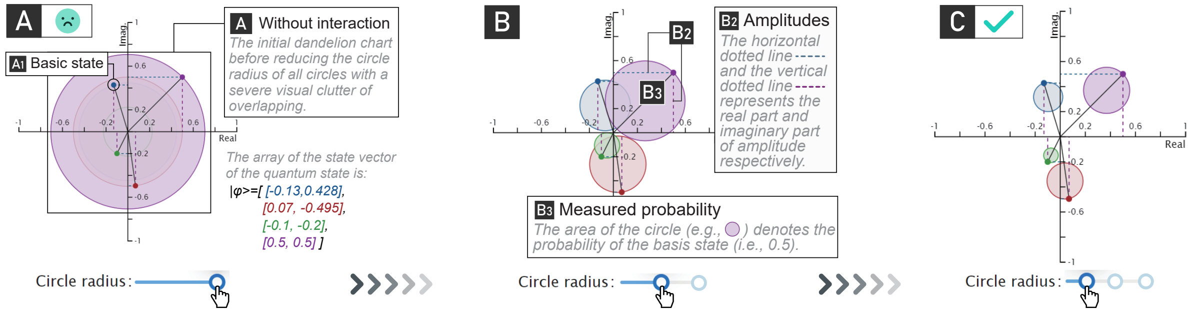


Fig. 3: The *dandelion chart* embedded in Probability Explanation View. (A) The *dandelion chart* before interaction with all circles overlapped with each other. (B) The *dandelion chart* after reducing the area of all circles by a factor of **0.5**, where the visual clutter is mitigated slightly. (C) The *dandelion chart* after reducing the area of all circles by a factor of **0.25**, where all circles are completely separated apart and can be compared clearly.

5.4.1 *Dandelion Chart*

Amplitudes. To visually represent a quantum state and the respective basis states, we leverage amplitudes of quantum states as they are the basic components of a specific quantum state [17, 58]. Recall that the amplitude of each state is intrinsically a complex number, consisting of a real and imaginary part, as illustrated by Equation 2. For each quantum state, we first apply a Cartesian coordinate system to represent the series of its amplitudes of each basis state based on Equation 1, where the x-axis encodes the real part, and the y-axis encodes the imaginary part. Thus, all the basis states of a quantum state are visualized as a set of points, as shown in Fig. 3 A$_1$. To further highlight amplitudes, the absolute values of real and imaginary parts are encoded by perpendicular lines in green and red from a point to the y-axis and x-axis (Fig. 3 B$_2$). Furthermore, we visualize the line connecting the point to the system's origin to highlight its position.

Probability explanation. According to Equation 3, the measured probability of each basis state can be calculated by the real and imaginary parts of the amplitudes. Meanwhile, based on geometry principles, the circle's area can be calculated using the radius, which is equal to the distance between the basis states' points and the origin of the system:

$$S_{circle} = \pi \cdot (|a|^2 + |b|^2), \tag{4}$$

where a and b are the real and imaginary parts of the amplitude. Thus, building on the Equations 3 and 4, we conclude that the area of the circle can represent the measured probability of a basis state as the area of the circle is proportional to the measured probability, as shown in Fig. 3 B$_3$. By this means, users are allowed to visualize the probability of the basis state in terms of their corresponding amplitudes indicated by the x- and y-coordinates of the points. However, there can exist a severe overlap between the circles (Fig. 3 A).

Notably, the prior work by Lamy [47] also utilized the rectangle area to allow the analysis of the measured probability of each basis state as well as entanglement and phase. Our proposed design, i.e., *dandelion chart*, can *explain* the measured probability regarding the *amplitudes* by the location of points in the Cartesian coordinate system, while preserving the capability of phase and entanglement visualization.

Visual clutter mitigation. We mitigate the visual clutter by scaling the area of circles through user interaction. By this means, all circles can be separated apart by decreasing all circles' radii, like the process from Fig. 3 A to Fig. 3 C.

If the radii of all the circles are reduced with the same factor k while keeping the point on the edge of the circle. Then the area of the circles satisfies the following equation:

$$S'_{circle} = \pi \cdot k^2 \cdot (|a|^2 + |b|^2), \tag{5}$$

where $k \in [0,1]$ is the factor for shrinking the area of circles. Meanwhile, based on Equations 3 and 5, then the area of the circle is still proportional to the measured probability due to the constant factor k. This means that users can scale the area of circles freely to mitigate the overlap while preserving the property of the representation of probabilities by the circles. Hence, *dandelion chart* can support probability explanations regarding amplitudes of the basis state through the user interaction of scaling the circles' radii. We name the design as "*dandelion chart*" due to the dandelion metaphor for each basis state like each entity in Fig. 3 C.

6 Case Study

In this section, we conducted two case studies on two popular quantum algorithms, *i.e.*, Grover's Algorithm [22] and Quantum Fourier Transform (*QFT*) [16], to demonstrate the usefulness of *QuantumEyes*. The users involved in the case studies are two quantum computing experts (E12 and E3) who also participate in the expert interviews in Section 7.

6.1 Case Study I - Grover's Algorithm

Grover's algorithm [22] is a quantum computing algorithm for searching an unsorted database, which is shown to be more efficient than classical algorithms. It works by repeatedly applying a process called amplitude amplification, which increases the probability of selecting the correct item(s) and decreases the probability of other items. We worked with E12, whose research interest includes applying Grover's Algorithm to speed up the unstructured searching problems. To find more insights behind the quantum circuit used in his research, E12 leveraged *QuantumEyes* to interactively explore Grover's Algorithm. Following the prior study [30], we implemented a 2-qubit Grover's Algorithm for the study.

Identifying the functionality block from the visualization. E12 began by examining the Probability Summary View and quickly noticed that the probability of State $|00\rangle$ was the largest at the beginning of the circuit. However, this dominance gradually diminished and was replaced by State $|11\rangle$ eventually. This transition can be observed through the length of the line segments and stacked areas at the final step (Fig. 1 A$_3$). He noted that this trasition occurred due to the functionality block of *amplitude amplification*, which effectively identified the target state, *i.e.*, State $|11\rangle$, despite having no prior knowledge of the specific basis state being sought (**R1**). Bearing this in mind, E12 became curious about the other functionality blocks of Grover's Algorithm, *i.e.*, the initialization and the *oracle*. With a clear goal, E12 found that the probabilities of the four basis states were identical, each having a probability of 0.25 as shown in Fig. 1 A$_1$. E12 identified the step following the two Hadamard gates (*i.e.*, H gates) at the end of the initialization, as all basis states are in a state of superposition with equal probability, precisely reflecting the characteristic of the initialization. E12 then noticed that the identified initialization was succeeded by a gate sequence of the "H-CX-H" combination. *"These three gates are commonly employed as an oracle that flips the signs of states, but I still have doubts about this and require further clarification to confirm my understanding."* We directed E12's attention to the *dandelion chart* for analyzing the amplitudes (**R2, R4**). Utilizing this function, E12 discovered the amplitudes of State $|11\rangle$ were flipped to negative values(Fig. 4 A$_2$). *"This is precisely what I anticipated. The flip of*

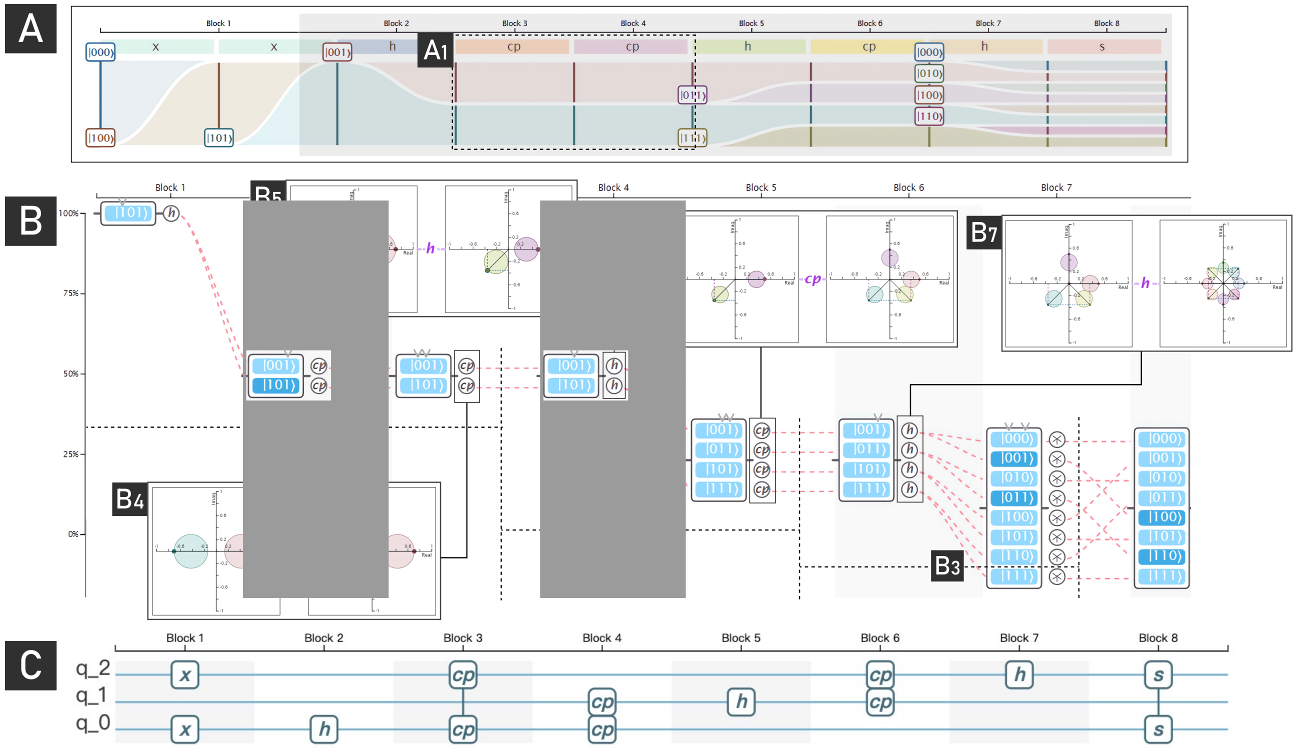


Fig. 4: The case for Quantum Fourier Transform. Three coordinated views (A-C) visualize the development of basis states for the global analysis, while State Comparison View with *dandelion chart* (B_4-B_7) explain how measured probabilities are determined by amplitudes for the local analysis. The *S* symbols in the quantum circuit diagram (C) indicate the SWAP gates.

State $|11\rangle$ aligns with the findings of the target state we speculated earlier. Moreover, the flip of the amplitude confirmed that the three quantum gates are an oracle for sure."

Uncovering the facts of the initialization and oracle. After the identification, E12 started to perform an in-depth analysis of each functionality block. By brushing the steps of initialization and the oracle from the stacked area chart, the State Evolution View was displayed as shown in Fig. 1 B_1. To delve further into the quantum gate's effect from a high-level perspective (**R2**), E12 clicked the "h" symbols of the Hadamard gate and displayed the visual explanations of the Hadamard gates (Fig. 1 C_1). Taking a close look at the appended view, E12 observed that the first qubit stayed still without any operation, while the second qubit was split into two states $|0\rangle$ and $|1\rangle$ in superposition. E12 commented *"I am truly impressed that this visualization can evidently explain why the final states are $|00\rangle$ and $|01\rangle$ through the decomposition of the gate operation."*

The expert then moved on to the analysis of the oracle, which is used to flip the signs of the target state (*i.e.*, $|11\rangle$). *"I am curious about how the target $|11\rangle$ was generated before the amplitude process"* (**R3**), E12 commented. Through hovering over the target state $|11\rangle$, E12 noticed an eye-catching red path to indicate how the $|11\rangle$ was generated, as shown in Fig. 1 B_1. He could confidently identify that the States $|01\rangle$ and $|11\rangle$ are the origin of the evolution, which were merged by the following Hadamard gate (Fig. 1 C_2).

Exploring the hidden insights of the amplification. E12 proceeded to analyze the functionality block of amplification, which is employed to amplify the probability of the flipped target state. By brushing the corresponding steps in the stacked area chart, E12 got a quick intuition of the operations of the CNOT gate (Fig. 1 C_3 and C_4) and the NOT gate (Fig. 1 C_5). To determine the reason for the sudden increase in the probability of State $|11\rangle$ (**R2**), E12 took a glance of Fig. 1 C_6 and quickly noticed that the Hadamard gate merged the first qubit's state (*i.e.*, $|0\rangle$ and $|1\rangle$) and generated a new qubit state (*i.e.*, $|1\rangle$). *"This is mainly because the first state $|0\rangle$ is negative, leading to the new state of $|1\rangle$ other than $|0\rangle$"*, E12 said, *"However, I cannot still understand why the probability changes into 1 instead of other numbers"* (**R4**).

Thus, as hinted by us, E12 further moved to the *dandelion chart* of the step by clicking the last step's background. After a glance, he noticed there are two states (*i.e.*, $|01\rangle$ and $|11\rangle$) at the left system and only one state (*i.e.*, $|11\rangle$) at the right with a symbol denoting the operation gate (Fig. 1 B_3). E12 found that the imaginary parts of all states are zero, as indicated by their zero y-coordinates. Furthermore, the real part of State $|11\rangle$'s amplitudes changed from around 0.7 in the left chart to -1.0 in the right-hand chart. *"I am surprised that the dandelion chart tells me that the flip of phase did not cause the change of the probability, the real part of the amplitude actually makes the state's probability twice its initial state."*

6.2 Case Study II - Quantum Fourier Transform

We worked with E3, whose research direction is Quantum Uncertainty, to understand a widely-used quantum algorithm, *i.e.*, Quantum Fourier Transform (*a.k.a.*, QFT) [51]. The QFT algorithm converts the amplitudes of a quantum state into the corresponding values in the frequency domain, which is similar to what the classical Fourier Transform does with signals. It forms a foundation for other quantum algorithms, such as Shor's Algorithm [52]. We implemented the quantum circuit following the guidelines of *Qiskit* [45] and presented *QuantumEyes* to E3 to perform our study.

Understanding the architecture of QFT algorithm. The expert E3 started by brushing the whole quantum circuit from the probability overview because he thought the QFT algorithm is an entity that cannot be split into different functionality blocks. Indicated by the first two X gates with probabilities of 1.0, E3 commented, *"These two X gates are for the state preparation because the lengths of the line segments remains the same during Block 1."* Meanwhile, he speculated the number to be mapped is 5 due to the decimal of State $|101\rangle$. After identifying the number to be mapped, E3 started to investigate the quantum circuit architecture of the QFT algorithm (**R2**). By exploring the State Evolution View along with the original circuits, E3 quickly found that the three key processes of the algorithm *"I can easily identify the three iterations of QFT* (as shown in Fig. 4 B_1, Fig. 4 B_2, and Fig. 4 B_3) *from the three continuous processes with the downward trend of probabilities from the middle view, each making the probability drop to 0.25"*. He also praised the advantage of the evolution view to intuitively reveal the temporal change of states' probabilities along the circuit, making the analysis of the gate's functionality more efficient and smooth.

Disclosing the implicit reasons of the measured probability. E3 then glanced at the probability summary of the QFT algorithm and found that the probability of the two States $|001\rangle$ and $|101\rangle$ did not change after the two Controlled Phase gates (Fig. 4 A_1). Thus, he planned to find more hidden insights about this phenomenon by drilling down to the local analysis using *dandelion chart*(**R4, R5**). According to the geometrical representation of Fig. 1 B_4, E3 noticed that the circle of State $|101\rangle$ rotates around 45 degrees anticlockwise after the Controlled Phase gate, making the amplitudes change but preserving the circle area. *"This design is fascinating to me because I can analyze the gate's effect from a perspective of geometry intuitively."* Next, he clicked the following Hadamard gate to find the reason for superposition using the *dandelion chart*. From Fig. 4 B_5, E3 noticed the both of the two original States $|101\rangle$ and $|001\rangle$ became two smaller circles. *"Before today, I can only observe the four states with the same probability of 0.25 after Hadamard gates. It is brilliant to build a mathematical intuition of the measured probability and the amplitudes."* After analyzing the individual quantum gate, E3 planned to investigate how the QFT algorithm represents a random quantum state by a series of continuous basis states (*i.e.*, $|000\rangle \cdots |111\rangle$). Hence, E3 clicked the last two quantum gates before the final SWAP gate and then adjusted the radius to separate all circles (**R6**), as shown in Fig. 4 B_6 and Fig. 4 B_7. *"This actually matches what I expected,"* E3 commented *"From the first chart, I realized that the Controlled Phase gate can only 'rotate' a state but never 'separate' a state into multiple states."* E3 further noticed that the four states are located in four different directions (*i.e.*, cardinal directions and diagonal). And then, the Hadamard gate generates each state into a new basis states in the opposite direction, making it possible to handle eight basis states for 3 qubits. *"The dandelion chart provides me a holistic picture of how the quantum gate changes the amplitudes of basis states, which makes the analysis of amplitudes more effective than ever before."*

7 Expert Interview

We further conduct a well-designed interview with actual domain experts to demonstrate the effectiveness and usability of *QuantumEyes* and the embedded *dandelion chart*.

7.1 Study Design

Participants and apparatus. We recruited 12 domain experts (**E1-12**) (12 males, $age_{mean} = 34.0$, $age_{sd} = 5.8$) from 6 different educational institutions (**E1-12**) in the U.S. to join our in-depth expert interview. These participants were selected by mainly considering their research background of quantum computing and check whether they have relevant research or development experience, guaranteeing the reliability of the collected feedback. More specifically, five participants (**E1, E9-12**) are working on Quantum Error Mitigation, six experts (**E4-7, E8, E13-14**) study Quantum Machine Learning (QML), two experts (**E2-3**) are working on Quantum Uncertainty, and one expert (**E7**)'s research direction is Quantum System Design. The interview was conducted via the online Zoom meeting. Also, all experts were asked to use a monitor with a resolution of 2560×1600 beforehand.

Procedures. The entire study was conducted on the online system *QuantumEyes*. We carried out the one-on-one, semi-structured study for all experts. Specifically, we first introduced the visual design of all views of *QuantumEyes* along with *dandelion chart*. Afterward, we invited all participants to accomplish six pre-defined tasks using *QuantumEyes*. The first four tasks are designed to evaluate the effectiveness of our proposed visualization designs for global analysis, including analyzing overall trend of basis state probabilities, identifying the gate effect and explaining the gate effect in terms of the changes of basis states and qubit states. The remaining two tasks aim to evaluate the effectiveness of dandelion chart's effectiveness for local analysis. The detailed task list can be found in Appendix H. We then asked them to verbally explain how the quantum states are evolving across the quantum circuit. The aforementioned process lasts approximately 40 minutes. After completing the tasks, all participants were encouraged to provide feedback on all the proposed visual designs in a think-aloud manner. Furthermore, we also invited participants to finish a post-study questionnaire (Table 1) and rate *QuantumEyes* using a 7-point Likert scale. The dimensions of the questionnaire are followed by the prior work by Ruan et al. [48]. The post-study interview lasted approximately 20 minutes, during which we recorded and took notes about the entire study process.

Q1	The workflow of global and local analysis can explain the quantum circuits comprehensively.
Q2	The system can effectively support the evolution analysis of each basis state.
Q3	The system can intuitively explain the gate effect via the visualization of qubit states.
Q4	The dandelion chart can effectively explain the measured probability based on the amplitudes.
Q5	The system is easy to learn.
Q6	The publicly-available QuantumEyes system is helpful for domain users.
Q7	I would like to use the QuantumEyes system to better understand quantum circuits in the future.
Q8	The user interaction of the system is smooth.
Q9	The user interaction is easy to use for domain users.
Q10	The overall design is easy to understand.
Q11	For global analysis, the three coordinated views are helpful in understanding the effects of quantum gates.
Q12	For local analysis, the dandelion chart is useful to visualize how amplitudes affect the probability intrinsically.

Table 1: The questionnaire consists of four parts, *i.e.*, the effectiveness for interpretability enhancement of quantum circuits (**Q1-4**), the usability of the visualization system (**Q5-7**), the user interaction (**Q8-9**), and the visual designs (**Q10-12**).

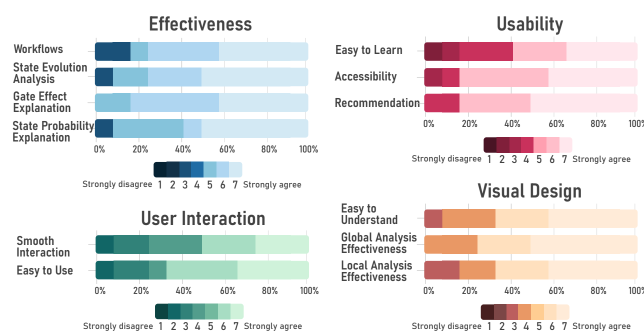


Fig. 5: The summary of the feedback of the questionnaire.

7.2 Result

We summarized all collected feedback regarding the four aforementioned aspects of evaluation as Fig. 5.

Effectiveness. Most participants appreciated the effectiveness of *QuantumEyes* to enhance the interpretability of quantum circuits ($rating_{mean} = 6.02, rating_{sd} = 1.18$). **E3-6** agreed that the workflow of global and local analysis is exactly what quantum computing users expect to see to explain the effects of quantum gates. Meanwhile, the trace-back analysis is praised by **E1-2** and **E9**. *"I need to manually calculate the state vectors to figure out how a state was produced and developed in my daily work before. This function provided by QuantumEyes is really fascinating to me"*, E2 said. Furthermore, most participants (**E1-7, E9-11**) highly appreciate the novel design *dandelion chart*, *"which is helpful to grasp the measured probability of basis states."* **E11** also commented that *QuantumEyes* can help him with circuit design and debugging due to the intuitive visualization of probability regarding the state's amplitudes.

Usability. The majority of participants applauded the usability of *QuantumEyes* in interactively enhancing the quantum circuit's interpretability ($rating_{mean} = 5.88, rating_{sd} = 1.65$). **E2-5** mentioned that the visualization system is user-friendly for quantum computing researchers and learners. Among them, **E4** commented, *"I can easily interact with the interface and accomplish all tasks, even though I do not have any background in visualization before."* **E7** and **E12** emphasized that they prefer the easy-to-understand visualizations, and *QuantumEyes* indeed provides the visualizations that they are familiar with in their daily routine tasks, like the original quantum circuit diagram and the visualization of the transformation of the qubit states.

User interactions. Most participants generally agreed that the user interactions in *QuantumEyes* is easy-to-use for quantum computing users ($rating_{mean} = 5.54, rating_{sd} = 1.66$). Among all feedback, **E3-4** and **E9** gave highly positive feedback for the interactions of decreasing the circle radius in *dandelion chart* to reduce the visual clutter. **E3** pointed out he feels struggle to adopt Bloch Sphere to inspect only the single-qubit state. Dandelion chart addresses the limitations perfectly while preserving the characteristic of displaying the quantum amplitudes. Meanwhile, E9 confirmed that reducing the circle area is feasible *"because users always need to focus on the circle with the largest area."* Furthermore, **E12** also expressed the desire to recommend *QuantumEyes* to his research group members due to the easy-to-use system interactions.

Visual designs. Most participants gave positive feedback about the visual designs in *QuantumEyes* ($rating_{mean} = 6.05, rating_{sd} = 0.99$). Specifically, **E5** mentioned that the visual designs for global analysis are informative. *"I like the visualization to show the how state evolves because it can directly tell when and how a basis state is developed. This characteristic would truly aid the analysis of quantum algorithms in my daily tasks."* Also, **E7** was willing to adopt *dandelion chart* for his own local analysis of the quantum states, *"To my surprise, this design is brilliant because everyone can find the rationale of probability changes without the complex matrix calculation, even for the beginners in quantum computing."*

Suggestions. In addition to the positive feedback, several participants also offered constructive suggestions. **E4** suggested that incorporating a feature to fold and unfold the blocks would be helpful for comparative analysis. **E9** also noted that *QuantumEyes* could be extended to visualize the temporal change of parameters in variational quantum circuits. **E10** expressed that a transition might be useful to highlight the difference when comparing a pair of *dandelion charts*.

8 Discussion

In this section, we first summarize the lessons learned during the development of *QuantumEyes* and *dandelion chart*. Then, we discuss the limitations of our proposed visual designs.

8.1 Lessons Learned

We reported the learned lessons from the development of *QuantumEyes*.

Indispensable necessity of visualization to interpret quantum computing. During the process of working with domain experts in the requirement collection and evaluation, they confirmed the great importance of interpreting quantum computing using visualization approaches. According to the actual use of *dandelion chart*, experts appreciated the impressive design, while also pointed out that quantum computing is not transparent for users to learn, which is exactly the domain where visualization can aid. Thus, they also mentioned that they preferred the designs with linked visual channels to offer the explanation intuitively.

Design considerations tailored for quantum computing users. By working with domain experts, we realize that lowering the learning costs of the proposed visual design for domain users is significant. In our study of design requirements, all participants preferred solutions that were easy to learn, simplifying the paradigm shifts from reading to understanding. For example, the overview of the probability is appreciated and used as the starting point of the system. Also, they praised the implementation of the original quantum circuit in *QuantumEyes* because the comparative analysis with the original circuit can significantly shorten their learning curves.

8.2 Limitations and Future Work

There are still several limitations of *QuantumEyes*.

Application scope. All the participants highly appreciate the effectiveness of *QuantumEyes* in helping quantum computing developers and users understand the working mechanism of static quantum circuits, which is the widely-used quantum circuits. However, with the growth of another type of quantum circuits, *i.e.*, variational quantum circuits (VQC), also gain more and more attention. Our novel design *dandelion chart* can be seamlessly applied to VQC applications to analyze quantum state evolution. However, the visual analytics system (i.e., *QuantumEyes*) as a whole cannot be directly applied to VQC for now. In future work, we plan to extend *QuantumEyes* to analyze variational quantum circuits and other features such as the generalisation of the visual feature "Path of state evolution" of *QuantumEyes* to a "Path of Bloch coefficient/Pauli probability evolution".

Scalability. The evaluation has demonstrated that our visualization works well for visualizing the states of two and three qubits. However, due to the limited screen space, the visualization components in *QuantumEyes* for global analysis, i.e., the Probability Summary View, State Evolution View and Gate Explanation View, may suffer from scalability issues with the increase of qubits in quantum circuits. The visualization component of *QuantumEyes* for local analysis, i.e., *dandelion chart*, has better scalability than the above three views, as it can reduce the radii of circles to explain the measured probabilities of more basis states. But when there are a large number of qubits in the quantum circuits, visual clutters may also appear. In the future, it is worth further exploration on how to enhance the scalability of *QuantumEyes*.

User-friendly interactions. The availability of *QuantumEyes* gained positive feedback from all participants. However, there are still several limitations regarding the user interactions we collected during the interview. First, to aid the obscure connection between the State Evolution View and the original circuit, we plan to implement the folding and unfolding of basis states and their corresponding quantum gates. Also, the comparison of two quantum states in *dandelion chart* can be improved by utilizing the transition of circles to highlight the effect of quantum gates. Further, to address the issue that the entangled states sometimes cannot be clearly distinguishable in the *dandelion chart*, we plan to add extra visual elements to highlight those basis states which are entangled using sector length distributions. Last, *QuantumEyes* is expected to enable the functionality of simulator customization and circuit data loading to enhance the system's flexibility.

9 Conclusion

We present *QuantumEyes* an interactive visualization system to enhance the interpretability of quantum circuits. By working closely with domain experts, we formulate six design requirements in terms of two analysis levels to guide the design of our system. Specifically, we propose three coordinated views (*i.e.*, a Probability Summary View, a State Evolution View and a Gate Explanation View) to support the global analysis of the quantum state evolution over the whole quantum circuit. Further, we propose a novel geometrical visual design *dandelion chart* for local analysis, enabling users to visually analyze the correlation of basis states' probability and amplitudes based on geometry principles. We conduct two case studies and expert interviews to demonstrate the effectiveness and usability of the proposed visualization approaches. The result shows that our approaches can effectively facilitate domain users to better understand quantum circuits.

References

[1] Suppressing quantum errors by scaling a surface code logical qubit. *Nature*, 614(7949):676–681, 2023. doi: 10.1038/s41586-022-05434-1 2

[2] AerSimulator. `https://qiskit.org/documentation/stubs/qiskit_aer.AerSimulator.html`, Access Date: 2023-3-14. 4

[3] D. Aharonov. A simple proof that toffoli and hadamard are quantum universal. *arXiv preprint quant-ph/0301040*, 2003. 5

[4] J. B. Altepeter, E. R. Jeffrey, M. Medic, and P. Kumar. Multiple-qubit quantum state visualization. In *Proceedings of 2009 Conference on Lasers and Electro-Optics and 2009 Conference on Quantum Electronics and Laser Science Conference*, pp. 1–2. IEEE, 2009. 2

[5] F. Amini, S. Rufiange, Z. Hossain, Q. Ventura, P. Irani, and M. J. McGuffin. The Impact Of Interactivity On Comprehending 2d And 3d Visualizations Of Movement Data. *IEEE Transactions on Visualization and Computer Graphics*, 21(1):122–135, 2015. doi: 10.1109/TVCG.2014.2329308 3

[6] F. Arute, K. Arya, R. Babbush, D. Bacon, J. C. Bardin, R. Barends, R. Biswas, S. Boixo, F. G. Brandao, D. A. Buell, et al. Quantum supremacy using a programmable superconducting processor. *Nature*, 574(7779):505–510, 2019. doi: 10.1038/s41586-019-1666-5 2

[7] J. C. Bardin, D. H. Slichter, and D. J. Reilly. Microwaves in quantum computing. *IEEE Journal of Microwaves*, 1(1):403–427, 2021. doi: 10.1109/JMW.2020.3034071 2

[8] I. Bengtsson and K. Życzkowski. *Geometry of quantum states: an introduction to quantum entanglement*. Cambridge university press, 2017. 2

[9] J. Biamonte, P. Wittek, N. Pancotti, P. Rebentrost, N. Wiebe, and S. Lloyd. Quantum machine learning. *Nature*, 549(7671):195–202, 2017. 2

[10] F. Bloch. Nuclear induction. *Physical Review*, 70(7-8):460, 1946. 2

[11] I. Buluta and F. Nori. Quantum simulators. *Science*, 326(5949):108–111, 2009. 3

[12] D. Castelvecchi. IBM's quantum cloud computer goes commercial. *Nature*, 543(7644), 2017. 2

[13] V. N. Chernega, O. V. Man'ko, and V. I. Man'ko. Triangle geometry for qutrit states in the probability representation. *Journal of Russian Laser Research*, 38:416–425, 2017. 3

[14] V. N. Chernega, O. V. Man'ko, and V. I. Man'ko. Triangle geometry of the qubit state in the probability representation expressed in terms of the triada of malevich's squares. *Journal of Russian Laser Research*, 38(2):141–149, 2017. 3

[15] V. N. Chernega, O. V. Man'ko, and V. I. Man'ko. Correlations in a system of classical-like coins simulating spin-1/2 states in the probability representation of quantum mechanics. *The European Physical Journal D*, 73:1–5, 2019. 3

[16] D. Coppersmith. An approximate fourier transform useful in quantum factoring. *arXiv preprint quant-ph/0201067*, 2002. 6

[17] D. 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*, vol. 400, pp. 97–117. The Royal Society London, 1985. 6

[18] P. A. M. Dirac. A new notation for quantum mechanics. *Mathematical Proceedings of the Cambridge Philosophical Society*, 35(3):416–418, 1939. doi: 10.1017/S0305004100021162 3

[19] R. P. Feynman. Quantum mechanical computers. *Optics news*, 11(2):11–20, 1985. 2

[20] M. Galambos and S. Imre. Visualizing the effects of measurements and logic gates on multi-qubit systems using fractal representation. *International Journal on Advances in Systems and Measurements Volume 5, Number 1 & 2, 2012*, 2012. 3

[21] Google Cirq. `https://quantumai.google/reference/python/cirq_web/bloch_sphere/BlochSphere`, Access Date: 2023-6-28. 2

[22] L. K. Grover. A fast quantum mechanical algorithm for database search. In *Proceedings of the Twenty-eighth annual ACM Symposium on Theory of Computing*, pp. 212–219, 1996. 6

[23] S. Hallgren. Polynomial-time quantum algorithms for pell's equation and the principal ideal problem. *Journal of the ACM (JACM)*, 54(1):1–19, 2007. 2

[24] V. Hassija, V. Chamola, V. Saxena, V. Chanana, P. Parashari, S. Mumtaz, and M. Guizani. Present landscape of quantum computing. *IET Quantum Communication*, 1(2):42–48, 2020. doi: 10.1049/iet-qtc.2020.0027 2

[25] T. Hey. Quantum computing: an introduction. *Computing & Control Engineering Journal*, 10(3):105–112, 1999. doi: 10.1049/ccej:19990303 2, 3

[26] C. Hughes, J. Isaacson, A. Perry, R. F. Sun, and J. Turner. *Quantum computing for the quantum curious*. Springer Nature, 2021. 3

[27] IBM Qiskit. `https://qiskit.org/documentation/stubs/qiskit.visualization.plot_bloch_vector.html`, Access Date: 2023-6-28. 2

[28] I. G. Karafyllidis. Visualization of the quantum fourier transform using a quantum computer simulator. *Quantum Information Processing*, 2:271–288, 2003. 3, 5

[29] Lattice Surgery. `https://latticesurgery.com/online-compiler`, Access Date: 2023-6-28. 3

[30] A. Li, S. Stein, S. Krishnamoorthy, and J. Ang. Qasmbench: A low-level quantum benchmark suite for nisq evaluation and simulation. *ACM Transactions on Quantum Computing*, 4(2):1–26, 2023. 6

[31] S. Lin, J. Hao, and L. Sun. Quflow: Visualizing parameter flow in quantum circuits for understanding quantum computation. In *Proceedings of 2018 IEEE Scientific Visualization Conference (SciVis)*, pp. 37–41. IEEE, 2018. doi: 10.1109/SciVis.2018.8823602 2, 3, 5

[32] H. Mäkelä and A. Messina. N-qubit states as points on the bloch sphere. *Physica Scripta*, 2010(T140):014054, 2010. 2

[33] M. Miller and D. Miller. GraphStateVis: Interactive visual analysis of qubit graph states and their stabilizer groups. In *Proceedings of 2021 IEEE International Conference on Quantum Computing and Engineering (QCE)*, pp. 378–384. IEEE, 2021. 3

[34] T. Munzner. A nested model for visualization design and validation. *IEEE Transactions on Visualization and Computer Graphics*, 15(6):921–928, 2009. doi: 10.1109/TVCG.2009.111 2

[35] A. Narayanan. An introductory tutorial to quantum computing. In *IEE Colloquium on Quantum Computing: Theory, Applications and Implications (Digest No: 1997/145)*, pp. 1–1. IET, 1997. doi: 10.1049/ic:19970789 2

[36] R. Orús, S. Mugel, and E. Lizaso. Quantum computing for finance: Overview and prospects. *Reviews in Physics*, 4:100028, 2019. 2

[37] L. M. Phillips, S. P. Norris, and J. S. Macnab. *Visualization in mathematics, reading and science education*, vol. 5. Springer Science & Business Media, 2010. 2

[38] M. Piattini. Training needs in quantum computing. In *QANSWER*, pp. 23–30, 2020. 2

[39] A. Politi, J. C. Matthews, and J. L. O'Brien. Shor's quantum factoring algorithm on a photonic chip. *Science*, 325(5945):1221–1221, 2009. 2

[40] J. Preskill. Quantum computing in the nisq era and beyond. *Quantum*, 2:79, 2018. doi: 10.22331/q-2018-08-06-79 2

[41] Q-Sphere. `https://quantum-computing.ibm.com/composer/files/new`, Access Date: 2023-6-28. 3

[42] Qiskit. `https://qiskit.org/`, Access Date: 2023-3-14. 4

[43] Quantum Code Visualizer. `https://gui.quantumcodes.io/`, Access Date: 2023-6-28. 3

[44] Quantum computing statistics: Benefits and negative effects. `https://seedscientific.com/quantum-computing-statistics/`, Access Date: 2023-2-14. 2

[45] Quantum fourier transform. `https://learn.qiskit.org/course/ch-algorithms/quantum-fourier-transform`, Access Date: 2023-2-27. 7

[46] Quirk. `https://algassert.com/quirk`, Access Date: 2023-3-1. 3

[47] Lamy, Jean-Baptiste. Dynamic software visualization of quantum algorithms with rainbow boxes. In *International Conference on Information Visualization Theory and Applications (IVAPP)*, 2019. 3, 6

[48] S. Ruan, Y. Wang, W. Jiang, Y. Mao, and Q. Guan. VACSEN: A Visualization Approach For Noise Awareness In Quantum Computing. *IEEE Transactions on Visualization and Computer Graphics*, 29(01):462–472, 2023. 3, 8

[49] S. Ruan, R. Yuan, Q. Guan, Y. Lin, Y. Mao, W. Jiang, Z. Wang, W. Xu, and Y. Wang. VENUS: A Geometrical Representation for Quantum State Visualization. *Computer Graphics Forum*, 2023. doi: 10.1111/cgf.14827 2, 3

[50] M. Sedlmair, M. Meyer, and T. Munzner. Design study methodology: Reflections from the trenches and the stacks. *IEEE Transactions on Visualization and Computer Graphics*, 18(12):2431–2440, 2012. doi: 10.1109/TVCG.2012.213 3

[51] P. W. Shor. Algorithms for quantum computation: discrete logarithms and factoring. In *Proceedings 35th annual symposium on foundations of computer science*, pp. 124–134. Ieee, 1994. 7

[52] P. W. Shor. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM Review*, 41(2):303–332, 1999. 7

[53] A. Steane. Quantum computing. *Reports on Progress in Physics*, 61(2):117, 1998. doi: 10.1088/0034-4885/61/2/002 2

[54] Z. Tao, Y. Pan, A. Chen, and L. Wang. ShorVis: A Comprehensive Case Study Of Quantum Computing Visualization. In *Proceedings of 2017 International Conference on Virtual Reality and Visualization (ICVRV)*, pp. 360–365. IEEE, 2017. doi: 10.1109/ICVRV.2017.00082 3

[55] Torchquantum. `https://torchquantum.readthedocs.io/en/latest/`, Access Date: 2023-3-14. 4

[56] M. Tory, A. E. Kirkpatrick, M. S. Atkins, and T. Moller. Visualization task performance with 2d, 3d, and combination displays. *IEEE Transactions on Visualization and Computer Graphics*, 12(1):2–13, 2005. doi: 10.1109/TVCG.2006.17 3

[57] F. Toyama, W. Van Dijk, and Y. Nogami. Quantum search with certainty based on modified grover algorithms: optimum choice of parameters. *Quantum Information Processing*, 12:1897–1914, 2013. 2

[58] K. Valiev. Quantum computers and quantum computations. *Physics-Uspekhi*, 48(1):1, 2005. 6

[59] W. Van Dam, S. Hallgren, and L. Ip. Quantum algorithms for some hidden shift problems. *SIAM Journal on Computing*, 36(3):763–778, 2006. 2

[60] J. van de Wetering. Zx-calculus for the working quantum computer scientist. *arXiv preprint arXiv:2012.13966*, 2020. 3

[61] K. L. Vavra, V. Janjic-Watrich, K. Loerke, L. M. Phillips, S. P. Norris, and J. Macnab. Visualization in science education. *Alberta Science Education Journal*, 41(1):22–30, 2011. 2

[62] Visualizing 2-Qubit Entanglement. `https://algassert.com/post/1716`, Access Date: 2023-6-28. 2

[63] C.-R. Wie. Bloch sphere model for two-qubit pure states. *arXiv preprint arXiv:1403.8069*, 2014. 2

[64] C.-R. Wie. Two-qubit bloch sphere. *Physics*, 2(3):383–396, 2020. 2

[65] R. Wille, L. Burgholzer, and M. Artner. Visualizing decision diagrams for quantum computing (special session summary). In *Proceedings of 2021 Design, Automation & Test in Europe Conference & Exhibition (DATE)*, pp. 768–773. IEEE, 2021. doi: 10.23919/DATE51398.2021.9474236 2

[66] M. M. Williams. *QCVis: A Quantum Circuit Visualization and Education Platform for Novices*. PhD thesis, Harvard University, 2021. 2, 3, 5

[67] A. C.-C. Yao. Quantum circuit complexity. In *Proceedings of 1993 IEEE 34th Annual Foundations of Computer Science*, pp. 352–361. IEEE, 1993. doi: 10.1109/SFCS.1993.366852 3

A Interviews in the Preliminary Study

To carefully inform the designs, we conducted an in-depth preliminary study to collect feedback for the design requirements. Specifically, we divided the study into two sessions, i.e., the interview and the prototype test. We listed all the questions we asked in the interview process, as shown in Table 2. The feedback from the questions is used to implement the visualization system prototype. All participants were presented with the same questions and were asked to answer the questions in a think-aloud manner.

Q1	What property of quantum circuits do you mostly use in your routine tasks?
Q2	Which part of a quantum circuit do you think is most difficult to understand?
Q3	Which component of quantum circuits do you think is most useful for people to understand a circuit?
Q4	How to understand the measured result of a quantum circuit?
Q5	What is your expected way yo explain quantum gates?
Q6	Which aspects do you think visualization can help you with for quantum circuit interpretability?
Q7	How to lower the learning costs for domain users to use a tailored tool?

Table 2: The pre-defined questions used in the preliminary study for the session of the design requirement collection.

B Bloch Sphere

A Bloch Sphere, as shown in Fig. 6, is a widely-used representation to visual quantum states in the quantum computing community. Bloch Sphere utilizes a point on a unit sphere to represent a quantum state, where the angles with the axes indicate the *amplitudes* of the quantum state. Despite the prevalence, Bloch Sphere has several limitations which need to be improved urgently:

- Bloch Sphere cannot support the multi-qubit state visualization, while the entanglement of multiple qubits is the power to achieve the quantum advantages.
- Bloch Sphere cannot visualize the probability of each basis state intuitively - the only way to acquire the probabilities from Bloch Sphere is a manual calculation based on the angles with the coordinates.
- Bloch Sphere is a 3D geometrical visualization, which has been proven to perform worse than two-dimension counterparts when conducting precise measurements.

In this work, we propose a novel geometrical visualization named *dandelion chart*, which can address the above issues through multiple correlated 2D shapes. In addition, a *dandelion chart* can mitigate the scalability issue based on the geometry principle using user interactions.

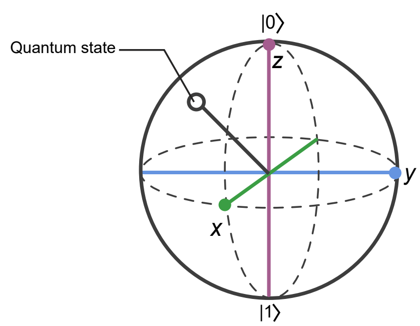


Fig. 6: The graphical illustration of Bloch Sphere.

C Usage of Dandelion Chart Package

According to the suggestions we collected from the post-study interview, several experts recommend we pack the methods of building a *dandelion chart* and publish it as a public-available web-based package. To this end, we published the package named *"dandelion_chart"*[3] to the online software registry, enabling a quick build for developers to visualize a specific quantum state.

The exported function to build *dandelion chart* requires six parameters, i.e., the array of the state vector of the quantum state, the array of the names of all basis states, the container to draw the design, the size of the chart, the position of the chart, and a factor to resize the circles in the chart. We also publish a function named *"generateStates()"* to generate all necessary state names based on the number of digits. For more detailed usage instructions and examples, please refer to the package homepage.

D System Interface

The system interface of *QuantumEyes* is shown in Fig. 7. Specifically, the system consists of four views, an original quantum circuit, and a control panel.

The summary view of a quantum circuit, as shown in Probability Summary View, provides an overview of the measured probability. For a more detailed analysis of the basis states across each step, State Evolution View is available. The effects of quantum gates and the transformation of qubit states are explained in Gate Explanation View. By combining State Comparison View with the geometrical visualization of *dandelion chart*, users can gain a better understanding of the measured probability in terms of amplitudes. Additionally, the original quantum circuits, the quantum computing users most familiar with, flatten the learning curves of using the visualization system *QuantumEyes*.

[3] https://www.npmjs.com/package/dandelion_chart

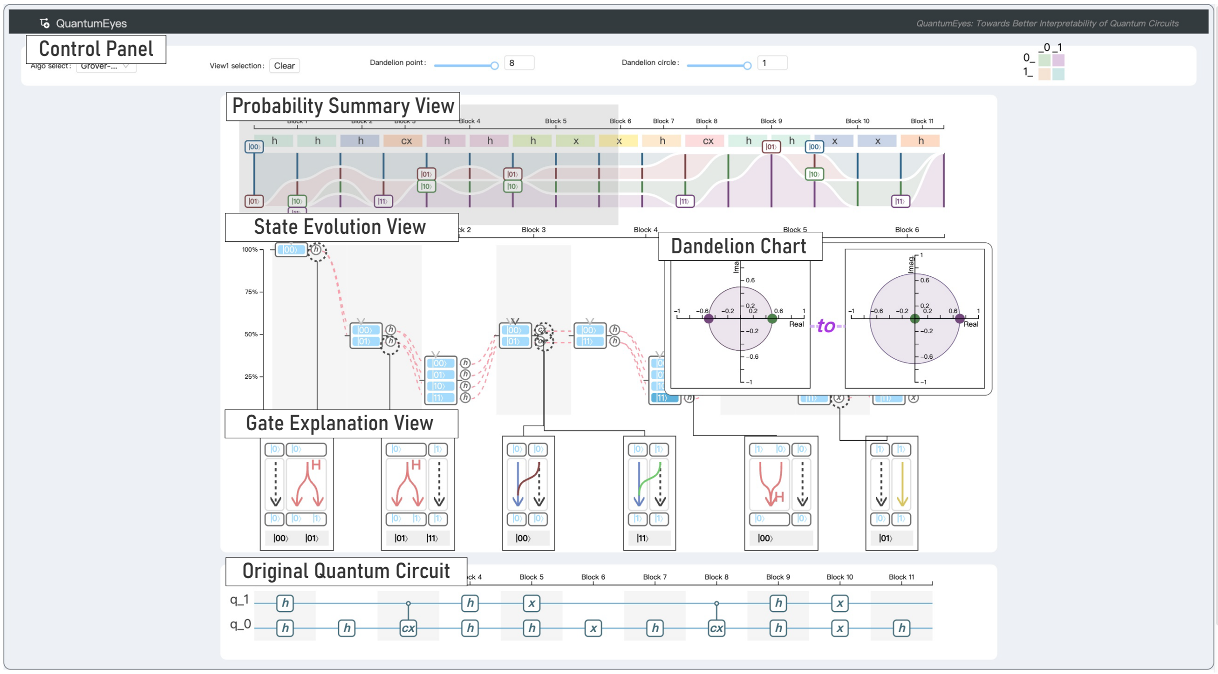


Fig. 7: The system interface of *QuantumEyes*, which consists of four views, an original quantum circuit, and a control panel.

E Matrix Decomposition

As shown in Fig. 8, we illustrate the matrix decomposition using an example of a transformation of CNOT gate. First, the state vector of the initial quantum state is $(0,1,1,0)^{\mathrm{T}}$, which can be split into the sum of the two basis state, i.e., $(0,1,0,0)^{\mathrm{T}}$ and $(0,0,1,0)^{\mathrm{T}}$. Meanwhile ,the basis states of the above two state vectors are $|01\rangle$ and $|10\rangle$. After the transformation of CNOT gate, the above two basis states are converted into $|11\rangle$ and $|10\rangle$, respectively. Thus, we completed the matrix decomposition by manipulating the basis states separately using the quantum gates.

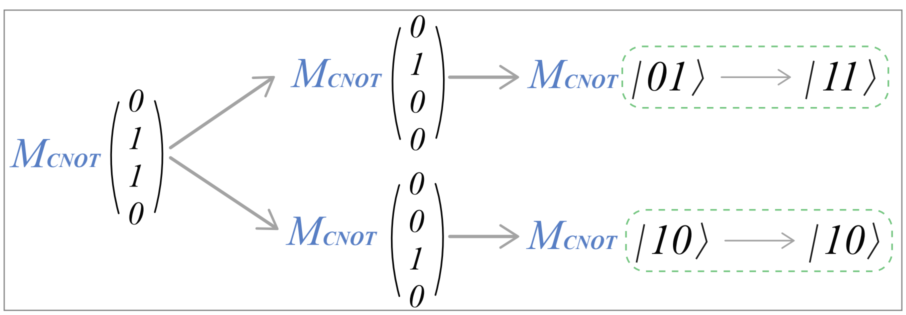


Fig. 8: The illustration of matrix decomposition. We use a transformation of CNOT gate acting on States $|01\rangle$ and $|10\rangle$ as an example. The matrix represents the state vector of the basis states of a quantum state.

F Introduction of Commonly-used Quantum Gates

To illustrate the operations of various quantum gates, we proposed several visual designs to depict the transformation acting on the qubit state. As shown in Fig. 10, we briefly introduce the commonly-used quantum gates regarding the abbreviation, visualization in Gate Explanation View, matrix representation, and the symbol used in the quantum circuit diagram. Specifically, we implemented the operations of six types of quantum gates, including two types of Hadamard gates, two types of CNOT gate, Not gate, and SWAP gate.

Hadamard gate is a basic quantum gate that operates on a single qubit, transforming it into a superposition state. It is represented by a matrix and when applied to a qubit in the $|0\rangle$ or $|1\rangle$ state, it transforms it into a superposition of the two states. The Hadamard gate is a fundamental building block in quantum algorithms and circuits.

CNOT gate, short for controlled NOT gate, is a fundamental quantum gate that operates on two qubits, a control qubit and a target qubit. It performs a NOT operation on the target qubit only when the control qubit is in the $|1\rangle$ state, otherwise, it leaves the target qubit unchanged. The CNOT gate can be represented by a 2×2 matrix and is often used as a basic building block for various quantum algorithms and circuits.

NOT gate, also known as the Pauli-X gate, is a fundamental quantum gate that operates on a single qubit, flipping its state from $|0\rangle$ to $|1\rangle$ or vice versa. The quantum NOT gate can be represented by a 2×2 matrix and is a basic building block for various quantum algorithms and circuits. It plays a similar role as the classical NOT gate in classical computing, but also has additional properties in the quantum realm, such as being able to create entangled states.

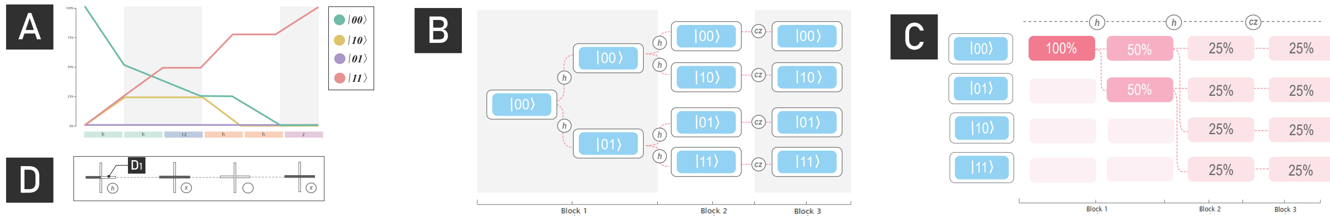


Fig. 9: The design alternatives of *QuantumEyes*. (A) The initial visualization summarizes the probability change across each step in the quantum circuit using the multiple-line chart, where the rectangles in the same color in the x-axis indicate the steps in a common block. (B) The design allows the analysis of the basis state's evolution via a tree diagram without the awareness of the measured probability of each basis state. (C) The visualization approach highlights the probability's change over steps, where the rectangles with different opacity denote the probability of the corresponding basis state. (D) The design to depict the qubit state before and after a gate operation, where the rectangles positioned horizontally denotes the initial state and the rectangles positioned vertically represent the final state.

SWAP gate is a fundamental quantum gate that operates on two qubits, allowing them to exchange their states. When applied to two qubits in the states $|a\rangle$ and $|b\rangle$, the SWAP gate transforms them into the states $|b\rangle\rangle$ and $|a\rangle$, respectively. The quantum SWAP gate can be represented by a 4×4 matrix and is a basic building block for various quantum algorithms and circuits. It is often used to implement quantum data exchange and to swap the states of two qubits in quantum registers.

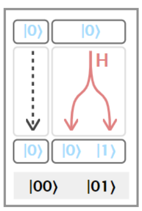
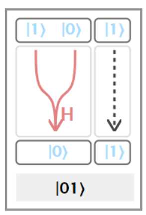
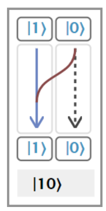
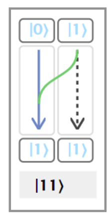
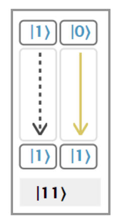
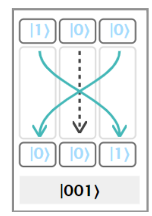
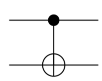
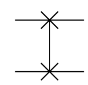
	Hadamard gate (for separating)	Hadamard gate (for merging)	Controlled NOT gate (control qubit is 0)	Controlled NOT gate (control qubit is 1)	NOT gate	SWAP gate
Abbreviation	H gate		CX gate		X gate	S gate
Visualization in Qubit Analysis View						
Matrix representation	$\frac{1}{\sqrt{2}}\begin{bmatrix}1 & 1\\1 & -1\end{bmatrix}$		$\begin{bmatrix}1&0&0&0\\0&1&0&0\\0&0&0&1\\0&0&1&0\end{bmatrix}$		$\begin{bmatrix}0&1\\1&0\end{bmatrix}$	$\begin{bmatrix}1&0&0&0\\0&0&1&0\\0&1&0&0\\0&0&0&1\end{bmatrix}$
Symbol	H				X	

Fig. 10: The introduction of six types of commonly-used quantum gates based on their functionality, including two types of Hadamard gates, two types of Controlled NOT gates, NOT gate, and SWAP gate. For a better illustration, we select the two-qubit transformation for Hadamard gate, Controlled NOT gate, and NOT gate, while we use the three-qubit transformation for the SWAP gate.

G DESIGN ALTERNATIVES

G.1 Probability Summary View

Initially, we attempted to use a multiple-line chart to show the probability change of each basis state in the circuit, as shown in Fig. 9A. However, we encountered problems with the lines crossing over and overlapping, which made it hard to see. The design also didn't make it easy to see the proportion of each basis state's probability. In the end, we decided to use a stacked area chart, which is able to (1) avoid any overlapping between multiple entities, and (2) visually reflect the proportion of each basis state using the fixed height of the vertical coordinate.

G.2 State Evolution View

We explored a couple of different design options before landing on the current visual design. Initially, we tried using a tree diagram to organize all of the basis states (Fig. 9B), but an expert pointed out that there were no discernible patterns regarding the gate's effect on the states, as they were all located evenly in each step. Next, we tried a design that utilized the opacity of corresponding rectangles to visualize the probability of each basis state (Fig. 9C), while the dotted lines depicted the state evolution. However, we encountered problems with severe line overlapping between adjacent steps, so this design was not scalable. Ultimately, we developed the current design, which enables to clustering of the basis state at each step based on their measured probability, making it easy to find the patterns of the quantum circuit and also differentiate the basis states to avoid any visual clutter.

G.3 Gate Explanation View

We also explored other options where a set of rectangles were used to represent the qubit states (Fig. 9 D). A rectangle was colored white if the quantum state is $|0\rangle$, black if it is $|1\rangle$, and half black and half white if it is in a superposition state (Annotation D_1). The initial states were placed vertically, while the final states were placed horizontally. However, this design was not preferred as it only showed the initial and final states without explicitly visualizing the gate operations. Therefore, we use the current visual design with a layout of "before-transformation-after," which can highlight the gate operations process more apparently. The design is more compact to enable the state visualization with more qubits.

H Pre-defined Tasks in Expert Interview

We conducted well-designed expert interview to demonstrate the effectiveness and usability of our visualization system *QuantumEyes*. Specifically, we asked each expert to perform the pre-defined tasks and rate the system based on the exploration of the tasks. All six tasks are categorized into two groups: i.e., the effectiveness evaluation of visual designs for global analysis (**T1-4**) to and local analysis (**T5-6**). We listed all tasks in Table 3 for a better illustration.

T1	Identify the overall trends of all basis states regarding the measured probability.
T2	Identify the generation of the basis states via the trace-back analysis.
T3	Describe the operations and effects of the quantum gates by comparing the basis states before and after.
T4	Explain the effect of the quantum gates from the perspective of the qubit state analysis.
T5	Explain how quantum gates change the amplitudes of the basis states.
T6	Explain how amplitudes of basis states change the corresponding measured probabilities.

Table 3: All pre-defined tasks are grouped into two categories, i.e., the effectiveness evaluation of visual designs for global analysis (**T1-4**) and local analysis (**T5-6**). Specifically, T1-4 are designed to evaluate the purpose of each view for global analysis (i.e., T1: overall trend perception, T2-3: gate effect explanation, T4: fine-grained gate explanation regarding individual qubit), and T5-6 are proposed to test dandelion chart's effectiveness of measured probability explanation.

I System Architecture

Fig. 11 illustrates the architecture of *QuantumEyes*, which consists of three tightly-connected modules: (1) data storage module, (2) data processing module, and (3) visualization module.

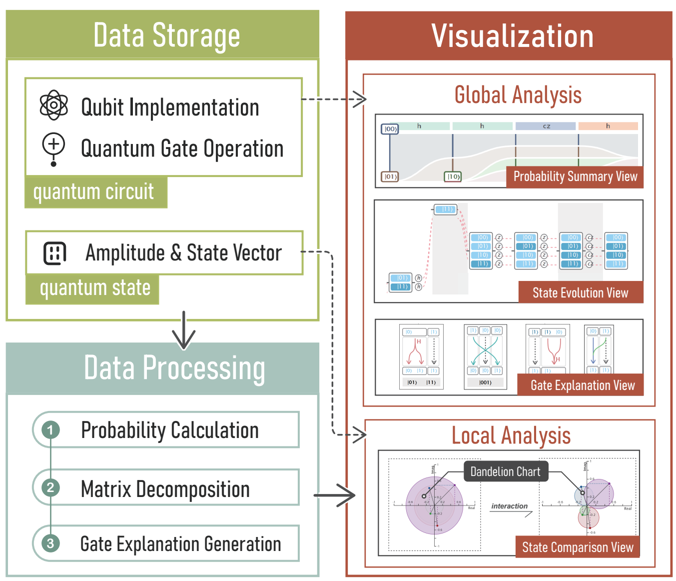


Fig. 11: The system architecture of *QuantumEyes* consists of three modules, i.e, a **data storage** module, a **data processing** module, and a **visualization** module.