A Co-design Process

To better clarify the co-design process, we illustrate the process in Figure 1. The steps connected by the green line indicate the stage of preliminary interview for the initial prototype, while those steps connected by the blue line represent the iterative tuning process in the stage of expert test. Specifically, we first conducted preliminary interviews with five quantum computing experts where we sought to understand the practical challenges and difficulties. Next, over the next four months, we revised our initial prototype iteratively according to the feedback collected from the expert test.

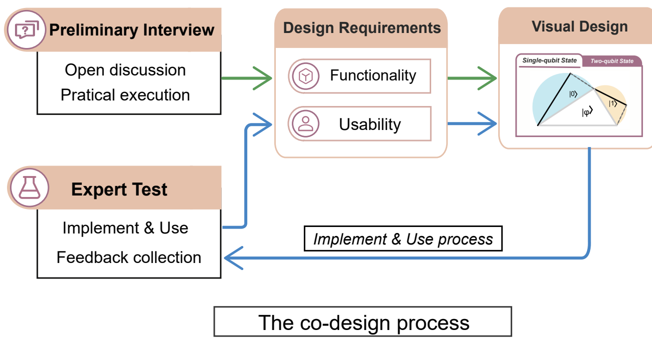

Fig. 1. The workflow of the co-design process with domain experts.

B *VENUS* Interface

According to the design requirements collected from the co-design process, domain experts expected that our visual design *VENUS* could be publicly-available to the quantum computing community (**R5**). Thus, we implemented the visual design of *VENUS* as a web-based interface, which consists of two components: the visualization module (Figure 2(A)) and control panel (Figure 2(B)). Users can easily access the *VENUS* interface at https://venus-interface.github.io/.

User interactions. During the stage of the expert test, we summarized and implemented a set of user interactions, and the details are as follows:

- *VENUS* interface supports two representation modes, *i.e.*, single-qubit mode and two-qubit mode. Users can switch to different modes via the buttons in the control panel (Figure 2(1)), and the widgets for other features will also update accordingly.
- To visualize the quantum states, users need to input the amplitudes of the state vector in the control panel (Figure 2(2)), where four numbers are required for single-qubit states, and eight numbers are required for two-qubit states. At least one value is required for visualization.
- During the iterative expert test process, **P1** reported that it is necessary to switch the display order of the qubit state, which enables users to inspect the probability of measuring the single qubit in a two-qubit state. For example, the display order will change from $[|00\rangle, |01\rangle, |10\rangle, |11\rangle]$ to $[|10\rangle, |11\rangle, |00\rangle, |01\rangle]$ if the order of the first qubit is changed from $0 \rightarrow 1$ to $1 \rightarrow 0$, while the order of the second qubit remains the same. To change the display order of the qubit state, users can change the order of the two qubits through the button in the control panel (Figure 2(3)).

C Suggestions for Improving *VENUS*

In addition to the positive feedback, we also received several valuable suggestions during the expert interviews.

Annotations for Entanglement Given that it is important to observe and analyze if two qubits are entangled or not. To this end, during the interview, E3 suggested that it is helpful to add special annotation on the visual elements to indicate whether the two qubits are entangled or not. For example, some visual hints to indicate the control and target of the entangled qubits would make *VENUS* more informative.

Noisy Result Comparison E2 also reported that it would be interesting to represent the comparison of noise and noise-free results using a glyph on the current state probability distribution. Since *VENUS* is built upon the quantum simulator, which cannot support the noise inspection generated in the real quantum devices, such as the cloud quantum computer provided by IBM Quantum platform. Thus, extending *VENUS* to support the comparison of noise and corresponding probability would enhance the generalizability of *VENUS*.

QNN-specific Visualization Hinted by E2, Quantum Neural Network is a popular topic in the area of quantum computing. However, the instructions of the single-qubit gate (*e.g.*, rotation gates) cannot be intuitively represented via the visual design of *VENUS*, such as the rotation angles of the current quantum states. In the future, we plan to focus on how to intuitively reflect the rotation angles in *VENUS*, preserving the advantages that naturally encode the quantum computing characteristics.

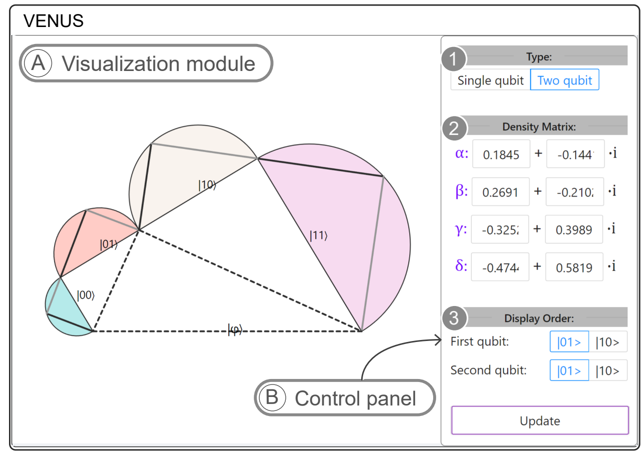

Fig. 2. The interface of *VENUS* tool. (A) The visualization module and (B) the control panel of *VENUS* interface.

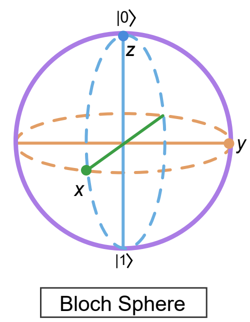

Fig. 3. A widely-used visualization called Bloch Sphere for quantum state representation.

D Bloch Sphere

Bloch Sphere is a famous visualization to represent single-qubit state in the quantum computing community, as shown in Fig. 3. Bloch Sphere leverages a point on the unit sphere to represent the *amplitudes* of a pure single-qubit state. However, Bloch Sphere has several severe issues that needs to be urgently addressed, which we discussed in the paper.